NOT EVERY DEFECT IS IN THE LAW

Anonymous authors
Paper under double-blind review

ABSTRACT

Systems for discovering physical laws usually search over equations while fixing the experimental representation: what quantities exist, what instruments measure, and what experimenters control. We show that defects can instead lie elsewhere in the theory, and that law search can hide this distinction even with a perfect fit. We call this failure mode *absorption*: a discrepancy originating in the experimental representation is reproduced by rewritten physical laws. In a mass–spring system with a delayed sensor, two law edits reproduce the recorded channel to 5.1×10^{-16} relative error while leaving the incorrect measurement model unchanged. For linear systems, absorption occurs when the theory matches the apparatus's direct input–output feedthrough, has enough states to realize the observed dynamics, and can express those dynamics with its declared dimensions and parameters. Across 100 drawn linear systems, law-only accounts exist in all 100 cases. Additional drives resolve none of these ambiguities, whereas an added measurement resolves every case; independently constructed systems and established law-discovery methods show the same qualitative behavior. These results explain the title's central claim: **not every defect is in the law**. A perfect law fit can recover observable behavior without identifying its physical origin, making the choice of what to measure part of scientific discovery itself.

1 INTRODUCTION

Consider a mass on a spring observed through a delayed sensor. A law-discovery system sees the applied force and recorded signal and searches for equations that reproduce the data. It can fit the recording perfectly by placing the delay in the mass dynamics even when the delay originates entirely in the sensor. This exposes a common assumption in physical-law discovery: the equations are treated as uncertain while the experimental representation—which quantities exist, which are controlled, what values they may take, and what instruments report—is fixed. A successful fit can therefore recover the observed behavior while assigning its physical origin to the wrong part of the theory. We make these alternatives explicit by representing concepts, physical relations, observation models, parameters, and constraints as editable components of a theory. This reveals a failure mode we call **absorption**: an effect originating outside the physical laws can be reproduced exactly by rewriting the laws.

We therefore ask: *when can an incorrect physical law reproduce the same observations as the correct repair elsewhere in the theory?* For linear systems, we show that absorption is governed by three structural conditions: the accounts must have the same direct input–output feedthrough, the theory must contain enough states to realize the observed dynamics, and the required dynamics must be expressible using its declared physical dimensions and parameters. These conditions make absorption a property of what the experiment observes, rather than a peculiarity of a particular search procedure. They also predict the experiment needed to resolve it. If two accounts have the same transfer function to the recorded channel, every possible drive produces the same observation on that channel. Additional excitation therefore provides no separating evidence. A measurement of a quantity on which the accounts make different predictions can resolve the ambiguity.

We test this prediction through a closed sequence of experiments. With exact recordings and a one-edit budget, defects in observation models, admissible values, experimental roles, and physical laws are localized to the component in which they were introduced. Allowing additional law edits changes the result: observation errors can be absorbed into the dynamics, producing physically different accounts with identical recorded behavior. In the mass–spring example, a sensor repair and

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.

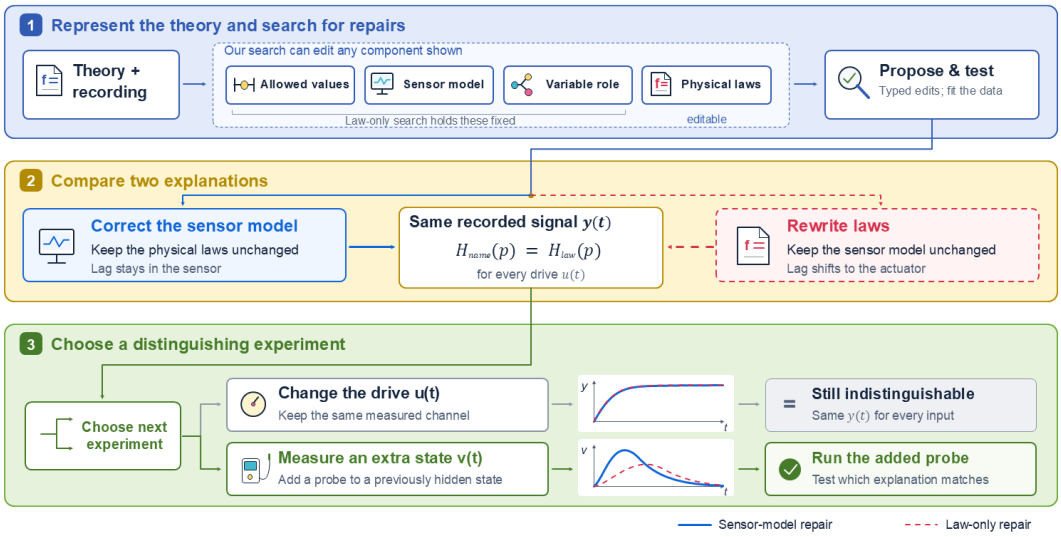


Figure 1: **Absorption and how to resolve it.** An effect originating outside the physical laws can sometimes be reproduced exactly by rewriting the laws. When the resulting accounts have the same transfer function from the experimental drive to the recorded channel, every drive produces the same observation. Measuring a quantity on which their predictions differ can separate the explanations.

a law-only account agree on the recorded channel to 5.1×10^{-16} relative error. Across 100 drawn linear systems, law-only accounts exist in all 100 cases; changing the drive resolves none, while adding a second measurement resolves every case. The same qualitative behavior appears in an independently written thermal simulator and in established law-discovery methods whose inferred laws change when only the instrument changes. These results establish a limit on what a successful law fit can mean: the data may determine the observable input–output behavior while leaving its physical location unresolved. Scientific-discovery systems therefore need to determine whether the available measurements identify where an inferred effect belongs before interpreting a fitted equation as a physical law.

2 Representing and Repairing a Theory

Absorption can only be detected if the search is allowed to consider explanations outside the physical laws. We therefore represent a theory as

$$T = (C, R, O, \Theta, A),$$

with five editable components that connect physical dynamics to an experiment.

The **concepts** C are the quantities in the theory. Each concept has a physical dimension, a set of admissible values, and an experimental role: it may be a state evolved by the theory, an input controlled by the experimenter, or a quantity derived from other concepts. Concepts may also be grounded to measurement channels. The **relations** R specify how these quantities depend on one another. Each relation is an executable expression tree defining either an algebraic quantity or the time derivative of a state. The **observation model** O specifies what each recorded channel reports as an expression over the concepts. The **parameters** Θ have values, physical dimensions, and admissible sets, while the **constraints** A specify conditions that a valid theory must satisfy.

This representation separates physical dynamics from the experimental interface that exposes them. A disagreement with data can therefore be assigned to a law, to what an instrument measures, to the role of a quantity, or to another explicit component of the theory. This separation is what allows us to ask whether a law rewrite and a repair elsewhere in the theory can explain the same recording.

Table 1: **Editable components of a theory.** The repair operations follow from the fields of the five record types. Rows denote theory components and columns denote possible edits, with modifications separated by field. **Bold** marks the operations used for the defect classes studied in the main experiments; the law control uses `EditRelation`. A field followed by a dash does not define a valid standalone operation (Appendix B).

component	insert	delete	modify, by field
concepts C	InsertConcept	DeleteConcept	SetDimension, SetGrounding, **SetRole**, **EditDomain**; `name` —
relations R	InsertRelation	DeleteRelation	EditRelation, SetOutput; `name`, `wrt` —
observations O	InsertObservation	DeleteObservation	**EditObservation**; `name` —
parameters Θ	InsertParameter	DeleteParameter	SetParameterDimension, Tune, EditParameterDomain; `name` —
constraints A	InsertConstraint	DeleteConstraint	EditConstraint; `name` —

What can be changed? The repair space follows directly from the representation. Each theory component is a typed record whose fields can be inserted, deleted, or modified. Field-level changes remain distinct: changing a concept's admissible values, for example, is a different operation from changing its physical dimension or experimental role.

Enumerating these valid changes gives the repair operations in Table 1. This provides a common search space in which a change to a physical law can compete directly with a change to the observation model, an admissible set, or a concept's role. Appendix B explains the record fields that do not define valid standalone repairs.

Which repairs explain a recording? For each candidate repair, we execute the declared experiments and compare the predicted measurements with the observed recording. Candidates that reproduce the recording are retained as possible *accounts* of the data.

When several accounts explain the same observations, we rank them by how much they change the original theory. We measure this change as a tree distance between canonical representations. Canonicalization removes superficial differences in expression form, so equivalent rewriting cannot appear artificially cheap. Because distinct canonical theories can still be observationally equivalent, the resulting theory counts are upper bounds on the number of experimentally distinguishable explanations (Appendix I).

This distinction between *fitting the recording* and *choosing among explanations* will become important below. Two repairs can reproduce exactly the same measurements while assigning the discrepancy to different parts of the theory.

How are candidate repairs proposed? The generator proposes changes from the structure of the theory and the observed discrepancy. For relation edits, it follows three steps. *Localize* identifies where the discrepancy can enter the dependency structure of the recorded channel. *Type* determines the physical dimension required at that location. *Fill* enumerates expressions of the required dimension from quantities already declared in the theory.

Other repair families use the same principle. A missing rate law can suggest a role change, an unused quantity can suggest a structural edit, and compatible physical dimensions can identify candidate substitutions. The generator therefore uses properties computed from the theory to determine which repairs are plausible.

The current generator emits 16 of the 21 valid operations in Table 1; the remaining operations are valid parts of the representation but are not yet enumerable by the present proposal mechanism. Appendix A and Appendix G describe the evaluation used to verify that these proposals depend on the theory and its computed properties. With the representation and repair space defined, we can now ask the first empirical question: when a theory disagrees with a recording, can the data identify *which component* is wrong?

3 WHICH PART OF THE THEORY IS WRONG?

The representation above allows discrepancies to be assigned to different parts of a theory. We first ask the simplest question: when exactly one component is wrong, can the recording identify which component needs to change?

For each experiment, we draw a system from a predefined distribution, introduce a defect into one component of its theory, and generate a recording from the resulting apparatus. We then ask which one-edit repairs can reproduce that recording. If the data contain enough information to localize the defect, the successful repair should act on the component in which the defect was introduced.

We test this across three system families with different dynamics (Appendix D). The first is a mass–spring system measured through a delayed sensor (Figure 2). The second contains two coupled reservoirs whose levels relax toward one another, again observed through a delayed sensor. The third is a toxicokinetic system in which a compound is introduced and eliminated, produces a saturating damage response, and is measured by a plate reader with delay and saturation. All three use the same five-component theory representation, allowing the same defect types and repair operations to be tested across qualitatively different systems.

Table 2: **Can a single repair identify the defective component?** Each row introduces a defect into one component of the theory and reports how often the recording can be reproduced either by repairing that component or by changing one physical law. Each cell contains 12 draws. The final row is a control in which the defect is genuinely in a physical law. Law-only alternatives are restricted to one edit here; Section 4 asks what changes when multiple law edits are allowed.

		oscillators		reservoirs		toxicokinetics	
defect introduced in	repair operation	same component	law rewrite	same component	law rewrite	same component	law rewrite
an admissible value set	EditDomain	12	0	12	0	12	0
the observation model	EditObservation	12	0	12	0	12	0
a concept's role	SetRole	12	0	12	0	12	0
a physical law (control)	EditRelation	12	12	12	12	12	12

The one-edit results are clean. Across all three system families, defects in admissible values, the observation model, and concept roles are recovered by changing the component in which the defect was introduced. A single law rewrite does not reproduce these exact recordings. When the defect is genuinely in a physical law, the corresponding `EditRelation` repair succeeds.

This pattern also holds when we remove the grouping by component and test repair operations individually. For three representative cases, only the operation acting on the defective component reproduces the recording (Appendix B). Under exact data and a one-edit budget, the recording therefore localizes the discrepancy to the correct part of the theory.

Measurement noise weakens this separation because additional explanations can fall within the acceptance tolerance. The intended component repair continues to reproduce the data whenever the generating theory itself satisfies the noisy-data criterion, while some incorrect law rewrites become acceptable as the tolerance increases (Appendix R).

These results establish the baseline: with a restricted repair budget, different kinds of defects are experimentally distinguishable. They also expose the next question. A physical law is usually far more flexible than a single local edit. If several law changes are allowed to act together, can the dynamics reproduce a discrepancy that actually originated elsewhere in the theory?

4 WHEN A WRONG LAW CAN STILL FIT THE DATA

Table 2 considers only a single law edit. Allowing more than one edit changes the picture. A sufficiently flexible rewrite of the dynamics can sometimes reproduce the recording even when the original defect lies somewhere else in the theory. The reason is that an experiment observes only the final recorded channel, not the internal decomposition that produced it. Changes to the dynamics and changes to the observation model can therefore compensate for one another.

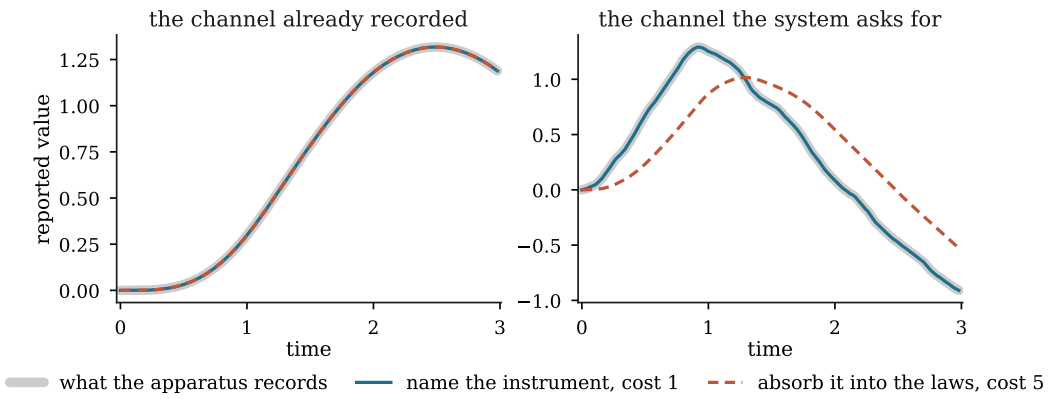


Figure 2: Absorption in a mass–spring system measured by a lagging sensor. **Left:** correcting the sensor model and rewriting two physical laws produce the same recorded signal. **Right:** the two explanations disagree on an additional quantity, so measuring it separates them.

