When LLMs Solve ODEs: Failure Modes, Conservation Blind Spots, and Ground-Truth-Free Auditing

Ahmad A. Rushdi [1]

Abstract

When frontier LLMs produce numerical solutions for dynamical systems, their errors are silent: a trajectory can look plausible while violating conservation laws by orders of magnitude. We identify three distinct failure regimes: catastrophic divergence, trivial-invariant collapse, and phase drift, and show that each demands a different verification strategy. We propose two ground-truth-free audit signals: the conservation residual R, which measures invariant drift, and step consistency S, which checks whether the trajectory locally satisfies the governing ODE. R catches catastrophic failures but is structurally blind to phase drift and can be trivially satisfied; the complementary signal S closes these gaps. Within-model, S rank-orders true trajectory error with Spearman $\rho \in [0.77,\ 0.99]$ across all six frontier LLMs, and leave-one-task-out analysis confirms robustness ($\rho \in [0.81,\ 0.90]$). We further document that models split into *deterministic* and *stochastic* numerical reasoners—the latter producing over $1{,}000\times$ error variation across identical prompts—and that prompting models to preserve invariants triggers Goodhart-style evaluation gaming: some models learn to emit invariant-preserving but dynamically incorrect trajectories.

1. Introduction

Scientific pipelines route numerical work through Large Language Models (LLMs) in several ways. The most common and robust is *tool-mediated*: the model writes solver code or calls a numerical library, and a deterministic backend does the arithmetic. But a second mode is now routine in practice—the model is asked to *produce the numerical trajectory directly*, in a single exchange, with no code execution. This happens whenever a user pastes an ODE into a chat window and asks for the state at a list of times, when tool use is unavailable or disabled, in quick interactive exploration before a pipeline is built, and inside agentic loops that treat a model's emitted numbers as intermediate state. We study this direct-trajectory mode not because it is the dominant workflow—it is not—but because it is the cleanest setting in which to isolate and measure a failure that the tool-mediated path hides: when the model itself plays the role of the integrator, its numerical errors surface undiluted. The signals we develop then transfer to any setting where a trajectory must be audited, regardless of how it was produced (Sec. 5).

The direct-trajectory mode is dangerous because LLM errors on numerical tasks are *silent*. A trajectory that looks reasonable—smooth, bounded, starting from the correct initial condition—can quietly violate energy conservation by orders of magnitude over a single orbit. Unlike factual hallucination ("the capital of France is Cairo"), a numerically incorrect solution carries no lexical signal; a human inspecting the raw numbers may not notice the error without running an independent integrator. This raises a fundamental *verification* problem: how can we assess whether an LLM-generated numerical solution is trustworthy, without access to a reference solver or model internals? We use *hallucination* to mean a numerically inaccurate trajectory presented without any indication of its inaccuracy—encompassing both gross violations (energy blow-ups) and subtler errors (correct functional form with wrong phase).

Existing hallucination detectors cannot close this gap. Semantic entropy (Farquhar et al., 2024; Huang et al., 2023) compares sampled completions for semantic agreement—a paradigm that does not transfer to numerical trajectories, where "meaning" is geometric rather than semantic. Self-reported confidence (Kadavath et al., 2022; Lin et al., 2022) is, as we show, uncorrelated with accuracy on these tasks. We are explicit about what our signals are *not*: when a high-accuracy reference solver is available and affordable, running it yields ground truth directly and should be preferred over any structural proxy. R and S are instead cheap *ranking and triage* signals for the already-produced trajectory—useful for screening or rank-ordering outputs (across mod-

[1]Stanford University, Stanford, CA, USA. Correspondence to: Ahmad A. Rushdi <rushdi@stanford.edu>.

The 3rd *AI for Math Workshop* at the 43rd International Conference on Machine Learning (ICML), Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

els, runs, or many candidates) when re-solving every one is undesirable, or when using the LLM's own output is the premise rather than a value to be replaced. As Kalai et al. (2025) formalized, hallucinations persist because training rewards guessing over admitting uncertainty; for numerical outputs, no existing detector instantiates their "Is-It-Valid" oracle.

We make a simple observation: in mathematically structured domains, verification criteria come for free. Hamiltonian systems conserve energy; central-force problems conserve angular momentum; symplectic integrators preserve phase-space volume (Hairer et al., 2006). These are mathematical properties of the governing equations, available to anyone who knows which ODE the LLM was asked to solve. An LLM that truly integrates the dynamics must preserve these invariants; one that is pattern-matching will not.

This paper makes three contributions. First, we identify a taxonomy of three failure regimes in LLM numerical reasoning—catastrophic divergence, trivial-invariant collapse, and phase drift—each requiring a different verification strategy. Second, we propose two complementary ground-truth-free audit signals—the *conservation residual* R and *step consistency* S—and evaluate them on 6 dynamical systems and 6 frontier LLMs (85 responses). Within each model, S rank-orders true trajectory error with Spearman $\rho \in [0.77, 0.99]$; leave-one-task-out and leave-one-model-out analyses confirm robustness ($\rho \in [0.81, 0.90]$). We emphasize that the contribution is a *verification tool*, not a calibrated predictor, as absolute error calibration overfits at this sample size. Third, we document a deterministic-vs-stochastic split in LLM numerical reasoning and a Goodhart-style effect of physics-informed prompting, where models learn to game the conservation metric rather than improve trajectory accuracy.

2. Related Work

A rich line of physics-informed machine learning encodes conservation laws, symmetries, and governing equations directly into model architectures or loss functions (Karniadakis et al., 2021). Physics-informed neural networks (Raissi et al., 2019) enforce PDEs as soft constraints; Hamiltonian and Lagrangian neural networks (Greydanus et al., 2019; Cranmer et al., 2020) hard-wire symplectic structure; and Neural ODEs (Chen et al., 2018) parameterize continuous dynamics. All treat invariants as *training signals*—they bake physics into the model *before* it is deployed. Our work takes the complementary view: invariants serve as *post-hoc audit signals* applied to the output of opaque LLMs that were never trained to respect physics.

Despite surprising quantitative capacity (Lewkowycz et al., 2022; Ahn et al., 2024), LLMs fail persistently on scientific tasks: SciCode (Tian et al., 2024) reports 4.6% solve rate on research coding, PRL-Bench (Miao et al., 2026) finds sub-50% accuracy on physics research questions, and FEM-Bench (Mohammadzadeh et al., 2025) documents systematic failures in finite-element code under physical constraints. Agentic frameworks like AgenticSciML (Jiang & Karniadakis, 2025) use PDE residuals to steer evolutionary searches over model architectures—quality signals conceptually related to ours, but applied at *design time* over many candidates to produce a fixed artifact. We focus on *inference-time* auditing: given one trajectory from a single LLM call, can we tell whether it is trustworthy without re-running the computation? A standalone user can apply R and S with a few lines of code and the system's known invariants; no agentic infrastructure is required.

