KNOWING WHEN TO STOP: IGP-BENCH AND THE MULTI-AGENT SYSTEM GALOIS FOR THE INVERSE GALOIS PROBLEM

Anonymous authors
Paper under double-blind review

ABSTRACT

Frontier agents can now sustain hours of autonomous work. Greater capability may bring greater progress, but it may also carry an agent farther along a wrong route. Existing long-horizon mathematics benchmarks are hard to build, saturate quickly, or grade partial progress only with a language model, which makes it difficult to reliably measure how much effort an agent spent on each route it tried. We introduce **IGP-Bench**, a benchmark of long-horizon algebraic geometry problems in inverse Galois theory. Its first release has 200 problems in three tiers: 39 with published explicit realizations, 55 whose realizations are known to exist but have not been made explicit, and 106 that remain open. On every problem, a successful route consists of objects that are hard to find but easy to verify, and GAP and Magma check each of them deterministically. Agents record their trajectory as an artifact graph, from which we score verified progress, format compliance, tool use, and **knowing-when-to-stop metrics**: a useful-resource fraction and a stopping-efficiency score comparing how far an agent goes down incorrect routes with how far it goes down correct ones. We test eight frontier models over 80 runs on the five Mathieu-group tasks, each run with a 10-hour wall-clock limit and 500,000 output tokens. Together these tasks offer 32,979 candidate routes, of which only 13 lead to a realization. Surprisingly, we find that Claude Opus 5 has the highest mean progress (76.2), followed by GPT-6 Sol (62.1) and GPT-6 Astra (59.4), and that the gap between them is not one of construction ability but of identifying the right route and of where the budget goes. These gaps motivate **GALOIS**, an open-source multi-agent system with a lead agent, a strategy reviewer, and specialist sub-agents for its 36 written skills, in which a dedicated route rejection agent screens candidate routes for known obstructions in the background, so wrong routes are ruled out quickly. In reference runs for IGP-Bench, GALOIS with a Claude Opus 5.5 lead produces Magma-verified polynomials for all five Mathieu tasks in 10 to 126 minutes, including M_{23} and M_{24}, for which no single agent produced a polynomial. Tasks, verifiers, evaluation code, and GALOIS will be publicly released.

1 INTRODUCTION

AI agents are capable of working for hours autonomously, and the tasks they complete are steadily increasing in both difficulty and length (Kwa et al., 2025). This allows agents to pursue research-level mathematics, but it also uncovers a problem that short tasks conceal: an agent that persists on difficult calculations will also waste resources on a dead end. For example, an account of the recent AI-assisted realization of M_{23} as a Galois group over $\mathbb{Q}$ reports that the most successful models were also the most stubborn in pursuing routes that ultimately failed (Huang et al., 2026; Zhang, 2026). When the coefficients of a curve had to be recognized from numerical data, the models pushed for higher precision, from 40 to 90 digits, until the machine ran out of memory. The problem was not precision but that coordinates were not Galois-equivariant, and once these coordinates were altered, 40 digits sufficed (Zhang, 2026; Chiou, 2026). Under a fixed budget, tokens need to be spent productively for the run to be successful; thus, this paper asks whether more capable models are able to allocate their resources and time efficiently.

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.

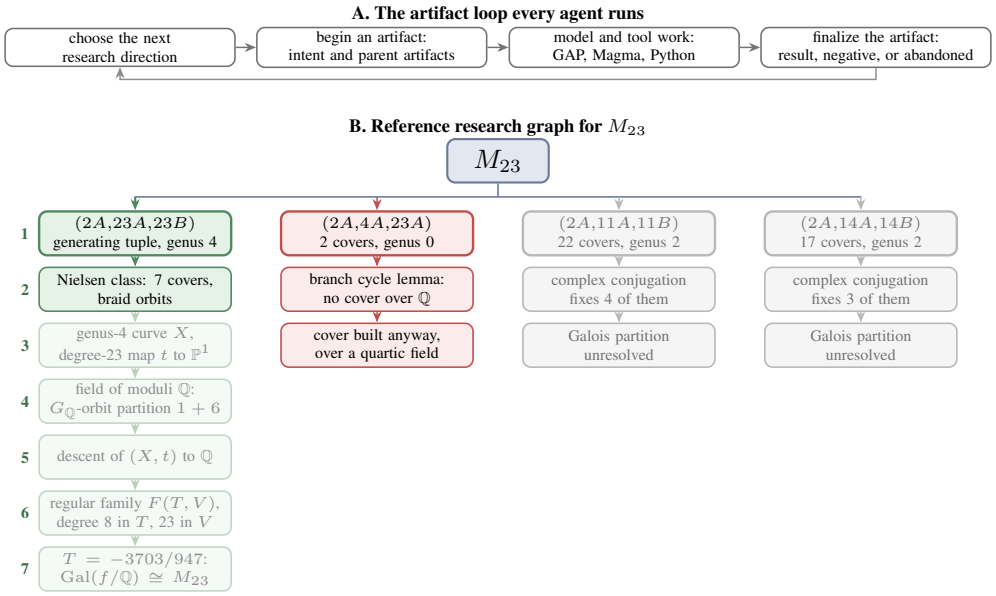


Figure 1: **How the benchmark sees an investigation. A.** Each agent works in a loop of self-declared artifacts, so every token and CPU-hour is attributed to an artifact and every artifact to a route. **B.** Class vectors an agent can choose on M_{23} and the verifiable objects along each. Green: the positive route (Huang et al., 2026), one box per stage of the M_{23} checkpoint ladder (Appendix D). Full boxes are stages some evaluated single-agent run verified, faded boxes stages none reached (Table 15). GALOIS's reference runs complete the column (Section 4). Red: a genus-0 route that the branch cycle lemma rules out from the character table alone, since inversion swaps $23A$ and $23B$, on which several single agents still built a cover over a quartic field (Appendix I.9). Grey: genus-2 routes with no cover over $\mathbb{Q}$ known and no obstruction proved.

The inverse Galois problem for a finite group G asks whether there exists a polynomial $f \in \mathbb{Q}[V]$ with a Galois group isomorphic to G. A *route* is a path to construct an f that satisfies these conditions and is indexed by a tuple of conjugacy classes, such as $(2A, 23A, 23B)$. The problem combines the abstraction of algebraic geometry (covers of the line, Hurwitz spaces, braid orbits) with the verifiability of computational mathematics: every intermediate object, from a generating tuple to the final polynomial, can be verified via GAP or Magma. It is also uniquely suited to studying stopping: a group offers many candidate routes, yet few of them are known to be successful. Since rigidity holds for few tuples, most covers are undefined over $\mathbb{Q}$, and some have genus beyond reach (Malle & Matzat, 2018; Serre, 1992). M_{23} is the most extreme case. Listed among Epoch AI's open problems (Epoch AI, 2026) and covered by *Scientific American* (Chiou, 2026), it resisted rigidity and braid-orbit generalizations, which had settled the previous 25 sporadic groups between 1984 and 1989 (Thompson, 1984; Matzat, 1987; Malle & Matzat, 2018). Of its 816 class triples, one tuple, $(2A, 23A, 23B)$, is known to realize through a genus-4 curve and a degree-23 cover (Huang et al., 2026) (Figure 1).

IGP-Bench turns this problem into an evaluation testbed. Each of its 200 tasks asks for a realization of a finite group over $\mathbb{Q}$ and defines a ladder of verifiable checkpoints. Agents annotate their work as a graph of artifacts, so every token and CPU-hour can be investigated. Each run is asked three questions: did the agent *identify* a correct route, *stop* on incorrect ones, and *continue* on the correct path? A verified progress score S answers the first and the third questions. Two knowing-when-to-stop metrics, the useful-resource fraction U_r and the stopping efficiency V_r, address the second question. None of the questions require an LLM judge. GALOIS is a multi-agent reference system: a lead agent plans, specialist agents execute written procedures for each stage of the work, and a dedicated route rejection agent tells the lead which routes to drop.

Across 80 runs on the five Mathieu groups, mean progress spans from 2.6 to 76.2. Claude Opus 5 leads GPT-6 Sol and GPT-6 Astra, and no single agent produces a polynomial for M_{23} or M_{24}. Construction ability does not explain this ordering, since Astra ties Opus on construction and GPT-6

Sol realizes far less. Where the budget goes differs more: Astra exhausts its $500{,}000$-token cap on M_{23} with under 7% of tokens on credited work, and GPT-6 Sol leaves incorrect routes more cheaply than any other system. With two runs per task, we provide confidence intervals and report allocation scores. GALOIS produces verified polynomials for all five tasks, including M_{23} and M_{24}, which no single agent finishes. Our contributions are IGP-Bench (Section 3), its corresponding knowing-when-to-stop metrics (Section 3.4), GALOIS (Section 4), and its evaluation pipeline (Sections 5–7).

2 RELATED WORK

Mathematics benchmarks and partial credit. FrontierMath and Riemann-Bench supply difficult problems with final evaluation and no partial credit, while HorizonMath includes relative credit for improving a published bound (Glazer et al., 2024; Garre et al., 2026; Wang et al., 2026). IMProofBench adds separately authored, automatically graded subproblems to research-level proofs, and TheoremBench measures machine-verified theorems in Lean (Schmitt et al., 2025; Pham et al., 2026). These benchmarks fix the intermediate structure in advance, whereas IGP-Bench's intermediate objects arise from the work itself and are accepted or rejected by a computer-algebra verifier, not a rubric or judge.

Research agents and multi-agent systems. PaperBench and EXP-Bench score components of research workflows with learned judges as a part of the grading (Starace et al., 2025; Kon et al., 2026). MetaGPT, AutoGen, and CAMEL organize role-specialized agents around a shared plan (Hong et al., 2024; Wu et al., 2023; Li et al., 2023), and FunSearch and AlphaEvolve have produced new constructions via search (Romera-Paredes et al., 2024; Novikov et al., 2025). GALOIS borrows the lead-and-specialist organization and adds verifier-grounded acceptance and explicit machinery for dropping routes.

Recent work establishes M_{23} as a Galois group over $\mathbb{Q}$ and describes the AI-assisted investigation behind that construction (Huang et al., 2026; Zhang, 2026). The present paper treats this realization as prior mathematical work and evaluates route selection, verified intermediate progress, and resource allocation. It does not claim the M_{23} realization as a contribution.

Cost, overthinking, and stopping. Kapoor et al. (2025) argue that accuracy-only agent benchmarks reward costly systems, so IGP-Bench attributes cost to individual lines of investigation. Reasoning models spend computation on problems that do not need it (Chen et al., 2025; Sui et al., 2025), and in agentic tasks extended reasoning displaces useful interaction with the environment (Cuadron et al., 2025). Closest to our question are the abstention benchmarks: AgentAbstain gives agents paired everyday tasks and records one pass or fail per task (Liu et al., 2026), and Luo et al. (2026) ask the same of web, terminal, and question-answering agents. IGP-Bench asks the question of investigations that last hours and have no single correct action: it attributes cost to routes, classifies each against a literature ledger, and reports where the budget went (Table 20 in the appendix).

3 IGP-BENCH

3.1 BACKGROUND

The goal is a polynomial with rational coefficients whose Galois group is G. A common approach builds a *cover*, a map from an algebraic curve to the line, and uses it to obtain a polynomial family $F(T, V)$. For a *regular* family over $\mathbb{Q}(T)$, suitable rational choices of T preserve the Galois group (Serre, 1992; Malle & Matzat, 2018). Each *route* is specified by a list of conjugacy classes, called its *class vector*. Group-theoretic tests can rule out routes before equations are computed. Passing these tests, or even proving that a realization exists, does not supply the polynomial. The benchmark therefore scores route identification and construction separately. Appendix B gives the mathematical definitions and construction steps.

3.2 TASKS

Three tiers. Each of the 200 tasks specifies a finite group G and asks for a polynomial $f \in \mathbb{Q}[V]$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong G$. Domain experts designed the tasks and assigned tiers from the cited literature,

checked on 25 September 2026 against Zywina's census of the small simple groups not known to occur over $\mathbb{Q}$, the LMFDB number-field tables, and Magma's Klüners–Malle database (Zywina, 2025; The LMFDB Collaboration, 2026). The 39 *Tier 1* tasks have a published polynomial or regular family. The 55 *Tier 2* tasks have an existence proof but no explicit polynomial in the cited sources. The 106 *Tier 3* tasks remain open: the six small simple groups the census leaves open and 100 further groups of Lie type, chiefly Suzuki and Ree groups. One of the six, $\mathrm{Sz}(8)$, is already the subject of a September 2026 realization claim (Appendix C), a sign of how actively contested these problems are. Examples and the catalog appear in Appendices A and C. Tier 2 and Tier 3 tasks are scored without reference answers (Appendix F). The campaign of Section 6 ran on the 19 September 2026 build; the audit moved 14 tasks, M_{23} among them, and replaced two (Appendix C).

Checkpoints. Most tasks use five checkpoints: a generating tuple, a braid orbit, a cover, a regular family, and a final polynomial. The first two check the group data that specify candidate covers; the last three check the construction itself. Six tasks use different sequences based on known constructions (Appendix F). At each checkpoint, a GAP or Magma verifier checks mathematical properties. Credit depends on passing these checks, and agents have no access to reference artifacts.

Route ledger. To score route identification and resource allocation, we use a *route ledger* that records class vectors supported by the literature and any known obstructions. Only literature-supported routes for the target group count as *positive*. All others, including routes of unknown status, count as *non-positive* and are treated as wrong for the allocation scores. The classification is conservative: a non-positive route may still succeed. Section 7 discusses the consequences, and Appendix F gives the registration rules.

What the agent sees. The agent receives the group, objective, checkpoint descriptions, and a fixed budget. It works in a persistent sandbox with a shell, Python, GAP, Magma, and an offline corpus of 17 documents (Appendix E). Sources that reveal a task's realization are excluded, and network access is disabled. The agent submits intermediate results in a prescribed format. A formatter checks syntax, but verifier outcomes are withheld until the run ends, so the agent must assess its own progress.

3.3 Example: M_{23}

The M_{23} task illustrates the three parts of an investigation that the benchmark scores.

Preliminary research. The agent enumerates candidate routes and tests their group data. For M_{23}, there are 816 class triples to consider. Verified facts about these candidates earn research credit.

Route identification. The literature supports the route $(2A, 23A, 23B)$. Its group data specify seven candidate covers, exactly one of which can be defined over $\mathbb{Q}$ (Huang et al., 2026). Verifying the group data for this route earns the route milestone; naming it alone does not.

Construction. The agent must compute equations for the cover, express them over $\mathbb{Q}$, and obtain and specialize a regular family (Appendix D). No single agent in our evaluation reached the exact cover equations on this route. Some instead constructed covers over larger number fields on routes already ruled out over $\mathbb{Q}$ (Figure 1). The ledger treats both ruled-out and unknown routes as non-positive, so the allocation metrics count the cost of researching either.

3.4 Metrics

Artifact graph. A *run* is one system's attempt at one task, recorded as a directed graph. Each node, or *artifact*, is a unit of work declared by the agent and contains its tool calls, outputs, and checkpoint submissions. Edges record dependencies, also declared by the agent. The evaluator charges model and tool costs to the active artifact and reports unattributed costs separately. Verifiers determine progress. An LLM judge checks whether artifact descriptions match the execution log; this description-fidelity measure does not affect the score.

Progress score. The progress score has three components:

$$S = R + K + C, \qquad 0 \le R \le 30, \quad K \in \{0, 30\}, \quad 0 \le C \le 40. \tag{1}$$

Research points R reward verified facts about candidate routes and constructions. Each fact is credited once, with weights based on the strength of the evidence, up to 30 points. The *route milestone* K awards 30 points once the agent verifies a positive route. *Continuation credit* C depends on the deepest verified checkpoint; a verified polynomial earns all 40 points, regardless of how it was found. Appendix F gives the acceptance rules and point schedules.

In this release, K is enabled for nine tasks. The other Tier 1 and Tier 2 tasks are scored on R and C only, with a maximum of 70, and Tier 3 tasks are scored under the open-case profile (Appendix I.3). Tables report R, K, and C as percentages of their respective maxima, and **POLY** as the fraction of runs that produce a verified polynomial.

Knowing-when-to-stop metrics. We record input and output tokens, wall-clock time, and tool CPU hours. The two allocation scores below are computed separately for output tokens, wall-clock time, and tool CPU hours; the text uses output tokens unless stated otherwise.

The *useful-resource fraction* U_r measures the share of cost assigned to credited work and its dependencies. Let A be the finalized artifacts, $r(v)$ the cost of artifact v, and $U \subseteq A$ the credited artifacts together with all their ancestors in the graph. Then

$$U_r = 100 \, \frac{\sum_{v \in U} r(v)}{\sum_{v \in A} r(v)}. \tag{2}$$

The *stopping efficiency* V_r compares screening and work on positive routes with work on non-positive routes. We group artifacts by class vector and use a benchmark-defined boundary between preliminary screening and route research. Let P_r be the total screening cost, and $\bar{C}_r^+$ and $\bar{C}_r^-$ the mean costs per positive and non-positive route after screening. Then

$$V_r = 100 \, \frac{P_r + \bar{C}_r^+}{P_r + \bar{C}_r^+ + \bar{C}_r^-}. \tag{3}$$

When defined, $V_r = 100$ means no research cost on non-positive routes. We use the evaluator's recorded route classifications where available and otherwise classify routes from the archived graphs. Appendix I reports disagreements between these procedures.

Because V_r uses the mean cost per route, an agent can score highly after pursuing many cheap non-positive routes; we therefore report route counts alongside it. U_r counts the full cost of a credited artifact and its ancestors, so grouping unrelated work into one artifact can inflate it. Both scores depend on the route ledger and on how agents record their work. A high V_r with low progress does not establish effective stopping. Appendix F gives undefined cases, classification rules, and the other reported metrics: format compliance, description fidelity, tool-call validity, and termination.

4 GALOIS

Section 3.4 measures whether agents leave wrong routes. GALOIS is an open-source multi-agent system in which a dedicated route rejection agent helps the other agents rule out wrong routes quickly: the agents work through the twelve stages of Appendix B, and a route is retired only on a listed obstruction (control loop in Figure 7, mechanisms and source files in Appendix H). A *skill* is a written procedure for one pipeline stage, a *specialist* is a model session that executes one skill, and a *source* is a class vector under investigation, what the benchmark calls a route.

Control loop. One run is one goal and one loop. A *lead mathematician* agent is spawned fresh at every step, reads the plan, the journal, the artifacts on disk, and an engine-generated context, and writes one decision: a step of up to six substeps (registered tool calls or specialist briefs), or done. A deterministic *format checker* validates the decision, and an optional read-only *strategy checker* judges it for grounding, goal progress, and tool fit, with at most two rejections per step so that a broken critic cannot block a run. The *runner* starts all substeps at once, tools as bounded subprocesses and specialists as headless model sessions, and records each result with a status and an outcome class. The lead plans every step.

Specialists and tools. Thirty-six skills written from the constructive literature (Table 8 in the appendix) cover the pipeline stage by stage, from the group dossier and Nielsen census to the regularity certificate and specialization, plus the four-point lane and obstruction methods, and drive 122 registered tools over GAP, Magma, PARI/GP, the Belyi package, and a FLINT-based power-series solver.

Route rejection. Stopping is engineered in three layers, from cheap to authoritative. *Screens and ranking* at entry admit only class vectors with positive generating counts and Galois-stable class multisets and rank them by Nielsen count and genus, but reject nothing. *Budgets and deferrals* bound each route: a Gröbner budget per coordinate chart, a typed precision ladder that raises precision once by a computed margin and refuses a second unmotivated raise, an evidence ledger that classifies a cover as promising, precision-limited, likely obstructed, or stalled, and a route transition that retires a source once every cover has a terminal disposition. A deferred route keeps its status unknown. *Rejection authority* is a whitelist: only an exact field-of-moduli obstruction, an audited passport obstruction, or a verified descent or regularity negative may retire a cover, and the two review agents (a background obstruction agent and a candidate reviewer) only advise. At the run level a no-progress guard, a spend cap, a circuit breaker on three consecutive failed steps, and a done gate that refuses a conclusion while a computation runs bound the whole run, which stops at the first certified specialization. The precision ladder transcribes the M_{23} episode of Section 1: it was added after development runs in which the lead re-raised precision on one cover until stopped by hand.