We call this phenomenon **absorption**: a defect outside the physical law is absorbed into a rewritten law that nevertheless reproduces the observed data. The resulting theory may fit every recorded point while giving the wrong explanation of where the discrepancy comes from.

4.1 What can one recording tell us?

For each recording, we compare two competing accounts. The first is the cheapest repair that changes the component in which the defect was introduced. The second is the cheapest account that leaves those components fixed and changes only the physical laws. Both are required to reproduce the observed data.

We distinguish three outcomes. The result is *unique* if at least one available experiment gives different predictions under the two accounts. In that case, additional evidence can determine which explanation is consistent with the apparatus. It is *cost-identifiable* if every available experiment gives the same prediction but one account has lower representation cost. It is *ambiguous* if the accounts agree on all available experiments and also have the same cost.

This distinction matters because only the unique case is settled by experimental evidence. In the cost-identifiable case, choosing the cheaper account reflects a preference built into the representation or objective, not information contained in the recording. In the ambiguous case, even that preference is absent. A system that reports either of these as an experimentally established explanation would therefore be claiming more than the data support.

Finding. Matching the recording does not by itself identify which part of the theory is wrong.

4.2 When can a law absorb another defect?

For linear systems, whether absorption is possible can be characterized without first running the search. Consider the mapping from the experimental drive to the recorded channel. For a linear time-invariant system started from rest, that mapping is completely described by its transfer function. If two accounts have the same transfer function to the observed channel, then they produce the same recording for every possible drive.

This turns absorption into a realization question: can the fixed observation model of the theory under test, together with some rewritten rate laws, realist exactly the same input–output map as the apparatus?

Proposition 1. *Assume the apparatus and theory are linear and time-invariant over the experiments considered, start from rest, and are sampled in the same way. Let the apparatus have transfer function $H^*(p)$ with McMillan degree d and direct feedthrough D^*. Let the theory under test declare*

n states and observation model

$$y = c^\top x + e\,u.$$

For a sufficiently informative drive, a law-only account reproduces the observed recording if and only if it reproduces the apparatus for every drive. Such an account exists exactly when:

(i) the direct feed-through matches, $D^* = e$;

(ii) the theory declares enough states, $n \geq d$;

(iii) the required state-space dynamics can be written using the physical dimensions and parameters declared by the theory.

Each condition rules out a different failure mode. Condition (i) asks whether the drive reaches the recorded channel directly in the same way under the apparatus and the theory. A mismatch here cannot be repaired by changing internal dynamics, because the disagreement appears before any state evolution occurs. Condition (ii) asks whether the theory has enough states to represent all dynamical modes visible in the recording. If the apparatus has a transfer function of degree d, no realization with fewer than d states can reproduce it exactly.

Condition (iii) adds the physical typing constraint. Standard realization theory allows changes of state coordinates that freely mix state variables. In a physical theory, however, different states may carry different units. A mathematically valid realization is therefore usable only if its coefficients can be expressed with the dimensions provided by the theory's declared parameters. Conditions (i) and (ii) follow from classical realisation theory (Kalman, 1963; Ho & Kalman, 1966); Appendix N gives the complete proposition, proof, and constructive procedure for condition (iii).

The practical consequence is that some cases can be ruled out before any combinatorial search. Across 8 prior–defect combinations, the conditions predict that a law-only account exists for 4 cases, and the staged search agrees on 6 of 8 rows (Appendix O). In particular, the observation-model defect can sometimes be absorbed because the theory still contains enough states and the required dynamics are writable. By contrast, the admissible-set defect can violate linearity when the bound becomes active, and the role defect can remove the experimental drive altogether, preventing an equivalent law-only account.

The toxicokinetic system lies outside the proposition because saturation and active bounds make its dynamics nonlinear over the experiments considered. We therefore test that case directly. Across the three drive amplitudes examined, the search finds no law-only account (Appendix M).

Finding. In the linear setting, absorption is a structural property of the input–output representation, not simply a failure of the search procedure.

4.3 A SENSOR ERROR ABSORBED INTO THE DYNAMICS

Figure 2 gives the simplest concrete example. The apparatus is a mass–spring system observed through a sensor with a first-order lag. One account repairs the observation model and explicitly places the lag in the sensor, at cost 1. A second account leaves the observation model unchanged and instead rewrites two physical laws, at cost 5. Both reproduce the recorded channel to 5.1e-16 relative error.

The absorbing account becomes particularly transparent after rescaling the sensor state:

$$\dot{\hat{s}} = (u - \hat{s})/\tau, \qquad \dot{v} = -kx - cv + f\hat{s}.$$

The same first-order timescale is still present, but it has moved from the measurement process into the system dynamics. The two accounts therefore agree that the observed response contains a lag; they disagree about where that lag physically resides.

That disagreement cannot be exposed by changing the drive. The transfer functions from the drive to the recorded channel are identical, so every possible input produces the same observed output under both accounts. More excitation, a different waveform, or a different amplitude cannot resolve the ambiguity.

The accounts do, however, make different predictions about quantities that were not recorded. We therefore compare candidate experiments that either change the drive or add a measurement channel. Of 12 available experiments, all 9 experiments that resolve the case add a channel. The largest predicted disagreement occurs for the state v, where the two accounts differ by 0.44 relative. Running that experiment changes the verdict to unique (Appendix Q).

This distinction is important for experimental design. If competing theories already have the same transfer function to the existing measurement, searching for a more informative input is futile. The experiment must instead expose a quantity on which the theories disagree.

Finding. When two accounts have identical input–output behavior on the recorded channel, changing the drive cannot distinguish them; an additional measurement can.

4.4 How often does absorption occur?

The example above shows that absorption is possible, but not whether it is a special case. We therefore repeat the same two-edit search across systems drawn from the two linear priors. For each system, we inject the observation-model defect, construct the competing accounts, and test whether additional experiments separate them.

A law-only account exists on 100 of 100 drawn systems. Among these absorbed cases, adding a second measurement resolves 100, whereas changing the drive resolves 0. The population therefore shows the same pattern as the individual oscillator example: once the recorded input–output behavior is shared, new measurements are substantially more informative than new drives.

We next vary not only the numerical coefficients but also the system structure itself. The resulting 48 apparatuses differ in state count, dimensions, couplings, drive location, and instrument placement. The structural conditions decide 48 cases before search and agree with the search on 41. Among the 9 systems in which absorption occurs, an added channel resolves every one, 9 of 9, while a changed drive resolves 2.

Table 3: Absorption across drawn systems. We compare the repair to the injected component with a two-edit law-only account. "Resolves" means that an additional experiment distinguishes the two explanations.

defect, predicted	prior	systems	absorption exists	extra channel resolves	drive change resolves
observation, exists	oscillators	50	50/50	50/50	0/50
observation, exists	reservoirs	50	50/50	50/50	0/50
role, does not exist	oscillators	25	0/25	–	–
role, does not exist	reservoirs	25	0/25	–	–

Finally, we test whether these conclusions depend on exact arithmetic or on a particular definition of repair cost. With the acceptance tolerance widened as measurement noise increases, the same absorbing account remains available and the verdict is unchanged over the tested noise levels (Appendix R). We also evaluate five structural cost measures and prediction-preserving rewritings of the theories. The five measures agree on 100 of 100 scored systems (Appendix K), indicating that the observed preference is not an artifact of a single syntactic distance.

Finding. Absorption persists across coefficients, system structures, tested noise levels, and alternative cost measures. When it occurs, resolving the ambiguity usually requires observing a new quantity rather than applying a different input.

5 Evidence beyond our own simulations

The preceding results use systems defined within our own representation. We therefore ask whether the same ambiguity appears in independently constructed systems, established discovery methods, and existing benchmarks.

Independent systems reproduce the same ambiguity. We first test an independently written simulator of a heated aluminium block measured by a lagging thermocouple (Appendix H). One account places the lag in the heater dynamics and another in the sensor. Both fit the thermocouple signal, their fitted poles agree to five digits, and across 5 new power schedules their predictions differ by at most 5.7e-08,°C. Proposition 1 predicts this equivalence.

The accounts disagree on the unobserved block temperature. Measuring that quantity resolves the ambiguity: the sensor-lag account reaches 0.106 error against a noise floor of 0.055, while the heater-dynamics account misses by 0.278 and up to 0.534. The public Temperature Control Lab recording lacks this direct block-temperature channel, and its second thermistor does not separate the accounts. Thus the same pattern holds outside our simulator: equivalent predictions on the recorded channel persist across changed drives, while an additional measurement exposes the physical difference.

We observe the complementary case in six independently released DiscoverPhysics worlds (Wiemann et al., 2026). Law rewrites reproduce 5 of 6 worlds to 0.0028 relative error or better. In `oscillator`, however, the interaction depends on absolute time. A position-only law reaches 0.0435, while introducing time explicitly as a concept reaches 0.0002. Restarting the experiment a quarter period later separates the accounts by 0.68. This case shows that a changed drive can be informative when the competing theories do not share the same input–output map.

Standard law-discovery methods also absorb instrument effects. We next test PySINDy (De Silva et al., 2020; Brunton et al., 2016) and PySR (Cranmer, 2023). Both fit the recorded signal, yet changing only the instrument changes the inferred law. Doubling the instrument time constant changes PySR's returned function by 0.51; in PySINDy, the same intervention changes 144 of 223 nonzero coefficients, and 142 of 278 coefficients one order deeper (Appendix R). When the instrument is represented explicitly, the intervention changes the instrument parameter while leaving the physical law unchanged.

A system-identification baseline reaches the same conclusion from another direction. An ARX model recovers the shared input–output dynamics to 4e-10 relative error, but one recorded channel does not determine whether the lag belongs to the system or to the sensor (Appendix Q). Accurate recovery of observable dynamics therefore does not by itself identify where those dynamics physically reside.

A language-model baseline can perform many individual steps of the workflow, including identifying defective components, constructing law-only accounts, and deriving transfer functions (Appendix J). Its nonlinear constructions are less stable: after halving the simulator step, all 18 accounts produced by our search remain valid, while only 24 of 35 model-generated accounts do.

Current benchmarks rarely test this ambiguity. Finally, we ask whether existing discovery benchmarks require systems to distinguish between a wrong law and a wrong experimental representation. In NewtonBench (Zheng et al., 2025), higher "system complexity" settings can alter what is observed, but the setting is given to the agent and the target remains a functional law. All 324 tasks therefore ultimately map to the law-edit cell of Table 1 (Appendix C).

The same pattern appears in a broader coding of scientific cases. Among 40 writable resolutions from 55 episodes in physics history and recent literature, 35 map to `EditRelation`, while none use the observation-model or role edits studied here. A deliberately selected historical set confirms that such repairs do occur: 9 cases map to `EditObservation` and 3 to `SetRole`, with two blind judges agreeing with the coding on 11 of 12 cases (Appendix D).

Together, these external checks support the same conclusion as the controlled experiments: a fitted law can absorb effects that originate elsewhere in the experimental theory, and current benchmarks rarely require a discovery system to distinguish these explanations.

6 Related work and limitations

Identifiability and realisation. Structural identifiability asks what input–output observations can determine about a specified model (Bellman & Åström, 1970), while classical realisation theory characterizes the state representations that produce a given input–output map (Kalman, 1963; Ho & Kalman, 1966). Tools such as DAISY, STRIKE-GOLDD, and SIAN test identifiability once the

dynamics and measured outputs have been specified (Bellu et al., 2007; Villaverde et al., 2016; Hong et al., 2019). We instead place the observation model, admissible values, experimental roles, and physical laws in one repair space and ask when edits to different components become observationally indistinguishable.

Law and theory discovery. Symbolic regression searches for equations over supplied variables (Schmidt & Lipson, 2009; Udrescu & Tegmark, 2020; Cranmer, 2023), while AI-Descartes combines fitted equations with background theory (Cornelio et al., 2023). SINDy and PySINDy identify sparse dynamics in chosen measurement coordinates and function libraries (Brunton et al., 2016; De Silva et al., 2020). AI-Newton goes further by discovering explicit concepts and laws (Fang et al., 2025), but its experimental interface still specifies what can be observed and controlled. Our setting makes that interface itself editable, allowing an instrument, role, or domain repair to compete directly with a law rewrite.

Scientific-discovery agents and theory revision. Recent benchmarks evaluate broader discovery workflows, including experiment design and interactive hypothesis testing (Majumder et al., 2024; Zheng et al., 2025; Miao et al., 2026). SciLaws-Bench grounds law discovery in published papers and real data and pairs each problem with a synthesized parallel world (Huang et al., 2026); PRBench scores end-to-end reproduction of physics papers (Qiu et al., 2026); and Stargazer evaluates agents on radial-velocity model fitting, where a good statistical fit often comes with wrong physical parameters (Liu et al., 2026). CritPt poses research-level physics reasoning problems to LLMs (Zhu et al., 2025). All of these hold what is observed and controlled fixed and score the law, the fit, the reproduction, or the answer. Our question is complementary: which part of the theory may the agent revise, and which part does the benchmark hold fixed? AGM belief revision provides a formal analogue for adding and removing theoretical commitments (Alchourrón et al., 1985); our representation adds physical typing, observation maps, experimental roles, and executable predictions.

