

Where the Ladder Must Stand

Yuan-Jie Liu*

(Dated: October 31, 2026)

Abstract

Mathematicians often say that an object *had to* exist: you cannot subtract in the natural numbers, so the integers appear; you cannot solve $x^2 + 1 = 0$ in the reals, so the complex numbers appear. The same objects are also called inventions, and both descriptions are usually offered without a test. This paper proposes one, in the form of three questions that can be answered yes or no, and reports what happened when the test was turned on the author’s own work. Question 1: can the demand be met on the same elements? Question 2: among all extensions, is there a smallest one, determined up to a unique relabelling? Question 3: is the result new, judged against a catalogue of known objects? If Question 2 is answered yes, the object is pinned down: any two smallest extensions are the same up to a relabelling, and the relabelling is unique. Three steps of the number tower pass Questions 1 and 2. The step from the rationals to the reals does not: its missing minimality is recorded as a named obligation rather than proved, and no theorem depends on it. A counting invariant proposed inside the same programme is refused, because it collapses to a difference of prime-factor counts and is therefore already definable on the same elements; the refusal is argued by hand, since that layer contains no formal demand. Question 3 broke: the implementation of “is it new?” compared the sizes of carriers, and the integers, the rationals and the complex numbers all have the same size, so the test refused everything it was meant to admit. The failure is reported at the same length as the successes, because it is the clearest evidence that the test has teeth. What survives is two working questions, one specified but unimplemented third, and a short list of what this approach cannot do, including the fact that every verdict is relative to a stated triple of ambient theory, refinement and catalogue. The mathematical core is formalised in Lean 4 with Mathlib: no `sorry`, one named axiom, and no theorem that consumes it.

“Whatever can be defined may be questioned.”

— *Galactic Empire* (Isaac Asimov)

* yuanjieliu65@gmail.com

I. THE QUESTION

a. A story with a wrong answer. Suppose a program announces that it has discovered a new number: α , with $\alpha^2 = 7$. Nothing about this is wrong, and nothing about it is a discovery. The rationals do not contain a square root of seven, so α does extend them; but the extension is the field everyone already knows, with α written instead of $\sqrt{7}$. Any test that only asks “was there a solution before?” will certify this as new, and it will certify a great deal else besides.

The same weakness appears in the standard justifications of the number systems. “The natural numbers cannot solve $x + 1 = 0$ ” is true. It is also true of the one-element set, of a free group, and of the field of rational functions. All of them contain a solution; none of them is the integers. So the sentence *there was no solution, therefore we added one* does not pick out an object. It picks out a crowd.

This paper is about the missing part of that sentence. We wanted a test that a reader could apply to someone else’s claim — and, more usefully, to their own — and get an answer that does not depend on how the object was built. We built the test, checked its mathematical core in a proof assistant, and then pointed it at our own programme. It rejected one of our constructions, refused to certify another, and broke on a third. We report all three.

b. The idea in one sentence. An object is *forced* when the demand cannot be met on the old carrier, and the extension that meets it is the *smallest possible* in a precise sense — smallest not in size, but in the sense that it maps into every other extension, uniquely.

The word “smallest” needs care, because it is where most claims of necessity quietly cheat. Take the demand “ $x + 1 = 0$ must become solvable”. If we are allowed to keep working among structures where addition is only a semigroup operation, we can add a formal symbol ε with $\varepsilon + 1 = 0$ and stop. The result solves the equation and is not the integers: its addition is not a group operation. If instead we first agree to work among commutative rings, the same demand produces the integers, and produces nothing else. The difference is not in the demand. It is in the company the demand is allowed to keep.

So the test has to fix the company first. This is the single most useful lesson we took from the exercise, and it is easy to state: *changing the ambient theory changes the object, and the change of theory is the real content of the step.*

Background: the same object, built many ways

The number systems are the oldest worked example of this problem, and they are worked precisely because the answer is not in the construction. The integers can be built as pairs of naturals modulo a difference, or as the group completion of the additive monoid; the rationals as ordered pairs of integers, or as the fraction field of the integers; the reals as Dedekind cuts (1872) or as equivalence classes of Cauchy sequences; the complex numbers as pairs of reals (Hamilton, 1837) or as the quotient ring $\mathbb{R}[x]/(x^2 + 1)$. Every pair of these constructions gives isomorphic objects. Landau's *Grundlagen der Analysis* (1930) builds the tower from Peano's axioms, with the standard uniqueness statements; Ebbinghaus and co-authors' *Numbers* (1991) surveys several constructions of each system together with their equivalence, and is, as far as we can tell, the book a mathematician reaches for when they want the whole tower in one place [1–3].