Results on the five tasks. We ran GALOIS on the five tasks with Claude Opus 5.5 at medium effort as lead, the strategy checker off, and no spend cap, each run set to stop at its first certified specialization (Appendix H). Table 9 in the appendix lists ten runs, two per task: a batch of five launched together on one commit, and the five single runs that first certified each task. All ten runs succeed, and every task ends with a verified polynomial in 10 to 126 minutes. On M_{11} and M_{24} both runs first took the top-ranked three-point source, $(4A, 4A, 5A)$ and $(3A, 3A, 12B)$, the exact field-of-moduli test returned *not defined over* $\mathbb{Q}$ on every retained cover, and the run moved to the queued four-point row, $(2A, 2A, 3A, 4A)$ and $(2A, 2A, 2A, 12B)$, and finished there. Both M_{23} runs reached the degree-23 polynomial on $(2A, 23A, 23B)$, in 41 and 43 minutes. Magma recomputation confirms every polynomial's Galois group. They are reference runs: the skills carry the mathematical knowledge of the constructive literature, so they set a time scale for the tasks rather than a like-for-like comparison with the single agents. They did not pass through the bridge, so Table 1 scores them from the run records, as marked there (Appendix H): $S = 100$ and a token V_r of 82.8, lowered by the M_{11} and M_{24} runs, which spent 27% and 53 to 65% of their tokens on the rejected sources. First-batch adapter runs through the bridge, with three backends, are in Appendix H (Figure 8).

5 EXPERIMENTAL SETUP

Tasks and systems. The evaluation uses the five Mathieu groups M_{11}, M_{12}, M_{22}, M_{23}, and M_{24}: four Tier 1 tasks and M_{23}, listed under Tier 2 because the catalog's source sheet had not yet incorporated its August 2026 realization, which is excluded from the corpus. Together they expose 27 checkpoints (the generic ladder for four tasks and the seven-stage ladder for M_{23}) and, counting unordered class multisets, $4{,}752$ three-point and $28{,}227$ four-point candidate class vectors, of which the ledger lists 13 as positive routes, about one in $2{,}500$. We evaluate eight models (a *system* is a model with its harness and effort setting) through native Claude, native Codex, or an OpenRouter-backed adapter, all behind the same metered, offline Model Context Protocol bridge: Claude Opus 5, GPT-6 Sol, GPT-6 Astra, GPT-5.6 Sol and Terra, Claude Sonnet 5, DeepSeek V4 Pro, and GLM-5.3 Flash, all at effort *xhigh*, with two main-campaign runs per task (Table 4, eligibility in Appendix I). Only the Opus run records carry a flag that known-answer stores were blocked, so terminal success is not measured under one protocol.

Sandbox, budgets, verification. Each run executes in a network-isolated workspace with Python, GAP, and Magma (Appendix G), with 10 hours, $500{,}000$ output tokens, eight CPUs, and task-dependent memory per problem. A budget limit ends the run, but finalized artifacts remain scoreable. Verifiers run after the agent's work ends and are not charged to its budget. Each returns `verified`, `not_verified`, or `unsupported`, and only the first advances a checkpoint (caps in Appendix G).

Table 1: **Main results on IGP-Bench**, averaged over each system's eligible task-runs. R, K, C as percentages of their maxima (30, 30, 40). S on the 100-point scale. POLY is the share of task-runs with a verified terminal polynomial. U_r and V_r are over output tokens. I_f is format compliance, η description fidelity, Valid and Term. tool-call validity and termination. — marks an unavailable value (not zero). Intervals are in Table 14 and eligibility in Appendix I. †Reference runs with a Claude Opus 5.5 lead, scored from the run records rather than by the evaluator (Section 4, Appendix H): C from the Magma-verified polynomials, R and K assigned for the exact class-vector enumeration and the ledger-positive routes, the rest computed over pipeline steps.

	Verified progress					Allocation		Protocol		Tool use		
System	Res. R	Route K	Cont. C	Total S	POLY (%)	Useful U_r	Stop V_r	Format I_f	Fidel. η	Valid (%)	Term. (%)	Runs n
Claude Opus 5	94.0	90.0	52.5	**76.2**	50.0	57.7	72.8	96.9	99.0	99.8	91.2	10
GPT-6 Sol	97.0	80.0	22.5	**62.1**	20.0	31.4	79.1	99.2	98.4	100.0	98.7	10
GPT-6 Astra	78.0	50.0	52.5	**59.4**	50.0	60.3	55.9	98.2	100.0	99.8	96.7	10
GPT-5.6 Sol	72.7	40.0	12.5	**38.8**	10.0	27.5	57.8	98.4	98.0	99.9	94.2	10
GPT-5.6 Terra	49.7	20.0	2.5	**21.9**	0.0	24.6	38.2	96.3	99.7	99.9	94.0	10
Claude Sonnet 5	35.0	0.0	2.5	**11.5**	0.0	26.9	30.6	93.9	97.3	99.8	95.5	10
DeepSeek V4 Pro	22.7	0.0	2.5	**7.8**	0.0	34.4	66.6	86.5	88.3	99.6	88.6	10
GLM-5.3 Flash	8.7	0.0	0.0	**2.6**	0.0	33.5	63.9	53.1	75.0	97.2	82.0	10
GALOIS†	100.0	100.0	100.0	**100.0**	100.0	82.8	82.8	—	—	—	—	10

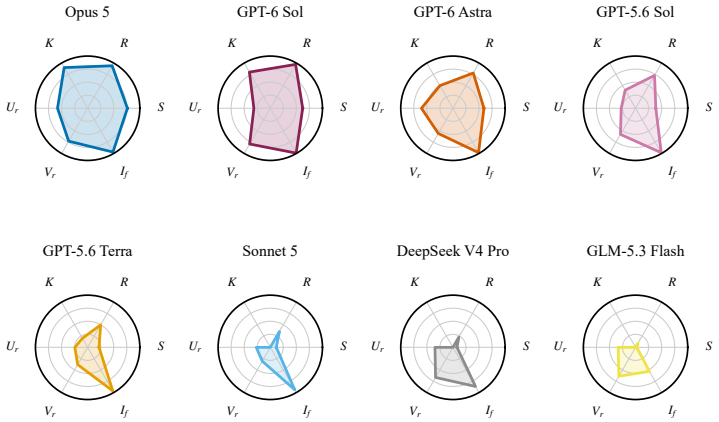


Figure 2: Six-metric profile of each of the eight single-agent systems (GALOIS is in Table 1) on a 0–100 scale: progress S, research R, route K, useful-resource fraction U_r, stopping efficiency V_r, and format compliance I_f. The leaders differ mainly in K and U_r. The V_r lobes of Sonnet, DeepSeek, and GLM reflect little route-level work.

6 RESULTS AND ANALYSIS

Overall performance. Table 1 gives the averages over the 80 eligible runs (Figure 3a shows the credit levels, Figure 2 each system's six-metric profile, and Figure 11 in the appendix the scores by task). Claude Opus 5 leads with $S = 76.2$. GPT-6 Sol is second ($S = 62.1$) with the highest research score ($R = 97.0$) but a verified polynomial in only two runs, and GPT-6 Astra is third ($S = 59.4$). GPT-5.6 Sol (38.8) reaches one polynomial in ten runs, and the remaining four score between 21.9 and 2.6 without one. Astra, Sonnet, and DeepSeek earn continuation credit in more runs than route credit: the second M_{23} checkpoint pays on any class vector, and four runs built polynomials on routes the ledger does not list (Appendix I).

Among the leaders, the gap is in route identification, not construction. On continuation, Astra ties Opus ($C = 52.5$, POLY 50%), while GPT-6 Sol converts far less ($C = 22.5$). On research, GPT-

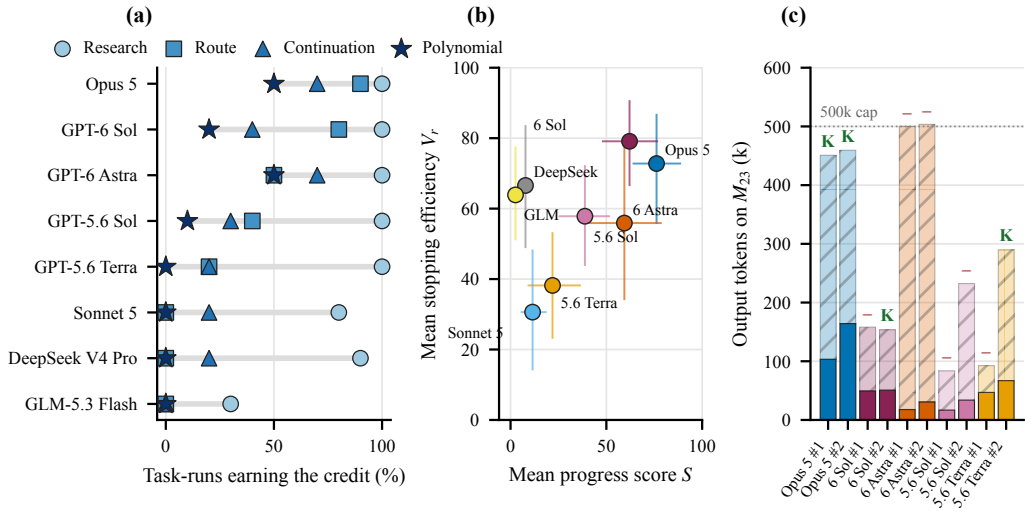


Figure 3: **Where credit is earned, and progress against stopping.** (a) Share of the task-runs of each single-agent system that earn research credit, identify a positive route, earn continuation credit, and produce a verified polynomial. The levels are not nested: the M_{23} stage-2 checkpoint and polynomials on non-positive routes earn continuation credit without route credit. (b) Mean S against mean V_r (tokens) for the eight systems with 95% bootstrap intervals over runs. The low scorers in the upper left research one to three routes per run (Table 13), so their high V_r reflects little route-level work. (c) Output tokens of every M_{23} run of the five leading systems, split into credited artifacts (solid) and the rest (hatched). K marks runs that identified the positive route. Both Astra runs exhaust the cap with almost nothing credited.

6 Sol is above Opus ($R = 97.0$ against 94.0) and Astra sixteen points behind. Route identification shows the largest observed gap: Opus earns K in 9 of 10 runs, GPT-6 Sol in 8 of 10, Astra in 5 of 10. With two runs per task these are observations rather than estimates: a bootstrap over runs gives a 95% interval of $[-5, 39]$ for the Opus–Astra difference in S and $[-10, 43]$ for the difference in V_r (per-system and task-clustered intervals in Table 14), so the ordering of the top three is not established by this campaign.

M_{23} as a case study. The leaders do separate on M_{23}, where each spent most or all of its budget. All sixteen single-agent runs (Table 15; the five leading systems in Figure 3c) fall into three patterns. Opus finds the positive route in both runs, spends 451k and 460k tokens with U_r of 22.9 and 35.7, and stalls at the cover equations ($C = 5$ of 40, 12.5% in the tables). GPT-6 Sol and Terra each find it once, and no other system does. GPT-6 Sol researches 25 and 22 routes, uses the full 10 hours in both, and has U_r of 31.2 and 33.0. Astra never finds the route: both runs end at the 500k-token cap after 8 and 13 researched routes, none positive, with U_r of 3.5 and 6.1. The leaders thus earn the same C on M_{23} and differ in where the budget went.

Stopping efficiency depends on the resource unit and the classification. Table 13 in the appendix reports V_r and U_r in tokens, tool CPU hours, and wall-clock time. In tokens, among the three leaders, GPT-6 Sol has the highest V_r (79.1), then Opus (72.8) and Astra (55.9). In CPU hours the order is GPT-6 Sol, Astra, Opus (61.1, 54.4, 53.0). The classification matters too: the evaluator's route verdicts and the retrospective census disagree on 12 of the 80 runs (Table 12), and under the census alone Astra's token V_r is 58.0 rather than 55.9. The route counts explain much of the rest: GPT-6 Sol researches 19 non-positive routes per run and leaves each one cheaply, and Astra commits to three and stays on them. DeepSeek and GLM research one to three routes per run, so their token V_r above 60 next to S below 8 reflects little route-level work. Aggregate stopping efficiency therefore does not rank the leaders reliably, while their M_{23} runs above do differ.

Routes break at continuation, and stopping is a separate axis. No single agent produces a polynomial for M_{23} or M_{24}: on M_{24} the four leading systems identify a positive route in every run and earn no continuation credit, and on M_{23} the runs that find $(2A, 23A, 23B)$ stall at the cover equations. Systems identify the right class vector far more often than they finish the construction (K in 28 of the 80 runs, a verified polynomial in 13), and the failure taxonomy of Appendix I.9, coded over 79 runs that differ from the eligible set by one, agrees: timed-out computations in 57% of runs, unsolved cover equations in 44%, descent to $\mathbb{Q}$ not achieved in 30%. The knowing-when-to-stop metrics carry information S does not: S correlates strongly with its components ($\rho = 0.92$ with R, 0.83 with K) and continuation with terminal success (0.76), but U_r and V_r correlate with S at only 0.44 and 0.38, and with each other at 0.28 (Appendix I.7).

Tier 3 tasks. Under the open-case profile (Appendix I.3), one Claude Opus 5 run per Tier 3 task averages $S = 45.1$: six of ten settled a candidate passport by an exact obstruction ($S = 60$), four earned research credit only, and none produced a cover (Figure 12). Three further exploratory Opus runs on open groups earned research credit only (Figure 13), so the open tier leaves ample headroom for stronger systems.

7 Discussion

Capability does not predict stopping, in either direction. The M_{23} episode of Section 1 suggested that the most capable models persist longest on wrong routes. The campaign does not support that reading: the two highest scorers, Opus and GPT-6 Sol, also leave non-positive routes most cheaply (V_r of 72.8 and 79.1 in tokens). Nor does it establish the converse, since the leaders' intervals overlap (Table 14). What the M_{23} runs do show is that the leaders fail in different places. Astra never identifies the route and spends its whole token cap on routes the ledger does not support. Opus and GPT-6 Sol find the route and stall at the cover equations, a continuation failure rather than a stopping failure. The failure taxonomy agrees: 27% of coded runs stopped with budget left and 62% overspent on a dead end (Appendix I.9). The remedy is not a shorter horizon but a stopping decision conditioned on route-level evidence. GALOIS builds that decision into the system: its route rejection agent tests candidate routes against known obstructions in the background, and its runs retired the top-ranked three-point routes of M_{11} and M_{24} on an exact obstruction, moved to four-point routes, and finished both, while its reference runs reach the M_{23} polynomial.

Scope of the metrics and the campaign. V_r is an average investment per non-positive route: it characterizes route-level spending rather than any single decision. "Non-positive" means "not literature-supported": four runs built verified polynomials on routes the ledger does not list and are scored as non-positive, and the grey routes of Figure 1 are dead ends by that definition only. The first campaign evaluates five related tasks with ten runs per system, two per task, across three harnesses, with answer stores blocked only for the Opus runs and two verifier issues (Appendix I). The remaining 195 tasks, every Tier 3 task among them, are not yet evaluated; six Tier 3 statuses were revised in the 25 September 2026 audit (Section 3.2). GALOIS is scored from its run records as a reference system: its runs did not pass through the bridge, used a newer lead model at a different effort with the strategy checker off, and followed development rounds that fixed tool and routing defects. A balanced campaign with uniform answer-store blocking, GALOIS runs through the bridge, and ablations of strategy review and route rejection comes next.

Conclusion. IGP-Bench makes the cost of a wrong route measurable: verifiers check every intermediate object, and the artifact graph attributes tokens and CPU time to routes. Runs that stop short of a polynomial still differ measurably in research, route identification, and continuation, so the benchmark records what an investigation establishes, not only whether it ends in success. The first campaign separates frontier agents well before saturation: Claude Opus 5 has the highest mean progress (76.2), followed by GPT-6 Sol (62.1) and GPT-6 Astra (59.4), and route identification, rather than construction ability, separates the leaders, together with where the budget goes. Current systems recognize the right structure far more reliably than they finish the construction, and a multi-agent system that rejects wrong routes early finished all five tasks. The results point toward mathematical agents judged not only on whether they solve a problem but on the verifiable knowledge they produce and how well they decide to persist, redirect, or stop; IGP-Bench measures exactly that.

ETHICS STATEMENT

IGP-Bench concerns constructive problems in pure mathematics and involves no human subjects, personal data, or sensitive content. Tasks and the fixed literature corpus are drawn from cited, published mathematical literature, which we redistribute only where licenses permit. Mathematical progress is scored by deterministic verifiers. The only LLM judge in our protocol checks whether an agent's reports match the execution log and plays no role in scoring progress, and the failure taxonomy of the appendix was coded with LLM assistance for qualitative diagnostics only. Since Tier 3 tasks are listed as open problems, a verified realization produced by an agent would constitute a new mathematical result, and we will disclose the system, model, and configuration responsible for such a result. Benchmark scores reflect a specific harness and should not be read as general capability rankings of the underlying models. We foresee no dual-use risk from the release of the benchmark or of GALOIS.

REPRODUCIBILITY STATEMENT

The 200 tasks with their checkpoint definitions, the GAP and Magma verifiers, the route ledger and scoring profiles, the evaluation and metric code, the agent manual and task briefs, the MCP benchmark bridge, and the full GALOIS source will be released publicly upon publication. An anonymized snapshot of the evaluation code, the per-run data behind every table and figure, and the scripts that regenerate them is provided as supplementary material. Section 5 and Appendices G–I give software requirements, sandbox configuration, budgets, per-system settings, and eligibility decisions. Appendices C–H document the catalog, the M_{23} ladder, the corpus, and the GALOIS mechanisms with their source files. The preserved run directories (artifact graphs, checkpoint submissions, tool logs, and abandoned artifacts) will be published with the code. Two limitations apply: Magma-dependent verifiers require a Magma license, and frontier-model outputs are nondeterministic and subject to provider-side changes, so agent trajectories can be reproduced only in distribution.

AI USE STATEMENT

In this work we used generative AI tools for drafting and revising text, for writing LaTeX and the scripts that process the per-run data and generate tables and figures, for literature search, and for checking the manuscript against the released data and code. We did not use generative AI tools to generate research ideas, to produce experimental results, or to score any run (the failure taxonomy was coded with LLM assistance and enters no score): all benchmark scores come from deterministic verifiers and the evaluation code, and the only model in the scoring pipeline is the description-fidelity judge described in Section 3.4, whose output does not enter the score. The evaluated agents are themselves LLM systems, which is the object of study. We reviewed all AI-assisted work: every number quoted in the text was checked against the released tables and data files, the mathematical statements were checked by the authors against the cited sources, and generated code was run and its outputs inspected. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Benjamin Antieau, Jürgen Klüners, and Gunter Malle. Jürgen Klüners and Gunter Malle on the inverse Galois problem and computation. Interview, https://antieau.github.io/2026/08/27/gunter-malle-juergen-klueners-ai.html, 2026. 27 August 2026.

Gennadii V. Belyi. On Galois extensions of a maximal cyclotomic field. *Mathematics of the USSR-Izvestiya*, 14(2):247–256, 1980.

Sebastian Biallas. A regular realization of the Suzuki group $\mathrm{Sz}(8)$ over $\mathbb{Q}$. Zenodo, https://doi.org/10.5281/zenodo.22730384; code and Lean certificate at https://github.com/sebastianbiallas/sz8-over-q, 2026. Published 13 September 2026; not yet independently verified by us.

Johan Bosman. A polynomial with Galois group $\mathrm{SL}_2(\mathbb{F}_{16})$. arXiv:math/0701442, 2007.

Johan Bosman. Modular forms applied to the computational inverse Galois problem. arXiv:1109.6879, 2011.

William Y. Chen. Noncongruence modular curves as Hurwitz spaces. arXiv:2510.12003, 2025.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT think that much for 2+3=? on the overthinking of long reasoning models. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. arXiv:2412.21187.

Lyndie Chiou. Mathematicians use AI to find mysterious symmetries, solving decades-old problem. *Scientific American*, September 2026. 22 September 2026, `https://www.scientificamerican.com/article/mathematicians-use-ai-to-find-mysterious-symmetries-solving-decades-old-problem/`.

Kevin Coombes and David Harbater. Hurwitz families and arithmetic Galois groups. *Duke Mathematical Journal*, 52(4):821–839, 1985.

Jean-Marc Couveignes. Tools for the computation of families of coverings. In *Aspects of Galois Theory*, volume 256 of *London Mathematical Society Lecture Note Series*, pp. 38–65. Cambridge University Press, 1999.