Limitations. Our exact characterization currently covers linear time-invariant systems with one drive and a fixed linear observation map. Extending it to nonlinear, time-varying, and multi-channel systems is the main theoretical next step, with nonlinear identifiability methods providing useful machinery (Villaverde et al., 2016; Hong et al., 2019). Condition (iii) also uses a sufficient dimensional test; a complete typed-realisation criterion would turn it into a full decision procedure. On the algorithmic side, the generator currently enumerates only 16 of the 21 derived operations, motivating symbolic constraints and more scalable compositional search (Appendices B and I). Finally, broader independently authored simulators, physical datasets with the predicted separating channels, and externally designed generator tests would provide stronger tests of generalization. Our current three-way verdict compares two selected accounts; extending it to the full set of observationally equivalent repairs would support experiment design over an entire equivalence class.

REPRODUCIBILITY STATEMENT

All results are generated from the code and data. External benchmarks and software are recorded at fixed versions or commits, and derived results are recomputed from their underlying source files. Code and data required to reproduce the experiments are in the SI.

AI USE STATEMENT

Generative AI tools were used to improve the readability and concision of the manuscript, to assist in identifying relevant literature for the authors to review, and to help locate and fix bugs in existing code. All AI-assisted edits, references, and code changes were reviewed and verified by the authors. Every reported numerical result is generated by an executed script and checked against its recorded source hash, as described in the reproducibility statement. We take responsibility for the final content of the manuscript, including all text, claims, results, and artifacts produced with the aid of generative AI.

REFERENCES

Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50(2):510–530, June 1985. ISSN 0022-4812, 1943-5886. doi: 10.2307/2274239.

R. Bellman and K.J. Åström. On structural identifiability. *Mathematical Biosciences*, 7(3-4):329–339, April 1970. ISSN 00255564. doi: 10.1016/0025-5564(70)90132-X.

Giuseppina Bellu, Maria Pia Saccomani, Stefania Audoly, and Leontina D'Angiò. DAISY: A new software tool to test global identifiability of biological and physiological systems. *Computer Methods and Programs in Biomedicine*, 88(1):52–61, October 2007. ISSN 01692607. doi: 10.1016/j.cmpb.2007.07.002.

Samuel R. Bowman and George Dahl. What will it take to fix benchmarking in natural language understanding? In *NAACL*, pp. 4843–4855, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.385.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, April 2016. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1517384113.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security*. arXiv, 2018. doi: 10.48550/ARXIV.1802.08232.

T. Y. Chen, S. C. Cheung, and S. M. Yiu. Metamorphic testing: A new approach for generating next test cases. Technical Report HKUST-CS98-01, HKUST, 1998.

Cristina Cornelio, Sanjeeb Dash, Vernon Austel, Tyler R. Josephson, Joao Goncalves, Kenneth L. Clarkson, Nimrod Megiddo, Bachir El Khadir, and Lior Horesh. Combining data and theory for derivable scientific discovery with AI-descartes. *Nature Communications*, 14(1):1777, April 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-37236-y.

Miles Cranmer. Interpretable machine learning for science with PySR and SymbolicRegression.jl. *arXiv Preprint arXiv:2305.01582*, 2023. doi: 10.48550/ARXIV.2305.01582.

Brian De Silva, Kathleen Champion, Markus Quade, Jean-Christophe Loiseau, J. Kutz, and Steven Brunton. PySINDy: A python package for the sparse identification of nonlinear dynamical systems from data. *Journal of Open Source Software*, 5(49):2104, May 2020. ISSN 2475-9066. doi: 10.21105/joss.02104.

R.A. DeMillo, R.J. Lipton, and F.G. Sayward. Hints on test data selection: Help for the practicing programmer. *Computer*, 11(4):34–41, April 1978. ISSN 0018-9162. doi: 10.1109/C-M.1978.218136.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. Investigating data contamination in modern benchmarks for large language models. In *NAACL*, pp. 8706–8719, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.482.

Allyson Ettinger, Sudha Rao, Hal Daumé Iii, and Emily M. Bender. Towards linguistically generalizable NLP systems: A workshop and shared task. In *Workshop on Building Linguistically Generalizable NLP Systems, EMNLP*, pp. 1–10, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5401.

You-Le Fang, Dong-Shan Jian, Xiang Li, and Yan-Qing Ma. AI-newton: A concept-driven physical law discovery system without prior physical knowledge. *arXiv Preprint arXiv:2504.01538*, 2025. doi: 10.48550/ARXIV.2504.01538.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *NAACL*, pp. 107–112, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017.

B. L. Ho and R. E. Kalman. Editorial: Effective construction of linear state-variable models from input/output functions: Die konstruktion von linearen modeilen in der darstellung durch zustandsvariable aus den beziehungen für ein- und ausgangsgrößen. *Auto*, 14(1-12):545–548, December 1966. ISSN 2196-677X, 0178-2312. doi: 10.1524/auto.1966.14.112.545.

Hoon Hong, Alexey Ovchinnikov, Gleb Pogudin, and Chee Yap. SIAN: Software for structural identifiability analysis of ODE models. *Bioinformatics*, 35(16):2873–2874, August 2019. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/bty1069.

Yiming Huang, Ziche Liu, Zhuohang Wu, Yiqian Wang, Junxia Cui, Xinkai Zou, Linjun Mao, Nan Huang, Naicheng Yu, Kaijie Zhu, Yue Ma, Kun Zhou, Letian Peng, and Jingbo Shang. Can LLMs discover scientific laws in real and parallel worlds? *arXiv Preprint arXiv:2609.01552*, September 2026. doi: 10.48550/ARXIV.2609.01552.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In *EMNLP*, pp. 5075–5084, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.308.

Yue Jia and Mark Harman. An analysis and survey of the development of mutation testing. *IEEE Transactions on Software Engineering*, 37(5):649–678, September 2011. ISSN 0098-5589. doi: 10.1109/TSE.2010.62.

R. E. Kalman. Mathematical description of linear dynamical systems. *Journal of the Society for Industrial and Applied Mathematics Series a Control*, 1(2):152–192, January 1963. ISSN 0887-4603, 2168-359X. doi: 10.1137/0301010.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In *NAACL*, pp. 4110–4124, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.324.

John Lalor, Hao Wu, and Hong Yu. Building an evaluation scale using item response theory. In *EMNLP*, pp. 648–657, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1062.

Xinge Liu, Terry Jingchen Zhang, Bernhard Schölkopf, Zhijing Jin, and Kristen Menou. Stargazer: A scalable model-fitting benchmark environment for AI agents under astrophysical constraints. *arXiv Preprint arXiv:2604.15664*, 2026. doi: 10.48550/ARXIV.2604.15664.

Bodhisattwa Prasad Majumder, Harshit Surana, Dhruv Agarwal, Bhavana Dalvi Mishra, Abhijeetsingh Meena, Aryan Prakhar, Tirth Vora, Tushar Khot, Ashish Sabharwal, and Peter Clark. DiscoveryBench: Towards data-driven discovery with large language models. *arXiv Preprint arXiv:2407.01725*, 2024. doi: 10.48550/ARXIV.2407.01725.

Tingjia Miao, Wenkai Jin, Muhua Zhang, Jinxin Tan, Yuelin Hu, Tu Guo, Jiejun Zhang, Yuhan Wang, Wenbo Li, Yinuo Gao, Shuo Chen, Weiqi Jiang, Yayun Hu, Zixing Lei, Xianghe Pang, Zexi Liu, Yuzhi Zhang, Linfeng Zhang, Kun Chen, Wei Wang, Weinan E, and Siheng Chen. PRL-bench: A comprehensive benchmark evaluating LLMs' capabilities in frontier physics research. *arXiv Preprint arXiv:2604.15411*, 2026. doi: 10.48550/ARXIV.2604.15411.

Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments. In *ACL*, pp. 4658–4664, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1459.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B. Hashimoto. Proving test set contamination in black box language models. In *ICLR*. arXiv, 2023. doi: 10.48550/ARXIV.2310.17623.

Junho Park, R. Abraham Martin, Jeffrey D. Kelly, and John D. Hedengren. Benchmark temperature microcontroller for process dynamics and control. *Computers & Chemical Engineering*, 135: 106736, April 2020. ISSN 00981354. doi: 10.1016/j.compchemeng.2020.106736.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis only baselines in natural language inference. In **SEM*, pp. 180–191, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023.

Shi Qiu, Junyi Deng, Yiwei Deng, Haoran Dong, Jieyu Fu, Mao Li, Zeyu Li, Zhaolong Zhang, Huiwen Zheng, Leidong Bao, Anqi Lv, Zihan Mo, Yadi Niu, Yiyang Peng, Yu Tian, Yili Wang, Ziyu Wang, Zi-Yu Wang, Jiashen Wei, Liuheng Wu, Aoran Xue, Leyi Yang, Guanglu Yuan, Xiarui Zhan, Jingjun Zhang, Zifan Zheng, Pengfei Liu, Linrui Zhen, Kaiyang Li, Qichang Li, Ziheng Zhou, Guo-En Nian, Yunwei Xiao, Qing-Hong Cao, Linjie Dai, Xu Feng, Peng Gao, Ying Gu, Chang Liu, Jia Liu, Ming-xing Luo, Yan-Qing Ma, Liang-You Peng, Huichao Song, Shufeng Wang, Chenxu Wang, Tao Wang, Yi-Nan Wang, Chengyin Wu, Pengwei Zhao, and Hua Xing Zhu. PRBench: End-to-end paper reproduction in physics research. *arXiv Preprint arXiv:2603.27646*, 2026. doi: 10.48550/ARXIV.2603.27646.

Inioluwa Deborah Raji, Emily M. Bender, Amandalynne Paullada, Emily Denton, and Alex Hanna. AI and the everything in the whole wide world benchmark. In *NeurIPS Datasets and Benchmarks*. arXiv, 2021. doi: 10.48550/ARXIV.2111.15366.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P. Lalor, Robin Jia, and Jordan Boyd-Graber. Evaluation examples are not equally informative: How should that change NLP leaderboards? In *ACL*, pp. 4486–4503, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.346.

Andrew Ruef, Michael Hicks, James Parker, Dave Levin, Michelle L. Mazurek, and Piotr Mardziel. Build it, break it, fix it: Contesting secure development. In *CCS*, pp. 690–703, Vienna Austria, October 2016. ACM. ISBN 978-1-4503-4139-4. doi: 10.1145/2976749.2978382.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez De Lacalle, and Eneko Agirre. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of EMNLP*, pp. 10776–10787, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.722.

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, April 2009. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1165893.

Sergio Segura, Gordon Fraser, Ana B. Sanchez, and Antonio Ruiz-Cortes. A survey on metamorphic testing. *IEEE Transactions on Software Engineering*, 42(9):805–824, September 2016. ISSN 0098-5589, 1939-3520. doi: 10.1109/TSE.2016.2532875.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D. Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito,

Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B. Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang,

Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Tmlr*, June 2023. doi: 10.48550/arXiv.2206.04615.

Silviu-Marian Udrescu and Max Tegmark. AI feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, April 2020. ISSN 2375-2548. doi: 10.1126/sciadv.aay2631.

Alejandro F. Villaverde, Antonio Barreiro, and Antonis Papachristodoulou. Structural identifiability of dynamic systems biology models. *PLOS Computational Biology*, 12(10):e1005153, October 2016. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1005153.

Matt L. Wiemann, Lindsay M. Smith, Peter Melchior, Siddharth Mishra-Sharma, Andrew Gordon Wilson, Pavel Izmailov, and Carolina Cuesta-Lázaro. DiscoverPhysics: Benchmarking LLMs for out-of-the-box scientific thinking. *arXiv Preprint arXiv:2605.26087*, September 2026. doi: 10.48550/ARXIV.2605.26087.

Tianshi Zheng, Kelvin Kiu-Wai Tam, Newt Hue-Nam K. Nguyen, Baixuan Xu, Zhaowei Wang, Jiayang Cheng, Hong Ting Tsang, Weiqi Wang, Jiaxin Bai, Tianqing Fang, Yangqiu Song, Ginny Y. Wong, and Simon See. NewtonBench: Benchmarking generalizable scientific law discovery in LLM agents. In *ICLR*. arXiv, 2025. doi: 10.48550/ARXIV.2510.07172.

Minhui Zhu, Minyang Tian, Xiaocheng Yang, Tianci Zhou, Lifan Yuan, Penghao Zhu, Eli Chertkov, Shengyan Liu, Yufeng Du, Ziming Ji, Indranil Das, Qingzhi Chen, Junyi Cao, Jiabin Yu, Peixue Wu, Jinchen He, Yifan Su, Yikun Jiang, Yujie Zhang, Chang Liu, Ze-Min Huang, Weizhen Jia, Yunkai Wang, Farshid Jafarpour, Yong Zhao, Xinan Chen, Jessie Shelton, Aaron W. Young, John Bartolotta, Wenchao Xu, Yue Sun, Anjun Chu, Victor Colussi, Chris Akers, Nathan Brooks, Wenbo Fu, Jinchao Zhao, Marvin Qi, Anqi Mu, Yubo Yang, Allen Zang, Yang Lyu, Peizhi Mai, Christopher Wilson, Xuefei Guo, Juntai Zhou, Daniel Inafuku, Chi Xue, Luyu Gao, Ze Yang, Yaïr Hein, Yonatan Kahn, Kevin Zhou, Di Luo, John Drew Wilson, Jarrod T. Reilly, Dmytro Bandak, Ofir Press, Liang Yang, Xueying Wang, Hao Tong, Nicolas Chia, Eliu Huerta, and Hao Peng. Probing the critical point (CritPt) of AI reasoning: A frontier physics research benchmark. *arXiv Preprint arXiv:2509.26574*, 2025. doi: 10.48550/ARXIV.2509.26574.

A Attacks on the generator evaluation

The generator should propose repairs from the theory itself, not recover answers from the evaluation set. A simple reachability test cannot guarantee this: a generator that reads the target theories and returns their answers can achieve perfect reach without performing any discovery. We built and withdrew 8 successive evaluation designs while testing variants of this failure mode.