That literature settles *what* the objects are. It does not settle the sentence that mathematicians actually use when they introduce them, which is causal rather than structural: *we could not subtract, so we made the integers*. The causal sentence is the interesting one, because it is the one that gets abused. A program that adds a formal square root of seven can say the same sentence, word for word, and it is wrong.

The philosophical version of the difficulty is older than the formal one. Benacerraf (1965) observed that if numbers are sets, then the natural number 4 could be the von Neumann 4 or the Zermelo 4, and nothing in arithmetic tells us which; so either the identification is arbitrary or numbers are not objects at all [4]. Structuralism replies that numbers are positions in a structure rather than objects with an inner nature [5]. We take no position on that debate. We take from it a design constraint: whatever test we propose must give the same answer for every construction of the same object. That constraint is what the second question in this paper enforces, and it is why the answer to the test is a statement about a universal property rather than about a set.

What has already been said about necessity

Mathematicians do have a standard name for *had to exist*: the object is *initial* (or *free*, or *universal*) in the relevant category. Mac Lane's textbook states and uses the fact that

an initial object is determined up to a unique isomorphism [6]. The specific instances are textbook too: the integers are the group completion of the additive monoid, the rationals are the fraction field, the complex numbers are the ring generated by a root of $x^2 + 1$. Model theory has a parallel notion: when a model completion exists it is unique up to equivalence, and the comparison embeddings are elementary and unique over the base model [7, 8].

So the ingredients are not new, and we say so in the paper rather than letting a referee say it. As far as we can tell from the literature cited here, what is missing is a *test*: a short list of questions that a reader can put to a claim of the form “this object had to exist”, including claims made by a program, together with an honest account of which questions can be answered mechanically and which cannot. The textbook statement is about a fixed category. A test has to survive a change of category, because that is exactly where the abuses live.

The question, stated plainly

With that, the question of the paper is narrow enough to answer. Given a structure and a demand, is there an object that the demand forces, and can we tell — mechanically, in a way a sceptical reader can repeat — when the answer is yes?

What we found, in advance

Four results, stated here so that a reader who stops after this page still knows what happened.

- *Three steps of the tower pass.* The integers, the rationals and $(\mathbb{C}, i)$ are each the smallest extension of the previous system once the ambient theory is fixed, and each is determined up to a unique relabelling. These are classical facts; the contribution is that one statement covers all three.
- *One step cannot be certified.* The reals are not proved to be the smallest complete ordered field over the rationals. We prove the failure that motivates the step and the cut property that describes the reals, and we register the missing statement as a named obligation that nothing depends on.

TABLE I. The problem this paper addresses, in the form we would want a referee to check. The last row distinguishes three levels of evidence rather than claiming one: checked, argued, asserted.

Item	Statement
Given	A structure S (objects with operations) and a demand D that S cannot meet.
Asked	Is there an object that D <i>forces</i> — one that every acceptable answer must be, up to a unique relabelling?
Obstacle 1	D may already be met on the same objects, in which case nothing is forced.
Obstacle 2	Many extensions may meet D , including trivial and chaotic ones; without a smallest one, “the” answer is not defined.
Obstacle 3	The answer may be an existing object under a new name.
Our answer	Three questions, one per obstacle: same-carrier solvability; existence of a smallest extension, unique up to relabelling; novelty against an explicit catalogue.
Delivered	Three levels of evidence, kept apart: machine-checked (three tower steps, the four failures, the two-score separation); argued by hand (the counting refusal); asserted (the minimality of $\mathbb{R}$). Question 3 is specified but not implemented, and our first implementation was refuted by our own examples.

- *One candidate is refused.* A counting invariant proposed inside our own programme collapses to a difference of prime-factor counts, hence is already definable on the same objects. The test refuses it, by hand.
- *The third question failed.* Our implementation of “is it new?” compared the number of elements, which makes the integers, the rationals and the complex numbers renamings of one another; the gate then refuses everything, including the objects the rest of the paper certifies.