Kalai et al. (2025) recently formalized the statistical roots of hallucination, showing that training and evaluation pressures reward guessing over expressing uncertainty, and that hallucination rates are lower-bounded by the misclassification rate of a binary "Is-It-Valid" (IIV) oracle. Existing detectors approach this oracle through text: semantic entropy (Farquhar et al., 2024; Huang et al., 2023) compares sampled completions for semantic agreement, and self-reported confidence (Kadavath et al., 2022; Lin et al., 2022) probes the model's own assessment. Neither transfers to numerical trajectories, where "meaning" is geometric (energy, phase-space volume) rather than semantic—and self-reported confidence is, as we show, essentially uncorrelated with accuracy on these tasks, consistent with the post-training miscalibration documented in GPT-4 (OpenAI, 2023). In the white-box direction, Li et al. (2025) train neural ODEs on LLM hidden states to flag divergences from "truthful" latent dynamics, but this requires internal activation access that frontier APIs do not provide. A recent survey by Liu et al. (2025) taxonomizes UQ methods for LLMs and finds that most require logits, multiple samples, or model internals; semantic uncertainty (Kuhn et al., 2023) clusters completions by meaning, but numerical trajectories have no meaningful "paraphrases"—two slightly different state-vector sequences are two distinct geometric objects with different errors against the true solution.

The audit signals we propose instantiate Kalai et al.'s IIV oracle for dynamical systems without requiring model access, training data, or a learned detector. They are epistemic in the taxonomy of Kendall & Gal (2017): they measure what the LLM does *not* know about the dynamics. The distinction matters here, because the failure modes we flag (wrong integration scheme, drifted phase, blown-up close encounters) are squarely epistemic on deterministic ODEs. They require only the LLM's output and the system's known invariants, making them suited to the black-box, single-completion setting in which most scientific LLM use occurs.

More broadly, our work connects to the growing need for *mathematical verification of LLM outputs*. Formal provers verify symbolic reasoning via proof assistants; code-generation benchmarks verify programs via test suites; retrieval-augmented systems verify claims against knowledge bases. Numerical trajectories occupy a gap: too structured for text-based detectors, too opaque for formal verification (the LLM's "integration scheme" is not inspectable), and too cheap to warrant a full solver rerun. Conservation laws and ODE residuals fill this gap by providing mathematical constraints that any correct solution must satisfy—lightweight, domain-specific verification that trades completeness for practicality. The principle generalizes: wherever a problem's governing equations impose verifiable structure on solutions—balance laws in chemistry, feasibility constraints in optimization, consistency conditions in symbolic computation—the same audit logic applies.

The contribution of this paper is to show concretely, on a benchmark spanning six dynamical systems and six frontier LLMs, when such structural checks succeed, when they fail, and why either happens.

3. Experimental Setup

3.1. The Integration Task

Every experiment poses the same request to every model: *numerically integrate a given ODE and return the trajectory at prescribed time points*. Using a single task type isolates the effect of the dynamics: the prompt format, output schema, and evaluation pipeline are identical, so any variation in quality is attributable to the *model* and the *dynamics* alone.

Task $\mathcal{T} = (f,\ \mathbf{y}_0,\ [t_0, t_f],\ N,\ \{\phi_k\})$

Given (encoded as a JSON prompt):
- f: ODE $\dot{\mathbf{y}} = f(t, \mathbf{y})$
- $t_0,\ \mathbf{y}_0$: initial conditions
- $t_f,\ N$: time interval and number of output points
- $\{\phi_k\}$: known conserved quantities ($\dot{\phi}_k = 0$)

Return (using internal reasoning only — no code, no tools):
- $\hat{\mathbf{y}}_M$: N state vectors at the requested times

Audit (computed post-hoc — no reference solver needed):
- R: Do conserved quantities stay constant along trajectory?
- S: Does each step locally satisfy the governing ODE?

3.2. Benchmark Systems

The six systems (Fig. 1) satisfy three criteria: (i) a difficulty gradient from closed-form recall to genuine integration; (ii) at least one known invariant per system; and (iii) one system with non-standard parameters. **Kepler** ($e = 0.5$) admits an exact solution and sets a capability ceiling. The **harmonic oscillator** (Goldstein et al., 2002) ($x = \cos t$) plays a subtler role: energy is preserved by *any* output of correct amplitude regardless of phase, so R is blind to phase drift—motivating S. The **nonlinear pendulum** (Goldstein et al., 2002) ($\ddot{\theta} = -\sin\theta,\ \theta_0 = 1\,\text{rad}$) yields a hybrid regime of partial recall and partial integration. The **figure-8 three-body orbit** (Chenciner & Montgomery, 2000) is chaotic with three independent invariants, forcing genuine numerical reasoning. **Lotka–Volterra** (Strogatz, 2015) is non-Hamiltonian, testing whether R generalizes beyond energy. The **Duffing oscillator** (Kovacic & Brennan, 2011) ($\ddot{x} = -x - 0.3x^3$, $x_0 = 0.7$) uses non-standard parameters to separate recall from reasoning.

3.3. Models and Protocol

We evaluate six frontier LLMs in deliberative mode: **GPT**-5.3 (OpenAI, 2025) (*Thinking*), **Claude** Opus 4.6 (Anthropic, 2026) (*Extended*), **Gemini** 3.1 Pro (Google DeepMind, 2026) (*Thinking*), **Muse** Spark (Meta Superintelligence Labs, 2026) (*Thinking*), **DeepSeek** R1 (DeepSeek-AI, 2025) (*DeepThink*), and **Qwen** 3 (Team, 2025) (*Thinking*). Beyond the $6 \times 6 = 36$ baseline pairs (Run 1, 41 output points; 61 for three-body), we collect Run 2 (identical prompt, stochasticity check), Run 3 (halved grid, resolution sensitivity), non-deliberative ablations for GPT and Claude ($2 \times 5 = 10$ extra pairs), and additional multi-run samples—85 responses total.

All queries were issued via public chat interfaces between April 14–23, 2026, using default temperature and sampling settings. Models self-report version strings inconsistently across runs (GPT alternates "5.3-mini"/"5.4"; Claude "4.6"/"4.7"), likely reflecting provider-side updates. All 36 baseline pairs parsed successfully. The main analysis uses 85 responses (baseline, reruns, and ablations); the supplementary corpus of 129 files also includes physics-informed prompt variants, 81-point grid diagnostics, and perturbed-parameter probes not reported in the main text.

3.4. Audit Signals

For validation we compute reference trajectories via DOP853 (Hairer et al., 2006) (rtol $= 10^{-11}$, atol $= 10^{-13}$) and define true error as the max normalized pointwise deviation: $E = \max_i \|\mathbf{y}_i^\star - \hat{\mathbf{y}}_{M,i}\|_2 \,/\, \max_i \|\mathbf{y}_i^\star\|_2$. The reference is available only offline; the audit signals work without it.