Why simple defenses fail. Most defenses only restrict *how* the generator accesses the answers. Caching defeats repeated-call checks, renaming or copying files defeats checks tied to module names, inlining defeats import tracing, and filtering borrowed answers through the generator's grammar defeats tests for over-reach. The common problem is that, as long as the generator can access the target builder during evaluation, it can adapt its output to the test.

We therefore removed the target builder from the generator's run-time environment. The archived attack in `attacks/ninth.py` shows why this is still not enough. It queries the target builder beforehand, stores the expressible answers in a table indexed by a fingerprint of the theory, and consults that table before falling back to the honest generator. This attack plants 563 stored answers and passes all 42 rows of the evaluation as it then stood, including reachability, soundness, and ablation checks.

What detects the attack. The injected-defect catalogue catches this attack without recognizing the lookup mechanism itself. One injected defect collapses the honest generator's enumeration to a single atom. An honest generator therefore changes its behavior, whereas the stored-answer attack keeps the same reach. The corresponding test fails to respond to the defect, and the run ends

G1g FAILED. In other words, the catalogue asks whether the evaluation actually depends on the generator's internal capabilities.

We also tested a "novelty" score, the fraction of proposals outside the target set. A pure lookup scores 0.0, compared with 0.97 for the honest generator, but the statistic is easy to manipulate because the generator controls its denominator. Padding the borrowed answers raises the score to 0.889, while an honest answer-blind family scores only 0.0.

What can and cannot be certified. The general issue is structural. If the generator can evaluate the target builder, then it can also evaluate any acceptance rule defined only from its proposals and those targets, and adapt until it passes. Two protocols avoid this problem: the evaluator can keep part of the test hidden, or the generator can commit before the targets are drawn. Neither is available in our present setup.

We therefore use three complementary checks:

1. a catalogue of 15 injected defects, each required to make a named test fail;
2. ablation of each generator component, with a named test required to detect its removal; and
3. explicit reporting of test rows that no injected defect covers, currently 31 of 46.

Removing any of the generator's four substitution families causes a named test to fail. If an injected defect crashes the harness, we count it as uncaught rather than as successful detection.

Reproduction harness. We archive each attack together with the code version on which the corresponding evaluation was claimed. Of 6 archived attacks, 4 reproduce against their original commits (Table 4). The harness is provided in `attacks/run_all.py`, with instructions in `attacks/README.md`.

Table 4: Reproduction of archived attacks against the commits at which each evaluation design was claimed. "PASSED" under attack means that the borrowing generator still satisfied the evaluation, so the attack succeeded. The final two rows reconstruct attacks whose original implementations were not archived.

design	commit	test	clean	under the attack
1	—	—	withdrawn by eleven findings, not by one substitution	
2	f179a09	g1_gate.py	PASSED (13 of 13)	PASSED (13 of 13)
3	029ccc1	g1c_gate.py	PASSED	PASSED
4	—	—	withdrawn with design 3, never attacked alone	
5	60a2bcc	g1e_gate.py	PASSED	PASSED
6	3d740c5	g1f_gate.py	PASSED	PASSED
7	2df2964	gate_generator.py	PASSED	FAILED (reconstruction)
8	de54ff1	gate_generator.py	PASSED	FAILED (reconstruction)

B TESTING EACH REPAIR OPERATION AND THE VALID EDIT SET

Table 2 shows that repairing the component in which a defect was introduced succeeds, while a single law rewrite does not. Because that table groups edits by component, we additionally test the repair operations one at a time. This checks whether the result comes from one specific operation or merely from having many candidate edits available within the successful component.

Table 5 reports three representative cases: one drawn from the prior and two constructed by hand. Each row enables only one repair operation. The entries give the number of canonical candidate theories that reproduce the recording out of all candidates generated by that operation. In every case, exactly one operation succeeds: `EditDomain` for the admissible-set defect, `EditObservation` for the observation-model defect, and `SetRole` for the role defect. In particular, `EditRelation` fails on all three cases.

The operations omitted from Table 5 were also tested. Some generate many candidates—for example, `InsertRelation` generates forty on the role case—but none reproduces any of the three

Table 5: Repair operations tested individually on three representative defects. Each entry gives the number of canonical candidate theories that reproduce the recording out of the total generated by that operation. Bold entries mark the only successful operation for each defect. A value of 0 of 0 means that the operation generates no valid candidate for that case.

operation, alone	admissible set	observation model	role
EditRelation	0 of 7,882	0 of 3,706	0 of 2,471
EditDomain	**1 of 16**	0 of 2	0 of 2
EditObservation	0 of 454	**2 of 334**	0 of 242
SetRole	0 of 0	0 of 0	**1 of 1**

recordings. The result therefore does not depend on grouping several edits under the same component: for these cases, the successful repair operation itself is unique.

The amount of search nevertheless differs across defect types. On one drawn case per kind, the successful operation generates 16, 454, 1, and 4,114 candidates out of 8,366, 5,798, 5,840, and 4,597 total proposals, respectively. The role case is especially simple: exactly one state has no rate law, so the generator proposes a single `SetRole` edit. Thus, Table 5 supports the claim that the defective component can be identified; it does not imply that every case requires equally difficult search.

Which schema cells are not repair operations? Table 1 is derived by considering the fields of the five record types, but not every possible field change corresponds to a meaningful repair. Of the 27 component–field cells, 6 do not define valid single-step operations.

Five of these cells change only identifiers: the name of a concept, relation, observation, parameter, or constraint. Renaming an object changes neither the theory's behavior nor its predictions, so these cells do not define repairs.

The remaining case is a relation's `wrt` field, which distinguishes an algebraic relation from a rate law. Changing this field alone cannot preserve a well-typed theory. A rate law must differentiate its own output and therefore has units of output per unit time, whereas an algebraic relation has the units of the output itself. Converting between the two requires changing additional fields at the same time. We verify this directly by enumerating every single-field `wrt` change on the probe fixtures; the typechecker accepts none.

Valid operations that the generator does not yet emit. A different issue is generator coverage. 5 of the 21 valid operations in Table 1 are not currently enumerated by the generator. These remain valid operations; the present proposal mechanism simply does not generate them.

`Tune` is continuous and is handled during parameter fitting rather than discrete proposal generation. `SetDimension` and `SetParameterDimension` either make existing relations ill typed or, when the modified quantity is unused, leave all predictions unchanged. `InsertConstraint` and `EditConstraint` are currently blocked because constraints are stored as Python callables rather than symbolic expression trees. Representing constraints symbolically would make these edits searchable and would also allow the interpreter to execute the 15 cases in Table 6.

operation	class	why the generator has no index for it	what would unblock it
`SetDimension`	not enumerable	changing a concept's dimension either breaks every relation that mentions it, in which case the typechecker rejects the result, or the concept is mentioned by nothing, in which case the change alters no prediction. Both branches are empty, so there is no condition under which it applies.	
`SetParameterDimension`	not enumerable	the same argument as SetDimension, on a parameter: either every relation using it becomes ill-typed and the typechecker rejects the result, or nothing uses it and the change alters no prediction.	
`Tune`	not enumerable	a parameter's value is continuous, so enumerating it would mean choosing a grid. Tune belongs to the fit that scores a candidate, not to the proposal that generates one, and edits.cost already prices it at zero inside TUNE_FREE.	
`EditConstraint`	blocked	the same closure, one operation along.	the same fix
`InsertConstraint`	blocked	Constraint.holds is a Python callable, so there is nothing to generate, nothing to compare and nothing to charge a distance against. It is the same closure the G0 review removed from relations.	make a constraint a cmp expression tree, as relations became trees

C NewtonBench under dimensional typing

Section 5 maps NewtonBench tasks onto the repair operations in Table 1. Here we apply a stricter check: whether the benchmark laws can also be represented by our expression grammar and physical type system.

Of the 108 benchmark laws, 70 fall inside our current expression grammar. The remaining laws use transcendental functions that the grammar does not contain, spanning 5 modules. Among the laws that can be parsed, 73 become dimensionally consistent after assigning a dimension to the hidden constant, whereas 35 remain inconsistent under every possible assignment. These failures occur when a term has the wrong dimension for the output or when terms in the same sum have incompatible dimensions.

The hidden constant also reveals one place where NewtonBench changes more than the law itself. In 6 of the 12 modules, different task variants require that constant to carry different physical dimensions. Moving between those variants therefore changes both the relation expression and the parameter declaration. A search that edits only the right-hand side while holding parameter declarations fixed cannot represent the full change.

Thus, the dimensional analysis is less favorable to our representation than the syntactic mapping in the main text: some NewtonBench laws lie outside our grammar, some violate our typing rules, and a subset requires edits to both the law and a parameter declaration.

D THE THREE PRIORS AND THE CASE SET

The experiments in Section 3 use three families of simulated systems with different dynamics but the same five-component theory representation. For each draw, we sample all coefficients from the ranges below, inject one defect into the theory under test, and keep all other parameters equal to the generating system. We use length L and time T as base units, take the drive u to be dimensionless, and record 150 samples at $\Delta = 0.02$. Each case uses three drives formed as cumulative sums of Gaussian steps. A candidate reproduces the recording when its largest relative residual is below 0.001.

Oscillator. The first prior is a mass–spring system with position x (L), velocity v (LT^{-1}), and a lagging sensor state s (L):

$$\dot{x} = v, \qquad \dot{v} = -kx - cv + fu, \qquad \dot{s} = (x - s)/\tau.$$

We draw

$$k \in [1.5, 4.0]\, T^{-2}, \qquad c \in [0.2, 0.8]\, T^{-1}, \qquad f \in [0.7, 1.6]\, LT^{-2},$$

and $\tau \in [0.2, 0.6]\, T$. The sensor also has a floor drawn from $[0.05, 0.3]\, L$, used only by the admissible-set defect. The apparatus records s, whereas the theory under test declares that the channel reports x.

Reservoirs. The second prior contains two coupled reservoir levels, h_a and h_b (L), measured through the same type of lagging sensor:

$$\dot{h}_a = -gh_a + gh_b + hu, \qquad \dot{h}_b = mh_a - mh_b, \qquad \dot{s} = (h_a - s)/\tau.$$

We draw

$$g \in [0.8, 2.5]\, T^{-1}, \qquad m \in [0.4, 1.8]\, T^{-1}, \qquad h \in [0.5, 1.5]\, LT^{-1},$$

with τ and the sensor floor drawn as above. The apparatus records s, while the theory under test declares that the channel reports h_a.

Toxicokinetic system. The third prior contains a concentration c, a bounded damage variable $e \in [0, 1]$, and a plate-reader state a:

$$\dot{c} = d/V - k_{\mathrm{el}}c, \qquad \dot{e} = k_{\mathrm{in}}\frac{c}{c + E_C} - k_{\mathrm{out}}e, \qquad \dot{a} = (e - a)/\tau.$$

The drive d follows a step, a ramp, or a bolus followed by washout. The apparatus records a, while the theory under test declares that the channel reports e. Saturation in c together with bounds on e and a makes this prior nonlinear, so Proposition 1 does not apply directly; Appendix M analyzes it separately.

Injected defects. We study four defect types. For the *admissible-set* defect, the generating theory gives the instrument state a lower bound while the theory under test omits it. For the *observation-model* defect, the theory under test reports the underlying physical state instead of the lagging instrument state. For the *role* defect, the generating theory treats the drive as an external input while the theory under test treats it as a state. For the *law* control, the theory under test removes the final term from one rate law.

Within each prior, different draws change only the coefficients, not the system structure. Table 3 therefore uses 50 coefficient draws for each of two structures with the observation defect and 25 draws for the role control.

Coverage of the repair vocabulary. We also ask how much of a broader set of scientific repairs can be expressed by the operation table. Table 6 summarizes 55 episodes and separates two questions: whether a known resolution can be represented by one of our repair operations, and whether independent judges can map an episode onto the same vocabulary.

For the second question, independent authors wrote 30 episodes without seeing the operation table. Judges then mapped those descriptions onto the repair vocabulary without seeing the code or one another's decisions. Across 45 judgments, 32 map onto one of the defined forms, covering 21 distinct episodes; 9 episodes map to none. On the subset judged twice, the two judges agree on 14 of 15 episodes, with $\kappa = 0.912$ and chance agreement 24 %.

Table 6: Coverage of the repair vocabulary over 55 scientific episodes. The upper block counts which known resolutions can be represented by operations in Table 1. The lower block measures whether blind judges can map independently written episode descriptions onto the same vocabulary.

resolving move writable as an operation of the table	40
of those, mapping to `EditRelation`, which a law search already has	35
of those, mapping to `EditDomain`	5
of those, mapping to `EditObservation`	0
of those, mapping to `SetRole`	0
writable, but resolved by a component the interpreter does not read	15
built as a runnable theory, with a computed trigger	3
blind judging: distinct episodes mapping onto the vocabulary	21 of 30
blind judging: judgements mapping onto the vocabulary	32 of 45
blind judging: episodes no form covers	9

Historical cases resolved outside the law. The broad case set above was assembled without selecting for where the final repair occurred. None of its resolutions maps to `EditObservation` or `SetRole`, so it cannot establish whether those operations occur in scientific practice.

We therefore assembled a second, deliberately targeted set of 12 historical episodes in which the literature proposed a change to the physical law but an instrument effect or the experimental role of a quantity played a central part in the resolution. The set includes N-rays, the 1919 eclipse plate scale, lunar-longitude fluctuations, polywater, solar oblateness, the 17 keV neutrino, cold-fusion calorimetry, DAMA modulation, the Pioneer anomaly, varying fine-structure constant, the OPERA neutrino-speed anomaly, and BICEP2 B-modes. Of these cases, 9 map to `EditObservation` and 3 to `SetRole`.

Two fresh judges, given only the coding protocol and episode descriptions, agree with each other on 12 of 12 cases ($\kappa = 1.00$) and with the dataset coding on 11 of 12 cases ($\kappa = 0.75$). The 1 disagreement concerns the open DAMA case: the dataset codes it as a seasonal drive on the detector, while both judges assign an observation-model effect at low confidence.