II. WHAT OTHERS HAVE DONE

Four bodies of work bear on this question, and the boundary between them and this paper is worth drawing explicitly.

a. Building the numbers, and proving the result unique. The constructions are nineteenth- to mid-twentieth-century mathematics: Dedekind’s cuts, Cantor’s sequences, Hamilton’s pairs, and the algebraic constructions that turn a semiring into a ring and a ring into a field. The systematic account, with uniqueness proofs, is [3]; the axiomatic account from Peano’s axioms is [1]. This literature is the ground truth for every example in this paper, and we claim nothing new about the objects themselves.

b. Universal properties and the uniqueness of the comparison. The general statement — an initial object is determined up to a unique isomorphism — is in [6], and it is used constantly: free groups, localisation, polynomial rings, quotients. Model theory supplies the same shape in a different language, through model completions and the uniqueness of the comparison map [7, 8]. Our main theorem is a parameterised form of this statement: the ambient theory and the refinement are arguments, so the same four-line argument covers the whole tower. We are explicit in the text that this is packaging, not new mathematics.

c. Formalisation. The reals have been built in proof assistants before — in HOL Light [9], and earlier in HOL88 — and the algebraic hierarchy used here is Mathlib’s [10], in the Lean theorem prover [11]; this development uses Lean 4. Practical advice about the instance parameters that make such a hierarchy usable is in [12]. What a proof assistant adds to a paper of this kind is not depth but *auditability*: the definitions cannot silently mean something else, and a claim that a proof is four lines can be checked as a fact about the file rather than taken on trust. In our experience this is worth the cost mainly because it makes mistakes visible — two of ours are in this paper.

d. Machine discovery. Programs that propose mathematics have a long history, from Lenat’s AM [13] to recent systems that use search or learning to generate conjectures and constructions [14–16]. Their pattern is consistent: generate candidates, then decide which ones are worth keeping. The decision is made by a human, by a heuristic of interestingness (as in AM), or by a checker that tests truth or performance rather than novelty. This paper is about the decision step. It does not generate anything, and it should be read as complementary to those systems: a generator supplies candidates, a criterion decides which

of them may be called forced.

e. Philosophy: discovery, invention, and what numbers could not be. The debate this paper’s question belongs to is Benacerraf’s [4], with the structuralist response in [5]. We do not try to settle it. We take one thing from it: if the answer is to be independent of the construction, the test must be invariant under change of construction. That is a formal requirement, and it is what Question 2 enforces. The philosophical reading of “forced” remains a reading; what we can check is that the object is determined.

f. Where this paper sits. Not in the construction of the objects, not in the uniqueness theorem, and not in the formalisation. It sits in the combination: a short test, stated so that it applies to any theory, applied honestly to the author’s own programme, with the negative results reported at the same length as the positive ones. The closest neighbour is the category-theoretic statement of initiality; the difference is that we ask what happens when the initiality claim is *false or unproved*, which is the situation a working mathematician is usually in.

III. A TEST WITH THREE QUESTIONS

We fix, before asking anything, three pieces of data: a theory (what the new object may carry), a way of reading the old structure inside the new one, and a demand (a constant to add, a property to impose, or both). With those fixed, the test is three questions.

a. Question 1: is the demand really unmet? The question is about the elements, not about the solution. The wording is doing work. Both the natural numbers and the integers are countable, so any ring structure on the integers can be carried over to the natural numbers by a bijection. The statement “the natural numbers admit no ring structure” is therefore *false*, and an early version of our own formalisation asserted it. The true statement keeps the old structure fixed: no ring structure on $\mathbb{N}$ whose addition and multiplication extend the usual ones solves $x + 1 = 0$. This episode is worth recording because the test caught its author.

b. Question 2: is the extension the smallest one? Among all extensions that meet the demand, is there one that maps into every other, and does exactly one such map exist? When the answer is yes we call the extension *minimal*. The name is traditional; the property is not smallness but universality, and universality is what makes the object determined. The

TABLE II. The three questions, what each one rules out, and what we actually delivered. “Answered” means a machine-checked statement about the examples in this paper; “specified” means we wrote down what the answer would have to be but did not implement it.