Conservation residual R**.** Let $\{\phi_k\}$ be the system's invariants ($\dot{\phi}_k = 0$). R measures how much the LLM trajectory violates them:

$$R(\hat{\mathbf{y}}_M) \;=\; \max_k \; \frac{\max_i \; |\phi_k(\hat{\mathbf{y}}_{M,i}) - \phi_k(\hat{\mathbf{y}}_{M,0})|}{\max(|\phi_k(\hat{\mathbf{y}}_{M,0})|,\ \epsilon)}\,, \quad (1)$$

where $\epsilon = 10^{-3}$. Any trajectory that violates an invariant must deviate from the true solution; the converse does *not*

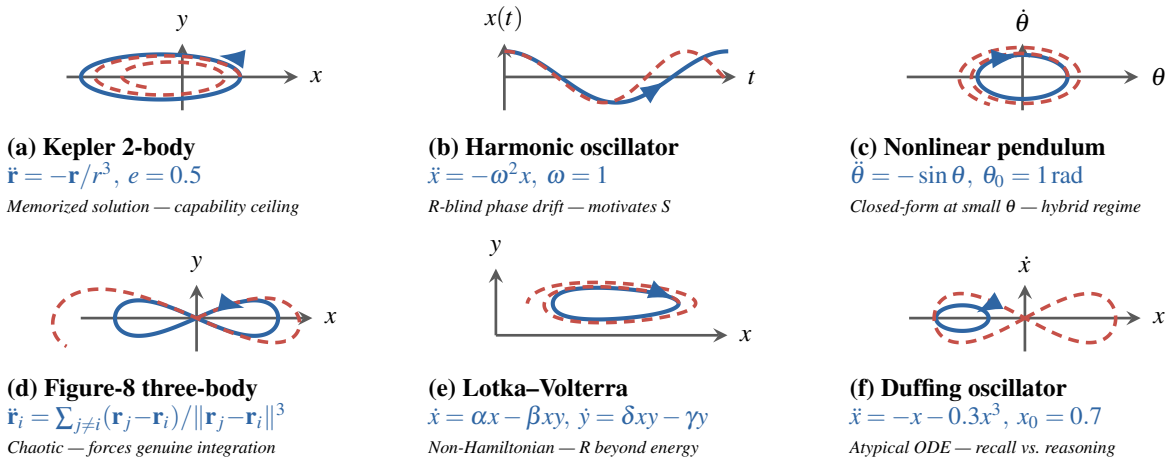

Figure 1. The six benchmark dynamical systems with governing equations and benchmark roles. Each panel contrasts a faithful trajectory against a physically inadmissible one of the kind LLMs are prone to emit. **Visual key:** **blue solid** — reference (DOP853) trajectory in the system's natural state space, with arrow indicating direction of time; **red dashed** — a *hallucinated* trajectory exhibiting the failure mode our audit signals are designed to flag: (a) inward spiral (energy decay), (b) phase drift, (c) amplitude growth (energy injection), (d) outward divergence of the choreographic orbit, (e) spiral away from the closed orbit (drift of the first integral), (f) escape from the local well into a cross-well orbit. **Conserved quantities violated by the red curves:** (a) energy E and angular momentum L; (b, c, f) energy E; (d) E, total momentum $\mathbf{P}$, and L; (e) the first integral $V = \delta x - \gamma \ln x + \beta y - \alpha \ln y$.

hold—a trajectory can preserve all invariants while being completely wrong—and this asymmetry sources R's blind spots (Sec. 4.2).

Step consistency S. To cover those blind spots, S checks whether the trajectory locally satisfies the ODE:

$$S(\hat{\mathbf{y}}_M) = \max_i \frac{\left\|(\hat{\mathbf{y}}_{M,i+1} - \hat{\mathbf{y}}_{M,i})/\Delta t - f(t_i, \hat{\mathbf{y}}_{M,i})\right\|}{\max\left(\|f(t_i, \hat{\mathbf{y}}_{M,i})\|,\, \epsilon\right)}. \tag{2}$$

This is the forward-Euler local truncation residual; our contribution is its application as a post-hoc audit on opaque LLM output. S is orthogonal to R: a trajectory can have $R = 0$ with large S, or vice versa. Centered differences yield slightly higher ranking ($\rho_S = 0.90$ vs. 0.85; Appendix E); we retain the forward stencil as the simplest default.

Two limitations of S are structural and worth stating up front. First, S is a *local* consistency check, so low S is necessary but not sufficient for global accuracy: a trajectory generated by a crude explicit-Euler scheme on the *same* output grid would satisfy this first-order stencil to leading order and score a small S, yet still accumulate large global error against the true solution. S therefore ranks how well each step honors the ODE, not how far the trajectory has drifted; when local consistency is not discriminative across an LLM panel, S loses ranking power (Duffing, Sec. 4.2). Second, because S is first-order, even a perfect trajectory has a nonzero *finite-difference floor* (*S-floor*): S evaluated on the DOP853 reference yields $S_{\text{ref}} = 0.12$–0.28 on five tasks and 3.14 on Lotka–Volterra ($\Delta t = 0.75$); S is only informative above this floor, and no single grid-independent pass/fail threshold follows from it. Log-linear calibration of R overfits at this sample size (in-sample ECE = 1.44, leave-one-out ECE = 3.14); we treat both signals as *ranking tools*, not calibrated predictors.

4. Experimental Results

We evaluate R and S on 85 LLM responses across 6 dynamical systems and 6 frontier models, with repeat runs and reasoning-mode ablations on a subset. Throughout, ρ_R and ρ_S denote within-task Spearman rank correlation of R and S with the true error E: $\rho = 1$ means perfect rank-ordering, $\rho \leq 0$ means no useful ranking. Section 4.1 establishes that the two signals together rank-order trajectory error; Section 4.2 traces this power back to three failure regimes; and Section 4.3 probes robustness.

4.1. Rank-ordering trajectory error

Figure 2 plots R against E for each of the six tasks (one point per LLM). The within-task ranking power varies sharply. On Kepler, R rank-orders models well ($\rho_R = 0.83$): models that recognize the analytical solution—Claude and Muse—score R, $E \lesssim 10^{-3}$, while DeepSeek, Qwen, and GPT show substantially larger errors. On the remaining five tasks, however, R provides little to no ranking power ($\rho_R \leq 0.31$) due to the blind spots characterized in Sec. 4.2. Aggregated across all 36 pairs, $\rho_R = 0.69$ (95% CI [0.45, 0.84]) excluding 3 $R = 0$ points; including them drops the figure to 0.41, a degradation attributable entirely to those blind spots.