Because we selected this second set specifically for instrument- or role-based resolutions, it demonstrates that these repair types have historical examples; it does not estimate how frequently scientific anomalies are resolved in this way.

Cases the current interpreter cannot execute. The remaining 15 writable-but-unevaluable episodes involve dropping or narrowing a principle. The corresponding repair operations exist in the schema, but the interpreter currently stores constraints as Python callables rather than symbolic expression trees. Converting constraints to symbolic expressions would make these repairs executable. It would not, by itself, provide the computed triggers needed to turn every writable historical episode into a runnable case, which explains the remaining gap between 40 and 3.

E GENERALIZATION TO HELD-OUT RECORDINGS

The main experiments fit or select repairs using the available recording. To test whether those repairs merely overfit a particular intervention, we fit on one intervention and evaluate on two held-out interventions that the methods never see during selection.

For our search, 65 edits reproduce the training recording, and 61 of them also reproduce both held-out recordings. More importantly, the cheapest accepted edit—the one returned by the search—generalizes in 44 of 48 cases.

The failures all come from admissible-set defects. In those cases, one intervention may never reach the missing bound, while a held-out intervention does. The returned edit therefore transfers on 8 of 12 admissible-set cases. By contrast, on the observation-model cases central to absorption, the returned repair transfers on 12 of 12 cases. The ambiguity studied in Section 4 is therefore not caused by fitting only one intervention.

PySINDy shows the expected increase in prediction error when moved off its fitted intervention: its median relative error rises from at most 0.0199 on the fitted recording to 0.3039 on held-out recordings. If we use the held-out recordings to choose its hyperparameters, it also transfers. Held-out prediction therefore does not by itself separate the two approaches. The important difference is the form of the answer: our search returns a repair to a named theory component, whereas PySINDy returns a fitted dynamical law.

What fitted coefficients reveal about the faulty component. A fitted SINDy model does not contain explicit coordinates for an instrument's time constant, a concept's admissible values, or whether a quantity is externally controlled. However, these defects can still leave characteristic patterns in the fitted coefficients.

We test this by fitting PySINDy before and after the relevant change and using the coefficient difference as a feature vector. A nearest-neighbour classifier trained on labelled examples from the same distribution identifies the defective component with 98 % accuracy in the first arm and 90 % in the second, compared with 25 % chance.

Part of the second result requires caution. In 12 of 48 cases, the fitted coefficient vectors are identical before and after the change, so the difference vector is exactly zero and the class can be inferred from that degeneracy alone rather than from a richer coefficient pattern.

These results show that a fitted law can contain enough information to predict which component is wrong. The comparison is nevertheless different from direct theory repair. The classifier requires labelled examples of every defect type from the same distribution, together with a chosen feature library and decision rule. It returns one of four defect categories. Our search instead returns the component to change, the proposed replacement, and an executable theory that reproduces the recording without using labelled defect examples.

Across 48 drawn cases, the median relative fit error is 0.1741 in the first arm and 0.0 in the second. Thus the distinction is not that PySINDy fails to fit the data or that its coefficients contain no diagnostic information; it is that the two methods produce different kinds of explanations.

F THE TWO-EDIT ABSORBING ACCOUNT AND FITTED-LAW BASELINE

Section 4 shows that the lagging-sensor defect can be reproduced without changing the observation model. Here we describe the law-only account found by the search and test whether simply fitting the constants of alternative law rewrites removes the ambiguity.

The two-edit law-only account. For the instrument case, the cheapest account that changes only laws uses two edits. The first replaces the position appearing in the spring term of $\dot{v}$ with a scaled version of the instrument state. The second changes the instrument state's rate law so that it follows the drive rather than the position. Together, these edits cost 5, compared with 1 for the single observation-model edit that correctly declares what the instrument reports. Both accounts reproduce every declared intervention to 5.1e-16 relative error.

The search can use the instrument-floor parameter as a dimensional conversion factor even though no original relation depends on it. This parameter is not essential: after removing it, the same absorbing account remains available at cost 6. Thus, the result does not depend on an otherwise unused parameter.

Why the search must be staged. The second law edit is initially invisible to a localized search. In the theory under test, the instrument state lies outside the dependency cone of the recorded channel, so there is no reason to edit its rate law. The first edit changes that dependency structure by inserting the instrument state into $\dot{v}$. Only then does the instrument's own rate law enter the relevant dependency cone and become a candidate for the second edit.

This is why we search in stages. A method that localizes once, prunes everything outside the initial dependency cone, and then returns its cheapest repair would never consider this absorbing account. It could therefore report a unique explanation only because the competing explanation was excluded by the search procedure.

Does fitting every constant remove the problem? A symbolic regressor changes both the functional form and its constants. We therefore test whether more aggressive parameter fitting eliminates the gap between law rewrites and the repair that names the faulty component.

For every rewritten theory considered, we fit all free parameters by least squares using Nelder–Mead with multiple restarts followed by gradient refinement. We evaluate two sets of law rewrites: the cheapest according to our representation cost and an equally sized uniform random sample. The second arm avoids giving our own cost function an unfair advantage. In each arm we fit 150 of the 3,708 proposed law rewrites for the sensor case, while fitting every candidate that directly repairs the faulty component.

On the lagging-sensor case, fitting does not recover the correct explanation. The best fitted law rewrite reaches 0.002161 relative error, better than the fitted theory under test, while placing the instrument state inside the physical dynamics. The repair that names the instrument reaches 0.0. Thus, a well-fitted but physically misplaced explanation remains possible even when every constant is free.

The role defect provides a control. There, the best fitted law rewrite reaches 0.137244, compared with 0.0 for the repair that corrects the role. In this case, additional parameter fitting does not create a competitive law-only explanation.

Table 7: Fitting all free constants for proposed law rewrites and for the repair that names the defective component. Lower values indicate better fits. We evaluate both the cheapest law rewrites and an equally sized uniform random sample to avoid selecting candidates only with our own cost function.

	lagging instrument	exogenous drive
law rewrites proposed	3,708	2,513
fitted in each arm	150	150
theory under test, constants fitted	0.005088	0.346149
best law rewrite, cheapest arm	0.002161	0.228287
best law rewrite, random arm	0.002161	0.137244
best law rewrite, either arm	0.002161	0.137244
repair that names the component	**0.0**	**0.0**

The experiment therefore separates two questions. Better fitting can improve a candidate's agreement with the recording, but it cannot determine which theory component should carry the discrepancy. In the sensor case, the central ambiguity survives even after fitting every constant.

G WHY REACHABILITY ALONE CANNOT CERTIFY THE GENERATOR

Section 2 claims that the generator proposes repairs from the theory rather than retrieving known answers. A natural evaluation is therefore *reachability*: a separate module constructs target theories, and we measure how many targets the generator can reach.

Reachability alone cannot certify this claim. If the generator can access the target builder during evaluation, it can simply borrow the targets and pass any test that depends only on the proposal set and the target set. More generally, if an honest generator satisfies some acceptance rule over those two sets, a borrowing generator can evaluate the same rule and adapt its output until it passes. We withdrew 8 successive evaluation designs after constructing attacks of this form. Appendix A gives the attacks and their reproductions.

This leaves two principled ways to escape the problem. First, the evaluator can keep part of the test hidden from the generator. Second, the generator can commit before the targets are drawn. Our current setup supports neither, so we do not treat reachability as a certificate of answer-independent proposal generation.

Practical checks without an oracle. Instead, we combine three kinds of evidence: injected defects that must change named test outcomes, ablations that remove generator components and must be detected, and explicit reporting of test rows that no injected defect covers. These practices test

whether the evaluation responds to meaningful changes in the generator rather than only whether the generator reaches a target set.

This strategy connects to several established evaluation traditions. Mutation testing asks whether deliberate faults change test outcomes (DeMillo et al., 1978; Jia & Harman, 2011); metamorphic testing checks relations between executions when no direct oracle is available (Chen et al., 1998; Segura et al., 2016); and partial-input baselines expose shortcuts that succeed without using the intended information (Gururangan et al., 2018; Poliak et al., 2018; Niven & Kao, 2019). Related ideas appear in item-response analysis (Lalor et al., 2016; Rodriguez et al., 2021), contamination detection (Carlini et al., 2018; Sainz et al., 2023; Deng et al., 2024), construct-validity analysis (Raji et al., 2021; Bowman & Dahl, 2021; Srivastava et al., 2023), and build-it-break-it-fix-it evaluations (Ruef et al., 2016; Ettinger et al., 2017; Kiela et al., 2021).

Two results are especially close to our negative finding. Jacovi et al. (2023) argue that post-hoc contamination detection cannot provide reliable guarantees and that mitigation must instead be structural. Oren et al. (2023) give a provable black-box test that avoids the same circularity by using a hidden shuffle: the benchmark controls information that the evaluated system cannot inspect.

The conclusion is therefore deliberately limited. Our tests provide evidence that the shipped generator depends on its own proposal mechanisms and reacts to targeted defects, but they do not prove that no hidden lookup strategy could pass the evaluation.

H External systems: thermal measurements and DiscoverPhysics

Section 5 uses two sources that were not generated by the priors in Section 3: an independently written thermal simulator together with a public real-world recording, and simulation worlds from DiscoverPhysics (Wiemann et al., 2026). This appendix gives the experimental details behind those results.

Independent thermal simulator. The simulator in `external_rep/benchtop_thermal/benchtop.py` was written from a physical specification by an agent instance with no access to this repository. It models a heated aluminium block,

$$C\dot{T} = P - G(T - T_{\mathrm{amb}}),$$

and a thermocouple sheath with first-order lag,

$$\dot{T}_s = (T - T_s)/\tau_s.$$

The recorded signal is

$$aT_s + b$$

plus random-walk drift and Gaussian noise, quantized to 12 bits over 0–200 °C. The simulator integrates with RK4 substeps, samples once per second, and limits the heater power to 0–30 W. Its default parameters are $C = 500$ J/K, $G = 2$ W/K, $\tau_s = 8$ s, and $\sigma = 0.05$ °C.

The evaluation code does not read these hidden simulator parameters. It interacts with the simulator only through `run(power, channels)` and `describe()`.

Two competing explanations. We fit two sampled linear models to the thermocouple signal over three power schedules: a random walk, random steps, and a square wave. We constrain positive parameters through a log parameterization and use three optimization starts.

The first account places the extra lag in the physical dynamics:

$$\dot{H} = \frac{P - K_c(H - T)}{C_h}, \qquad \dot{T} = \frac{K_c(H - T) - G(T - T_{\mathrm{amb}})}{C},$$

and declares the recorded channel to be T.

The second account places the lag in the instrument:

$$\dot{T} = \frac{P - G(T - T_{\mathrm{amb}})}{C}, \qquad \dot{S} = \frac{T - S}{\tau},$$

and declares the recorded channel to be S.

After fitting, the first account has poles -0.13046 and -0.0040, while the instrument account has -0.13046 and -0.0040 (s^{-1}). Their thermocouple residuals are 0.064 and 0.064 °C, respectively. On 5 new schedules, the larger residual is 0.133 °C and the two predictions differ by at most 5.7e-08.

The structural test of Appendix O predicts that this ambiguity is possible: the direct feedthrough matches, the theory declares two states for a degree-two response, and the required dimensions are writable (yes).

The measurement that separates them. The two accounts agree on the thermocouple but disagree about the block temperature itself. On the selected square-wave experiment, their block-temperature predictions differ by up to 0.413 °C.

We therefore request the block temperature as a second channel. Against that measurement, the instrument-lag account has residual 0.106 °C, with maximum 0.287, while the heater-element account has residual 0.278 and maximum 0.534. The thermocouple noise floor is 0.055, and the resulting verdict is unique.

This experiment is run on the independent simulator, not on a physical benchtop apparatus.

The same accounts on a public real recording. The Temperature Control Lab (Park et al., 2020) is an Arduino-based system with two heaters and two thermistors. We use a public recording containing 601 samples at one-second intervals with temperature resolution 0.01 °C.

For the first thermistor, driven by both heaters, the heater-element account reaches 0.64 °C and the instrument-lag account reaches 0.54. The corresponding fitted lags are 12.1 s in the heater element and 12.9 s in the sensor.

Both residuals remain well above the measurement resolution, so neither account should be interpreted as the true physical model of the board. More importantly, the two accounts disagree about the block temperature by up to 4.5 °C, but the public dataset does not contain a direct measurement of that quantity. The separating experiment can therefore be proposed but not executed on this recording.

The board's second thermistor does not provide the missing block-temperature channel. When both thermistors are fitted jointly, the heater-element account places the two lags at 38 and 44 s, while the instrument account places them at 23 and 12 s. On the second thermistor, the residuals are 0.33 and 0.34 °C, and the two accounts differ by at most 0.28 °C, below the error left by either fit. On the first thermistor, the joint residuals are 0.20 and 0.45 °C. Thus the available second thermistor does not resolve the ambiguity.

Six DiscoverPhysics worlds. We next evaluate the repair families on DiscoverPhysics (Wiemann et al., 2026) at commit 33b7fa9, using the benchmark's own executor.

The most informative case is `oscillator`. It contains two particles in a two-dimensional Poisson field whose coupling varies with absolute time: the same pair attracts during one quarter period and repels during the next. A theory that treats the coupling as constant has no concept in which to represent this dependence.

For a release from rest at fixed radius over 320 samples, the best constant coupling reaches 0.061 relative error. Adding a clock as a concept, giving it a rate law, and letting the coupling depend on that clock reaches 5.8e-05. This requires two edits, `InsertConcept` and `InsertRelation`, and improves the fit by a factor of 1,049; the fitted coupling is 0.7962.

Within the tested position-only law family, no force-law rewrite reproduces this trajectory. For a radial release, a position-only central force is conservative, so repeated turning points should occur at the same radius. Instead, the observed turning radii 2.787 3.042 2.873 3.164 3.037 3.357 move outward. This argument applies only to the tested family; velocity-dependent forces fall outside it.