Alejandro Cuadron, Dacheng Li, Wenjie Ma, Xingyao Wang, Yichuan Wang, Siyuan Zhuang, Shu Liu, Luis Gaspar Schroeder, Tian Xia, Huanzhi Mao, Nicholas Thumiger, Aditya Desai, Ion Stoica, Ana Klimovic, Graham Neubig, and Joseph E. Gonzalez. The danger of overthinking: Examining the reasoning-action dilemma in agentic tasks. In *Conference on Language Modeling*, 2025. arXiv:2502.08235.

Pierre Dèbes and Jean-Claude Douai. Algebraic covers: field of moduli versus field of definition. *Annales Scientifiques de l'École Normale Supérieure*, 30(3):303–338, 1997.

Luis Dieulefait, Enric Florit, and Núria Vila. Seven small simple groups not previously known to be Galois over $\mathbb{Q}$. *Mathematics*, 10(12):2048, 2022.

Epoch AI. The inverse Galois problem for the Mathieu group M_{23}. FrontierMath: Open Problems, `https://epoch.ai/frontiermath/open-problems/inverse-galois`, 2026. Listed as unsolved by January 2026; marked "solved (human + AI)" as of 25 September 2026.

Michael D. Fried. Fields of definition of function fields and Hurwitz families—groups as Galois groups. *Communications in Algebra*, 5(1):17–82, 1977.

Michael D. Fried and Helmut Völklein. The inverse Galois problem and rational points on moduli spaces. *Mathematische Annalen*, 290:771–800, 1991.

Suhaas Garre, Erik Knutsen, Sushant Mehta, and Edwin Chen. Riemann-Bench: A benchmark for moonshot mathematics. *arXiv preprint*, 2026. arXiv:2604.06802.

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*, 2024. arXiv:2411.04872.

Louis Granboulan. Construction d'une extension régulière de $\mathbb{Q}(T)$ de groupe de Galois M_{24}. *Experimental Mathematics*, 5(1):3–14, 1996.

Robert Guralnick and Gunter Malle. Rational rigidity for $E_8(p)$. *Compositio Mathematica*, 150: 1679–1702, 2014.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, 2024.

Xiaoyu Huang, Blake Jackson, Kyu-Hwan Lee, Bjorn Poonen, Rachel Pries, and Shaowu Zhang. The Mathieu group M_{23} is a Galois group over $\mathbb{Q}$. *arXiv preprint*, 2026. arXiv:2608.08538.

Sayash Kapoor, Benedikt Stroebl, Zachary S. Siegel, Nitya Nadgir, and Arvind Narayanan. AI agents that matter. *Transactions on Machine Learning Research*, 2025. arXiv:2407.01502.

Emir Eray Karabiyik. The groups $\mathrm{SL}_2(\mathbb{F}_{13})$ and $\mathrm{SL}_2(\mathbb{F}_{19})$ are Galois over $\mathbb{Q}$. arXiv:2609.03946, 2026. 3 September 2026.

Michael Klug, Michael Musty, Sam Schiavone, and John Voight. Numerical calculation of three-point branched covers of the projective line. *LMS Journal of Computation and Mathematics*, 17 (1):379–430, 2014.

Patrick Tser Jern Kon, Qiuyi Ding, Jiachen Liu, Xinyi Zhu, Jingjia Peng, Jiarong Xing, Yibo Huang, Yiming Qiu, Jayanth Srinivasa, Myungjin Lee, Mosharaf Chowdhury, Matei Zaharia, and Ang Chen. EXP-Bench: Can AI conduct AI research experiments? In *International Conference on Learning Representations*, 2026.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Dietrich, Elizabeth Hasin, Josh Isaacs, Grant Jones, et al. Measuring AI ability to complete long tasks. *arXiv preprint*, 2025. arXiv:2503.14499.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: Communicative agents for "mind" exploration of large language model society. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Xun Liu, Yi Evie Zhang, Vira Kasprova, Parisa Rabbani, Pardis Sadat Zahraei, Tianyu Zhang, Ali Ebrahimpour-Boroojeny, and Varun Chandrasekaran. AgentAbstain: Do LLM agents know when not to act? *arXiv preprint*, 2026. arXiv:2607.10059.

Han Luo, Bingbing Wen, and Lucy Lu Wang. Agentic abstention: Do agents know when to stop instead of act? *arXiv preprint*, 2026. arXiv:2606.28733.

Gunter Malle. Polynomials with Galois groups $\mathrm{Aut}(M_{22})$, M_{22}, and $\mathrm{PSL}_3(\mathbb{F}_4) \cdot 2_2$ over $\mathbb{Q}$. *Mathematics of Computation*, 51(184):761–768, 1988.

Gunter Malle and B. Heinrich Matzat. *Inverse Galois Theory*. Springer Monographs in Mathematics. Springer, 2nd edition, 2018.

B. Heinrich Matzat. *Konstruktive Galoistheorie*, volume 1284 of *Lecture Notes in Mathematics*. Springer, 1987.

B. Heinrich Matzat and Andreas Zeh-Marschke. Realisierung der Mathieugruppen M_{11} und M_{12} als Galoisgruppen über $\mathbb{Q}$. *Journal of Number Theory*, 23(2):195–202, 1986.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. AlphaEvolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint*, 2025. arXiv:2506.13131.

QuocViet Pham, Elvir Karimov, Andrey Galichin, and Ivan Oseledets. TheoremBench: Evaluating LLMs on theorem proving in formal mathematics. *arXiv preprint*, 2026. arXiv:2606.09450.

Stefan Reiter. Galoisrealisierungen klassischer Gruppen. *Journal für die reine und angewandte Mathematik*, 511:193–236, 1999.

Matthieu Romagny and Stefan Wewers. Hurwitz spaces. In *Groupes de Galois arithmétiques et différentiels*, volume 13 of *Séminaires et Congrès*, pp. 313–341. Société Mathématique de France, 2006.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625:468–475, 2024.

Johannes Schmitt, Gergely Bérczi, Jasper Dekoninck, Jeremy Feusi, Tim Gehrunger, Raphael Appenzeller, Jim Bryan, Niklas Canova, Timo de Wolff, Filippo Gaia, Michel van Garrel, Baran Hashemi, David Holmes, Aitor Iribar Lopez, Victor Jaeck, Martina Jørgensen, Steven Kelk, Stefan Kuhlmann, Adam Kurpisz, Chiara Meroni, Ingmar Metzler, Martin Möller, Samuel Muñoz-Echániz, Robert Nowak, Georg Oberdieck, Daniel Platt, Dylan Possamaï, Gabriel Ribeiro, Raúl Sánchez Galán, Zheming Sun, Josef Teichmann, Richard P. Thomas, and Charles Vial. IMProofBench: Benchmarking AI on research-level mathematical proof generation. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2025. arXiv:2509.26076.

Jean-Pierre Serre. *Topics in Galois Theory*, volume 1 of *Research Notes in Mathematics*. Jones and Bartlett, 1992.

Jeroen Sijsling and John Voight. On computing Belyi maps. *Publications Mathématiques de Besançon. Algèbre et Théorie des Nombres*, pp. 73–131, 2014.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Patwardhan. PaperBench: Evaluating AI's ability to replicate AI research. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, 2025.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models. *Transactions on Machine Learning Research*, 2025. arXiv:2503.16419.

The LMFDB Collaboration. The L-functions and modular forms database. `https://www.lmfdb.org`, 2026. Number-field tables by transitive Galois group, queried 25 September 2026.

John G. Thompson. Some finite groups which appear as $\mathrm{Gal}\, L/K$, where $K \subseteq \mathbb{Q}(\mu_n)$. *Journal of Algebra*, 89(2):437–499, 1984.

Helmut Völklein. *Groups as Galois Groups: An Introduction*, volume 53 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1996.

Erik Y. Wang, Sumeet Motwani, James V. Roggeveen, Eliot Hodges, Dulhan Jayalath, Charles London, Kalyan Ramakrishnan, Flaviu Cipcigan, Philip Torr, and Alessandro Abate. Measuring progress in reasoning toward mathematical discovery with automatic verification. *arXiv preprint*, 2026. arXiv:2603.15617.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint*, 2023. arXiv:2308.08155.

Shaowu Zhang. Where to point the telescope: AI, mathematical judgment, and M_{23}. Proofs and Prompts, `https://proofsandprompts.com/2026/08/21/where-to-point-the-telescope/`, 2026. Published 21 August 2026.

David Zywina. Inverse Galois problem for small simple groups. Note, `https://pi.math.cornell.edu/~zywina/papers/smallGalois.pdf`, 2025. Last updated August 2025.

A SAMPLE TASKS

Figure 4 shows one task from each of the three tiers.

Tier 1 M_{11}**, degree 11**	**Tier 2** Co_3**, degree 276**	**Tier 3** $\mathrm{Sz}(32)$**, degree 1025**
Problem: Construct $f(V) \in \mathbb{Q}[V]$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong M_{11}$.	**Problem:** Construct $f(V) \in \mathbb{Q}[V]$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong \mathrm{Co}_3$.	**Problem:** Construct $f(V) \in \mathbb{Q}[V]$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong \mathrm{Sz}(32)$.
Answer: $f(V) = V^{11} - 4V^{10} - 75V^7 + 540V^6 - 210V^5 - 22500V^3 + 56000V^2 - 82100V - 57600$	**Answer:** — (existence known by rigidity of $(2A, 5B, 30A)$; no polynomial published)	**Answer:** — (unknown whether $\mathrm{Sz}(32)$ occurs over $\mathbb{Q}$)
MSC: 12 Field theory and polynomials; 20 Group theory	**MSC**: 20 Group theory; 12 Field theory and polynomials	**MSC**: 12 Field theory and polynomials; 11 Number theory

Figure 4: One IGP-Bench task from each tier. The question is identical in all three: given G, construct $f(V) \in \mathbb{Q}[V]$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong G$. In Tier 1 an explicit polynomial is published. In Tier 2 the group is known to occur over $\mathbb{Q}$ but no polynomial is published. In Tier 3 it is unknown whether the group occurs at all, as of the catalog's 19 September 2026 source sheet. Every submitted answer is checked deterministically in GAP or Magma.

B THE CONSTRUCTIVE PIPELINE

B.1 MATHEMATICAL BACKGROUND

Regular realizations. Fix a finite group G. Almost every known realization of a nonabelian simple group over $\mathbb{Q}$ comes from a *regular* realization over $\mathbb{Q}(T)$, a polynomial $F(T, V) \in \mathbb{Q}[T, V]$ with $\mathrm{Gal}(F/\mathbb{Q}(T)) \cong G$ and $\mathbb{Q}$ algebraically closed in the splitting field, which Hilbert irreducibility specializes to polynomials over $\mathbb{Q}$ with the same group (Serre, 1992; Völklein, 1996; Malle & Matzat, 2018).

Covers and route screening. The route to such an F is geometric. By Riemann's existence theorem, a G-cover of the line branched over r points corresponds to a *generating tuple* $(g_1, \ldots, g_r)$ with $g_1 \cdots g_r = 1$ and $\langle g_1, \ldots, g_r \rangle = G$, up to simultaneous conjugation. Fixing the conjugacy classes $C = (C_1, \ldots, C_r)$ of the g_i fixes a *Nielsen class* $\mathrm{Ni}(G; C)$, on which the braid group acts, and its orbits are the components of the Hurwitz space parametrizing the covers (Fried, 1977; Fried & Völklein, 1991; Romagny & Wewers, 2006). A G-cover whose branch divisor is defined over $\mathbb{Q}$ can be defined over $\mathbb{Q}$ only if its class vector is stable under the absolute Galois group acting on classes and branch points together and its braid orbit is Galois-invariant. The condition at the real place is that the orbit be fixed by complex conjugation. Rational classes over rational branch points are the simplest case. For M_{23} the classes $23A$, $23B$ and the branch points $\pm\sqrt{-23}$ are swapped together by $\mathrm{Gal}(\mathbb{Q}(\sqrt{-23})/\mathbb{Q})$. Necessary conditions for a route are thus decided early, by group-theoretic data. Sufficiency, as the M_{23} fixed point shows, is not.

Rigidity and explicit construction. When G has trivial center, $r = 3$, the classes are rational, and $|\mathrm{Ni}(G; C)| = 1$, the tuple is *rigid* and a regular realization over $\mathbb{Q}$ exists without any equations. This criterion of Belyi (1980), Thompson (1984), and Matzat (1987), with its braid-orbit generalizations, settled 25 of the 26 sporadic groups. A polynomial, however, needs the cover itself, as computed explicitly for M_{11}, M_{12}, M_{22}, and M_{24} (Matzat & Zeh-Marschke, 1986; Malle, 1988; Granboulan, 1996). The modern route is numerical uniformization, recognition of exact equations, descent to $\mathbb{Q}$, and a specialization (Couveignes, 1999; Klug et al., 2014; Sijsling & Voight, 2014). The early screens in the pipeline below eliminate most class vectors.

B.2 PIPELINE STAGES

Figure 5 shows the twelve stages of the constructive pipeline that GALOIS follows and on which the IGP-Bench checkpoints sit.

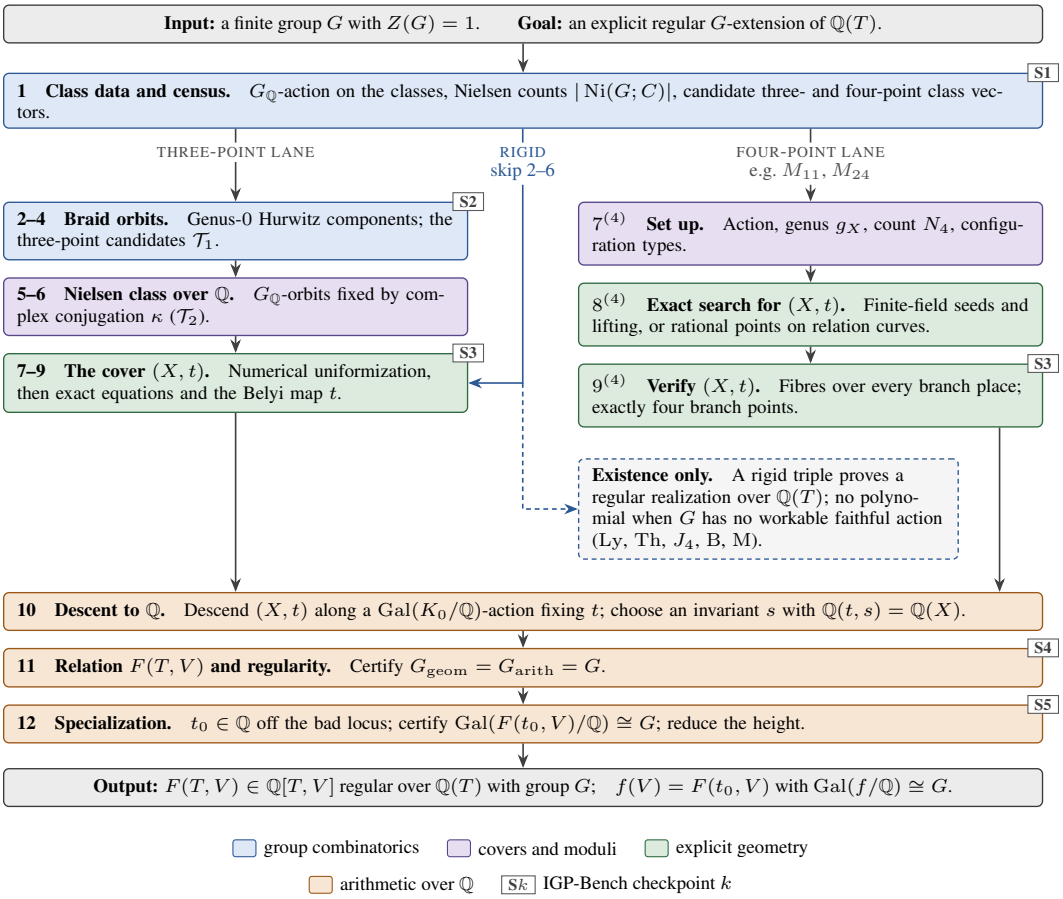


Figure 5: The constructive pipeline behind GALOIS and IGP-Bench verification. Stage 1 is common to every group; the route then splits by the number of branch points and rejoins at Stage 10, where descent to $\mathbb{Q}$ and certification are identical for both lanes. Only Stage 7 is numerical. A rigid triple ($|\mathrm{Ni}(G;C)| = 1$) skips Stages 2–6; for a group with no faithful action of workable degree it proves existence only (no polynomial). White tags mark the five IGP-Bench verifier checkpoints of the generic ladder: S1 generating tuple, S2 braid orbit, S3 geometric cover, S4 regular family over $\mathbb{Q}(T)$, S5 terminal polynomial over $\mathbb{Q}$.

C TASK CATALOG

Table 2 summarizes the catalog by family, and Table 3 lists every catalog task of the first release with its permutation degree (the degree of the registered action), its tier, its ladder, and the number of literature routes the ledger holds for it. Degrees in parentheses are the target polynomial degrees of the four passport-only tasks, which are defined by branch data only and have no registered action (the catalog records 14 and 20, the degrees of the PSL_2 quotient actions, for the two SL_2 tasks). Ledger routes are counted as direct routes plus routes attached through an almost-simple ambient group, and 12 of the 141 ledger routes concern groups outside the catalog and are not counted in the table. Seventy entries carry a *needs review* flag in the catalog because their tier rests on a source outside the team's source sheet or on an unchecked condition; the tier audit below checked every task against the census, the LMFDB tables, and the cited sources. The ten Tier 3 groups were chosen by a Magma screen of class triples (cyclotomically closed triples, structure constants, Riemann–Hurwitz

Table 2: **The IGP-Bench catalog.** Tasks by group family and tier (Tier 1: explicit realization published. Tier 2: existence proved, no explicit polynomial in the sources. Tier 3: listed as open). 94 tasks use the generic five-step ladder and six use bespoke ladders. The route ledger holds 141 literature routes (103 three-point, 38 four-point, 34 rigid) for 64 of the groups. In the present release the route milestone K is configured for the five Mathieu tasks and the four bespoke-ladder tasks, and the other Tier 1 and Tier 2 tasks are scored on R and C until their ledger routes are registered as positive. Tiers are those of the 19 September 2026 build used in the campaign. The audit of 25 September moves 14 tasks and replaces two (Appendix C).

Family	**Tier 1**	**Tier 2**	**Tier 3**	**Total**	**Examples**
Sporadic	7	16	0	23	M_{11}, M_{23}, J_1, Co_3, Fi_{22}
Alternating	9	0	0	9	A_5–A_{13}
PSL, PGL, SL	10	32	5	47	$\mathrm{PSL}_2(p)$ for $7 \le p \le 113$, $\mathrm{SL}_2(13)$, $\mathrm{PSL}_4(3)$
Other groups of Lie type	5	8	5	18	$\mathrm{PSU}_3(3)$, $G_2(4)$, ${}^2F_4(2)'$, $\mathrm{Sz}(8)$
Rubik's cube groups	0	2	0	2	$4 \times 4 \times 4$, $5 \times 5 \times 5$
Cyclic (smoke test)	1	0	0	1	C_5
All	32	58	10	100	

genus, sampled generating Nielsen classes), excluding groups with a rational rigid generating triple and ranking by the fewest generating classes and lowest genus. Tiers in Table 3 are those of the 19 September 2026 build used in the campaign, with * marking the individually listed tasks that the audit below moves.

Table 3: The first 100 of the 200 release tasks.