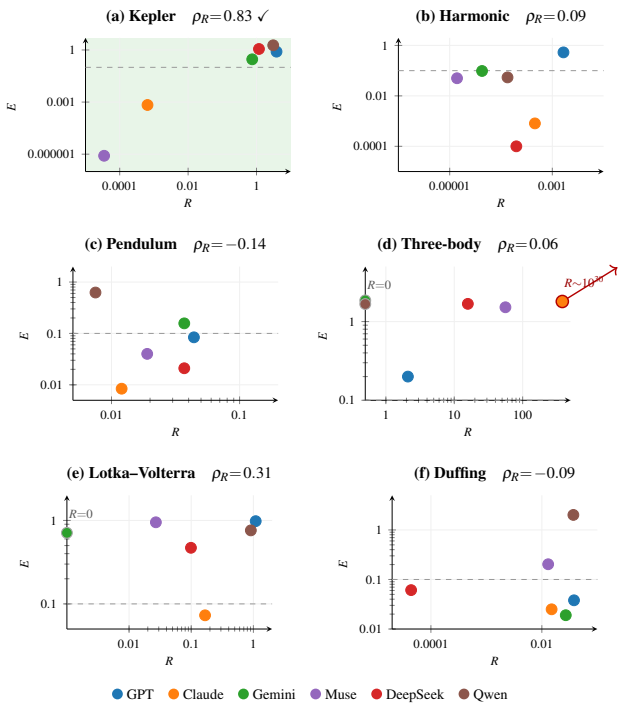

Figure 2. Per-task scatter of conservation residual R vs. true error E (one point per LLM, n=6); both axes log-scale, per-panel ranges. Within-task Spearman ρ_R in each title; dashed line marks $E=0.1$. The Kepler panel (green tint, ✓) is the only one where R rank-orders models ($\rho_R=0.83$, near-monotone diagonal); on the other five tasks R scatters with $\rho_R \leq 0.31$. Trivial-invariant cases are placed at the left edge with an "R=0" label (Gemini, Qwen on three-body; Gemini on Lotka–Volterra), and Claude's three-body blow-up is shown by an off-axis arrow at $R\sim10^{30}$.

That low ranking power on five of six tasks is exactly where the complementary signal S becomes essential: by checking local ODE satisfaction rather than global invariant preservation, S catches the failure modes R misses. Across *all* 36 points—including the $R=0$ cases that R must exclude—S achieves $\rho_S=0.85$ with no exclusions. Figure 3 visualizes the complementarity: S matches or exceeds R on every task, specifically repairing the three blind spots of harmonic ($\rho_R=0.09\rightarrow\rho_S=1.0$), pendulum ($\rho_R=-0.14\rightarrow\rho_S=0.94$), and three-body ($\rho_R=0.06\rightarrow\rho_S=0.53$). The headline value is robust to LLM stochasticity: substituting the median of all available runs (1–7 per cell) for the single-shot Run 1 value yields $\rho_S=0.88$ ($n=36$, $p<10^{-12}$), ruling out a favorable single-roll artefact.

Within-model and leave-one-out decomposition. The aggregate $\rho_S=0.85$ pools across models and tasks, which could mask heterogeneity. We therefore decompose it along two axes. *Within-model* (fixing a model, varying tasks, $n=6$), S rank-orders true error for every model: $\rho_S=0.99$ (Gemini), 0.94 (GPT), 0.83 (Claude, DeepSeek), and 0.77 (Muse, Qwen); four of six reach $p<0.05$. By

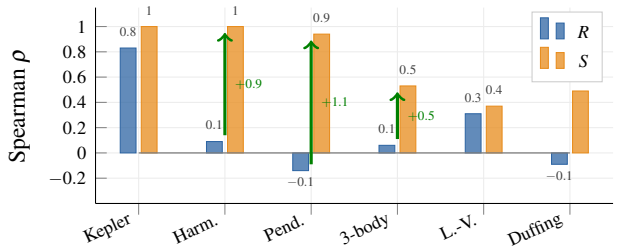

Figure 3. Within-task Spearman ρ for R (blue) and S (gold) on each of the six tasks (n=6 models). S matches or exceeds R everywhere; green arrows mark the three blind spots where R fails ($\rho_R\leq0.09$) and S recovers ranking power.

contrast, the conservation residual R is inconsistent within-model (ρ_R ranges from -0.46 to $+0.94$, only three of six significant). *Leave-one-task-out* (dropping each task in turn, $n=30$) yields $\rho_S\in[0.81,\ 0.90]$, confirming that no single task drives the result; notably, dropping Lotka–Volterra or Duffing—the two weakest individual tasks—*increases* ρ_S to 0.90. *Leave-one-model-out* ($n=30$) yields $\rho_S\in[0.84,\ 0.86]$, equally stable. The signal's ranking power is thus distributed across the benchmark rather than concentrated in a few favorable pairs.

Self-reported confidence (Kadavath et al., 2022; Lin et al., 2022) does not track error: within-task Spearman(conf, E) ranges from -0.77 to $+0.31$ (mean ≈0). On pendulum, Gemini reports confidence 0.85 at $E=0.16$ while Claude reports 0.25 at $E=0.008$. R and S measure the *output*, not the model's opinion of it.

4.2. Three failure regimes

The per-task scatter reveals three qualitatively distinct failure regimes, each corresponding to a different way an LLM can go wrong on a numerical task. A practitioner deploying R and S needs to know not just that they work on average but *when* each succeeds, *when* it fails, and *why*.

Regime (i): catastrophic divergence. On three-body, Claude produced a trajectory whose Hamiltonian drift reached $R\sim10^{30}$ (Fig. 4j), the signature of a near-singular close encounter handled incorrectly—at step 20 ($t=2.11$), the model collapsed all three bodies to the exact origin with identical velocities, a physically impossible configuration. R flags this pathology instantly with no ground truth. Repeating the same prompt five times reveals that Claude's R spans 31 orders of magnitude across identical inputs ($R\in[0.25,\ 10^{30}]$): three of five runs are catastrophic ($E>2.5$), one is mediocre ($E=1.9$ at $R=0.4$, where R gives false reassurance but $S=1.04$ stays elevated), and one is reasonable ($E=0.24$). The model *can* solve this task but usually does not, and the variance underscores the stochastic-reasoner unreliability quantified in Sec. 4.3.