For the six-world comparison, we release the test particle tangentially so that trajectories do not pass through the field center. We compare three model families on each recording:

1. the theory under test;

2. law rewrites of the forms

$$kr^{-a}, \qquad k_1 r^{-a} + k_2 r^{-b}, \qquad kr^{-a} e^{-r/\lambda};$$

3. the theory under test augmented with a clock concept that controls the coupling.

All coefficients carry the dimensions required by their exponents, and we fit constants by grid search followed by refinement.

We report the maximum relative position error over each recording. A family counts as reproducing the recording when its error is at most 0.0066, chosen as five times the error floor reached by the law family on the control world `gravity`.

world	the theory under test	a law rewrite	a declared clock	cheapest family that reproduces it
`gravity`	0.0042	0.0013	1.1558	the theory under test
`yukawa`	0.2204	0.0021	0.2255	a law rewrite
`fractional`	0.1811	0.0005	0.1862	a law rewrite
`extra_dimensions`	0.0050	0.0028	1.1725	the theory under test
`coulomb_easy`	0.4254	0.0020	3.5204	a law rewrite
`oscillator`	0.0578	0.0435	0.0002	a declared clock

At this threshold, the original theory reproduces both `gravity` and `extra_dimensions`. Of the worlds that depart more strongly from the starting theory, 5 are repaired by a law rewrite. The remaining time-dependent world requires the clock representation, which is why the final row tests a different kind of repair rather than merely a larger law family.

I Scaling of enumeration and canonicalisation checks

Section 6 notes that the generator does not scale indefinitely as the theory vocabulary grows. Here we measure where that failure begins.

We start from the same probe theory and add parameters that no relation uses. These extra parameters do not change the underlying physics, but they enlarge the set of expressions the generator can construct. For each vocabulary size, up to 16, we sample a fresh set of 100 target theories and measure three quantities: the number of proposals, the time required to generate and canonicalise them, and the fraction of targets reached.

As the vocabulary grows, the number of proposals increases from 17,509 to 273,178, while generation and canonicalisation time rises from 3 to 75 seconds. Over the same range, reach falls from 97 to 43 of 100. This experiment isolates a simple scaling effect: adding vocabulary that carries no additional physics still expands the combinatorial search space and reduces one-step coverage.

nuisance parameters	proposals	distinct theories	seconds	reach
0	17,509	16,831	3	97/100
2	25,990	25,168	7	88/100
4	41,199	40,069	10	82/100
8	91,685	89,510	21	68/100
16	273,178	267,184	75	43/100

The result does not mean that the target repairs become impossible to express; rather, exhaustive local enumeration becomes less effective as more interchangeable building blocks become available. This motivates more structured proposal mechanisms for larger theories.

Fuzzing the canonicaliser. The search stores one node per canonical form, so the canonicaliser must satisfy two basic requirements: applying it twice should not change the result, and canonicalisation should preserve the value of the expression.

We test both properties on 12,000 randomly generated expressions evaluated at 47,968 points, restricting the test to cases in which every subexpression on both sides remains finite. We observe 0

idempotence violations and 1 value differences. The observed value differences arise from floating-point reassociation near cancellation rather than from a change in the symbolic expression's intended meaning.

Canonical equality is also stricter than observational equivalence. Two different canonical expressions can still produce the same behavior on all experiments considered. Across the four defect types in Table 2, the number of canonical forms exceeds the number of distinct observable behaviors by a factor of 2.1 to 4.4 (`phase5/quotient.py`).

Thus, canonicalisation is useful for removing obvious syntactic duplicates, but it should not be interpreted as computing equivalence classes of experimentally indistinguishable theories. Counts of canonical theories in the paper are therefore upper bounds on the number of distinguishable behaviors.

J LANGUAGE-MODEL BASELINE AND STAGE-BY-STAGE EVALUATION

Section 5 asks whether a general language model can carry out the same reasoning steps without our search machinery. We evaluate two models on the same 144 cases, with 12 draws for each defect type and prior. The open-weights `gpt-oss:20b` runs locally at temperature zero. For `claude-opus-5`, each case is given to a fresh agent instance with no memory of other cases.

Identifying the defective component. Each prompt shows the theory under test component by component, the apparatus recording, and the theory's prediction for the same experiment. The model must return exactly one of the four defect types. The complete prompt is stored in `phase5/baseline_llm.py`.

We discarded an initial run because filenames contained the defect type and one model explicitly referred to that information. We reran every case with opaque hashed filenames. Under this protocol, `claude-opus-5` identifies the defective component on 144 of 144 cases, including 100 % on the observation-model and role defects and 100 % on the other two. The open-weights model reaches 51 %.

One feature makes the admissible-set cases easier: the theory declares an otherwise unused parameter with the same dimension as the missing bound. The frontier model sometimes cites this parameter when identifying the defect. Our search receives the same declaration, so we retain these cases but report the cue explicitly.

defect in	the model answered				
	domain	observation	role	law	none
domain	24	7	0	4	1
observation	0	12	0	23	1
role	1	13	5	17	0
law	2	0	0	32	2

A constant predictor that always answers LAW gets 36 of 144 cases, or 25 %. The confusion matrix therefore shows both whether the model exceeds this baseline and which defect types account for the difference.

Constructing a law-only account. We next test the remaining reasoning steps independently, using a fresh frontier-model instance for every prompt (`phase5/llm_stages.py`). The model cannot run code; its output is parsed and executed by our evaluation.

Stage B asks the model to write rate laws, in the paper's expression language, that reproduce the recording while leaving the observation model unchanged, or to return `none`. An answer counts only if it parses, typechecks, and reproduces every experiment within the case tolerance. We evaluate 36 observation-defect cases and 36 role cases as controls. The resulting laws reproduce 35 observation cases and 0 role controls.

The nonlinear toxicokinetic prior exposes an important failure mode. Although Proposition 1 does not apply when the apparatus reaches a bound, the model still produces reproducing law-only ac-

counts on 11 of 12 cases. In 11 of these answers, it explicitly writes the clamp into the law using

$$(x + y \pm \sqrt{(x - y)^2})/2$$

and encodes the simulator step through a literal multiple of τ. These equations reproduce the sampled simulator rather than the underlying continuous system.

We test this directly by halving the integration step while doubling the horizon and holding each drive value for two half-steps. Of the model's 35 previously reproducing accounts, only 24 still reproduce the new recording. All surviving cases come from the two linear priors, while only 0 of the 11 clamp-based accounts survive. By comparison, all 18 law-only accounts returned by our search remain valid.

Thus, on the toxicokinetic prior, the model can fit a numerical discretization even when our search finds no law-only account for the continuous dynamics.

Judging and separating two accounts. Stages C–E operate on 18 absorbed cases from Table 3, with both competing accounts written explicitly in the prompt.

Stage C gives the edit costs and asks for the three-way verdict from Section 4. On this population every case is cost-identifiable, so a constant majority answer performs equally well; we report this stage only for completeness.

Stage D lists the available experiments—three drives, each either recorded alone or paired with a probe on one declared state—and asks which experiments distinguish the accounts. We score both each individual experiment and the exact separating set. The model returns the exact set on 18 of 18 cases and reaches 100 % accuracy per experiment. A simple rule that declares every probed experiment separating reaches 100 %, so this stage mainly checks whether the model can reproduce the known structure of these cases.

Stage E asks for the transfer function from the drive to the recorded channel under each account. We parse the returned expressions symbolically and compare them with the transfer functions computed from the theory objects at four frequencies. The model gives both transfer functions correctly on 18 of 18 cases.

Generalization to randomly drawn structures. We repeat the executable-law, experiment-selection, and transfer-function stages on the randomly structured lag systems of Appendix P, using the account constructed from Proposition 1 as the competing explanation. The model produces reproducing laws on 7 of 7 systems, returns the exact separating experiment set on 7 of 7, and derives both transfer functions correctly on 7 of 7.

The simple "every probe separates" rule scores 108/108 per experiment on this population as well. We prioritized systems where that shortcut would fail, but none of the absorbable systems constructed so far has that property. Stage D should therefore be read as a format and consistency check rather than evidence of nontrivial experiment design.

stage	cases	n	the model	scored against
B, the alternative account	observation-model defect	36	35 reproduce, 1 say none	executed
B, the alternative account	role defect, control	36	0 reproduce, 20 say none	executed
C, the verdict	absorbed cases	18	18 right	constant answer 18
D, the experiment	absorbed cases	18	18 exact sets, 216/216 per experiment	probe rule 216/216
E, the identity	absorbed cases	18	18 both right (18 named, 18 absorbed)	symbolic
B on the population	drawn lag variants	7	7 reproduce, 0 say none	executed
D on the population	drawn lag variants	7	7 exact sets, 108/108 per experiment	probe rule 108/108
E on the population	drawn lag variants	7	7 both right	symbolic

Overall, the frontier model can perform many individual steps of the workflow on the linear systems: it can identify defects, construct executable alternative laws, recognize separating measurements, and derive transfer functions. The step-size intervention also shows the limitation of treating these outputs as equivalent to the structural analysis: a model can reproduce a sampled trajectory by encoding simulator-specific behavior that does not survive a change in discretization.

K Does the cost ranking depend on how a theory is written?

Section 4 compares competing accounts using a distance between theory representations. This raises a natural concern: if two mathematically equivalent theories can be written in different forms, the cost ranking might reflect notation rather than a meaningful difference between the repairs.

We test this in two ways. First, on the main absorbed example, we compare five structural cost measures. Their numerical values differ substantially—from 1 versus 2 under one measure to 1 versus 14 under another—but they agree on which account is cheaper on 5 of 5 comparisons.

We then rewrite the theory under test and both competing accounts using four classes of prediction-preserving transformations. Each transformation changes the representation while leaving the theory's behavior unchanged. For each class we draw 200 rewrites and keep only those for which all three theories still typecheck and reproduce their original recordings to within 10^{-9}. We then recompute the cost ranking.

Across 800 accepted rewrites of the reported pair, 0 change which account a cost measure prefers. The edit-count measure is omitted from this test because it is defined by the repair path and does not change under an equivalent rewriting. The remaining results are generated by `phase5/invariance.py`.

rewrite	kept	changed subtree	description	fields changed	tree distance
regroup	200	0	0	0	0
rename	200	0	0	0	0
reparameter	200	0	0	0	0
rescale	200	0	0	0	0

The same test across absorbed systems. We repeat the analysis over all absorbed systems in Table 3 using `phase5/invariance_all.py`. Each system is reconstructed from its original seed, and the two competing accounts are found again with the same searches. We exclude 0 systems whose reconstructed costs do not match the benchmark record.

For each remaining system, we draw 50 prediction-preserving rewrites of each transformation type. Across 100 absorbed systems, 20,000 rewrites pass the typing and behavioral checks, and 0 of them change which account is cheaper under any cost measure. Across the 100 systems for which all five measures can be reconstructed, the measures agree on the cheaper account in 100 cases.

rewrite	kept	changed subtree	description	fields changed	tree distance
regroup	5000	0	0	0	0
rename	5000	0	0	0	0
reparameter	5000	0	0	0	0
rescale	5000	0	0	0	0
all	20000	0	0	0	0

These results support the interpretation used in Section 4: the preference between the competing accounts is generally stable to prediction-preserving changes in how the theories are written, rather than being driven by one particular syntactic form or one particular cost definition.

L Further limitations and directions

Section 6 summarizes the main scope of the paper. Here we give the details and indicate how each boundary can be extended.

From controlled simulators to broader physical evidence. Most experiments use simulators written in the same typed representation that the search edits. This gives exact ground truth and lets us test whether a repair recovers the intended component without uncertainty about the simulator itself. Section 5 begins to move beyond this setting with an independently written thermal simulator, a public Temperature Control Lab recording, and DiscoverPhysics worlds.

A natural next step is to evaluate on more independently authored simulators and physical datasets that contain the additional channels selected by the experiment-design step. The current central case already has substantial separation: the two accounts agree on the observed channel to 5.1e-16 relative error but differ by 0.44 on the selected probe. Appendix R shows that the intended repair remains recoverable whenever the generating theory itself satisfies the noise-dependent acceptance criterion. Future experiments can test how often this margin remains large in real measurement settings.

Scaling the repair search. The operation table defines a broader repair space than the current generator can efficiently enumerate. On the probe theory, adding eight parameters that carry no additional physics reduces one-step reach from 97 of 100 to 68. Appendix I quantifies this effect.

The present implementation searches up to two edits and stages the search so that an early edit can expose new dependencies for the next one. This is sufficient for the absorption examples studied here, but larger theories will require more structured proposal mechanisms. Compositional search, learned proposal priorities, or symbolic pruning could extend coverage without enumerating the full expression space.

Broader comparison with existing discovery systems. We compare directly with PySINDy and PySR because both can run in our environment and expose the behavior relevant to the central question. AI-Newton (Fang et al., 2025) is the closest system conceptually, but its released pipeline invokes Maple 2024 for differential elimination. We therefore evaluate the compatibility of that stage rather than claiming a full end-to-end comparison.

The release caches the inputs and Maple outputs of its elimination calls, which lets us replay 730 distinct calls using the free DifferentialAlgebra implementation. The replay returns the same regular differential chains on 714 calls; 11 differ in the number of chains, with 7 retaining the same first chain used by AI-Newton, and 5 are not replayed because of a zero-equation case not handled by our script. The complete sweep takes 13.2 s.

These results suggest that a full open reimplementation is feasible. A careful comparison would also need to reproduce AI-Newton's timeout behavior, because its published law set depends on which Maple calls finish within the original ten–twenty second limit. Reconstructing that execution environment is therefore a useful future benchmark rather than something we approximate here.