Task	Group	Degree	Tier	Ladder	Ledger routes
Sporadic groups					
m11	M_{11}	11	1	generic, bespoke (8)	3
m12	M_{12}	12	1	generic	5
m22	M_{22}	22	1	generic	7
m22_2	M_{22}:2	22	1	generic	4
hs	HS	100	1	generic	4
hs_2	HS:2	100	1	generic	3
m24	M_{24}	24	1	generic	3
m23	M_{23}	23	2*	generic, bespoke (7)	1
j1	J_1	266	2	generic	1
j2	J_2	100	2	generic	2
j2_2	J_2:2	100	2	generic	1
m12_2	M_{12}:2	396	2	generic	1
mcl	McL	275	2	generic	1
mcl_2	McL:2	275	2	generic	1
he	He	2058	2	generic	2
he_2	He:2	2058	2	generic	2
ru	Ru	4060	2	generic	2
suz	Suz	1782	2	generic	1
suz_2	Suz:2	1782	2	generic	1
co3	Co_3	276	2	generic	1
co2	Co_2	2300	2	generic	1
fi22	Fi_{22}	3510	2	generic	2
fi22_2	Fi_{22}:2	3510	2	generic	2
Alternating groups					
psl2_5	$A_5 \cong \mathrm{PSL}_2(5)$	5	1	generic	2
a6	A_6	6	1	generic	1
a7–a13	$A_7, \ldots, A_{13}$	7–13	1	generic	0
Projective and special linear groups					
psl2_7	$\mathrm{PSL}_2(7)$	7	1	generic	4
pgl2_7	$\mathrm{PGL}_2(7)$	8	1	generic	2
psl2_11	$\mathrm{PSL}_2(11)$	11	1	generic	8
pgl2_11	$\mathrm{PGL}_2(11)$	12	1	generic	4
psl2_13	$\mathrm{PSL}_2(13)$	14	1	generic	2
pgl2_13	$\mathrm{PGL}_2(13)$	14	1	generic	0
psl2_8	$\mathrm{PSL}_2(8)$	9	1	generic	0
psl3_3	$\mathrm{PSL}_3(3)$	13	1	generic	2
psl3_4	$\mathrm{PSL}_3(4)$	21	1	generic	3
psl5_2	$\mathrm{PSL}_5(2)$	31	1	generic	2
psl2_p (24 tasks)	$\mathrm{PSL}_2(p)$, $17 \le p \le 113$ prime	$p+1$	2	generic	2 each
pgl2_17	$\mathrm{PGL}_2(17)$	18	2*	generic	0
psl2_25	$\mathrm{PSL}_2(25)$	26	2*	generic	1
psl2_49	$\mathrm{PSL}_2(49)$	50	2*	generic	1
psl2_121	$\mathrm{PSL}_2(121)$	122	2	generic	1
psl3_5	$\mathrm{PSL}_3(5)$	31	2	generic	0

Task	Group	Degree	Tier	Ladder	Ledger routes
psl3_9	$\mathrm{PSL}_3(9)$	91	2	generic	1
sl2_13	$\mathrm{SL}_2(13)$	(56)	2	bespoke (7)	0
sl2_19	$\mathrm{SL}_2(19)$	(40)	2	bespoke (7)	0
psl2_16	$\mathrm{PSL}_2(16)$	17	3*	generic	0
psl2_27	$\mathrm{PSL}_2(27)$	28	3*	generic	0
psl2_125	$\mathrm{PSL}_2(125)$	126	3*	generic	0
psl3_7	$\mathrm{PSL}_3(7)$	57	3*	generic	0
psl4_3	$\mathrm{PSL}_4(3)$	40	3*	generic	0
Other groups of Lie type					
psu3_3	$\mathrm{PSU}_3(3)$	28	1	generic	1
psp4_3	$\mathrm{PSp}_4(3)$	27	1	generic	2
psp4_4	$\mathrm{PSp}_4(4)$	85	1*	generic	2
psp6_2	$\mathrm{PSp}_6(2)$	28	1	generic	2
o8_2	$O_8^+(2)$	120	1	generic	2
psu4_3	$\mathrm{PSU}_4(3)$	112	2	generic	1
psu5_2	$\mathrm{PSU}_5(2)$	165	2	generic	1
psu6_2	$\mathrm{PSU}_6(2)$	672	2	generic	1
psu3_7	$\mathrm{PSU}_3(7)$	344	2	generic	0
g2_3	$G_2(3)$	351	2	generic	1
g2_4	$G_2(4)$	416	2	generic	1
3d4_2	${}^3D_4(2)$	819	2	generic	1
2f4_2	${}^2F_4(2)'$	1600	2	generic	1
sz_8	$\mathrm{Sz}(8)$	65	3	generic	0
sz_32	$\mathrm{Sz}(32)$	1025	3	generic	0
psu3_4	$\mathrm{PSU}_3(4)$	65	3	generic	0
psu3_5	$\mathrm{PSU}_3(5)$	50	3*	generic	0
psu3_8	$\mathrm{PSU}_3(8)$	513	3	generic	0
Rubik's cube groups and smoke test					
r4	$4 \times 4 \times 4$ cube group	(96)	2	bespoke (5)	0
r5	$5 \times 5 \times 5$ cube group	(150)	2	bespoke (6)	0
c5	C_5	5	1	generic	1

Tier audit of 25 September 2026. After the campaign, every task was checked against Zywina's census of the simple groups of order at most 10^8 not known to occur over $\mathbb{Q}$ (Zywina, 2025), the LMFDB number-field tables (The LMFDB Collaboration, 2026), Magma's Klüners–Malle database, and the cited sources. Six of the ten Tier 3 groups had been realized: $\mathrm{PSL}_2(16)$ has a published explicit polynomial (Bosman, 2007) and moves to Tier 1, and $\mathrm{PSL}_2(27)$, $\mathrm{PSL}_2(125)$, $\mathrm{PSU}_3(5)$, $\mathrm{PSL}_3(7)$ (Dieulefait et al., 2022) and $\mathrm{PSL}_4(3)$ (Reiter, 1999) move to Tier 2. Seven Tier 2 groups have a published explicit polynomial over $\mathbb{Q}$ and move to Tier 1: M_{23} (Huang et al., 2026), $\mathrm{PSL}_2(19)$ (Karabiyik, 2026), $\mathrm{PSL}_2(25)$ and $\mathrm{PSL}_2(49)$ (Bosman, 2011), and $\mathrm{PSL}_2(17)$, $\mathrm{PSL}_2(23)$, and $\mathrm{PGL}_2(17)$, for which the LMFDB lists number fields with the corresponding transitive Galois group. $\mathrm{SL}_2(13)$ has an explicit polynomial in the files accompanying Karabiyik (2026), with coefficients too large for the Magma verifier, and its tier awaits a team decision. $\mathrm{PSp}_4(4)$ moves from Tier 1 to Tier 2 because its cited source realizes $\mathrm{PSp}_4(4)$:2 only. The two remaining certified-open groups of the census, $\mathrm{PSL}_3(8)$ and $\mathrm{PSU}_3(9)$, enter Tier 3, and $\mathrm{PSL}_2(109)$ and $\mathrm{PSL}_2(113)$, whose route and evidence duplicate the other $\mathrm{PSL}_2(p)$ tasks, leave the catalog. After the rebuild the counts are 39 Tier 1 (40 with $\mathrm{SL}_2(13)$), 55 (54) Tier 2, and 6 Tier 3, still 100 catalog tasks. In Table 3 three of the aggregated $\mathrm{PSL}_2(p)$ tasks move to Tier 1 and two leave. Of the six Tier 3 groups, $\mathrm{Sz}(8)$ was reconfirmed open on 27 August 2026 (Antieau et al., 2026), has a conditional realization assuming parity, ABC, and Chowla (Chen, 2025), and is the subject of a regular-realization claim of 13 September 2026 (Biallas, 2026) that has not yet been independently verified.

Open-case profile and research catalog. For Tier 3 the repository carries a development implementation of a separate 30/30/40 profile. Its route component certifies a geometric route by verifier predicates (a product-one tuple in the pinned action, certified generation, a complete braid orbit, a route-geometry certificate) instead of matching a literature ledger, in the current implementation only an exactly certified polynomial earns continuation credit, the implemented research predicates cap R at 16, and a realization over $\mathbb{Q}$ is reported as a separate flag rather than folded into the score. Six development tasks exist ($\mathrm{Sz}(32)$, $\mathrm{PSL}_3(8)$, $\mathrm{PSU}_3(4)$, $\mathrm{PSU}_3(8)$, $\mathrm{PSU}_3(9)$, and $\mathrm{Sz}(8)$) with a preregistered pilot of ten runs, two per group under a five-hour limit, that has not been launched. Alongside the task catalog, a research-only registry lists 104 simple-group identities: 97 Suzuki groups $\mathrm{Sz}(2^{2m+1})$ and large Ree groups ${}^2F_4(2^{2m+1})$ with q up to 2^{103}, whose open status is family-level and provisional (Guralnick & Malle, 2014), and seven small seeds, four of which are also catalog tasks. Together the two lists name 200 distinct simple groups.

D THE M_{23} LADDER AND WORKED ROUTE

The bespoke ladder. The M_{23} task has seven checkpoint stages instead of the generic five, with cumulative continuation credit C of $0, 0, 5, 15, 20, 25, 30, 40$ points for the deepest verified stage $0, \ldots, 7$: (1) a generating tuple of degree-23 permutations with product one and a valid genus (S1); (2) a complete list of pairwise inequivalent Nielsen representatives for an ordered class vector, enumerated in GAP under a cap of 2×10^6 (S2, 5 points); (3) a geometric cover, either a $\mathbb{Q}$-equation with branch cycles or a number-field cover with marking data (S3, 15); (4) a certificate that the field of moduli is $\mathbb{Q}$ (S4, 20); (5) a descended cover over $\mathbb{Q}$ with its descent isomorphisms (S4, 25); (6) a regular family over $\mathbb{Q}(T)$, checked through exact branch data and a good finite-characteristic reduction with group M_{23}, because the verifier does not attempt Magma's generic geometric-Galois-group computation in degree 23, which does not complete in practical time (S4, 30); (7) a degree-23 polynomial, irreducible with square discriminant and Galois group M_{23} certified with Magma's `GaloisProof`, or a specialization certificate for a verified family (S5, 40). The route milestone K on M_{23} is the *key case*: $K = 30$ when a verified braid-orbit, Nielsen-count, or Nielsen-representative result has the class vector $(2A, 23A, 23B)$ as its subject. Stage 2 credit is awarded for a complete representative list on *any* class vector, which is why several systems earn $C = 5$ on M_{23} without K (Table 15). Research points come from verified facts in six categories (finite counts and structure for three- and four-point vectors, rationality, arithmetic), with weights from 1 (product one) to 6 (a reduced Hurwitz component, a Hurwitz model, an existence theorem), capped at 30.

The worked route. The positive route runs through the four-point class vector $(23A, 23A, 23B, 23B)$ and its three-point degenerations $(2A, 23A, 23B)$. The steps below are the ones a solver has to complete.

1. **Candidate discovery.** M_{23} has 16 nontrivial conjugacy classes and $\binom{18}{3} = 816$ class triples up to order. No Galois-stable triple has a Nielsen class of size one, so rigidity fails, and the smallest positive Nielsen count is 7, for $\{2A, 23A, 23B\}$ (Huang et al., 2026). The routes the literature had tried before, $(2A, 8A, 8A)$ among them, are recorded in the benchmark's negative-case census.
2. **Braid orbits.** Seeds (a, a, b, aba^{-1}) with $a \in 23A$, $b \in 23B$, $ab \in 2A$ are moved by the pure braid generators $\beta_{12}, \beta_{13}, \beta_{14}$ in GAP until each orbit closes. The seven seeds give seven orbits of the four-point Nielsen class.
3. **Genus and cusps.** On each orbit Z_i, viewed as a degree-$|Z_i|$ cover of the λ-line branched at $0, 1, \infty$, the generators act with cycle counts $(c_1, c_2, c_3) = (2, 1, 1)$, so by Riemann–Hurwitz $g = 1 + \frac{1}{2}\big(|Z_i| - c_1 - c_2 - c_3\big) = 1 + \frac{1}{2}(2 - 4) = 0$. The cusps record the degenerate triples $(23A, 23B, 23B)$ and $(2A, 23A, 23B)$ obtained by colliding branch points.
4. **Promotion of the triple.** Every orbit representative is verified to have the form (a, a, b, aba^{-1}) with $\langle a, b\rangle = M_{23}$, the class set $\{2A, 23A, 23B\}$ is Galois-stable, and the number of verified orbits equals $|\,\mathrm{Ni}(M_{23}; 2A, 23A, 23B)| = 7$. This is the route milestone K.
5. **Branch points over $\mathbb{Q}$.** Since $23A$ and $23B$ are swapped by $\mathrm{Gal}(\mathbb{Q}(\sqrt{-23})/\mathbb{Q})$, the branch locus is the zero set of $x(x^2 + 23)$, with $0 \mapsto 2A$ and $\pm\sqrt{-23} \mapsto 23A, 23B$, so that the class map is Galois-equivariant, which it would not be with $\{0, 1, \infty\}$.
6. **Galois action on the covers.** The absolute Galois group acts on the seven covers. One might expect it to act transitively, but it has a fixed point, which therefore has field of moduli $\mathbb{Q}$ and, by the results of Coombes & Harbater (1985) and Dèbes & Douai (1997), descends to a cover over $\mathbb{Q}$ (Huang et al., 2026). Complex conjugation acts by $[(g_1, g_2, g_3)] \mapsto [(g_1^{-1}, g_3^{-1}, g_2^{-1})]$.
7. **Analytic model.** The curve X has genus 4 and is not hyperelliptic. A uniformization $X_{\mathbb{C}} \cong \Gamma\backslash\mathbb{D}$ produces four holomorphic differentials as power series at a suitable point.
8. **Exact equations.** The four series give the canonical map to $\mathbb{P}^3$, whose image is cut out by a quadric P and a cubic Q with coefficients in $\mathbb{Q}(\sqrt{-23})$, recognized from the numerical data and checked exactly. This is where the precision episode of Section 1 happened: in triangle-group coordinates the coefficients were not recognized at 40, 60, or 90 digits, while in coordinates defined through the Belyi map, which are Galois-equivariant, 40 digits sufficed for P, and Q was recovered from the same 40-digit data by a separate reduction algorithm (Zhang, 2026).
9. **The cover.** The degree-23 map $t\colon X \to \mathbb{P}^1$ is found from the points above the branch locus.

10. **Descent.** The cover is rewritten over $\mathbb{Q}$ and a second function s is chosen with $\mathbb{Q}(X) = \mathbb{Q}(t, s)$.
11. **Relation and regularity.** A polynomial $F(T, V) \in \mathbb{Z}[T, V]$ of degree 8 in T and 23 in V with $F(t, s) = 0$ is computed and certified: irreducibility, branch behavior, and $G_{\mathrm{geom}} = G_{\mathrm{arith}} = M_{23}$, the latter through a reduction modulo 31 checked in Magma.
12. **Specialization.** Rational values t_0 off the bad locus give $f(V) = F(t_0, V)$ with $\mathrm{Gal}(f/\mathbb{Q}) \cong M_{23}$, reduced with `polredabs` and certified in Magma. The published example is unramified outside $\{2, 3, 23\}$.

In the present evaluation the systems that identify the route complete steps 1–4 and some of 5–6. None completes step 8.

E Literature Corpus

The offline corpus (`igpbench-general-math-v4`) contains 17 documents, searched through a deterministic offline BM25 index. It is deliberately background material: textbooks, course notes, and expository articles on Galois theory, Hurwitz spaces, and the relevant groups, together with three evaluator-written guides. A document is admitted only after a manual check that it contains no final polynomial, coefficient vector, family, or specialization for an active task, no task-specific branch-cycle tuple, Belyi map, or calibration fixture, no worked construction copyable by transcription, and no link or appendix with such data. Explicitly excluded are the M_{11} polynomial papers and König's M_{11} family, the M_{23} realization (Huang et al., 2026) and its supplements, the $\mathrm{SL}_2(13)$ and $\mathrm{SL}_2(19)$ constructions, the "Rubik's cube as a Galois group" preprint, and any A_5 polynomial (one section of Crespo's article is redacted). An audit script checks the corpus for known title and phrase markers of the excluded sources.

1. J. S. Milne, *Fields and Galois Theory*, course notes, v5.10 (2022).
2. J. S. Milne, *Group Theory*, course notes, v4.01 (2025).
3. K. Conrad, *Galois Correspondence*, expository note.
4. *Outline of Galois Theory Development*, Stanford Math 210B course handout.
5. L. Chariker, *The Inverse Galois Problem, Hilbertian Fields, and Hilbert's Irreducibility Theorem*, REU paper (2007).
6. M. D. Fried, *Nielsen Classes and Hurwitz Spaces*, web notes.
7. The Stacks Project Authors, *Algebraic Curves* (chapter 0BRV), 2026 snapshot.
8. R. A. Wilson, *Sporadic Groups*, chapter 5 of the finite simple groups course notes.
9. P. M. Bradley, *Counting G-Orbits on the Induced Action on k-Subsets*, PhD thesis, Manchester (2014).
10. T. Crespo, *Galois Representations, Embedding Problems and Modular Forms*, Collectanea Math. 48 (1997), with the example polynomials of Section 3.2 redacted.
11. M. Kuschkowitz, *Quadratische Formen, Einbettungsprobleme und die Hasse–Witt-Invariante*, Master's thesis, TU Braunschweig (2010).
12. D. Sen, K. Bauer, P. Zvengrowski, *A Generalized Goursat Lemma*, J. Algebra Appl. 12 (2013).
13. R. Fernando, *Structure and Generation Properties of the Rubik's Cube Group*, expository notes (2017).
14. The GAP Group, *GAP Tutorial*, release 4.16.1 (2026).
15. IGP-Bench, *Magma, GAP, and Polynomial-Galois Workflow Guide*, evaluator documentation.
16. IGP-Bench, *Composita, Central Lifts, and Discriminant Square Classes*, evaluator documentation.
17. IGP-Bench, *IGPBench Task Literature Map*, an agent-searchable reading map.

F Metric Definitions

Checkpoint sequences. The generic ladder has five stages: generating tuple (S1), braid orbit (S2), geometric cover (S3), regular family over $\mathbb{Q}(T)$ (S4), and terminal polynomial (S5). Six tasks—M_{11}, M_{23}, $\mathrm{SL}_2(13)$, $\mathrm{SL}_2(19)$, and two Rubik's cube groups—use task-specific ladders based on

known constructions. Appendix D gives the seven-stage M_{23} ladder. At each checkpoint, acceptance is a predicate checked by GAP or Magma; no match to a stored reference artifact is required.

Research and construction credit. Research points R come from independently verified auxiliary facts about tuples, orbits, families, and polynomials. Each fact is credited once, and the total is capped at 30. On the generic ladder, weights range from 0.5 to 3 according to the strength of the evidence, from statistical evidence to an existence theorem. Task-specific schedules, including the one for M_{23} in Appendix D, use different weights. Continuation credit C follows the deepest verified checkpoint: 15 for a cover, 30 for a regular family, and 40 for a terminal polynomial on the generic ladder. Task-specific ladders assign intermediate credit differently. A polynomial found by any method receives $C = 40$ once verified.

Route ledger and milestone. The ledger records 141 distinct literature routes for 64 groups, together with known obstructions. A positive route must support a realization over $\mathbb{Q}(T)$ for the target group. A route in an overgroup such as $\mathrm{Aut}(M_{22})$ is not positive for M_{22}. On the generic ladder, the route milestone $K = 30$ requires one artifact to contain both a generating tuple on a registered positive route and a verified braid-orbit computation on it. It is awarded at most once. Task-specific acceptance rules apply where configured; for M_{23}, Appendix D gives the accepted braid-orbit, Nielsen-count, and Nielsen-representative certificates.

Resource usage. We record the model and reasoning configuration, input tokens with cache accounting where available, output tokens, benchmark elapsed time, and measured tool CPU hours. Separately reported reasoning-token counts are included when available and are not added again when already included in output tokens. Tool CPU usage excludes model-server inference. Missing measurements are distinguished from zero usage.

Format compliance. A deterministic formatter checks artifact records and final submissions against public formatting requirements, allowing at most two repair reminders. We define

$$I_f = \frac{100\, n_0 + 50\, n_1 + \frac{100}{3}\, n_2}{n_0 + n_1 + n_2 + n_{\mathrm{fail}}},$$

where n_i counts submissions accepted after exactly i reminders and n_{fail} counts submissions that remain unaccepted. Retries do not count as new submissions. The score is undefined when no submissions are recorded. Format acceptance does not imply mathematical correctness.

Description fidelity. A judge compares finalized result and negative-result records with the execution log and classifies each as correct, incorrect, or undetermined, with counts N_c, N_i, N_u. We define $\eta = 100\, N_c/(N_c + N_i)$, undefined when no record has a determinate judgment, and report all three counts alongside η. Object provenance, declared dependencies, recording completeness, and final-submission consistency are reported separately.

Tool calling. Let T be all tool calls and $T_{\mathrm{v}} \subseteq T$ the calls with parseable actions and valid arguments. Validity is $|T_{\mathrm{v}}|/|T|$ and termination is the fraction of valid calls that run to completion (a process that exits with a nonzero status still counts as terminated). Termination measures tool reliability, not how the run ended.

Figure 6 works through U_r and V_r on a toy artifact graph.