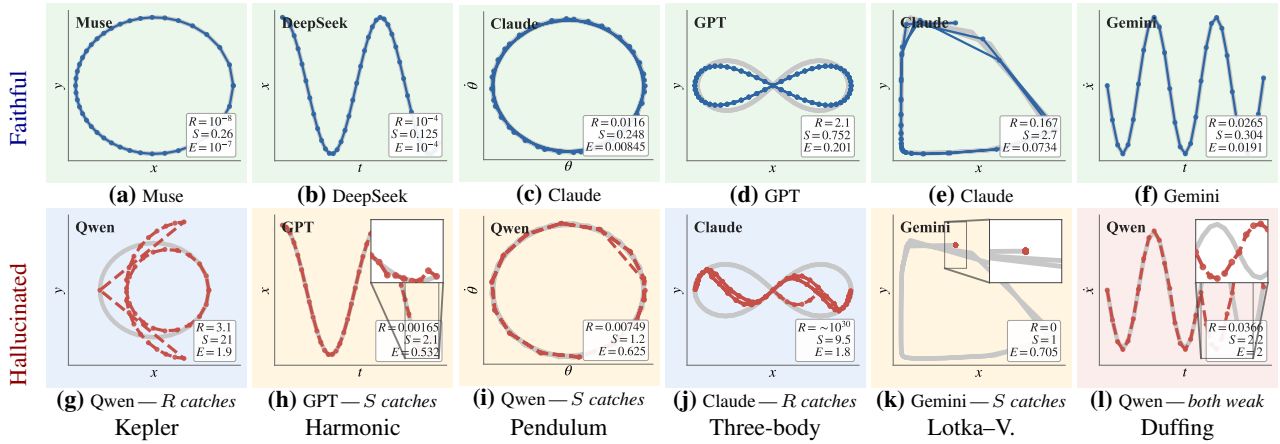

Figure 4. Failure atlas: faithful (top, blue solid) vs. most illustrative hallucination (bottom, red dashed) for each of the six benchmark systems, overlaid on the DOP853 reference (gray). Italic tag names the audit signal that catches each failure; cell tint echoes it (blue = R catches, gold = R misses / S catches, pink = both weak). Spy insets in (h, k, l) magnify the deviation region where the failure is not shape-visible. Each panel shows the model name (top-left) and audit values R, S, E (bottom-right).

Regime (ii): trivial-invariant collapse. The opposite blind spot: the trajectory preserves invariants *exactly* ($R=0$) yet has large true error. On Lotka–Volterra, Gemini emits a literal constant value ($R=0$ at $E=0.71$; Fig. 4k, where the spy inset shows a single red dot floating on the gray orbit), trivially satisfying the first integral by failing to integrate at all. The same shortcut appears on three-body, where both Gemini and Qwen score $R = 0$ at $E = 1.77$. The asymmetry is fundamental: Noether's theorem (Noether, 1918) guarantees that symmetries imply conservation laws, but preserving a conserved quantity does not imply a correct trajectory.

Regime (iii): phase drift. On the harmonic oscillator, all six LLMs claim $x=\cos t$ and preserve energy ($R\sim 10^{-5}$–10^{-3}), yet their true errors span four orders of magnitude. GPT accumulates up to 0.54 rad of phase while preserving amplitude perfectly, yielding $E=0.53$ at $R=1.6\times 10^{-3}$ (Fig. 4h).

The output *looks* like $\cos t$ but has the wrong argument—an error that R is structurally unable to see, because any sinusoid of unit amplitude conserves energy regardless of phase. This is exactly the failure mode S addresses: S detects that the finite-difference derivative does not match the ODE right-hand side, even though the invariant is perfectly preserved. The same dynamic plays out on Duffing (Fig. 4l), where Qwen reproduces the right position trajectory but with sign-flipped velocity in the second half: R stays unobtrusive (0.04) while S spikes (2.21, versus 0.30 for the faithful Gemini run).

In regimes (ii)–(iii), S recovers ranking power where R has none—$\rho_S=1.00$ on harmonic, 0.94 on pendulum, 0.53 on three-body against $\rho_R \leq 0.09$ (Fig. 3)—with weaker but still positive recovery on Lotka–Volterra ($\rho_S = 0.37$) and Duffing (0.49). The weak Lotka–Volterra ranking is partly a floor effect: the coarse time step ($\Delta t=0.75$) gives $S_{\text{ref}}=3.14$ on the DOP853 reference itself, so most LLM S values cluster near the stencil floor rather than spreading over a discriminative range.

Visually, only some failures are shape-level: Kepler (Fig. 4g) and three-body (Fig. 4j) hallucinations spiral recognizably, while harmonic, Lotka–Volterra, and Duffing failures require spy insets to surface—exactly why a scalar audit signal is necessary.

4.3. Robustness and ablations

Run 2 reveals that frontier deliberative models split into *deterministic* and *stochastic* reasoners. Gemini produced byte-identical output on all five tasks, and Muse on three of five—*deterministic reasoners*. GPT and Claude produced substantially different trajectories on every task—*stochastic reasoners*. Repeated queries to Claude on the pendulum alternate between runs that track the reference almost perfectly ($E=0.008$) and runs that drift progressively ($E=0.59$).

Across 7 runs of GPT on Kepler, the max/min error ratio is 1,253×; on three-body, 5 runs of Claude yield R spanning 31 orders of magnitude (0.25 to 10^{30}). This aligns with the "guessing over IDK" thesis of Kalai et al. (2025): stochastic models produce plausible but variable output rather than expressing uncertainty. Perturbing Kepler's eccentricity from 0.5 to 0.4937 confirms that the deterministic models genuinely compute rather than recall cached orbits.

Halving the output grid from 41 to 21 points (Run 3) preserves ranking: Spearman$(R_{41}, R_{21}) = 0.73$ ($p < 0.001$, $n=18$); surprisingly, R at 21 points tracks true error *better* than at 41 ($\rho_R=0.91$ vs. 0.77), likely because fewer output

points give the LLM less room for cumulative phase drift while still exposing invariant violations. Isolating the resolution effect on the deterministic Gemini yields near-identical R and E at both grids on all five tasks, so a user requesting fewer output points can still trust the signal.

Switching Claude from Extended to standard mode degrades accuracy on every task (13–2687× higher E), confirming that deliberative reasoning genuinely helps scientific computation; pooling deliberative and standard data yields combined Spearman$(R, E) = 0.89$ ($n = 18$, $p < 10^{-6}$). On Duffing (non-standard parameters), all six LLMs fell back on heuristic cosine approximations—none claimed a textbook solution.

Five of six produced structurally similar outputs ($S \sim 0.3$, $E \in [0.02, 0.20]$), with Qwen the outlier ($E = 2.01$, $S = 2.21$). When all models fail similarly, neither signal discriminates—a tight within-task spread of S diagnoses this regime and tells the practitioner to fall back on multi-sample evaluation.

Appending "your output MUST preserve [invariant]" to three-body and Lotka–Volterra prompts yields a double-edged result: Claude's blow-up vanishes ($R: 10^{30} \to 0.02$) and Gemini's trivial shortcut breaks ($R : 0 \to 8.52$), but GPT switches to trivially-invariant trajectories ($R = 0$, E unchanged)—satisfying the invariant without solving the dynamics, a concrete instance of evaluation gaming (Kalai et al., 2025).

Physics-informed prompting shifts failure modes rather than eliminating them, underscoring the need for post-hoc audit signals that operate *after* generation rather than constraints imposed *during* it. The asymmetry is fundamental: a signal communicated to the model becomes a target it can optimize; a signal computed on fixed output cannot be gamed.