#	Question	Rules out	Status here
1	Can the old carrier support the demand?	changing the carrier and calling it a solution	answered
2	Is there a smallest extension, unique up to relabelling?	the trivial extension, the chaotic extension, and every construction that depends on choices	answered
3	Is it new, given a catalogue of known objects?	renaming an existing object	specified, not implemented; our first implementation failed (§VI)

next section states what follows.

c. Question 3: is it new? New relative to what is already known — not relative to the system’s own state, which can always be arranged to report novelty. In practice this means an explicit catalogue of known objects, supplied from outside. Section VI reports what happened when we tried to implement this question; the short version is that our implementation was wrong in a way that only the examples revealed.

What “forced” means here, and what it does not

The word carries at least four different claims, and a paper that does not say which one it means can be attacked from any direction. In order of strength:

- *Metaphysical*: the object could not have failed to exist. We do not touch this.
- *Conceptual*: the object is already contained in the concept we started from. We do not touch this either; our demands are data, not concepts.

- *Practical*: no working alternative exists, given the state of the field. That is a claim about a community, not about mathematics.
- *Determined*: every acceptable answer is the same object up to a unique relabelling. This is the one we test.

The fourth is weaker than the first three, and it is the only one that can be settled by reading a proof. It is also the one Benacerraf’s argument leaves standing: if numbers are positions in a structure, then “the same position” is exactly determination up to relabelling, and nothing stronger is on offer. We therefore do not claim to have shown that the integers had to exist. We claim to have shown that, given the demand and the company it keeps, nothing else could have answered it.

That is not a hedge; it changes what counts as a refutation. A reader who wants meta-physical necessity will find the test silent, and should. A reader who wants to know whether a program’s “discovery” is a renaming will find it decisive.

IV. WHY THE SECOND QUESTION SETTLES THE OBJECT

Question 2 has a clean answer, and the answer is the reason the test is worth running. If an extension is smallest in the sense above, then:

1. it maps into every other extension, and the map is unique;
2. any two smallest extensions are the same up to a relabelling of their elements, and that relabelling is itself unique;
3. the smallest extension has no symmetries of its own beyond the identity.

None of this mentions rings, order, or completeness. The three statements use four properties of the surrounding category — an identity map, a composition, an underlying-function map, and the fact that maps with the same values are equal — and nothing else. That is why one argument covers the integers, the rationals and the complex numbers at once, and why the argument can be checked once rather than three times.

We should be plain about how much this is worth. Once you *define* “smallest” as “maps into everything, uniquely”, the first and third statements are restatements of the definition,

and the proof is four lines. The content is not the proof. The content is that this is the right definition to have chosen: it is the only one on the table that does not depend on how the object is constructed. Two people who build the complex numbers differently — as pairs of reals, as a quotient ring, as a matrix algebra — get the same answer to Question 2, and their constructions are related by a relabelling that is itself unique. That is what it means for the object, rather than the construction, to be determined.

a. A worked example, and a warning. For the step from the reals to the complex numbers, the demand is “add a constant j with $j \cdot j = -1$ ”. The extension $(\mathbb{C}, i)$ is smallest, and it has no symmetry beyond the identity. Now drop the constant from the demand and ask only that *some* square root of -1 exists. The same field is still an extension, but it is no longer smallest: the comparison is no longer unique. Identity and complex conjugation are two distinct maps over $\mathbb{R}$, and both are allowed; pinning i is what removes the second one. Finding a solution and pinning it down are different achievements, and only the second produces an object. A test that accepts the unpinned version will accept a field with a symmetry group, which is exactly the ambiguity it was supposed to remove.

b. One thing the argument does not settle. The demand “the order is complete” behaves differently from “ $j^2 = -1$ ”, and it is worth saying why in words rather than symbols. An equation like $j^2 = -1$ is preserved by every ring map: if j satisfies it, so does the image of j . A condition like “every bounded non-empty set has a least upper bound” is not preserved by ring maps, and it has no term model — there is no expression you can write down, quotient by a relation, and be done. What follows is that the smallest extension cannot be *computed* the way the complex numbers can. It does not follow that no smallest extension exists. Our first draft drew the stronger conclusion; the referee was right and we withdrew it (Sec. V B).