Tasks without a known answer. Tier 2 tasks have no reference polynomial and Tier 3 tasks no positive route, but progress on them is still measured. Research points are property-based and apply unchanged: a verified Nielsen count or rationality obstruction earns the same credit on Co_3 as on M_{11}. For Tier 2 the ledger records the tuple whose rigidity proves existence, so route identification and every checkpoint up to the cover are scored as in Tier 1, and the verifier needs no terminal reference. For Tier 3, a tuple that passes the structural verifier without being on any ledger route is recorded as a *new route*, earning research credit at once and route credit after independent verification. A verified realization of an open group would be a new theorem, which we would disclose together with the system that produced it. In the present release the route milestone is configured only where the ledger's routes have been registered as positive (the five Mathieu tasks and the four

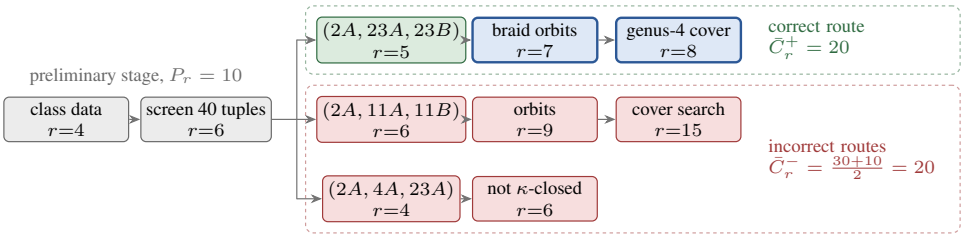


Figure 6: **The two stopping metrics on a toy artifact graph.** Boxes are finalized artifacts with their cost $r(v)$, and blue boxes with bold borders are verified checkpoints. Grey is preliminary screening ($P_r = 10$), green the route the ledger classifies as positive, red the routes it classifies as non-positive. $(2A, 4A, 23A)$ is dropped because its class vector is not closed under complex conjugation κ, which swaps $23A$ and $23B$. Credited artifacts and their ancestors form U, here the grey boxes and the green route, so $U_r = 100 \cdot 30/70 \approx 43$: less than half the budget supported verified credit. The stopping efficiency compares screening plus an average correct route with the same plus an average incorrect route, $V_r = 100 \cdot (10 + 20)/(10 + 20 + 20) = 60$: an average wrong route cost two thirds as much as everything that led somewhere. $V_r = 100$ would mean no wrong route was researched at all.

other bespoke-ladder tasks) and, for the Tier 3 tasks, through the open-case profile of Appendix I.3, which awards K for an exact obstruction of a candidate passport or a verified cover on one. The remaining tasks have a maximum score of 70 until that registration is complete.

Route classification and undefined cases. Literature-based routes are classified as correct by structural verification against a registered ledger case. New routes require independent verification. Let $n_\pm$ and $C_r^\pm$ be the number and total post-screening cost of correct and incorrect researched routes. Then $\bar{C}_r^\pm = C_r^\pm / n_\pm$ if $n_\pm > 0$ and 0 otherwise. If only incorrect routes are researched, $V_r = 100\, P_r/(P_r + \bar{C}_r^-)$. V_r is reported as N/A when no route enters the research stage or the denominator is zero. A tuple screened only during preliminary work contributes to P_r but to neither route count, and preliminary and research costs are never double counted. Where the evaluator did not resolve every route verdict, V_r is a retrospective classification in which a route counts as positive only if its class vector is in the literature census. The retrospective values are listed per run in the released data. Appendix I states which value each run uses. Unattributed cost is excluded from the denominator of U_r, so a run that leaves most of its work outside artifacts is scored on the part it attributed.

G Experimental Setup Details

Sandbox. Each run executes in an evaluator-owned workspace under Bubblewrap filesystem isolation and a seccomp network policy that blocks all agent network access except Magma's license check. The workspace provides Python 3.12 (SymPy, mpmath), GAP 4.15.1 with `ctbllib`, `atlasrep`, `tomlib`, and `mapclass`, Magma V2.29-9, and ripgrep. Jobs run one problem per Slurm array task with 8 cores on an institutional HPC cluster. Three limits apply on different sides of the protocol: the agent's shell commands are subject to a per-command timeout (7,200 s in the main campaign, while many coded runs, including Opus and campaign-02 runs, record an 1,800 s tool limit), the verifiers' Magma calls to 21,600 s, and braid-orbit and Nielsen enumerations to $2{\times}10^6$ classes.

Verification. A call over a cap returns `unsupported`. The description-fidelity judge of Appendix F is Claude Sonnet 5 invoked through the Claude Code CLI with tools disabled.

Reporting. Campaign samples are drawn deterministically from the catalog by tier and seed, and every run of a campaign shares one configuration file.

Table 4: Configuration of the eligible runs per system. Every run is a main-campaign run with 10 hours, a 500k output-token cap, eight CPUs, and task-dependent memory (64–192 GB), at most two runs per task. The earlier 6-hour repetitions of GPT-5.6 Sol and Terra and the Claude Haiku campaign are excluded (Appendix I).

System	Model id	Harness	Effort	Runs per task	n
Claude Opus 5	claude-opus-5	Claude native (Claude Code)	xhigh	2	10
GPT-6 Sol	gpt-6-sol	Codex native	xhigh	2	10
GPT-6 Astra	gpt-6-astra	Codex native	xhigh	2	10
GPT-5.6 Sol	gpt-5.6-sol	Codex native	xhigh	2	10
GPT-5.6 Terra	gpt-5.6-terra	Codex native	xhigh	2	10
Claude Sonnet 5	claude-sonnet-5	Claude native (Claude Code)	xhigh	2	10
DeepSeek V4 Pro	deepseek-v4-pro-0813	OpenRouter adapter	xhigh	2	10
GLM-5.3 Flash	z-ai/glm-5.3-flash	OpenRouter adapter	xhigh	2	10

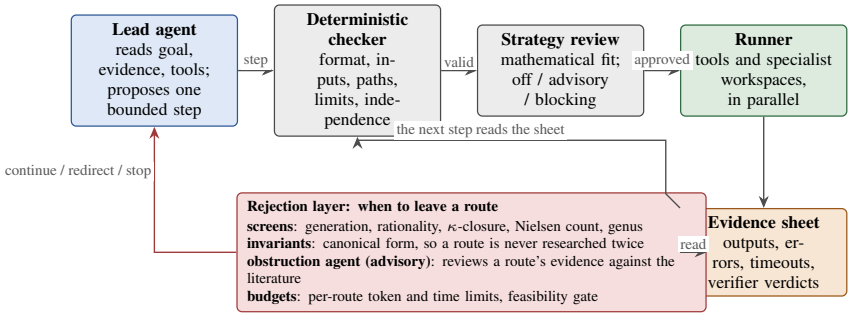


Figure 7: **The GALOIS control loop.** The lead agent proposes one bounded step, a deterministic checker validates it, an optional strategy reviewer judges its mathematical fit, and a runner executes it through registered tools or specialist workspaces. Everything that happens, including timeouts and errors, is written to the evidence sheet that the lead reads at the next step. The rejection layer turns evidence into a stop signal: cheap screens reject candidates before any expensive computation, normalization invariants stop the same route from being researched twice, an obstruction agent reviews a route against the literature and returns advice (only exact certificates may reject, Table 6), and per-route budgets end a route whose projected cost exceeds what is left.

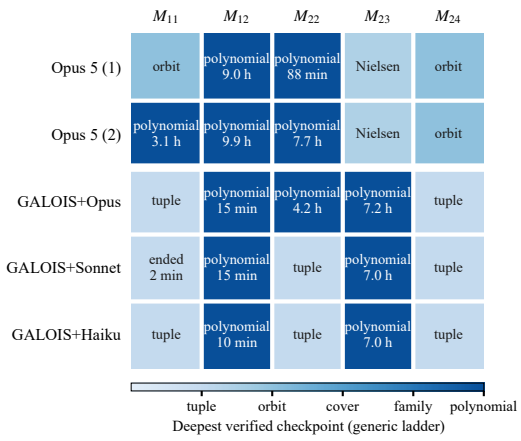


Figure 8: **Deepest verified checkpoint per task, adapter runs**: the two single-agent Claude Opus 5 runs and one first-batch GALOIS adapter run per task with each of three backends. Polynomial cells give the time to verifier acceptance. The GALOIS runs ended on the no-progress guard rather than on the 10-hour budget, and Sonnet's M_{11} run ended after two minutes.

H GALOIS DETAILS

Score timelines on IGP-Bench. Figure 9 shows the progress score over time for the M_{22} and M_{23} runs of Figure 8, from fifteen-minute captures of the GALOIS runs and the score timelines of the two single-agent Opus 5 runs. On M_{23} the single agents plateau at the key case ($S = 65$) while every GALOIS backend reaches the terminal polynomial after about seven hours. The score gap in the other direction is the missing route-milestone credit discussed in Section 4.

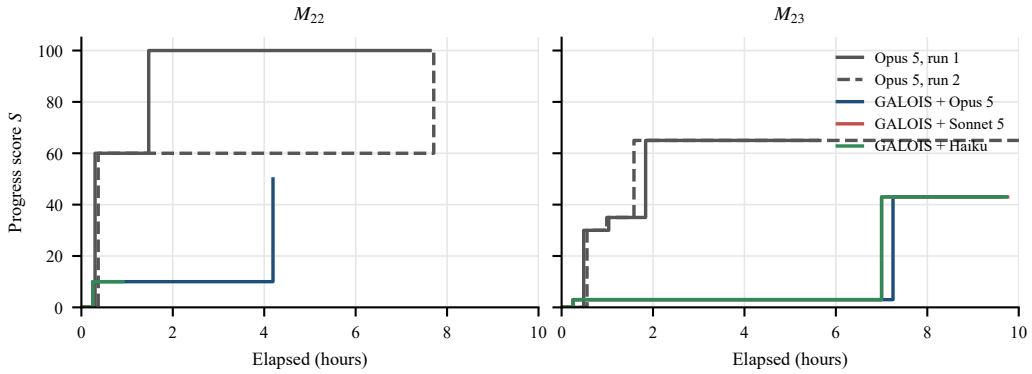


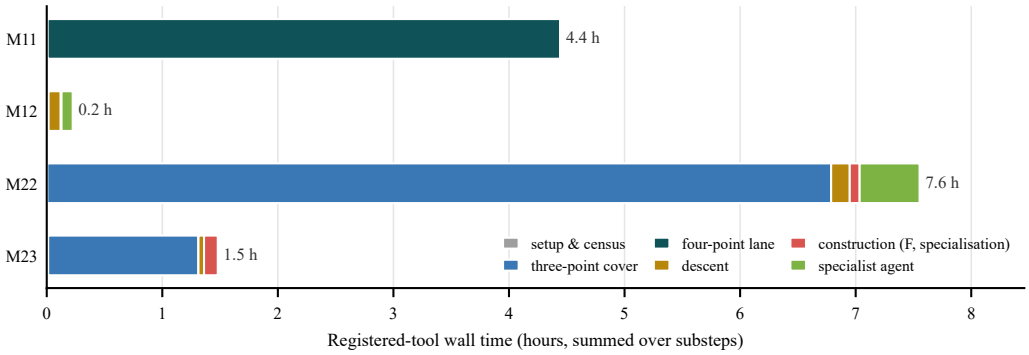
Figure 9: **Progress score over time on M_{22} and M_{23}** for the GALOIS runs of Figure 8 and the two single-agent Opus 5 runs.

Reference run. Separately from the benchmark runs, the pipeline was run end to end on M_{11}, M_{12}, M_{22}, and M_{23} with Claude Opus 5 at medium reasoning effort and without the benchmark bridge. It produced explicit polynomials for all four groups: M_{11} on the four-point route $(2A, 2A, 3A, 4A)$ in 46 steps and 4.7 hours, M_{12} on $(3A, 3A, 6A)$ in 12 steps and 13 minutes, M_{23} on $(2A, 23A, 23B)$ in 16 steps and 1.1 hours, and M_{22} on $(2A, 11A, 11B)$ in 15 steps and 7.6 hours. The M_{22} polynomial was verified afterwards by the IGP-Bench terminal-polynomial verifier (Magma GaloisGroup and GaloisProof) because the pipeline's own final acceptance was interrupted. Reported model cost was 14 to 46 USD per group, dominated by lead-agent calls (Figure 10, top). Compute was dominated by single Magma certificates, with about seven of M_{22}'s 7.6 tool-hours spent in the three-point-cover stage, almost all of it on one monodromy certificate (Figure 10, bottom). These runs are not budget-matched to the benchmark campaign and are reported for cost and route information only. The description in Section 4 follows the inspected revision of the GALOIS repository. The engine is deterministic scripting except for three agent roles (lead, strategy checker, specialist) and two bounded review agents. State, plan, and journal each have a single writer, and every mutation is atomic under a lock. Table 5 lists the stopping rules with their defaults, Table 6 separates what may reject a route from what may only advise, and Table 7 collects the development runs cited in the main text.

Skills by stage. Table 8 lists the 36 skills of the run commit with the pipeline stage each serves. Every skill is a written procedure for one stage of Appendix B, addressed to an arbitrary registered group with trivial centre and an arbitrary class vector: the procedures and the twelve worked examples bundled with them are written symbolically or on self-test instances (A_5, S_3, synthetic passports), and the GAP tools take the group from the engine's registry, which at the run commit held 27 groups, 22 of them with a pinned permutation representation (the five Mathieu groups, J_1, J_2, HS, Co_3, $\mathrm{Sz}(8)$, $\mathrm{PSU}_3(3)$, $\mathrm{PSU}_4(3)$, $\mathrm{PSU}_5(2)$, and $\mathrm{PSL}_2(p)$ for nine primes). The tool registry holds 38 three-point, 72 four-point, and 12 legacy tools.



(a) Reported model cost by agent role.



(b) Registered-tool wall time by pipeline stage.

Figure 10: **Cost profile of the GALOIS reference run** (Claude Opus 5, medium effort).

Table 5: Stopping and deferral rules in GALOIS. Defaults from `engine/lock.py` and `config.json`, with the source module of each rule.

Rule	Trigger and consequence	Module
First-success stop	an accepted polynomial with a certified specialization ends the run; nothing is spent on further tuples	`success.py`
No-progress guard	a step counts as progress only if it ran a tool on a new candidate key or a specialist wrote an artifact; a strictly raised precision counts at most three times; after k no-progress steps ($k = 2$ default, 8 in recent campaigns) the run halts with a boundary report	`driver.py`
Spend cap	\$30 per run by default, checked before every lead spawn; the obstruction agent starts nothing within \$1 of the cap	`driver.py`
Circuit breaker	three consecutive failed steps hand the streak to the lead once, a second trip halts	`driver.py`
Done gate	a *done* decision is refused while a long tool lease or an unresolved wall-clock stop exists; after two refusals the run halts rather than record an unsupported conclusion	`driver.py`
Goal-reached and negative stops	a *continue* after a certified realization, or on a single candidate with a validated terminal negative, is refused; twice, then halt	`driver.py`
Step and attempt caps	40 steps per run; 6 lead attempts per step; 2 strategy REVISEs per step, then auto-pass	`lock.py`
Typed precision ladder	on an *insufficient digits* failure, rerun once at the computed requirement plus a 24-digit margin (cap 120); a second unmotivated raise is refused and the cover is deferred with a revisit condition	`scheduler.py`
Evidence ledger	a numerical cover is classified promising, precision-limited, likely obstructed, stalled, or ambiguous; dispositions reorder work and reject nothing	`evidence_ledger.py`
Route transition	when every retained cover of a source is rejected or deferred, the source is retired and the queue advances (next source, then the four-point rows, then the next tier)	`scheduler.py`
Per-tool limits	registry timeouts from 1 minute (candidate review) to 12 hours (numerical covers); a cap hit is a result, not an error, and never a negative	`tools.py`
Four-point feasibility	a 30-second calibration declares a seed search infeasible when projected coverage is below 10^{-3} of the search space	`four_point_seed_search.py`
Taint guard	a consumer of a blocked or incomplete upstream artifact is refused, which makes the dead end explicit	`acceptance.py`

Table 6: Rejection authority in GALOIS. Only the first group may retire a cover or a passport (the branch data of a cover: its degree and the classes of its branch cycles); everything else returns `may_reject = false`.

May reject	exact field-of-moduli obstruction in Magma (invariants of a bound genus-0 candidate, symmetrized over branch permutations; verdict *not defined over* $\mathbb{Q}$ on a selected cover); audited degree-11 passport template; elementary passport necessities replayed in GAP (Riemann–Hurwitz, dihedral bound, branch-cycle lemma); verified descent or Stage-11 regularity negatives replayed independently
May only advise	entry screens and route ranking (Nielsen count, genus, real-pairing heuristics); cheap precheck; the 14-method obstruction package (genus-one and hyperelliptic moduli, ramification-divisor invariants, field-of-moduli degree, normalization-invariant search, and others); the parallel obstruction agent (at most two 15-minute assessments per candidate, one 5-minute experiment, eight 10-minute research rounds); the candidate reviewer (records repair, obstruction experiment, defer, reject-with-validation, escalate precision, switch lane, or continue); the strategy checker

Table 7: Development runs recorded in the GALOIS repository. These are not IGP-Bench runs: budgets are in dollars and wall time, the M_{23} route is in the system's knowledge base, and the numbers are quoted from the run reports rather than recomputed.

Run	Outcome	Source
M_{24}, Opus lead, run 2 (2026-09-23)	halted by the no-progress guard after two unproductive lead steps; 14 steps, 52 min, $4.25	`docs/2026-09-23-m24-opus-medium-hpc-run.md`
M_{24}, Opus lead, run 3	guard raised to 8; three-point source deferred under the budget rule at step 17; four-point route $(2A, 2A, 2A, 12B)$ certified; 31 steps, 84 min, $11.45	same
M_{24}, run 4	$(3A, 3A, 12B)$ rejected in-run on all four covers by an exact field-of-moduli certificate; 36 steps, 80 min, $8.75	`docs/2026-09-23-m24-3A3A12B-rejection-certificate.md`
Six groups (2026-09-19)	150 steps, 93 lead calls, $220; the lead copied the engine's proposal in 20 of 61 cases and never scheduled a four-point candidate; only M_{23} completed	`docs/2026-09-19-lead-follows-engine-plan.md`

Table 8: The 36 skills of GALOIS at the run commit, by pipeline stage (Appendix B). Each is a written procedure for one stage, addressed to any registered group and class vector. Stage numbers with (4) belong to the four-point lane.