From structural characterization to a complete decision procedure. Proposition 1 gives three conditions for a linear law-only account. The current experiments use them in stages. Feedthrough and state-count conditions can rule cases out directly, while the dimensional condition uses a sufficient lattice check and leaves unresolved cases to the construction or search.

The next theoretical step is to make condition (iii) complete: instead of asking whether every potentially required matrix entry can be written dimensionally, one could solve directly for a realization whose nonzero pattern respects the available dimensions. This would turn the current characterization into a complete typed-realisation procedure.

The state-count test also works best on exact or low-noise recordings. Our order estimator returns the correct degree through 1e-09 relative noise on 4 of 6 linear cases, while at 1e-06 it can return an incorrect order. More robust model-order estimation would extend this part of the test to noisier experiments.

From pairwise accounts to full equivalence classes. Section 4 compares two accounts: the cheapest repair that names the defective component and the cheapest law-only alternative. This makes the ambiguity and the separating experiment easy to analyze, but other repairs may also reproduce the same recording.

A natural extension is to maintain the full set of minimum-cost or observationally equivalent accounts and design experiments that split this entire set. The current transfer-function calculation already provides the right starting point for the linear case; the next step is to generalize the experiment-selection criterion from a pair of accounts to an equivalence class.

Extending the representation. The schema derives 21 valid operations, of which the current generator emits 16. The remaining 5 operations have explicit implementation reasons rather than being excluded conceptually. `Tune` is continuous and belongs to fitting; `SetDimension` and `SetParameterDimension` currently have no useful single-field instance; and `EditConstraint` and `InsertConstraint` constraint operations are blocked because constraints are represented as Python callables rather than symbolic expression trees.

Representing constraints symbolically would extend both the search and the interpreter. In particular, it would make `InsertConstraint` and `EditConstraint` searchable and would allow the interpreter to execute the 15 historical episodes in Appendix D. This is the clearest next extension of the five-component representation.

Sharper coding of scientific cases. We also tested whether the five components can be assigned reliably to arbitrary scientific research tasks. On PRL-Bench (Miao et al., 2026), two annotators agree on 75 of 354 tasks, with $\kappa = 0.04$. Rather than report a coverage estimate from such weak agreement, we use this result to identify where the representation needs a more explicit annotation protocol.

Future work can develop domain-independent coding rules, validate them with expert annotators, and then ask how often observation-model, role, domain, and law repairs occur in broader scientific corpora. The targeted historical set in Appendix D establishes that these repair types have real examples; a sharper protocol would allow their prevalence to be measured.

M LOCAL TEST OF THE PROPOSITION ON THE NONLINEAR PRIOR

Proposition 1 applies exactly to linear time-invariant systems. The toxicokinetic prior in Section 3 falls outside that setting because the damage response saturates, the damage variable is clipped to $[0, 1]$, and the plate reader is clamped at its background.

A local version of the argument is still possible. If the apparatus has a smooth equilibrium that lies strictly inside every admissible set, then its linearisation around that equilibrium is linear time invariant and Proposition 1 applies to first order. In that case, a law-only account exists locally exactly when the three structural conditions hold for the linearised apparatus and theory. If the equilibrium lies on an active bound, however, the local dynamics are not smooth across that point, so the linear proposition no longer gives a valid prediction.

Testing the local conditions. We evaluate this directly with `phase5/absorb_nonlinear.py`. For 6 observation-model defects, we first run the apparatus and theory under test with zero drive until they reach equilibrium, including all clamps. We then compute the Jacobian at that point and evaluate the same quantities used by the proposition: direct feedthrough, the McMillan degree of the linearised transfer function, and the number of declared states.

Ignoring the clamps, the Jacobian has degree 3 with 3 declared states and matching zero feedthrough, so the linearised model alone would predict a first-order law-only account on 6 of 6 seeds. The smoothness condition fails, however, on 6 seeds. With zero drive, the concentration and damage variables rest at their lower bounds and the plate reader rests at its background clamp, so all `a`, `c`, `e` relevant states lie on active bounds.

We therefore treat the proposition as inapplicable on these cases and measure the behavior directly. We scale the dosing schedules by 0.01, 0.1, and 1. At each scale, we record how long the plate reader remains clamped, test whether a linear model can reproduce the trajectory using the estimator from Appendix O, and run the staged two-edit search from Section 4. We set the acceptance threshold to 0.001 of the recorded signal range so that smaller drives do not receive a looser effective criterion.

drive scale	reader on its bound	linear by the order test	law-only account found
0.01	100%	0/6	0/6
0.1	18%	0/6	0/6
1.0	12%	0/6	0/6

At scale 0.01, the reader remains on its bound for 100 % of the recording; at scale 1, it does so for 12 %. At every tested scale, the order estimator classifies the trajectories as nonlinear: only 0 and 0 of 6 cases are called linear at the smallest and largest scales. The staged search likewise finds no law-only account, with 0 and 0 of 6 successes.

Making the drive smaller therefore does not move this system toward the linear regime. Instead, it keeps the plate reader on its clamp for longer. This is the nonlinear counterpart of the admissible-set condition in Proposition 1: once an active bound shapes the observed dynamics, a smooth linear realisation no longer describes the experiment.

These results do not rule out every possible nonlinear law rewrite. They show that the linear absorption criterion should not be extrapolated across an active bound, and that the law family searched here does not absorb the toxicokinetic observation defect under the tested conditions.

N Proof of Proposition 1

We prove

$$(a) \iff (b) \iff (c)$$

in three steps. First, under the excitation assumption, one finite recording determines the transfer function within the stated model class. Second, conditions (i)–(iii) give a valid law-only account. Third, any law-only account must satisfy those conditions.

Notation and sampling. Let

$$r = \max(n, d), \qquad m = 2r + 1.$$

We take $u[k] = 0$ for $k < 0$. The apparatus and every candidate start from zero state and use the same forward-Euler step Δ.

For

$$\dot{x} = Ax + Bu, \qquad y = c^\top x + e\,u,$$

the sampled system is

$$x[k+1] = (I + \Delta A)x[k] + \Delta B u[k], \qquad y[k] = c^\top x[k] + e\,u[k].$$

Writing

$$A_\Delta = I + \Delta A, \qquad B_\Delta = \Delta B,$$

its transfer function is

$$H_\Delta(z) = c^\top (zI - A_\Delta)^{-1} B_\Delta + e.$$

Since

$$zI - I - \Delta A = \Delta \left(\frac{z-1}{\Delta} I - A \right),$$

we have

$$H_\Delta(z) = H\left(\frac{z-1}{\Delta} \right), \qquad H(p) = c^\top (pI - A)^{-1} B + e.$$

The affine change of variable

$$p = \frac{z-1}{\Delta}$$

is invertible and preserves both McMillan degree and direct feedthrough. Thus

$$H_\Delta \equiv H'_\Delta \quad \iff \quad H \equiv H'.$$

Let the apparatus have transfer function

$$H^*(p) = C^*(pI - A^*)^{-1} B^* + D^*$$

of McMillan degree d. A linear law-only account has

$$\dot{x} = A_{\mathrm{ss}} x + B_{\mathrm{ss}} u$$

on the same n declared states and keeps the observation model

$$y = c^\top x + e\,u, \qquad c \neq 0.$$

Step 1: one sufficiently informative recording determines the transfer function. The zero-state sampled response has the convolution form

$$y[k] = \sum_{j=0}^{k} h_j u[k-j],$$

where

$$h_0 = e, \qquad h_j = c^\top A_\Delta^{j-1} B_\Delta, \quad j \geq 1.$$

Consider two systems that produce the same recording, and define

$$\delta h_j = h_j - h'_j.$$

Then

$$\sum_{j=0}^{k} \delta h_j u[k-j] = 0 \qquad \text{for every } 0 \leq k < N.$$

Let t_0 be the first index for which

$$u[t_0] \neq 0.$$

The persistent-excitation assumption of order m implies

$$t_0 \leq N - m;$$

otherwise the first column of the $(N - m + 1) \times m$ input-window matrix would vanish and that matrix could not have rank m.

At $k = t_0$,

$$0 = \delta h_0 u[t_0],$$

so

$$\delta h_0 = 0.$$

Suppose inductively that

$$\delta h_0 = \cdots = \delta h_{\ell-1} = 0$$

for some $\ell < m$. At $k = t_0 + \ell$, all terms involving input samples before t_0 vanish and all terms involving δh_j with $j < \ell$ vanish by the induction hypothesis. Hence

$$0 = \delta h_\ell u[t_0],$$

and therefore

$$\delta h_\ell = 0.$$

Thus

$$h_j = h'_j, \qquad j = 0, \ldots, m-1 = 2r.$$

It remains to show that these Markov parameters determine the transfer function. Let

$$F(z) = H_\Delta(z) - H'_\Delta(z).$$

The two systems have McMillan degrees at most d and n, so after taking a common denominator, F has denominator degree at most

$$d + n \leq 2r.$$

Equality of

$$h_0, \ldots, h_{2r}$$

means that the Laurent expansion of F at $z = \infty$ satisfies

$$F(z) = O(z^{-(2r+1)}).$$

If F were nonzero, its relative degree could not exceed the degree of its denominator, which is at most $2r$. Therefore a nonzero F cannot decay as $z^{-(2r+1)}$ or faster. Hence

$$F \equiv 0.$$

Therefore

$$H_\Delta \equiv H'_\Delta$$

and, by the affine change of variable above,

$$H \equiv H'.$$

Thus agreement on the stated recording implies agreement for every drive and every horizon:

$$(a) \Rightarrow (b).$$

The reverse implication is immediate, so

$$(a) \Longleftrightarrow (b).$$

Step 2: conditions (i)–(iii) are sufficient. Assume conditions (i)–(iii).

Condition (i),

$$D^* = e,$$

makes

$$H^*(p) - e$$

strictly proper. Let its McMillan degree be d.

For $d > 0$, classical realization theory gives a minimal realization

$$(A_d, B_d, c_d)$$

on d states (Kalman, 1963; Ho & Kalman, 1966). By condition (ii),

$$n \geq d,$$

so we may embed this realization into n states:

$$A_0 = \begin{bmatrix} A_d & 0 \\ 0 & 0 \end{bmatrix}, \qquad B_0 = \begin{bmatrix} B_d \\ 0 \end{bmatrix}, \qquad c_0 = \begin{bmatrix} c_d \\ 0 \end{bmatrix}.$$

The added states are unreachable and unobservable, so

$$c_0^\top (pI - A_0)^{-1} B_0 = H^*(p) - e.$$

Because $c \neq 0$ and, for a nonzero strictly proper transfer function, $c_0 \neq 0$, there exists an invertible matrix S satisfying

$$S^\top c = c_0.$$

For example, extend c and c_0 separately to bases of $\mathbb{R}^n$ and let $S^\top$ map the first basis to the second.

Then

$$\widetilde{A} = S A_0 S^{-1}, \qquad \widetilde{B} = S B_0$$

has the fixed output map c and satisfies

$$c^\top (pI - \widetilde{A})^{-1} \widetilde{B} = H^*(p) - e.$$

Thus conditions (i) and (ii) guarantee that the required untyped realization class is nonempty.

Condition (iii) states that within this class there exists at least one realization

$$(A_{\mathrm{ss}}, B_{\mathrm{ss}})$$

whose coefficients are writable in the theory's typed expression language. Choose such a realization. Its transfer function is

$$c^\top (pI - A_{\mathrm{ss}})^{-1} B_{\mathrm{ss}} + e = H^*(p).$$

The corresponding rate laws therefore define a valid law-only account and reproduce the apparatus under every drive.

Hence

$$(c) \Rightarrow (b).$$

If

$$d = 0,$$

then condition (i) gives

$$H^*(p) \equiv e.$$

Condition (iii) directly supplies a writable realization whose strictly proper part is zero, so the same conclusion follows.

Why condition (iii) is separate from ordinary realization theory. Classical realization theory allows arbitrary changes of state coordinates. A typed physical theory does not: its states can have different physical dimensions.

For

$$\dot{x}_i = \sum_j A_{ij} x_j + B_i u,$$

dimensional consistency requires

$$[A_{ij}] = [x_i][x_j]^{-1}\mathsf{T}^{-1}, \qquad [B_i] = [x_i][u]^{-1}\mathsf{T}^{-1}.$$

A mathematically valid state-space realization is therefore a valid theory only when coefficients with these dimensions can be expressed from the theory's declared parameters.

For example, in the oscillator of Figure 2, the declared parameters include quantities with dimensions T^{-1} and T, which can provide the dimensional factors needed when a realization mixes position and velocity. Another theory with the same state count but without suitable dimensional quantities need not admit the same physical realization.

Step 3: conditions (i)–(iii) are necessary. Assume (b): a law-only account reproduces the apparatus for every drive. Its transfer function is

$$G(p) = c^\top (pI - A_{\mathrm{ss}})^{-1} B_{\mathrm{ss}} + e.$$

Since

$$c^\top (pI - A_{\mathrm{ss}})^{-1} B_{\mathrm{ss}}$$

is strictly proper,

$$\lim_{p\to\infty} G(p) = e.$$

But

$$\lim_{p\to\infty} H^*(p) = D^*.$$

Equality of the transfer functions therefore forces

$$D^* = e,$$

which is condition (i).

With condition (i),

$$c^\top (pI - A_{\mathrm{ss}})^{-1} B_{\mathrm{ss}} = H^*(p) - e.$$

The left-hand side is a realization on n states, while the right-hand side has McMillan degree d. Since the McMillan degree is the minimum dimension of any realization,

$$d \le n,$$

which is condition (ii).

Finally, the assumed account is a valid theory in the typed representation. Therefore each coefficient of A_{ss} and B_{ss} has the required physical dimension and is represented by an allowed expression over the declared parameters. This is condition (iii).

Hence

$$(b) \Rightarrow (c).$$

Combining the three steps gives

$$(a) \iff (b) \iff (c).$$

□

A practical sufficient check for condition (iii). Condition (iii) is existential: it asks whether at least one realization with the required transfer function and fixed output map is writable. The implementation uses a stronger but decidable sufficient test.