V. WHAT THE TEST FOUND

We ran the test on four steps of the number tower, and on two further pieces of our own programme: an invariant built from prime factorisations, and a coordinate built from reversible dynamics. Table III gives the summary; the two rows marked “not applicable” are those pieces, and the reason is that neither is a demand on a structure — one is a definition, the other a coordinate system — so Questions 1 and 2 have nothing to bite on.

TABLE III. What passed, what did not, and what could not be judged. “Same carrier” is Question 1, “smallest” is Question 2, “new” is Question 3. “Proved” means machine-checked modulo the standard-maps condition of Sec. VII; “asserted” means written down as an obligation and not proved; a dash means the question does not apply or was not used. No entry in the “New” column was produced by an implementation: the novelty numbers printed by the shipped trace were entered by hand.

Case	Same carrier	Smallest	New
$\mathbb{N} \rightarrow \mathbb{Z}$	proved	proved	not used
$\mathbb{Z} \rightarrow \mathbb{Q}$	proved	proved	not used
$\mathbb{Q} \rightarrow \mathbb{R}$	proved	<i>asserted</i>	not used
$\mathbb{R} \rightarrow \mathbb{C}$	proved	proved	not used
powers vs roots	not applicable	not applicable	not applicable
counting invariant	<i>fails</i> (refused by hand)	—	—
vibration coordinate	not applicable	not applicable	not applicable

A. Three steps that pass

The integers are the smallest extension of the naturals once we agree to work among commutative rings; the rationals are the smallest extension of the integers once we agree to work among fields; and $(\mathbb{C}, i)$ is the smallest extension of the reals once we pin the square root of -1 . The underlying facts are classical (compare [3]); the formulation as a smallest extension, with the old structure pinned, is our packaging. What the test adds is that the three are instances of one statement, and that in each case the answer does not depend on the construction. For the integers, the side condition that the extension really contains the old system also gives characteristic zero for free, which is a pleasant sign that the definition is carrying its weight.

B. One step we could not certify

For the step from the rationals to the reals we prove the failure — the rationals are not complete — and we prove that every real is generated by a cut of rationals below it. We

do *not* prove that the reals are the smallest complete ordered field over the rationals. That statement is written down as a named obligation in the formal development, and nothing in the development uses it.

We think the obligation is true and provable; a page of standard arguments about complete ordered fields should do it. We did not do it. The consequence is that the fourth step is certified less strongly than the first three, and we prefer to say so in the text rather than to let a reader discover it in the appendix. The asymmetry is about proof debt and about what can be computed, not about a metaphysical difference between adjoining a root and completing an order.

C. One candidate we refused

Our programme also proposed an invariant of integers, meant to capture the “tension” between two numbers: count the primes that divide one but not the other, subtract the count in the opposite direction, and call the result $\tau(a, b)$. The proposal was that the surface $\tau = 0$ organises arithmetic, and that τ is a new object in its own right.

The axiomatic part worked out cleanly. Under four natural assumptions, the invariant is unique up to sign and equals a difference of counts, $\tau(a, b) = \omega(a) - \omega(b)$, where $\omega(n)$ is the number of distinct prime factors of n . That is the whole content: τ knows how many prime factors each side has and never which ones. Since ω is a century-old function and the formula is definable on the same carrier, the test returns *refused* at Question 1. There is no new object here; there is a theorem about ω , and a clear statement of why the construction cannot say more.

Two honest qualifications. The refusal is argued by hand: the counting part of the development contains no demand and no catalogue, so “Question 1 returns refused” is a mathematical observation, not a machine-checked verdict. And the refusal does not go through Question 3; our first draft said the novelty score was zero there, which does not follow and is not available, since no catalogue exists.

The episode is still useful. It is the case where the test said no to us, and it came with a reason: the invariant collapses to a count, and counts are already on the same carrier. A weaker test would have produced a paper claiming a discovery.

VI. THE MEASUREMENT THAT FAILED

Question 3 is the one we could not deliver, and we describe the failure because it is instructive.

The design had two scores. One was a weighted average of five qualities — coverage, relatedness, closure gain, minimality, novelty — meant to *rank* candidates. The other was a product of four — novelty, consistency, generativity, compression — meant to *admit* them. The distinction is real: a weighted average gives a high mark to a candidate whose novelty is zero but whose other qualities are excellent, which is precisely the $\sqrt{7}$ story; a product gives it zero. We checked the arithmetic of this distinction in the proof assistant: there is a score with novelty zero and generativity $4/5$, and a snapshot with the same entries whose product is zero (`weighted_sum_admits_renaming`, `product_rejects_renaming`). We stress what this does and does not show. It shows that the two formulas are not interchangeable. It does not model renaming at all: no catalogue, no candidate, no map between objects appears in those two statements.