Stage	Skill	Procedure
Group data and census		
1	`group-setup`	Verify the group's pinned permutation representation from the engine registry, its order and class count against the ATLAS table, and emit the labelled class table with power maps and rationality flags.
1	`group-setup-research`	Register a group the ATLAS libraries do not know (names that AtlasRep or ctbllib know are registered without an agent).
1	`class-vector-census`	Enumerate every unordered multiset of three and of four nontrivial classes with the exact product-one count by the Frobenius character formula, the genus, and a feasibility flag.
1	`dossier-extension`	Group dossier: the three-point census with exact Nielsen counts and the admissible $\{M, M, N, N\}$ four-point vectors consumed by Stages 2 to 6.
1	`real-pairing-filter`	Necessary-condition filters on class vectors (real-pairing conditions, genus cap) and a plausibility ranking; advisory, never a route certificate.
1	`candidate-ranking`	Exploratory ranking of census rows by raw Nielsen mass and source genus; advisory input to the route portfolio.
Orbits and Nielsen classes		
2–4	`orbit-decomposition`	Decompose the ordered product-one fibre of a four-class vector into pure-braid orbits with witnesses (GAP job).
2–4	`genus-screen`	Independently re-derive the λ-line genus of every pure-braid orbit component from its degree and cycle counts.
2–4	`targeted-orbit-closure`	Close seed tuples under targeted pure-braid moves, compute λ-line genus and cusps, and recognise Häfner forms.
2–4	`hafner-bridge`	Häfner's bridge from three-point triples to length-two genus-zero four-point pure orbits (optional).
2–4	`three-point-orbits`	Exact braid-orbit inventory of a three-point class vector on generating product-one tuples (optional).
5–6	`nielsen-representatives`	$\mathbb{Q}$-rational branch configuration with the equivariance check, exact Nielsen count, one certified representative per orbit, and the action of complex conjugation with the real orbits.
6	`cover-galois-orbits`	Galois-orbit partition of a cover set by simultaneous lattice recognition, the separating invariant of Stage 6.
Numerical cover, exact curve, covering map		
7	`belyi-numerical-cover`	Numerical differentials and triangle data of one cover in the Belyi-package disc coordinate.
7	`belyimap-frame-transform`	Coordinate-transform certificate from the package uniformiser to the working frame, with the echelon and Weierstrass record.
8	`curve-reconstruction`	Exact curve model over a number field from the accepted coordinate series, with a case-by-case escalation route.
7–9	`genus-zero-exact-reconstruction`	Exact-first lane for genus-zero covers: the rational function whose factorisation the three ramification partitions prescribe.

continued on next page

Table 8 continued

Stage	Skill	Procedure
7–9	`certified-branch-cycles`	Certify the monodromy permutations of an exact genus-zero map by ball arithmetic, tying it to the selected Nielsen representative.
9	`covering-function`	The exact covering map $t\colon X \to \mathbb{P}^1$ from the exact curve and its distinguished points (Magma).
Descent, family, certificates		
10	`descent-and-auxiliary-function`	Descent datum of the pair (X, t) to $\mathbb{Q}$, the fixed field, the auxiliary function s with $\mathbb{Q}(X) = \mathbb{Q}(t, s)$, and the primitivity certificate.
11	`polynomial-construction`	Build $F(T, V)$ from the $\mathbb{Q}$-model bundle: solve, CRT, reduce, verify, certify.
11	`monodromy-certificate`	The regularity certificate, items 1 to 7, for an accepted $F(T, V)$.
12	`specialization-certificate`	Certified specialisation: sweep a height box, choose t_0, prove irreducibility and the Galois group, reduce, and check field equivalence and ramified primes.
Four-point lane		
7(4)–12(4)	`four-point-construction`	Lane index for the four-point arithmetic lane and the legacy bounded exact rational-function construction.
7(4)	`four-point-configuration`	Bind a selected four-class vector to its construction problem: action, genus, normaliser and forced-rational-place checks, exact Nielsen recount, chamber counts, chart recommendations.
8(4)	`four-point-ramification-seed`	Compile a genus-zero ramification chart and find verified finite-field seeds with the target monodromy group.
8(4)	`four-point-local-recovery`	Hensel-lift a verified seed, search residue discs, reconstruct over $\mathbb{Q}$ or a small number field with held-out digits, and verify exactly.
8(4)	`four-point-family-reconstruction`	The p-adic one-parameter family through a verified seed, bidegree-bounded relations, Magma analysis of the relation curve, and exact evaluation at a rational point.
9(4)	`four-point-exact-export`	Export an exact four-point model as polynomial, model, and branch artifacts for the verification stages.
Obstructions (rejection evidence)		
any	`inverse-galois-obstructions`	Select and run bounded exact obstruction experiments: class compatibility, real structures, intrinsic four-point invariants, local points.
any	`exchange-orbit-descent-obstruction`	Criterion on the $G_{\mathbb{Q}}$-action on the exchange orbits of a three-point passport, read through an intrinsic invariant of a bound exact cover.
any	`three-point-passport-obstruction`	Low-degree intrinsic invariants for unresolved three-point passports, separating marked real obstructions from branch-symmetric passport obstructions.
any	`genus-one-moduli`	Exact j-invariant diagnostics of a genus-one source curve.
any	`hyperelliptic-moduli`	Exact Igusa or Shioda invariants of genus-2 and genus-3 hyperelliptic source curves for nonrational-moduli obstructions.
any	`ramification-divisor-invariants`	Exact projective invariants of ramification-point blocks on a genus-zero source, for field-of-moduli obstructions without full cross-ratio traces.
any	`invariant-relation-certification`	Turn a conjectured relation for a geometric invariant into an exact open-chart identity.

Task-specific content in the runtime checkout. Each run starts from a checkout built by `scripts/make_blind_snapshot.sh`: the engine, schemas, templates, knowledge files, skills without their self-test folders, tools, the scripts that runtime code names, and one obstruction reference, with no git history, documentation, reports, tests, or earlier runs. We searched this file set at the run commit for the class vectors of the five tasks. The eight knowledge files name no Mathieu group, and the per-group case cards of the development runs, which record each task's certified route and queue order, were moved out of the knowledge directory on 25 September 2026 and are not in the checkout. The routes do appear elsewhere. $(2A, 23A, 23B)$ is in 23 files, as the example string for the class-vector tag in the help text of scripts and templates of the Stage 5 to 12 skills (19 files) and in development notes in four engine modules, and $(2A, 11A, 11B)$ in nine files of the same kind. The finished routes of M_{11}, M_{12}, and M_{24} each appear in one to three engine docstrings as the cases on which a tool was measured. The docstring of the exact field-of-moduli tool names the M_{11} three-point route it decides, and one obstruction script is specialised to the ramification profile of the M_{24} three-point route. No file marks any vector as positive, rejected, or preferred, and the lead's queue is computed by the census and ranking tools at run time. A lead with file access could nevertheless have read these names. The read records of the runs in Table 9 have not been checked against this list.

Run configuration. The runs of Section 4 use Claude Opus 5.5 at medium effort through the Claude CLI as lead, the strategy checker off, the obstruction agent on, a stop at the first certified specialization, and no spend cap. Each run starts from a fresh checkout holding only the runtime files, and heavy tools run on the same 16-core, 110 GB workstation. The skills encode the mathematical methods of the constructive literature, from the census and rigidity screens through

numerical uniformization, recognition, and descent to certification. That knowledge is part of the system, and these runs serve as reference runs that set a time scale for the tasks rather than as a like-for-like comparison with the single agents. An audit of every tool call of every agent session (`scripts/audit_lead_reads.py`) found no read outside the checkout in any run of Table 9.

Table 9: GALOIS runs on the five tasks. The first five were launched together at 03:26 UTC on 26 September 2026 on one commit (88d718b); the last five are the single runs on the commits that fixed the routing defects found in the development rounds (Table 10). "Rejected" gives the three-point source retired by the exact field-of-moduli test (*not defined over* $\mathbb{Q}$ on every retained cover) before the run moved to the four-point row it finished on. Steps are executed pipeline steps, Lead and Obstr. the lead-agent and obstruction-agent calls, Min the wall-clock minutes from run creation to the final decision, USD the model cost reported by the CLI, Tokens the output tokens of all agent calls, and V_r the token stopping efficiency over pipeline steps (scoring paragraph below). Magma recomputed every polynomial's Galois group (`data/galois_verification.txt`).

Run	Rejected	Finished route	Endpoint	Steps	Lead	Obstr.	Min	USD	Tokens	V_r
M_{11} k	$(4A, 4A, 5A)$ c2, c3	$(2A, 2A, 3A, 4A)$	polynomial	46	27	10	78	26.40	128k	73.5
M_{12} k	none	$(3A, 3A, 6A)$ c1	polynomial	13	6	6	13	6.23	38k	100
M_{22} k	none	$(2A, 11A, 11B)$ c2	$F(T, V)$	22	13	5	126	24.64	252k	100
M_{23} k	none	$(2A, 23A, 23B)$ c3	polynomial	16	12	3	43	7.97	38k	100
M_{24} k	$(3A, 3A, 12B)$ c1–c4	$(2A, 2A, 2A, 12B)$	polynomial	37	16	8	79	15.34	88k	35.1
M_{11} g	$(4A, 4A, 5A)$ c2, c3	$(2A, 2A, 3A, 4A)$	polynomial	45	24	10	78	22.33	119k	73.0
M_{12} h	none	$(3A, 3A, 6A)$ c1	polynomial	13	9	5	10	7.27	38k	100
M_{22} j	none	$(2A, 11A, 11B)$ c2	$F(T, V)$	25	12	8	59	14.51	114k	100
M_{23} h	none	$(2A, 23A, 23B)$ c3	polynomial	17	9	4	41	7.27	41k	100
M_{24} f	$(3A, 3A, 12B)$ c1–c4	$(2A, 2A, 2A, 12B)$	polynomial	31	11	11	43	12.69	74k	46.7

Table 10: The development record: the 17 recorded runs before the batch of Table 9, from the repository's run summaries. Round C ran the five tasks in parallel under a 30 USD cap with heavy tools on a cluster; the later rounds ran locally without a cap.

Round	**What happened**	**Fix or lesson**
A (1 run, M_{24})	$(3A, 3A, 12B)$ rejected on all four covers, $(2A, 2A, 2A, 12B)$ certified; 31 steps, 2 h 28 min, \$12.51	first success
C (5 runs, parallel)	M_{12} and M_{23} certified, M_{22} to $F(T, V)$; M_{11} and M_{24} halted at the cap after correct rejections of $(4A, 4A, 5A)$, $(4A, 4A, 6A)$, and $(3A, 3A, 12B)$	a tool reported a required precision proportional to the digits supplied, so the precision ladder climbed without end on genus-one sources; the engine dropped a rejected source's four-point child rows from the queue
D, E (4 runs, M_{11}, M_{24})	cancelled or halted by the no-progress guard while the routing fixes were completed	four-point tools listed whenever the queue holds a four-point row; child rows kept after a source is rejected
F, G (2 runs)	M_{24} in 43 min and M_{11} in 78 min certified (Table 9)	playbook rule to keep the first relation-pair batch in its required order
H, I, J (5 runs)	M_{12} and M_{23} certified; three M_{22} runs testing versions of a rule for a verified curve whose map is not yet found, of which only the last reached $F(T, V)$	compute the map from its exact divisor rather than fitting it numerically

Scoring the GALOIS runs. These runs have no artifact graph and no evaluator record, so the GALOIS row of Table 1 is scored from the run records by `scripts/collect_galois_runs.py` (output in `data/galois_runs.csv`), with the paper's definitions applied to pipeline steps instead of artifacts. A step belongs to the class vector its plan entry names, and steps 1 to 3 (group setup, exact Nielsen census, route portfolio) are the preliminary screening. The finished route of every run is on the ledger's positive list ($(2A, 2A, 3A, 4A)$, $(3A, 3A, 6A)$, $(2A, 11A, 11B)$,

$(2A, 23A, 23B)$, $(2A, 2A, 2A, 12B)$), and the two rejected sources are not. C is 40 for every run, since every polynomial is Magma-verified (`data/galois_verification.txt`). R and K are assigned in full: the exact census, Nielsen representatives, and orbit data the runs compute exceed the research cap, and each run carries a verified generating tuple on its positive route. U_r and V_r are over the output tokens of the agent calls attributed to each step, with U_r taken as the share in preliminary steps and steps of the finished route, an upper bound on the artifact definition, which with at most one non-positive route per run coincides with V_r. Format compliance, description fidelity, tool-call validity, and termination are not reported for these runs. Under these definitions the ten runs average $S = 100$, POLY 100, and $U_r = V_r = 82.8$. The scores are not directly comparable with the evaluator's: the runs had no token cap, a different lead model and effort, and no bridge, and the route classification was done by us from the plan entries.

I Experimental Results

I.1 Run eligibility, provenance, and known protocol issues

The analysis snapshot combines the most recent eligible run summaries in the shared campaign archive: 10 Opus, 10 GPT-6 Sol, 10 Astra, 10 GPT-5.6 Sol, 10 Terra, 10 Sonnet, 10 DeepSeek, and 10 GLM task-runs (80 in total), two runs per task. We use only Opus at xhigh effort, omit the post-hoc Astra M_{11} replacement (a replicate chosen after seeing that the preregistered run scored 4, whose substitution would raise Astra's mean S from 59.4 to 69.0), and omit the earlier 6-hour repetitions of GPT-5.6 Sol and Terra so that every system has two runs per task. Claude Haiku is left out of the aggregate: its only valid runs are five from the earlier 6-hour campaign (its 10-hour batch ran without Magma on the agent path), too few for a comparable row, and its verified progress was near zero. GLM was the newest complete upload at the analysis cutoff. GPT-6 Sol's runs come from a two-wave campaign whose upload holds only 15-minute captures, so its run records were copied from the cluster. That campaign ran without the description-fidelity judge, so its η comes from re-running the pinned judge on the archived records and its I_f is recomputed from the runs' format logs. The same procedure supplies η for the DeepSeek and GLM runs, whose judge call had failed on the cluster, and for the campaign-02 M_{23} runs of GPT-5.6 Sol and Terra, where the judge had been switched off, and I_f for the ten campaign-02 GPT-5.6 runs, whose campaign summary carried no format fields. Five GLM runs finalized no record with a determinate judgment and have no η. All runs are scored under the literature-positive route rule of Section 3.4: the route milestone and its research top-up (the research points attached to the milestone) count only ledger routes in the target group whose class vector the evaluator lists as positive. Runs scored at an earlier benchmark revision could also match routes in an almost-simple overgroup. We re-scored every such run whose milestone had been earned on a non-positive route by replaying its archived record through the current evaluator. Two runs changed: the Sonnet M_{22} batch-2 run (60 to 0, since its route in $\mathrm{Aut}(M_{22})$ is no longer eligible) and the GPT-5.6 Sol campaign-02 M_{22} run (100 to 46, since its $\mathrm{Aut}(M_{22})$ route $(2B, 5A, 12A)$ loses the milestone and top-up while its Magma-verified terminal polynomial keeps the rule-independent continuation credit). Every other earlier milestone also matched a positive route, or is the M_{23} key case, and is unchanged.

Provenance of terminal polynomials is audited. Every certified Opus polynomial (five in the eligible set) was built by the run itself. The Opus run records carry the flag `no-known-answer-stores-v1`, meaning Magma's polynomial database, stored number fields, and stored Belyi maps were masked, and no other system's records carry this flag. Astra's M_{12} and M_{22} polynomials are constructions on routes of its own choice (Gröbner or finite-field search with p-adic lifting, then GaloisProof). GPT-5.6 Sol's one polynomial is the campaign-02 M_{22} construction on the $\mathrm{Aut}(M_{22})$ route. Two verifier issues were raised with the maintainers: a non-monic integral multiple of a correct M_{11} polynomial was rejected for Sol (campaign-02) and for Astra (batch 1), so both lost terminal credit on runs whose own GaloisProof had passed, and Opus's second M_{11} run, which realized M_{11} through an M_{12} route, earned no route credit because that route has no interface for a route recorded in an overgroup. The V_r value of a run is the evaluator's wherever one was preserved, in the per-run summary or in the census sheet (all Astra runs, eight DeepSeek runs, six GLM runs, one Opus run, 25 runs in all), and the census value otherwise (51 runs). Four GLM runs recorded no route research and have no V_r.

I.2 CREDIT LEVELS AND SCORES BY TASK

Figure 3 in the main text shows the credit levels each system's runs reach and progress against stopping efficiency, Figure 11 its per-task S and U_r, and Table 17 gives the numbers.

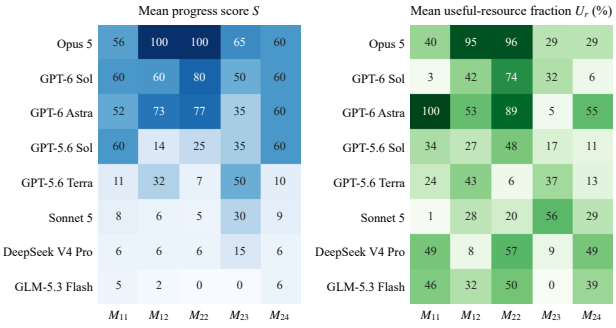


Figure 11: **Progress and useful fraction by task.** Mean progress S and mean useful-resource fraction U_r (tokens) per system and task. M_{23} and M_{24} absorb large budgets from every system without yielding a construction.

I.3 TIER 3 TASKS UNDER THE OPEN-CASE PROFILE

Tier 3 tasks have no literature-positive route, so the route milestone of Section 3.4 is undefined for them. The *open-case* profile keeps the $30/30/40$ layout and the generic continuation ladder and replaces K by a *passport resolution*. A *candidate passport* of G is an unordered class vector $C = \{C_1, C_2, C_3\}$ that is Galois-stable (closed under $C \mapsto C^k$ for k coprime to $\operatorname{lcm} \operatorname{ord} C_i$), contains at least $|G|/|Z(G)|$ product-one triples (one free conjugation orbit of a generating triple), and has Riemann–Hurwitz genus at most $g_{\max}$ in the registered action; $g_{\max} = 2$ by default, raised to the smallest candidate genus when no candidate of genus ≤ 2 exists (Sz(8): 4; $\mathrm{PSL}_2(27)$: 9). Every three-point regular realization of G over $\mathbb{Q}(T)$ of source genus at most $g_{\max}$ has its branch-cycle description in a candidate passport. $K = 30$ is awarded once, for the first artifact that *resolves* a candidate: (i) an obstruction recomputed by the evaluator in GAP, that no inner Nielsen class of the passport is fixed by any of the four real structures of a conjugation-stable configuration of three branch points (fact N7: no cover in the passport is defined over $\mathbb{R}$, hence none over $\mathbb{Q}$), or that every product-one triple generates a proper subgroup (fact N4); or (ii) a verified cover over $\mathbb{Q}$ with the passport (checkpoint S3 or higher). A resolution completes R to 30, as the route milestone does on Tier 1 and Tier 2 tasks; R additionally credits a correct passport census (C1) and correct counts of inner Nielsen classes and real classes (C2). Numerical results, including unrecognized invariants of numerically computed covers, earn nothing.

Figure 12 and Table 11 report one run of Claude Opus 5 (maximum reasoning effort, 500,000 output tokens, eight CPUs) per Tier 3 task. The runs were launched in parallel with a five-hour wall budget; a harness restart cost the first 2 h 12 min and the agents were stopped gracefully at the original deadline, so the effective agent time is 2 h 48 min per task, and finalized artifacts were scored as after a wall-time stop. Six runs resolved a candidate by obstruction within the first hour and a half; all ten reached the braid-orbit checkpoint and none claimed a cover. A literature audit of 2026-09-25 (Zywina's list of small simple groups) reclassifies $\mathrm{PSU}_3(5)$, $\mathrm{PSL}_3(7)$, $\mathrm{PSL}_4(3)$, $\mathrm{PSL}_2(27)$, and $\mathrm{PSL}_2(125)$ as realized over $\mathbb{Q}$ by non-regular constructions without a published polynomial and $\mathrm{PSL}_2(16)$ as having a published polynomial; the certified-open groups among the ten are Sz(8), $\mathrm{PSU}_3(4)$, $\mathrm{PSU}_3(8)$, and Sz(32). The open-case scores do not depend on the tier label. For comparison, three exploratory Claude Opus 5 runs on further open groups (September 2026, uneven budgets, stopped by hand) score 15 ($\mathrm{PSU}_3(4)$), 12 ($\mathrm{PSL}_3(8)$), and 12 ($\mathrm{PSU}_3(8)$) under the same profile: they computed censuses and Nielsen classes but resolved no candidate passport, and re-verifying the $\mathrm{PSL}_3(8)$ run showed that its passport $(3A, 3A, 4A)$ has 24 real inner Nielsen classes, twice the 12 the pipeline analyzed, because the pipeline applies only the all-real structure when every class is rational.

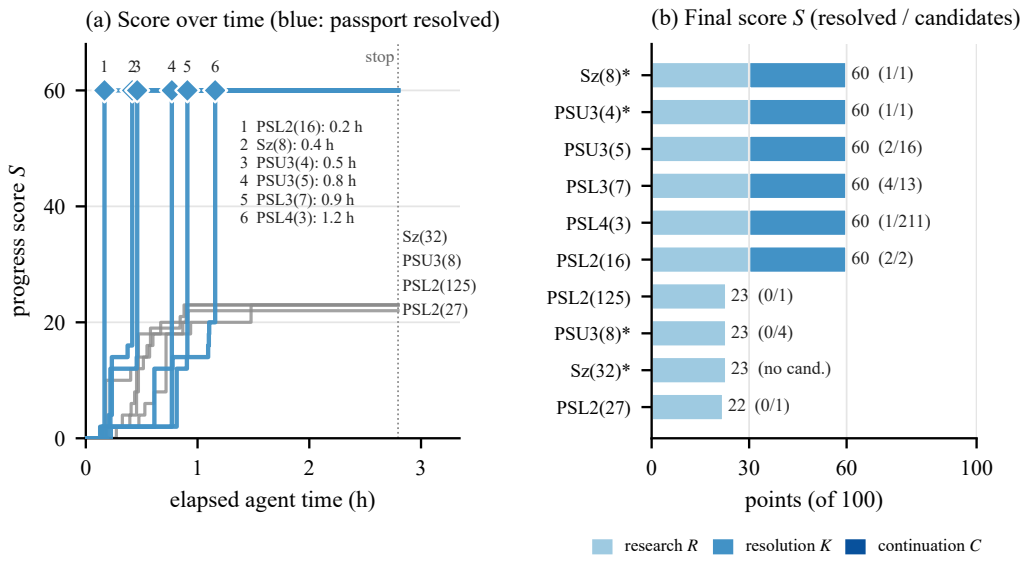


Figure 12: **Ten Tier 3 runs of Claude Opus 5 under the open-case profile.** (a) Progress score S over elapsed agent time, one step line per task, recorded by the harness at every artifact event; diamonds mark the artifact that resolved a candidate passport (blue lines), grey lines are the four runs that resolved none. All agents were stopped at 2 h 48 min. (b) Final score per task as research R, resolution K, and continuation C; the number at the bar end is S and, in parentheses, resolved over candidate passports of genus at most $g_{\max}$. Groups marked $*$ are on Zywina's list of open cases; the others were reclassified as realized by the 2026-09-25 audit. No run claimed a cover, so $C = 0$ throughout.