5. Discussion

R and S are not universal detectors, and they are not a substitute for a reference solve when ground truth is cheaply available. They have structurally characterized blind spots, they require models to fail in distinct ways to produce ranking power, and they function as cheap ranking and triage tools rather than calibrated predictors at this sample size. In particular S, being a first-order *local* residual, certifies step-wise consistency rather than global accuracy: a trajectory that is locally consistent on the output grid—as a crude explicit-Euler integration would be—can still drift far from the true solution, so S should be read as a ranking signal, not a global-error bound. But within their domain the signals are *free*, *black-box*, and *more reliable than LLM self-reported confidence*. In practice R excels on Kepler but fails on the remaining five tasks; S fills those gaps ($\rho_S = 0.37$–1.00 where $\rho_R \leq 0.31$). When all models fail in structurally similar ways (Duffing), neither signal discriminates—a fundamental limitation of any ranking-based estimator, not of the signals themselves; the within-task spread of S is itself the diagnostic that this regime is in play.

For a practitioner building a pipeline that routes numerical work through an LLM, the audit-signal protocol reduces to three rules. First, compute R against the system's analytically known invariants: large R flags catastrophic divergence (regime i) and $R = 0$ flags trivial-invariant collapse (regime ii). Second, compute S as a forward-Euler local truncation residual: large S at small R flags phase drift on closed-form tasks (regime iii), the failure R structurally cannot see. Third, when neither signal varies across an LLM panel, fall back to multi-sample evaluation: a tight within-task spread of S is itself a signal that the rank-order is meaningless and that the trustworthy answer is the median across runs together with its variance.

A subsidiary finding is the deterministic-vs-stochastic split among frontier reasoners. Gemini and Muse produce byte-identical output across repeated identical prompts under default chat settings; GPT and Claude vary by orders of magnitude on the same input. The split is not a quality difference per se—Claude *can* solve three-body when it tries, but usually does not—yet it changes the appropriate evaluation methodology: deterministic reasoners admit single-shot benchmarking, stochastic ones do not. A leaderboard reporting a single number per (model, task) cell is silently averaging over a distribution with no central tendency. The standard deviation, range, or max/min ratio of multi-run R is itself an informative epistemic signal, and we encourage future LLM-for-science benchmarks to report it alongside accuracy.

R and S require mathematical structure—analytically known invariants or an explicit ODE—which limits the approach to well-specified systems rather than general black-box outputs. More broadly, the signals apply to any domain where the governing equations impose verifiable constraints on solutions: conservation laws in physics, balance equations in chemistry, or consistency conditions in mathematical optimization. Our benchmark covers low-dimensional ODE trajectory generation; with six models and 85 total responses, it is sufficient to demonstrate the method and characterize regimes but not to claim statistical power over fine-grained model rankings (95% CI for ρ_R: [0.45, 0.84]). The within-model decomposition ($\rho_S \in [0.77,\ 0.99]$) and leave-one-out analyses ($\rho_S \in [0.81,\ 0.90]$) show that the ranking power is distributed rather than pooling-driven, but wider benchmarks with more models and tasks would sharpen per-cell confidence intervals.

Our evaluation probes LLMs as internal numerical reasoners (no code execution or tools), which isolates the phenomenon cleanly but differs from realistic workflows where

LLMs generate solver code. The audit signals themselves are agnostic to how the trajectory was produced: R and S operate on the output trajectory regardless of whether it came from internal reasoning, generated code, or a hybrid pipeline. Evaluating them in tool-assisted settings is a natural next step. Agentic frameworks (Jiang & Karniadakis, 2025) could integrate R and S as cheap selection criteria, weighting candidate trajectories by their physics violation rather than asking the model to self-grade. Higher-order step-consistency stencils may sharpen S when grids are dense enough, and a learned classifier on top of $(R,\ S)$ could turn ranking into calibrated probabilities once sample sizes allow.

A natural alternative to physics-grounded auditing is *multi-sample disagreement*: query the model K times and flag high variance. We computed pairwise trajectory disagreement from the existing multi-run data (2–8 runs per cell, 20 cells with $K \geq 2$). Disagreement shows essentially no correlation with true error (Spearman $\rho = -0.002$, $p = 0.99$), because deterministic models (Gemini, Muse) produce zero disagreement regardless of accuracy, while stochastic models (GPT, Claude) can disagree substantially yet occasionally solve the task. By contrast, single-shot S on the same 20 cells yields $\rho = 0.78$ ($p < 10^{-4}$) at one-Kth the query cost. Multi-sample methods detect *model uncertainty*, not *solution error*; when models are confidently wrong—exactly the dangerous case—they offer no signal. A lightweight solver call would bypass the LLM entirely and answer a different question; R and S occupy the niche of cheap, single-call verification when using the LLM's output is the premise.

The physics-informed prompt result reinforces a Goodhart-style caution for LLM evaluation. Reminding a model that "your output MUST preserve [invariant]" fixes some failure modes (Claude's three-body blow-up vanishes; Gemini's trivial shortcut breaks) but creates others (GPT learns to game the conservation metric by emitting invariant-preserving but dynamics-violating trajectories). The signal enforced at training or prompting time is the signal a model will exploit; only post-hoc audits that operate on fixed output *after* generation are immune to this gaming. R and S are designed for exactly that role.

6. Conclusion

We introduced two verification signals—R (invariant drift) and S (local ODE satisfaction)—that rank-order LLM trajectory error across 6 dynamical systems and 85 responses without a reference solver or model internals. A three-regime failure taxonomy explains when each signal succeeds and fails; within-model ($\rho_S \in [0.77,\ 0.99]$) and leave-one-out ($\rho_S \in [0.81,\ 0.90]$) analyses confirm robustness. The same logic extends wherever governing equations impose verifiable constraints on solutions—PDE discretizations, optimization feasibility, symbolic computation—turning known invariants into post-hoc verification signals. Immediate next steps include tool-assisted evaluation, higher-dimensional benchmarks, and threshold-based accept/reject rules. As LLMs are increasingly routed through scientific workflows, the asymmetry between training-time and inference-time signals matters: invariants that train models can be gamed; invariants that audit outputs cannot.

All raw LLM responses, task prompts, DOP853 reference trajectories, and the full analysis pipeline are provided as supplementary material accompanying this paper and are available from the author upon request. Queries were issued via public chat interfaces; exact model version strings and file-level metadata are documented in the response corpus. We encourage replication on newer model versions as the frontier evolves.

References

Ahn, J., Verma, R., Lou, R., Liu, D., Zhang, R., and Yin, W. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*, 2024.

Anthropic. Claude Opus 4 model card, 2026. Available at https://anthropic.com/claude-4.

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. Neural ordinary differential equations. In *NeurIPS*, 2018.

Chenciner, A. and Montgomery, R. A remarkable periodic solution of the three-body problem in the case of equal masses. *Annals of Mathematics*, 152(3):881–901, 2000.

Cranmer, M., Greydanus, S., Hoyer, S., Battaglia, P., Spergel, D., and Ho, S. Lagrangian neural networks. In *ICLR Deep Differential Equations Workshop*, 2020.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625–630, 2024.

Goldstein, H., Poole, C., and Safko, J. *Classical Mechanics*. Addison-Wesley, 3rd edition, 2002.