Modelling renaming is where the implementation failed. To decide whether a candidate was new, the code compared the *sizes* of carriers: if a candidate’s carrier had the same cardinality as a known object’s carrier, it was called a renaming. The integers, the rationals and the complex numbers are all countably infinite. One countable object in the catalogue therefore makes all three renamings of it, and the gate assigns all three a novelty of zero. The gate refuses exactly the objects the rest of the paper certifies.

We did not patch this. Comparing sizes can only produce false refusals, never false admissions, so the risk runs in the opposite direction from the one our first draft described. A correct comparison is between structures, not carriers, which requires the catalogue to carry its own maps; that is a specification, and we have not implemented it. Consequently no admission decision in this paper uses Question 3, and the executable trace shipped with the code prints novelty values that were entered by hand. We put this in the table caption rather than in a footnote, and the caption says it.

VII. WHAT THE TEST CANNOT DO

a. It assumes the standard maps. The four properties that the main argument uses do not say that the maps between structures are the structure-preserving ones. A degenerate choice — one map between any two structures — satisfies all four, and in that setting every object is both smallest and largest, so the test says nothing about anything. Our examples use the standard maps by construction, but the assumption is not provable inside the framework, only checkable in each instance. We mark every statement that needs it as “modulo C0”. This is the price of stating the criterion for an arbitrary theory instead of for rings.

b. It does not generate the demands. The hard parts of a discovery engine — deciding whether a demand is satisfiable, proposing a candidate, deciding minimality, deciding consistency — are inputs here, not code. We prove what surrounds them: that a solvable demand never needs a new carrier, that a contradiction is never expanded, that the old system embeds in the new one, and that two discoveries of the same demand agree up to a unique relabelling. The engine is a judge, not an author.

c. It does not decide its own questions. Each question is a proposition about a fixed triple of data. Deciding it is per-instance work, and for Question 1 it is the same work as solving equations in a structure. Nothing here is a decision procedure, and we do not claim that the three questions can be answered mechanically in general.

d. It has only refused one of our candidates. The other case, the reals, is an uncertified obligation rather than a refusal. A test that has said no once is a start, not a track record. The strongest evidence we could offer would be a refusal of a candidate proposed by someone else, and believed by them; we do not have that yet.

e. The test is relative to the data. Every verdict in this paper is relative to three choices: the ambient theory, the refinement, and — for Question 3 — the catalogue. Change any of them and the verdict can change. This is not a defect to be patched; it is what makes the verdict checkable. But it does bound the claim: we have not shown that an object is forced unconditionally, only that it is forced relative to a stated triple. A reader who wants the unconditional word will not find it here, and the honest response is that we do not know what the unconditional word would mean.

VIII. DISCUSSION

The practical lesson is about the order in which to explain an object. The usual order is: here is the object, here are the properties that make it necessary. The order that survived our test is the reverse. First the failure that cannot be repaired in place. Then the company the repair is allowed to keep, because the same failure repaired in different company gives different objects. Then the smallest repair, which is determined once the company is fixed. Whatever is left over after those three steps is not forced, and should be called a construction.

The second lesson is that a negative result is cheap and worth publishing. Our programme’s counting invariant was, for a while, the centrepiece of a different paper. Running it through Question 1 took an afternoon and produced a theorem about ω instead. That is a better outcome than a paper claiming a discovery, and it cost less.

The third lesson is about machine-checking. The proof assistant did not settle any philosophical question, and it did not make the mathematics deep. It did three useful things: it forced the definition of “gap” to be stated in a way that is not false; it made the four-line nature of the main argument visible, so we could not pretend it was more; and it turned one of our claims about two scoring formulas into a theorem that a reader can check in a minute. It also made our mistakes visible — the ring-structure error, the carrier-counting error — which is the main reason to use it in a paper of this kind.