Table 11: **Tier 3 tasks under the open-case profile.** One run of Claude Opus 5 (maximum effort) per task, 2 h 48 min effective agent time, scored at the graceful stop. R, K, C, and S are raw points (maxima 30, 30, 40, 100); *level* is the deepest verified checkpoint; *cand.* is the number of candidate passports of genus at most $g_{\max}$ in the evaluator's census and *res.* how many the run resolved; *resolution* names the passport, its genus, and the obstruction fact; *tok.* is output tokens in thousands. Class names follow the GAP character-table library.

Group	status†	$g_{\max}$	cand.	res.	R	K	C	S	level	resolution (passport, genus, fact); tok.
Sz(8)	open	4	1	1	30	30	0	60	S2	$(2a, 5a, 5a)$, 4, N7; 371k
$PSU_3(4)$	open	2	1	1	30	30	0	60	S2	$(3a, 3a, 4a)$, 2, N7; 348k
$PSU_3(5)$	realized	2	16	2	30	30	0	60	S2	$(2a, 5a, 6a)$, 0, N4; $(2a, 4a, 10a)$, 0, N7; 373k
$PSL_3(7)$	realized	2	13	4	30	30	0	60	S2	$(2a, 4a, 7b)$, 0, N7; $(2a, 4a, 14a)$, 0, N7; 402k
$PSL_4(3)$	realized	2	211	1	30	30	0	60	S2	$(3a, 5a, 8a)$, 2, N7; 363k
$PSL_2(16)$	polynomial	2	2	2	30	30	0	60	S2	$(2a, 5a, 5b)$, 0, N4; $(3a, 5a, 5b)$, 1, N4; 288k
$PSL_2(125)$	realized	2	1	0	23	0	0	23	S2	none; N7 only on $(7a, 7b, 7c)$ of genus 37; 344k
$PSU_3(8)$	open	2	4	0	23	0	0	23	S2	none; N7 only on passports of genus 22 and 64; 467k
Sz(32)	open	2	0	0	23	0	0	23	S2	none; no three-point candidate of genus ≤ 12; 290k
$PSL_2(27)$	realized	9	1	0	22	0	0	22	S2	none; 441k
mean					27.1	18.0	0.0	45.1		

†Catalog tier label re-audited on 2026-09-25: *open* = on Zywina's list of unrealized small simple groups; *realized* = proved to occur over $\mathbb{Q}$ by a non-regular construction, no published polynomial; *polynomial* = an explicit polynomial is published.

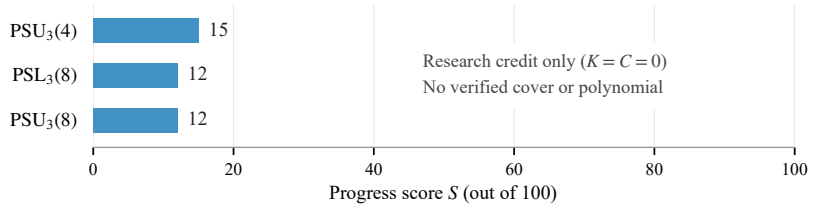


Figure 13: **Preliminary research on three further open groups**: exploratory Claude Opus 5 runs under the open-case profile, one run per group with uneven budgets. All credit is for research ($K = C = 0$): even the strongest single agent is far from saturating the open tier.

I.4 Uncertainty, stopping statistics, the M_{23} runs, and how runs ended

Table 13 gives the allocation scores in three resource units and under both classification procedures, Table 14 the bootstrap intervals, Table 15 every M_{23} run, Table 12 the runs on which the evaluator's route milestone and the retrospective census disagree, and Table 16 how runs ended.

Table 12: Runs on which the evaluator's route milestone K and the retrospective census disagree about whether a positive route was researched. In eight runs the census counts a positive route that earned no milestone. In four the milestone was earned on a route the census does not count.

System	Task	Run	K	Census correct	Census wrong	V_r tok.
GPT-5.6 Sol	M12	sol_A	0	1	18	78.3
Claude Opus 5	M23	opus_A	100	0	1	17.6
Claude Opus 5	M11	opus_A2	0	2	10	89.8
GPT-5.6 Sol	M12	sol_A2	0	1	7	37.5
GPT-5.6 Sol	M24	sol_A2	100	0	6	40.9
GPT-5.6 Terra	M11	terra_A2	0	1	3	67.0
GPT-6 Astra	M11	astra_A	0	1	1	28.6
GPT-6 Astra	M12	astra_A2	0	1	3	63.2
GPT-6 Astra	M22	astra_A2	0	1	1	47.9
GPT-6 Sol	M22	sol6_A	100	0	17	60.5
GPT-6 Sol	M12	sol6_A2	0	1	28	83.1
GPT-6 Sol	M22	sol6_A2	100	0	11	49.5

Table 13: Stopping statistics per system: routes per run and non-positive routes per run from the retrospective classification of the artifact graphs, V_r under the paper's rule (evaluator value where recorded, census otherwise) in three resource units and under the census alone in tokens, U_r in three units, and mean tool CPU hours per run. The first V_r column is the value used in Table 1. V_r means exclude runs without a defined value: four GLM runs in every unit and one Sonnet run in CPU hours.

		Routes/run		V_r (paper rule)			V_r	U_r			
System	n	all	wrong	tok.	CPU	time	census tok.	tok.	CPU	time	CPU h
Claude Opus 5	10	10.5	9.3	72.8	53.0	66.0	72.8	57.7	60.2	56.3	2.2
GPT-6 Sol	10	20.5	19.3	79.1	61.1	75.6	79.1	31.4	29.7	30.7	0.7
GPT-6 Astra	10	3.8	2.9	55.9	54.4	59.4	58.0	60.3	62.8	56.8	0.4
GPT-5.6 Sol	10	9.3	8.8	57.8	29.2	46.4	57.8	27.5	23.3	23.7	2.6
GPT-5.6 Terra	10	6.9	6.6	38.2	34.2	31.6	38.2	24.6	13.8	16.9	3.8
Claude Sonnet 5	10	3.9	3.9	30.6	8.2	20.3	30.6	26.9	28.5	26.3	2.4
DeepSeek V4 Pro	10	2.7	2.7	66.6	44.5	51.2	66.6	34.4	20.7	25.9	2.5
GLM-5.3 Flash	10	1.4	1.4	63.9	48.9	55.4	63.9	33.5	35.9	31.9	0.5

Table 14: Per-system means with 95% bootstrap intervals (10,000 resamples). Over runs: percentile intervals from resampling the task-runs with replacement. Task-clustered: the five tasks are resampled with replacement and every run of a sampled task is kept.

		over runs			task-clustered	
System	n	S	K	V_r (tokens)	S	V_r (tokens)
Claude Opus 5	10	76.2 [64, 88]	90.0 [70, 100]	72.8 [56, 87]	[59, 93]	[57, 89]
GPT-6 Sol	10	62.1 [48, 76]	80.0 [50, 100]	79.1 [67, 91]	[54, 72]	[64, 93]
GPT-6 Astra	10	59.4 [40, 78]	50.0 [20, 80]	55.9 [34, 77]	[46, 72]	[27, 82]
GPT-5.6 Sol	10	38.8 [26, 51]	40.0 [10, 70]	57.8 [44, 72]	[23, 55]	[42, 73]
GPT-5.6 Terra	10	21.9 [9, 36]	20.0 [0, 50]	38.2 [23, 53]	[9, 38]	[22, 54]
Claude Sonnet 5	10	11.5 [6, 18]	0.0 [0, 0]	30.6 [14, 48]	[6, 20]	[13, 48]
DeepSeek V4 Pro	10	7.8 [5, 11]	0.0 [0, 0]	66.6 [49, 83]	[6, 11]	[46, 84]
GLM-5.3 Flash	10	2.6 [0, 6]	0.0 [0, 0]	63.9 [51, 77]	[0, 5]	[49, 76]

Table 15: Every M_{23} run. Routes and wrong routes from the retrospective classification, V_r in tokens and in tool CPU hours, U_r in tokens. $C = 12.5$ is the 5-point stage-2 credit of the bespoke ladder (Appendix D). N/A marks a run with no route research, for which V_r is undefined.

System	Run	S	K	C	Routes	Wrong	V_r tok.	V_r CPU	U_r tok.	Tokens (k)	Hours
Claude Opus 5	batch 2, run 1	65	100	12.5	1	1	17.6	9.3	22.9	451	5.6
	batch 2, run 2	65	100	12.5	10	9	78.6	46.1	35.7	460	10.0
GPT-6 Sol	batch 2, run 1	35	0	12.5	25	25	43.8	4.3	31.2	158	10.0
	batch 2, run 2	65	100	12.5	22	21	81.1	8.7	33.0	154	10.0
GPT-6 Astra	batch 2, run 1	35	0	12.5	8	8	4.7	0.9	3.5	500	7.4
	batch 2, run 2	35	0	12.5	13	13	6.9	3.5	6.1	504	8.2
GPT-5.6 Sol	batch 2, run 1	35	0	12.5	11	11	39.7	4.1	20.0	83	3.5
	batch 2, run 2	35	0	12.5	5	5	24.4	1.5	14.5	232	9.8
GPT-5.6 Terra	batch 2, run 1	35	0	12.5	7	7	13.4	0.1	50.7	92	3.9
	batch 2, run 2	65	100	12.5	14	13	62.4	0.8	23.2	290	10.0
Claude Sonnet 5	batch 2, run 1	24	0	12.5	5	5	77.7	63.3	48.3	451	4.4
	batch 2, run 2	35	0	12.5	2	2	40.1	0.0	64.0	411	5.8
DeepSeek V4 Pro	batch 2, run 1	20	0	12.5	3	3	83.1	12.1	2.1	242	1.9
	batch 2, run 2	10	0	12.5	11	11	98.2	96.4	15.5	211	2.6
GLM-5.3 Flash	batch 2, run 1	0	0	0.0	0	0	N/A	N/A	0.0	509	2.4
	batch 2, run 2	0	0	0.0	2	2	48.1	97.5	0.0	446	5.0

Table 16: How the eligible runs ended: the reason recorded in the run summary, and, derived from the recorded counters, how many runs reached 99% of their wall or token budget. The two disagree for Opus, one of whose M_{12} runs is recorded as completed at 9.9 hours, for Sonnet, whose M_{22} run is recorded as completed at 497k tokens, and for GLM, seven of whose runs reached or exceeded the 500k cap by up to 5% and one of which used the full 10 hours, all recorded as completed.

	Recorded termination				Derived from counters	
System	Completed	Wall limit	Token limit	Other	Wall $\geq 99\%$	Tokens $\geq 99\%$
Claude Opus 5	7	2	1	0	3	1
GPT-6 Sol	4	3	3	0	3	3
GPT-6 Astra	7	1	2	0	1	2
GPT-5.6 Sol	9	1	0	0	1	0
GPT-5.6 Terra	4	6	0	0	6	0
Claude Sonnet 5	10	0	0	0	0	1
DeepSeek V4 Pro	4	0	6	0	0	6
GLM-5.3 Flash	10	0	0	0	1	7

I.5 RESULTS BY TASK

Table 17: Verified progress by model and task. Components are percentages of their maxima; S is out of 100.

Model	Task	R	K	C	S	POLY	n
Claude Opus 5	M11	70.0	50.0	50.0	56.0	50.0	2
Claude Opus 5	M12	100.0	100.0	100.0	100.0	100.0	2
Claude Opus 5	M22	100.0	100.0	100.0	100.0	100.0	2
Claude Opus 5	M23	100.0	100.0	12.5	65.0	0.0	2
Claude Opus 5	M24	100.0	100.0	0.0	60.0	0.0	2
GPT-6 Sol	M11	100.0	100.0	0.0	60.0	0.0	2
GPT-6 Sol	M12	85.0	50.0	50.0	60.5	50.0	2
GPT-6 Sol	M22	100.0	100.0	50.0	80.0	50.0	2
GPT-6 Sol	M23	100.0	50.0	12.5	50.0	0.0	2
GPT-6 Sol	M24	100.0	100.0	0.0	60.0	0.0	2
GPT-6 Astra	M11	56.7	50.0	50.0	52.0	50.0	2
GPT-6 Astra	M12	60.0	50.0	100.0	73.0	100.0	2
GPT-6 Astra	M22	73.3	50.0	100.0	77.0	100.0	2
GPT-6 Astra	M23	100.0	0.0	12.5	35.0	0.0	2
GPT-6 Astra	M24	100.0	100.0	0.0	60.0	0.0	2
GPT-5.6 Sol	M11	100.0	100.0	0.0	60.0	0.0	2
GPT-5.6 Sol	M12	46.7	0.0	0.0	14.0	0.0	2
GPT-5.6 Sol	M22	16.7	0.0	50.0	25.0	50.0	2
GPT-5.6 Sol	M23	100.0	0.0	12.5	35.0	0.0	2
GPT-5.6 Sol	M24	100.0	100.0	0.0	60.0	0.0	2
GPT-5.6 Terra	M11	36.7	0.0	0.0	11.0	0.0	2
GPT-5.6 Terra	M12	56.7	50.0	0.0	32.0	0.0	2
GPT-5.6 Terra	M22	23.3	0.0	0.0	7.0	0.0	2
GPT-5.6 Terra	M23	100.0	50.0	12.5	50.0	0.0	2
GPT-5.6 Terra	M24	31.7	0.0	0.0	9.5	0.0	2
Claude Sonnet 5	M11	26.7	0.0	0.0	8.0	0.0	2
Claude Sonnet 5	M12	20.0	0.0	0.0	6.0	0.0	2
Claude Sonnet 5	M22	16.7	0.0	0.0	5.0	0.0	2
Claude Sonnet 5	M23	81.7	0.0	12.5	29.5	0.0	2
Claude Sonnet 5	M24	30.0	0.0	0.0	9.0	0.0	2
DeepSeek V4 Pro	M11	20.0	0.0	0.0	6.0	0.0	2
DeepSeek V4 Pro	M12	20.0	0.0	0.0	6.0	0.0	2
DeepSeek V4 Pro	M22	20.0	0.0	0.0	6.0	0.0	2
DeepSeek V4 Pro	M23	33.3	0.0	12.5	15.0	0.0	2
DeepSeek V4 Pro	M24	20.0	0.0	0.0	6.0	0.0	2
GLM-5.3 Flash	M11	16.7	0.0	0.0	5.0	0.0	2
GLM-5.3 Flash	M12	6.7	0.0	0.0	2.0	0.0	2
GLM-5.3 Flash	M22	0.0	0.0	0.0	0.0	0.0	2
GLM-5.3 Flash	M23	0.0	0.0	0.0	0.0	0.0	2
GLM-5.3 Flash	M24	20.0	0.0	0.0	6.0	0.0	2

I.6 Additional quantitative views

Figure 14 splits each system's mean S into its three credit components, Figure 15 plots progress against measured cost and time.

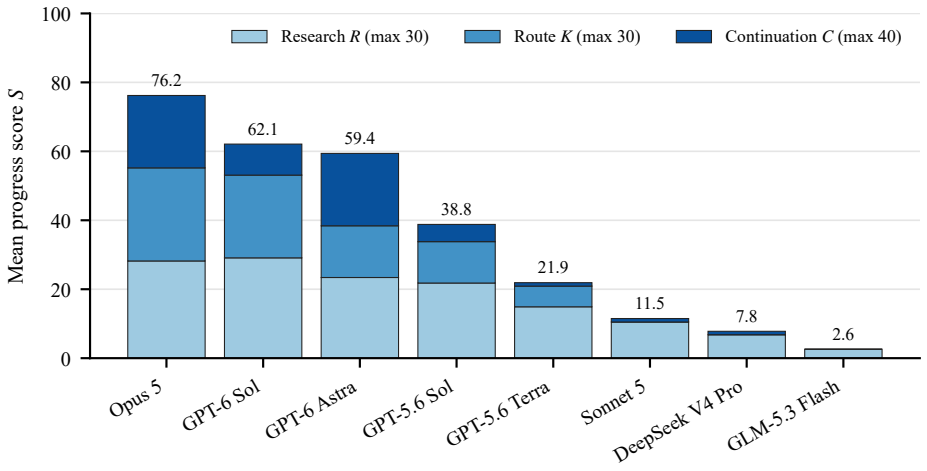


Figure 14: Composition of the mean progress score S (points, 80 eligible runs, eight single-agent systems) into research ($0.3R$), route ($0.3K$), and continuation ($0.4C$) credit. Opus and GPT-6 Sol differ mainly in continuation credit, Opus and Astra in route credit.

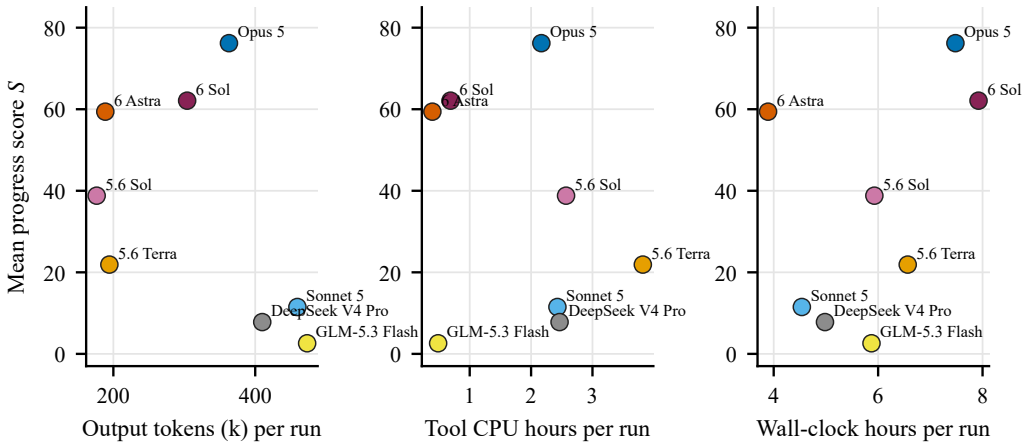


Figure 15: Progress against measured cost. Cost and time are observational because campaign budgets are not fully balanced across systems.

I.7 Correlation among metrics

Figure 16 gives the run-level Spearman correlation among the metrics.

I.8 An artifact graph

Figure 17 shows the artifact graph of a Claude Opus 5 run that realizes M_{22}.

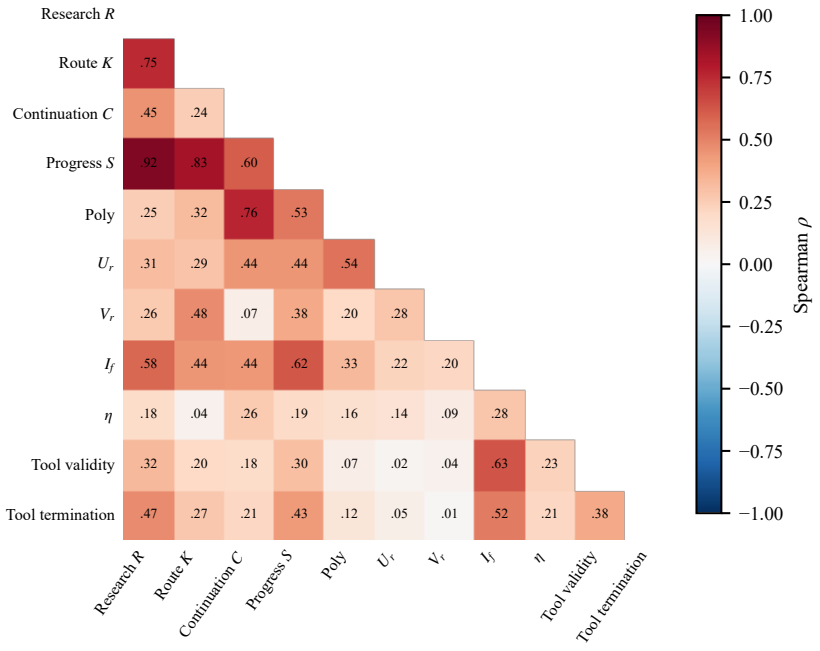


Figure 16: Run-level Spearman correlation among IGP-Bench metrics over the 80 eligible task-runs, on pairwise complete runs (V_r is defined on 76 runs, I_f on 79, and η on 75).