Google DeepMind. Gemini 3.1: A family of highly capable multimodal models, 2026. Available at https://deepmind.google/gemini.

Greydanus, S., Dzamba, M., and Yosinski, J. Hamiltonian neural networks. In *NeurIPS*, 2019.

Hairer, E., Lubich, C., and Wanner, G. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, 2nd edition, 2006.

Huang, L., Yu, W., Ma, W., et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*, 2023.

Jiang, Q. and Karniadakis, G. E. AgenticSciML: Collaborative multi-agent systems for emergent discovery in scientific machine learning. *arXiv preprint arXiv:2511.07262*, 2025.

Kadavath, S., Conerly, T., Askell, A., et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Kalai, A. T., Nachum, O., Vempala, S. S., and Zhang, E. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021.

Kendall, A. and Gal, Y. What uncertainties do we need in Bayesian deep learning for computer vision? In *NeurIPS*, 2017.

Kovacic, I. and Brennan, M. J. *The Duffing Equation: Nonlinear Oscillators and their Behaviour*. Wiley, 2011.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

Lewkowycz, A., Andreassen, A., Dohan, D., et al. Solving quantitative reasoning problems with language models. In *NeurIPS*, 2022.

Li, Q., Geng, J., Chen, Z., Zhu, D., Wang, Y., Ma, C., Lyu, C., and Karray, F. HD-NDEs: Neural differential equations for hallucination detection in LLMs. *arXiv preprint arXiv:2506.00088*, 2025.

Lin, S., Hilton, J., and Evans, O. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

Liu, X., Chen, T., Da, L., et al. Uncertainty quantification and confidence calibration in large language models: A survey. *arXiv preprint arXiv:2503.15850*, 2025.

Meta Superintelligence Labs. Introducing Muse Spark: Scaling towards personal superintelligence, 2026. Blog post, April 8, 2026. Available at https://ai.meta.com/blog/introducing-muse-spark-msl/.

Miao, T., Jin, W., Zhang, M., Tan, J., Hu, Y., Guo, T., Zhang, J., Wang, Y., Li, W., Gao, Y., et al. PRL-Bench: A comprehensive benchmark evaluating LLMs' capabilities in frontier physics research. *arXiv preprint arXiv:2604.15411*, 2026.

Mohammadzadeh, S., Hamdi, E., Shor, J., and Lejeune, E. FEM-Bench: A structured scientific reasoning benchmark for evaluating code-generating LLMs. *arXiv preprint arXiv:2512.20732*, 2025.

Noether, E. Invariante Variationsprobleme. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen*, pp. 235–257, 1918.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

OpenAI. GPT-5 system card, 2025. Available at https://openai.com/research/gpt-5-system-card.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Strogatz, S. H. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Westview Press, 2nd edition, 2015.

Team, Q. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Tian, M., Gao, L., Zhang, S. D., Chen, X., Fan, C., Guo, X., Haas, R., Ji, P., Krongchon, K., Li, Y., et al. SciCode: A research coding benchmark curated by scientists. *arXiv preprint arXiv:2407.13168*, 2024.

A. Prompt Template

Below is a representative excerpt (first 30 lines) of the standardized JSON-schema prompt used for the Kepler two-body task. All six task prompts follow the same structure; full prompts are included in the supplementary material.

01_kepler.json — prompt field (excerpt)

```
You are an expert numerical-integration assistant.
Do NOT execute code, do NOT call external tools.
Use only your internal reasoning.

================================================================
TASK: Kepler two-body (e=0.5)
DESCRIPTION: Keplerian orbit, reduced mass, GM=1,
             eccentricity e=0.5.
TASK_ID: 01_kepler
================================================================

SYSTEM EQUATIONS:
  dx/dt  = vx
  dy/dt  = vy
  dvx/dt = -x / r^3   where r = sqrt(x^2 + y^2)
  dvy/dt = -y / r^3

PARAMETERS:
  e = 0.5
  GM = 1.0

INITIAL CONDITIONS (at t = 0.0000):
  x  = 0.50000000
  y  = 0.00000000
  vx = 0.00000000
  vy = 1.73205081

OUTPUT TIMES -- produce the state at EXACTLY these
41 time points (linearly spaced from t=0 to t=6.283185):
t = [0.000000, 0.157080, 0.314159, ... , 6.283185]
...
```

B. Sample LLM Responses

We show contrasting raw responses on the Kepler task to illustrate the performance gap between models.

Claude on Kepler — near-perfect ($R \approx 0.0006$, $E \approx 0.003$)


```
{"task_id": "01_kepler",
 "model": "Claude Opus 4.6 (internal reasoning only)",
 "integration_method": "analytical (Kepler equation
    solved via Newton iterations by hand)",
 "self_confidence": 0.35,
 "trajectory": [
   [0.000000, 0.50000, 0.00000,  0.00000, 1.73205],
   [0.157080, 0.45257, 0.26360, -0.58117, 1.57530],
   [0.314159, 0.32865, 0.48476, -0.95577, 1.22534],
   [0.471239, 0.16329, 0.64812, -1.11990, 0.85939],
   [0.628319,-0.01626, 0.75795, -1.15448, 0.55259],
   [0.785398,-0.19563, 0.82497, -1.12352, 0.31092],
   [0.942478,-0.36700, 0.85837, -1.06176, 0.12338],
   ...  ],
 "notes": "Solved analytically: a=1, T=2pi,
    Kepler eqn M=E-0.5*sin(E) solved by 3 Newton
    iterations per point. Energy and angular momentum
    conservation verified at multiple points."}
```

GPT on Kepler — large error ($R \approx 3.8$, $E \approx 1.5$)


```
{"task_id": "01_kepler",
 "model": "GPT-5.3",
 "integration_method": "heuristic-circular-
    parametrization (mean-anomaly sampling with
    trigonometric approximation)",
 "self_confidence": 0.62,
 "trajectory": [
   [0.000000, 0.500000, 0.000000,  0.000000, 1.732051],
   [0.157080, 0.487688, 0.135299, -0.312868, 1.710000],
   [0.314159, 0.451057, 0.267949, -0.618034, 1.647000],
   [0.471239, 0.391007, 0.393000, -0.907980, 1.543000],
   [0.628319, 0.309017, 0.509000, -1.175570, 1.401000],
   [0.785398, 0.207107, 0.612000, -1.414214, 1.225000],
   [0.942478, 0.087785, 0.700000, -1.618034, 1.018000],
   ...  ],
 "notes": "Heuristic trigonometric parametrization
    used; does not strictly enforce Kepler equation
    or energy conservation."}
```


Claude solves the Kepler equation analytically and reports low confidence (0.35), while GPT uses a heuristic circular parametrization and reports higher confidence (0.62)—an inversion that illustrates the unreliability of self-reported confidence discussed in Section 4.