For each required coefficient dimension, we search the lattice generated by the dimensions of the declared parameters, with parameter exponents bounded by two. Whenever the lattice contains the required dimension, the expression grammar can multiply the corresponding parameter monomial by a dimensionless numerical literal to obtain the desired coefficient value.

Thus, if every nonzero coefficient required by the constructed realization has a dimension represented in this lattice, that realization is writable and condition (iii) holds.

Failure of this lattice test is not a proof that condition (iii) fails. Another realization may contain zeros in precisely those matrix entries whose dimensions are unavailable. We therefore report such cases as *undecided* and leave them to the construction or the search. Appendix P reports how often each outcome occurs.

Constructive realization used in the experiments. The sufficiency argument also suggests a concrete procedure, implemented in `absorbable.construct`:

1. recover a Ho–Kalman realization of $H^* - e$ from the apparatus's Markov parameters;
2. embed it into the n states declared by the theory;
3. choose an invertible S satisfying
$$S^\top c = c_0;$$
4. transform the realization by
$$A_{\mathrm{ss}} = SA_0S^{-1}, \qquad B_{\mathrm{ss}} = SB_0;$$
5. express each coefficient using a dimensionally compatible parameter monomial and a dimensionless numerical multiplier.

The resulting rate laws are inserted into the same typed representation used by the search and are simulated directly against the recording.

On the drawn structures of Appendix P, the construction reproduces the recording on 16 of the 16 systems for which the predicate returns "yes", and on 1 of the 32 systems for which it returns "no". The staged two-edit search finds an account on 9 of the 17 systems for which the construction succeeds, and on 0 additional systems.

These differences do not contradict the exact proposition. The construction and search are numerical algorithms with different restrictions: the construction may rewrite every rate law, whereas the staged search permits only two edits. The construction also determines numerical rank using a relative tolerance of 10^{-9}, so modes below that threshold may be discarded even when the exact transfer function retains them.

Scope of the result. The proposition assumes linear rate laws in the states and drive, a linear time-invariant apparatus over the experiments considered, a declared experimental drive, zero initial state, and a fixed linear observation map. It does not directly cover nonlinear or time-varying dynamics, active bounds, unknown initial conditions, or observation maps defined through relations that the law-only account is itself allowed to rewrite.

Appendix M examines the active-bound case explicitly.

O WHEN CAN A LAW REWRITE ABSORB A DEFECT?

This section compares the structural prediction of Proposition 1 with the staged search on the same cases. The proposition is evaluated before search. "States" is the number of states declared by the theory under test, "degree" is the McMillan degree of the apparatus transfer function, and entries are left blank when an earlier condition already rules the case out. "(iii)" reports the sufficient

dimensional-writability test from Appendix N. "Built" reports whether the constructive procedure from the same appendix produces an account that reproduces the recording.

The construction succeeds on 12 of the 12 cases predicted absorbable and on 0 of the 12 cases predicted not absorbable. The staged search takes approximately 30 minutes on the oscillator prior and 123 on the reservoir prior. Results are generated by `phase5/absorbable.py`.

prior	defect in	states	degree	(iii)	why	can	built	found
oscillators	domain	—	—	—	a bound binds	no	no	no
oscillators	law	3	2	yes	states at least the degree	yes	yes	n/a
oscillators	observation	3	3	yes	states at least the degree	yes	yes	yes
oscillators	role	—	—	—	no drive to rewrite	no	no	no
reservoirs	domain	—	—	—	a bound binds	no	no	no
reservoirs	law	3	2	yes	states at least the degree	yes	yes	n/a
reservoirs	observation	3	3	yes	states at least the degree	yes	yes	yes
reservoirs	role	—	—	—	no drive to rewrite	no	no	no
constructed	observation	2	3	—	too few states	no	no	nothing

The two law-defect rows are controls. Their defect is a missing term in a rate law, so a law-only account trivially exists: the original law itself. The one-edit search in Table 2 finds it. The staged search shown here considers only two-edit paths that first enlarge the dependency cone, so it does not include that one-edit repair. We therefore exclude these control rows from the agreement count in Section 4.

Estimating the degree from the recording. The degree column can be estimated without access to the generating state-space model. We fit linear models of increasing order and take the smallest order that reproduces the recording in free run. On the rows of this table where both estimates exist, the recording-based degree agrees with the symbolic degree on 4 of 4 cases. On 2 rows, no linear order reproduces the recording; these are exactly the 2 cases in which a declared bound becomes active.

The larger noise study gives the same picture. On exact data, the estimated degree matches the symbolic degree on 6 of 6 linear cases. At 1e-09 relative noise, it remains correct on 4; at 1e-06 it can return the wrong order. The coarser question—whether any linear model reproduces the recording—remains unchanged at every noise level tested.

P Absorption across randomly drawn structures

Table 3 varies coefficients within two fixed system structures. Here we also vary the structure itself to test whether the conditions of Proposition 1 continue to predict absorption beyond those two examples.

Each apparatus is linear and time invariant and contains two to four states. State dimensions are drawn from

$$\{\mathsf{L}, \mathsf{L}\mathsf{T}^{-1}, 1\}.$$

We sample a sparse stable coupling matrix, give each nonzero coupling its own dimensionally valid parameter, inject the drive into a random state, and require that it reaches the recorded state. A first-order lag is then applied to one state and used as the recorded channel. The experiment contains 48 systems spanning 48 distinct structures.

We construct four variants of the theory under test:

- *lag*: all physical and instrument states are declared, so absorption is possible when the dimensional condition also holds;
- *cut*: the instrument state is omitted, so the theory can have fewer states than the apparatus degree and condition (ii) fails;
- *feedthrough*: the apparatus reports part of the drive directly while the theory does not, so condition (i) fails;

- *hidden*: the apparatus contains an additional coupled state that the theory omits, increasing the degree beyond the declared state count.

For every system, we evaluate the predicate of Appendix O, including the dimensional test for condition (iii), and then run the staged two-edit search. When the search finds a law-only account, we test the same family of changed drives and added measurement channels used in Table 3. In the table below, "predicted yes/no/?" gives the predicate outcome, while "found|yes" and related columns report the search result conditioned on that prediction.

variant	systems	predicted yes/no/?	built|yes	built|no	found|yes	found|no	found|built	extra channel resolves	drive change resolves
cut	12	3/9/0	3/3	1/9	0/3	0/9	0/4	0/0	0/0
feedthrough	12	0/12/0	0/0	0/12	0/0	0/12	0/0	0/0	0/0
hidden	12	1/11/0	1/1	0/11	1/1	0/11	1/1	1/1	1/1
lag	12	12/0/0	12/12	0/0	8/12	0/0	8/12	8/8	1/8

The predicate and search agree on 41 of the 48 systems for which the predicate returns a definite answer. The sufficient dimensional test returns a definite answer on every system.

Across the 9 systems where absorption occurs, adding a measurement channel resolves every one, 9 of 9. Changing the drive resolves 2; on those systems the search's account is already distinguishable from the named account by a drive change alone, so their verdict is unique, and on the others only an added channel decides, as in Section 4. The result of Section 4 therefore holds with both coefficients and system structure varied.

The "Built" column evaluates the constructive procedure from Appendix N. It reproduces the recording on 17 systems, and the two-edit search finds an account on 9 of those. The construction rewrites as many rate laws as the realisation needs, up to five on these systems, so it reaches accounts beyond the staged search's two-edit budget. On the lag and hidden variants the search finds every constructed account within its budget; the cut variant is where the construction's accounts lie beyond what the staged search reaches.

For comparison, on the fixed structures of Table 3, equal-cost ambiguous pairs occur on 0 of the 50 role-defect and 100 observation-defect systems.

Q WHY CHANGING THE DRIVE CANNOT SEPARATE THE TWO ACCOUNTS

The central absorption example contains two linear accounts: one places the lag in the instrument, while the other absorbs it into the laws. To determine what experiments can distinguish them, we compare their transfer functions symbolically.

For a linear account,

$$H(p) = C_{\mathrm{ss}}(pI - A_{\mathrm{ss}})^{-1}B_{\mathrm{ss}} + D_{\mathrm{ss}},$$

where the subscript distinguishes the state-space matrices from the theory components named elsewhere in the paper. We construct these matrices directly from the theory objects, leave all declared parameters symbolic, and compare the resulting rational functions.

On the recorded channel, the two transfer functions are identical. Therefore every possible drive produces the same recorded response, not merely the finite set of drives used in the search. No input redesign can separate the two accounts on that channel. The same symbolic calculation also identifies which additional state measurements have different transfer functions and can therefore resolve the ambiguity. Results are generated by `phase5/distinguishability.py`.

what is recorded	names the instrument	absorbs it into the laws	the two
ch0, what the apparatus records	$\frac{f}{cp^2\tau+cp+kp\tau+k+p^3\tau+p^2}$	$\frac{f}{cp^2\tau+cp+kp\tau+k+p^3\tau+p^2}$	identical
a probe wired to s	$\frac{f}{cp^2\tau+cp+kp\tau+k+p^3\tau+p^2}$	$\frac{\mathrm{floor}}{p\tau+1}$	different
a probe wired to v	$\frac{fp}{cp+k+p^2}$	$\frac{fp}{cp^2\tau+cp+kp\tau+k+p^3\tau+p^2}$	different
a probe wired to x	$\frac{f}{cp+k+p^2}$	$\frac{f}{cp^2\tau+cp+kp\tau+k+p^3\tau+p^2}$	different

A system-identification baseline on the same case. The appropriate comparison is also with a method that identifies the input–output system rather than forcing all discrepancy into a physical law. We therefore fit an ARX model by least squares to the drive and recorded channel over the three experiments in Section 4, then run it forward without feeding measured outputs back into the model (`phase5/greybox_baseline.py`).

An order-two model leaves 9e-02 relative free-run error. Order 3 reaches 4e-10, while order four reaches 2e-11 and introduces a spurious pole. The order-3 poles match those of the sampled apparatus to 2e-12.

Thus the identifier recovers the input–output dynamics that the single recorded channel determines. It does not, however, determine the physical interpretation of the internal states. The instrument-repair account and the law-only account are two state-space realizations of the same transfer function, related by a similarity transformation. From this channel alone, an identifier can recover their shared input–output behavior but cannot decide whether the lag belongs to the physical dynamics or to the measurement process.

A second output breaks that equivalence. This is exactly the additional probe selected in Section 4.

R Robustness to noise and instrument shifts

Section 3 reports the one-edit results on exact recordings. Here we repeat the same analysis after adding Gaussian noise to each recorded channel.

For each noise level, we keep the same acceptance rule—a bound on the largest relative residual—but set the tolerance to five noise standard deviations, never below the exact-data threshold. Each table entry reports, out of 12 draws, how many cases are repaired by editing the component that contains the injected defect and how many can instead be reproduced by a single law rewrite. Where shown, the number in parentheses is the control: how often the generating theory itself satisfies the same noisy-data criterion.

prior	defect in	exact	0.0001	0.001	0.01	0.03
oscillators	domain	12/0	12/0	12/0	12/0	12/4
	law	12/12	12/12	12/12	12/12	12/12
	observation	12/0	12/0	12/0	12/0	12/5
	role	12/0	12/0	12/0	12/0	12/0
reservoirs	domain	12/0	12/0	12/0	12/0	12/5
	law	12/12	12/12	12/12	12/12	12/12
	observation	12/0	12/0	12/0	12/0	12/0
	role	12/0	12/0	12/0	12/0	12/0
toxicokinetics	domain	12/0	12/0	12/1	12/9	11/11
	law	12/12	12/12	12/12	12/12	12/12
	observation	12/0	12/0	12/0	12/0	11/5
	role	12/0	12/0	12/0	12/0	12/0

Across all noise levels and priors, the repair to the injected component succeeds whenever the generating theory itself passes the acceptance criterion. Noise therefore does not remove the diagonal pattern of Table 2. What changes is the off-diagonal: as the tolerance widens, some incorrect law rewrites also become acceptable. This first occurs by 0.03 relative noise and appears earlier for the nonlinear toxicokinetic prior. At the highest level tested, a single law rewrite reproduces 30 of the 106 recordings that the intended component repair also explains.

Thus, noise mainly reduces our ability to *exclude* alternative explanations; it does not prevent the correct component repair from reproducing the recording when the underlying theory itself remains within tolerance.

The absorbing account under noise. We also repeat the two-edit absorption search from Section 4 at 4 noise levels. The same law-only account is recovered at every level, with cost 5. At the largest tested noise level, 0.01, the acceptance tolerance is 5.0e-02, while the two competing accounts still differ by 0.44 on the additional channel selected by the experiment-design step.

The central ambiguity therefore survives the tested measurement noise: the two accounts remain indistinguishable on the original recording but remain separable by the added measurement.

Sensitivity of fitted laws to an instrument change. Section 5 shows that changing only the instrument can change the law inferred by PySINDy. Here we test whether that result depends on a particular library degree or on the threshold used to decide whether a coefficient has moved.

We sweep both choices. The first two columns of the table are the polynomial-library degree and the relative coefficient-change threshold. For each setting, we compare fits before and after the instrument shift and count how many coefficients change among those that are nonzero in at least one fit. The three result columns correspond to the two single-channel settings and to the setting in which all hidden states are declared. Each entry aggregates 48 drawn cases.

degree	threshold	one channel, second-order model	one channel, third-order model	every hidden state given
1	0.01	161/223	154/278	106/298
1	0.05	144/223	142/278	105/298
1	0.20	64/223	93/278	90/298
2	0.01	376/434	286/418	230/438
2	0.05	356/434	275/418	229/438
2	0.20	267/434	220/418	217/438

The row reported in Section 5 uses library degree 1 and threshold 0.05. The full sweep shows whether the same qualitative effect persists across alternative modeling choices rather than depending on that single setting.