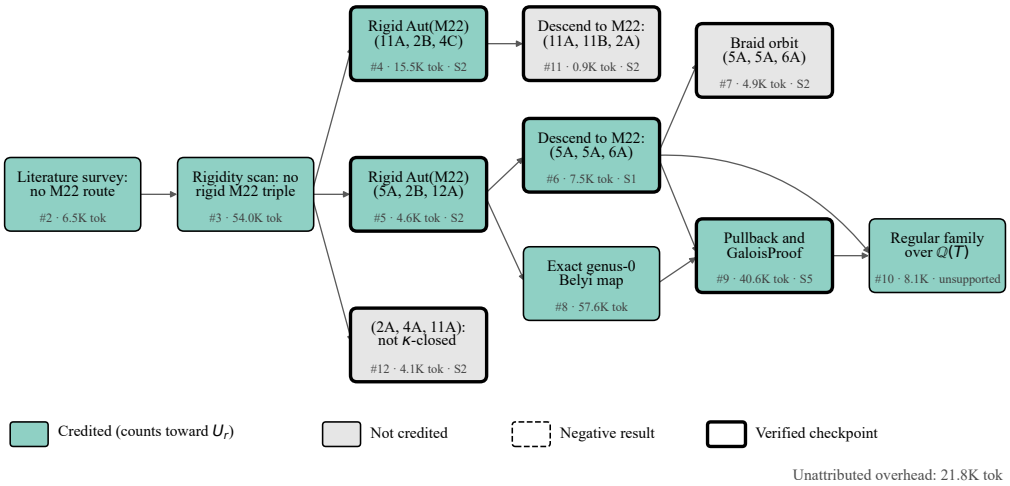


Figure 17: Artifact graph of a Claude Opus 5 run that realizes M_{22} ($S = 100$). Boxes are artifacts laid out by dependency depth, with output tokens and the certified checkpoint strength. Filled boxes count toward U_r and bold borders are verified checkpoint claims. The run finds no rigid rational triple in M_{22}, certifies two rigid triples of $\mathrm{Aut}(M_{22})$, descends one to M_{22}, solves its Belyi map exactly, and certifies the pulled-back polynomial (S5).

I.9 FAILURE TAXONOMY

The taxonomy covers 79 task-runs: the eligible runs except the GPT-6 Sol campaign-02 M_{12} rerun, which finished after the coding pass. Each run was coded against a fixed scheme of 20 failure types in six phases by LLM-assisted reading of its artifact graph, checkpoint submissions, and tool logs. The labels were not independently validated, so we use the taxonomy only for qualitative diagnostics. Every percentage quoted in the main text is the *Runs* column of Table 18, and Figure 18 shows the per-system rates.

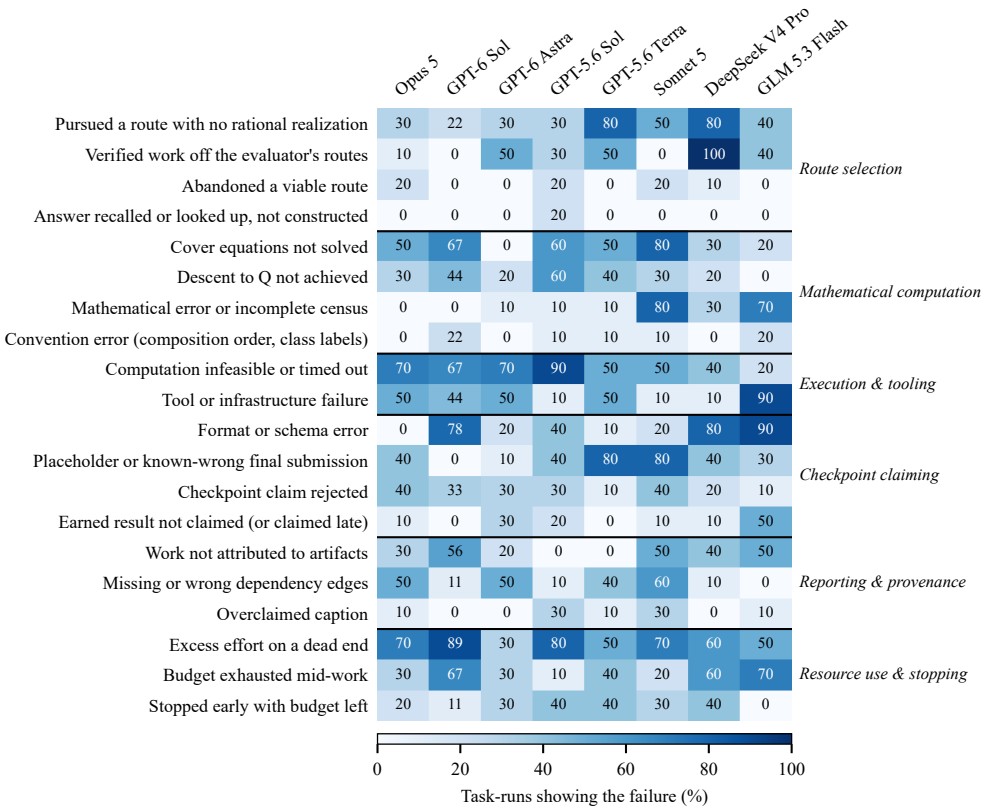


Figure 18: Failure-mode heatmap over the 79 coded task-runs. Cells show the percentage of each system's coded task-runs exhibiting the failure.

Table 18: Failure modes across IGP-Bench runs, by phase of the constructive pipeline. *Runs* is the percentage of the 79 analysed task-runs showing the failure; *Share* is its percentage of all recorded failure instances. Failures were identified from each run's artifact graph, evaluator verdicts, format log and honesty-judge checks, then merged into shared types.

Failure type	Runs (%)	Share (%)	Example
Route selection			
Pursued a route with no rational realization	46	7.4	(2A,4A,11A) for M11: field of moduli $\mathbb{Q}(\sqrt{-11})$ (Sol, Opus)
Verified work off the evaluator's routes	35	5.7	Opus M24: rational (3A,3B,4C) is on no recorded route
Abandoned a viable route	9	1.4	Sol M24: earned K on (12A,2A,2A,2A), then dropped it
Answer recalled or looked up, not constructed	3	0.4	Sol M12: Magma `PolynomialWithGaloisGroup(12,295)`
Mathematical computation			
Cover equations not solved	44	6.8	Opus M23: 7-equation Belyi system; Gröbner basis unfinished
Descent to Q not achieved	30	4.7	Sol M24: rigid (23A,3A,2A) cover built only over $\mathbb{Q}(\sqrt{-23})$
Mathematical error or incomplete census	27	4.7	Sonnet: rigidity census kept only triples with $n(C) = \lvert G\rvert$
Convention error (composition order, class labels)	9	1.4	Sonnet M12: involution labelled 2a from GAP class names, not ATLAS
Execution & tooling			
Computation infeasible or timed out	57	8.8	Magma Gröbner/F4 runs hitting the 1,800 s tool limit
Tool or infrastructure failure	39	7.0	Terra M12: every shell command failed after 2.3 h
Checkpoint claiming			
Format or schema error	42	6.5	Sol M11/M22: non-polynomial final submission rejected
Placeholder or known-wrong final submission	41	6.3	Sol M22: submitted a trinomial it had shown has group A_{22}
Checkpoint claim rejected	27	4.5	Sol M12: claimed M12 cover has monodromy of order 1,920
Earned result not claimed (or claimed late)	16	2.5	Opus M12: Nielsen-class data never claimed as a checkpoint
Reporting & provenance			
Work not attributed to artifacts	30	4.7	Sonnet M22/M23: 97–98% of output tokens in no artifact
Missing or wrong dependency edges	29	4.5	Parent edges omit artifacts actually used (judge check 3)
Overclaimed caption	11	1.8	Sol M22: caption calls a rejected regular family certified
Resource use & stopping			
Excess effort on a dead end	62	10.6	Opus M22: kept open for 6 h after reaching $S = 100$
Budget exhausted mid-work	41	6.3	Opus M22: token cap hit mid-way through the cover solve
Stopped early with budget left	27	4.1	Terra M11/M23/M24: stopped after about 1 h of 6 h

I.10 ALL TASK-RUNS

Table 19 lists every eligible task-run behind the averages of Tables 1 and 17, one row per run.

Table 19: IGP-Bench scores for every eligible task-run (80 rows), before averaging over repetitions and tasks. Columns as in Table 1, each for a single run: R, K and C as a percentage of their maxima; POLY is 100 when the run produced a verifier-certified terminal polynomial. Every row is a main-campaign run (10 h, 500,000 output tokens); *run 1* and *run 2* are the two repetitions of a task. Eligibility and provenance follow Appendix I; N/A marks a metric that is undefined for the run (Appendix F).

		Verified progress (%)					**Allocation**		**Protocol**		**Tool use (%)**	
Task	**Run**	R	K	C	S	POLY	U_r	V_r	I_f	η	Valid	Term.
Claude Opus 5 (10 runs)												
M11	run 1	100.0	100.0	0.0	**60.0**	0.0	21.0	89.8	98.5	100.0	100.0	96.1
	run 2	40.0	0.0	100.0	**52.0**	100.0	58.8	89.8	100.0	100.0	100.0	95.6
M12	run 1	100.0	100.0	100.0	**100.0**	100.0	94.8	100.0	86.4	100.0	99.4	87.4
	run 2	100.0	100.0	100.0	**100.0**	100.0	95.5	95.1	95.0	100.0	100.0	83.8
M22	run 1	100.0	100.0	100.0	**100.0**	100.0	95.1	40.7	100.0	100.0	100.0	81.2
	run 2	100.0	100.0	100.0	**100.0**	100.0	96.2	76.7	100.0	90.0	99.3	88.2
M23	run 1	100.0	100.0	12.5	**65.0**	0.0	22.9	17.6	96.9	100.0	100.0	91.3
	run 2	100.0	100.0	12.5	**65.0**	0.0	35.7	78.6	97.7	100.0	100.0	94.9
M24	run 1	100.0	100.0	0.0	**60.0**	0.0	29.0	78.2	97.9	100.0	100.0	97.1
	run 2	100.0	100.0	0.0	**60.0**	0.0	28.2	61.4	96.7	100.0	99.6	96.1
GPT-6 Sol (10 runs)												
M11	run 1	100.0	100.0	0.0	**60.0**	0.0	2.4	92.8	99.1	100.0	100.0	98.8
	run 2	100.0	100.0	0.0	**60.0**	0.0	3.8	90.4	98.9	100.0	100.0	98.6
M12	run 1	100.0	100.0	100.0	**100.0**	100.0	64.5	95.9	99.2	100.0	100.0	98.3
	run 2	70.0	0.0	0.0	**21.0**	0.0	19.0	83.1	100.0	100.0	100.0	97.9
M22	run 1	100.0	100.0	100.0	**100.0**	100.0	83.3	60.5	99.0	100.0	100.0	97.6
	run 2	100.0	100.0	0.0	**60.0**	0.0	64.5	49.5	98.8	88.9	100.0	99.5
M23	run 1	100.0	0.0	12.5	**35.0**	0.0	31.2	43.8	100.0	100.0	100.0	99.8
	run 2	100.0	100.0	12.5	**65.0**	0.0	33.0	81.1	100.0	97.6	100.0	99.5
M24	run 1	100.0	100.0	0.0	**60.0**	0.0	3.0	96.9	98.4	98.2	100.0	99.1
	run 2	100.0	100.0	0.0	**60.0**	0.0	9.6	97.1	99.0	98.9	100.0	98.3
GPT-6 Astra (10 runs)												
M11	run 1	13.3	0.0	0.0	**4.0**	0.0	99.3	28.6	100.0	100.0	100.0	95.7
	run 2	100.0	100.0	100.0	**100.0**	100.0	100.0	100.0	100.0	100.0	100.0	97.4
M12	run 1	100.0	100.0	100.0	**100.0**	100.0	65.3	45.8	95.0	100.0	99.0	97.1
	run 2	20.0	0.0	100.0	**46.0**	100.0	41.5	63.2	95.0	100.0	99.0	99.0
M22	run 1	100.0	100.0	100.0	**100.0**	100.0	100.0	62.9	100.0	100.0	100.0	99.2
	run 2	46.7	0.0	100.0	**54.0**	100.0	77.4	47.9	100.0	100.0	100.0	97.3
M23	run 1	100.0	0.0	12.5	**35.0**	0.0	3.5	4.7	97.2	100.0	100.0	98.2
	run 2	100.0	0.0	12.5	**35.0**	0.0	6.1	6.9	97.2	100.0	99.8	98.4
M24	run 1	100.0	100.0	0.0	**60.0**	0.0	60.8	99.1	100.0	100.0	100.0	92.7
	run 2	100.0	100.0	0.0	**60.0**	0.0	49.0	100.0	97.6	100.0	99.8	91.8
GPT-5.6 Sol (10 runs)												
M11	run 1	100.0	100.0	0.0	**60.0**	0.0	46.6	83.4	97.9	100.0	100.0	93.8
	run 2	100.0	100.0	0.0	**60.0**	0.0	20.9	83.5	97.1	100.0	99.3	94.4
M12	run 1	43.3	0.0	0.0	**13.0**	0.0	26.7	78.3	99.0	100.0	100.0	96.6
	run 2	50.0	0.0	0.0	**15.0**	0.0	27.1	37.5	98.3	100.0	99.8	89.7
M22	run 1	13.3	0.0	0.0	**4.0**	0.0	5.8	58.4	98.4	100.0	100.0	92.8
	run 2	20.0	0.0	100.0	**46.0**	100.0	90.5	42.7	100.0	100.0	100.0	93.9
M23	run 1	100.0	0.0	12.5	**35.0**	0.0	20.0	39.7	97.7	100.0	100.0	96.3
	run 2	100.0	0.0	12.5	**35.0**	0.0	14.5	24.4	97.5	100.0	99.8	95.3
M24	run 1	100.0	100.0	0.0	**60.0**	0.0	8.3	89.6	100.0	100.0	100.0	95.5
	run 2	100.0	100.0	0.0	**60.0**	0.0	14.6	40.9	97.9	80.0	100.0	93.7
GPT-5.6 Terra (10 runs)												

continued on next page

Table 19 continued

		Verified progress (%)					Allocation		Protocol		Tool use (%)	
Task	**Run**	R	K	C	S	POLY	U_r	V_r	I_f	η	Valid	Term.
M11	run 1	33.3	0.0	0.0	**10.0**	0.0	16.5	45.2	96.4	100.0	100.0	98.4
	run 2	40.0	0.0	0.0	**12.0**	0.0	30.9	67.0	92.9	100.0	100.0	93.7
M12	run 1	13.3	0.0	0.0	**4.0**	0.0	3.2	54.8	98.1	100.0	100.0	93.5
	run 2	100.0	100.0	0.0	**60.0**	0.0	82.7	62.8	92.9	100.0	99.1	95.3
M22	run 1	13.3	0.0	0.0	**4.0**	0.0	4.7	5.1	94.7	100.0	100.0	90.7
	run 2	33.3	0.0	0.0	**10.0**	0.0	7.6	8.3	97.4	97.1	100.0	93.8
M23	run 1	100.0	0.0	12.5	**35.0**	0.0	50.7	13.4	100.0	100.0	100.0	91.5
	run 2	100.0	100.0	12.5	**65.0**	0.0	23.2	62.4	98.8	100.0	99.8	94.5
M24	run 1	36.7	0.0	0.0	**11.0**	0.0	17.1	51.0	95.7	100.0	99.8	96.7
	run 2	26.7	0.0	0.0	**8.0**	0.0	9.1	11.9	96.2	100.0	100.0	91.7
Claude Sonnet 5 (10 runs)												
M11	run 1	33.3	0.0	0.0	**10.0**	0.0	1.0	1.0	88.9	80.0	99.6	93.6
	run 2	20.0	0.0	0.0	**6.0**	0.0	0.3	0.0	93.8	100.0	100.0	95.5
M12	run 1	0.0	0.0	0.0	**0.0**	0.0	39.7	40.3	90.9	100.0	100.0	96.1
	run 2	40.0	0.0	0.0	**12.0**	0.0	16.8	0.0	94.4	92.9	99.6	92.9
M22	run 1	0.0	0.0	0.0	**0.0**	0.0	0.7	69.9	94.7	100.0	99.5	96.4
	run 2	33.3	0.0	0.0	**10.0**	0.0	40.1	19.9	95.7	100.0	100.0	96.4
M23	run 1	63.3	0.0	12.5	**24.0**	0.0	48.3	77.7	97.4	100.0	99.6	96.4
	run 2	100.0	0.0	12.5	**35.0**	0.0	64.0	40.1	93.8	100.0	100.0	95.8
M24	run 1	20.0	0.0	0.0	**6.0**	0.0	1.1	11.3	94.7	100.0	99.8	95.2
	run 2	40.0	0.0	0.0	**12.0**	0.0	56.5	46.0	94.4	100.0	99.7	97.0
DeepSeek V4 Pro (10 runs)												
M11	run 1	20.0	0.0	0.0	**6.0**	0.0	16.9	44.5	88.9	100.0	100.0	91.9
	run 2	20.0	0.0	0.0	**6.0**	0.0	81.1	82.6	80.0	66.7	100.0	88.4
M12	run 1	20.0	0.0	0.0	**6.0**	0.0	0.3	23.6	80.0	66.7	100.0	89.0
	run 2	20.0	0.0	0.0	**6.0**	0.0	16.5	26.4	90.0	75.0	98.5	94.8
M22	run 1	0.0	0.0	0.0	**0.0**	0.0	18.4	81.5	75.0	75.0	99.4	89.6
	run 2	40.0	0.0	0.0	**12.0**	0.0	95.3	87.7	87.5	100.0	99.4	90.5
M23	run 1	50.0	0.0	12.5	**20.0**	0.0	2.1	83.1	95.0	100.0	100.0	92.5
	run 2	16.7	0.0	12.5	**10.0**	0.0	15.5	98.2	100.0	100.0	100.0	76.5
M24	run 1	20.0	0.0	0.0	**6.0**	0.0	58.0	99.3	83.3	100.0	100.0	87.7
	run 2	20.0	0.0	0.0	**6.0**	0.0	40.0	38.8	85.7	100.0	98.4	85.2
GLM-5.3 Flash (10 runs)												
M11	run 1	0.0	0.0	0.0	**0.0**	0.0	83.2	81.8	75.0	100.0	98.6	88.9
	run 2	33.3	0.0	0.0	**10.0**	0.0	9.6	64.8	62.5	100.0	97.0	78.1
M12	run 1	0.0	0.0	0.0	**0.0**	0.0	0.0	N/A	0.0	N/A	96.6	87.8
	run 2	13.3	0.0	0.0	**4.0**	0.0	63.7	56.2	66.7	66.7	98.5	84.4
M22	run 1	0.0	0.0	0.0	**0.0**	0.0	0.0	N/A	N/A	N/A	100.0	75.4
	run 2	0.0	0.0	0.0	**0.0**	0.0	99.9	45.2	61.1	N/A	98.3	90.3
M23	run 1	0.0	0.0	0.0	**0.0**	0.0	0.0	N/A	33.3	N/A	94.9	86.5
	run 2	0.0	0.0	0.0	**0.0**	0.0	0.0	48.1	46.3	33.3	92.8	79.2
M24	run 1	40.0	0.0	0.0	**12.0**	0.0	78.1	87.1	72.2	75.0	97.3	78.7
	run 2	0.0	0.0	0.0	**0.0**	0.0	0.0	N/A	61.1	N/A	98.4	70.4

J Related Benchmarks

Table 20: Related benchmarks. *Partial credit*: intermediate progress is scored. *Deterministic*: without a human or learned judge. *Task-derived*: intermediate states arise from doing the task rather than being authored in advance. *Stopping*: whether, and how, knowing when to stop is measured. ✓ yes, △ partly, × no, — not applicable without partial credit.

Benchmark	**Math research**	**Partial credit**	**Deterministic**	**Task-derived**	**Stopping**
FrontierMath, Riemann-Bench	✓	×	—	—	×
HorizonMath	✓	△	✓	×	×
IMProofBench	✓	✓	△	×	×
TheoremBench	△	✓	✓	△	×
PaperBench	×	✓	×	△	×
EXP-Bench	×	✓	△	△	×
AgentAbstain	×	×	△	×	act or abstain, one bit per task
IGP-Bench	✓	✓	✓	✓	cost per route, U_r, V_r