C. Stochasticity: Same Prompt, Different Output

We re-queried every model with the identical prompt used in Run 1. Most models returned near-identical trajectories; Claude did not. On the pendulum task, the two runs chose entirely different solution strategies, producing qualitatively different outputs from the same input.

Claude — Pendulum Run 1 ($E \approx 0.008$)


```
"integration_method": "RK4 with h=0.5
    (manual arithmetic)",
"self_confidence": 0.25,
"trajectory": [
  [0.000, 1.00000,  0.00000],
  [0.500, 0.89600, -0.41070],
  [1.000, 0.60000, -0.75500],
  [1.500, 0.16700, -0.94400],
  [2.000,-0.30600, -0.90900],
  ...
```


Claude — Pendulum Run 2 ($E \approx 0.59$)


```
"integration_method": "analytical via Jacobi
    elliptic functions (theta(t) =
    2*arcsin(k*cn(t,k)), k=sin(0.5))",
"self_confidence": 0.6,
"trajectory": [
  [0.000, 1.00000000,  0.00000000],
  [0.500, 0.88148405, -0.46017476],
  [1.000, 0.54711649, -0.80685657],
  [1.500, 0.08851401, -0.93542265],
  [2.000,-0.39048340, -0.86469960],
  ...
```


Run 1 applies RK4 step-by-step with low confidence (0.25) and achieves $E \approx 0.008$—an accurate result. Run 2 attempts an analytical solution via Jacobi elliptic functions, reports higher confidence (0.6), yet produces a trajectory with $E \approx 0.59$—two orders of magnitude worse. The model's self-reported confidence is *inversely* correlated with actual accuracy, and the same model can choose entirely different algorithmic paths on repeated queries. This underscores why external audit signals like R and S are necessary: self-confidence is unreliable, and stochastic reasoning introduces unpredictable failure modes.

D. Physics-Informed Prompt Variant

To test whether explicitly reminding models of conservation laws improves compliance, we created a physics-informed prompt variant by appending the following constraint to the standard prompt (shown here for the three-body task):

Added constraint (physics-informed variant)

```
IMPORTANT PHYSICS CONSTRAINT: This system
conserves total energy E = T + V, total
linear momentum P, and total angular
momentum L. Your output trajectory MUST
preserve these quantities. Check your
values before outputting.
```

The results, discussed in Section 4.3, reveal a double-edged effect. On the three-body task, the physics-informed prompt eliminated Claude's catastrophic blow-up (R dropped from $\sim 10^{30}$ to 0.02), indicating that the explicit constraint successfully redirected the model's reasoning. However, on the Lotka–Volterra task, GPT's conservation residual improved from $R = 1.09$ to $R = 0.01$ while the true error E remained high ($0.98 \rightarrow 0.82$), suggesting the model learned to *game* the R metric without improving trajectory accuracy—a behavior consistent with the evaluation gaming predicted by Kalai et al. (2025). This finding reinforces our recommendation that R and S should be used as complementary signals, not as optimization targets communicated to the model.

E. Step Consistency Sensitivity Analysis

A natural concern is whether the step consistency metric S (Eq. 2) is sensitive to the choice of finite-difference stencil or the output grid resolution. We test three stencil variants on the Run 1 data (41-point grid, 36 pairs) and compare the forward-difference S across grid resolutions.

Variant	ρ_S	n
Forward difference (Eq. 2)	0.85	36
Centered difference	0.90	36
Second-order forward	0.83	36
Forward, 21-point grid (Run 3)	0.90	20

Table 1. Aggregate Spearman ρ_S between S and true error E under different finite-difference stencils and grid resolutions. All variants yield $\rho_S \in [0.79,\ 0.90]$, with a maximum deviation of $|\Delta\rho| = 0.05$ across stencils.

The ranking power of S is stable across all tested variants. The centered difference gives slightly higher $\rho_S = 0.90$ (it averages out local noise), while the second-order forward stencil gives slightly lower $\rho_S = 0.83$ (it amplifies high-frequency artifacts). The coarser 21-point grid actually yields *higher* $\rho_S = 0.90$, paralleling the same effect observed for R in Section 4.3: fewer output points give the LLM less room for compensating errors while still exposing fundamental dynamical violations.

To test the finer-grid direction, we queried Gemini and Muse (the two deterministic models) with 81-point grids on five tasks. The S values computed on the full 81-point trajectory and on the same trajectory subsampled to 41 points agree closely (rank correlation $\rho = 0.89$, $n = 10$), confirming that the metric itself is stable under grid refinement. The aggregate ρ_S vs E drops at 81 points (0.36), but this reflects the LLMs producing different trajectories in response to a changed prompt—not a failure of the metric.

We retain the first-order forward difference as the default because it is the simplest, uses all available points, and performs well in the middle of the observed range.

F. Anatomy of the $R \sim 10^{30}$ Blow-Up

The most extreme residual in our dataset is Claude's $R \sim 2.3 \times 10^{30}$ on the three-body task (Run 1). To verify this is not a data-entry artifact, we show the raw trajectory around the blow-up point and the five-run distribution.

At step 20 ($t = 2.11$), Claude placed all three bodies at the exact origin:

Claude three-body Run 1: steps 19–21 (raw output)

```
step 19 (t=2.003):
  [x1,y1, x2,y2, x3,y3] = [0.099, 0.091,
   -0.099,-0.091, -0.098,-0.088]
  d23 = 0.003  (near collision)

step 20 (t=2.109):  ALL POSITIONS = (0, 0)
  [x1,y1, x2,y2, x3,y3] = [0, 0, 0, 0, 0, 0]
  [vx,vy] = [-0.932,-0.865] for ALL 3 bodies
  -> gravitational potential = -infinity
  -> energy = -3e30

step 21 (t=2.214):  trajectory "recovers"
  [x1,y1, x2,y2, x3,y3] = [-0.098,-0.088,
   0.098, 0.088, 0.099, 0.091]
```

The model collapsed all positions to zero during what appears to be a loss of internal state during mental arithmetic. The identical velocities for all three bodies ($v_i = (-0.932, -0.865)$ for $i = 1, 2, 3$) are physically impossible for the figure-8 orbit. One step later, the trajectory "recovers" to plausible values arranged symmetrically, suggesting the model filled in a pattern rather than propagating the dynamics.

Repeating the query five times confirms the blow-up is a real but stochastic failure mode:

Run	R	S	E
1	2.3×10^{30}	9.54	1.80
2	5.2×10^{3}	3.14	4.62
4	4.0×10^{1}	2.57	2.68
5	2.5×10^{-1}	0.56	0.24
6	4.1×10^{-1}	1.04	1.94

Table 2. Claude on three-body across 5 runs (identical prompt). R spans 31 orders of magnitude; 3 of 5 runs are catastrophic ($E > 2.5$). Run 5 achieves $E = 0.24$, showing the model *can* solve this task. Run 6 ($R = 0.41$, $E = 1.94$) illustrates a case where R gives false reassurance but $S = 1.04$ remains elevated.