a. What would change our mind. Three things. A genuine discovery that passes Questions 1 and 2 and is refused by a repaired version of Question 3 would show that the third question is wrong, not merely unimplemented. A demand whose extensions do not form a category would show that the second question is not available in general. And a proof that the reals are not the smallest complete ordered field over the rationals would make our fourth row false rather than unproved. We expect the third to go the other way: the obligation looks provable, and we would be glad to withdraw the asymmetry.

b. How this relates to other work. The uniqueness half of Question 2 is category theory folklore [6], and its model-theoretic cousin — the uniqueness of the comparison supplied by a model completion — is in [7]. What is added here is the packaging: the same statement for an arbitrary theory, with the demand and the ambient theory as parameters, and the two negative verdicts that follow from it. Automated conjecturing systems [13–16] generate statements and are judged by whether the statements are true and useful. They supply

candidates; this paper is about the other end, deciding which candidates may be called forced.

IX. METHODS

a. Formal setting. A theory assigns to each type the type of structures on it; a model is a carrier together with a structure. A framework supplies maps between models, an identity, a composition, an underlying-function map, and extensionality (maps with equal values are equal). Associativity and the unit laws for composition are derived from these, not assumed. A demand supplies, for each model, a type of witnesses, a predicate on witnesses, and a transport of witnesses along maps that is functorial. An extension of a model is a target model, an injective map from the old carrier, and a witness satisfying the predicate. It is *minimal* when it maps into every other extension, uniquely. The three conclusions of Sec. IV are Theorem 1 of the development (`IsMinimal.hom_unique`, `IsMinimal.carrierEquiv`, `IsMinimal.rigid`).

b. Trust base. The development is Lean 4 [11] with Mathlib [10], with no `sorry`. There is one named axiom, the initiality of the reals, and no theorem consumes it. Separately, `native_decide` occurs in two files of the counting layer; it introduces `Lean.ofReduceBool` into the trusted fragment, and it supports finite numerical checks that this paper does not report. The statements this paper does report from that layer are proved by rewriting and arithmetic, not by evaluation; our earlier draft stated this the other way round and the correction is recorded in the repository.

c. What is checked, and what is only argued. Checked in the development (each has a named declaration; the names are collected in the audit file): the main theorem; the four failures; the three minimality results; the characteristic-zero side condition; the exclusion lemmas for two of the three pairs of verdicts; the engine’s four positive properties; the torsion decomposition of the power map; the collapse of the counting invariant to a count; the two-score separation. Argued but not checked: the preservation of equations versus closure conditions; the refusal of the counting invariant; the comparison between a candidate and the old world in the gap branch. Assumed: the standard-maps condition, and the initiality of the reals.

d. Data and code. The Lean development and the seven-round self-audit of this manuscript are in the repository, tagged `paper-forced-objects-v1.2`; the Lean sources are those of commit `f671974`, unchanged since then. The build pins Mathlib by git revision in `lakefile.pinned.toml`, and the same pinned file drives the Lean CI job: on every push that touches the sources it resolves the dependencies at that revision, fetches the Mathlib oleans and builds `Nods`, so the development is built from a clean checkout along a path a reader can repeat. The artifact is scoped rather than implied: `artifact/SCOPE.md` lists exactly the paths that belong to this paper and `scripts/export_artifact.sh` copies them out, so work elsewhere in the repository is not shipped with the paper. Every declaration named in the text can be located by its identifier, and the audit records which claims were withdrawn or downgraded, and why.

ACKNOWLEDGMENTS

The development was carried out in Lean 4 with Mathlib, and thanks are due to the Mathlib community. The manuscript was put through seven rounds of adversarial self-audit, run as independent referee lines by the author’s tooling; the record, including the withdrawal of two factual claims and the demotion of the third question, is kept with the source.

COMPETING INTERESTS

The author declares no competing interests.

-
- [1] E. Landau, *Grundlagen der Analysis* (Akademische Verlagsgesellschaft, Leipzig, 1930).
 - [2] R. Dedekind, *Stetigkeit und irrationale Zahlen* (Vieweg, Braunschweig, 1872).
 - [3] H.-D. Ebbinghaus, H. Hermes, F. Hirzebruch, M. Koecher, K. Mainzer, J. Neukirch, A. Prestel, and R. Remmert, *Numbers*, Graduate Texts in Mathematics, Vol. 123 (Springer, New York, 1991).
 - [4] P. Benacerraf, What numbers could not be, *The Philosophical Review* **74**, 47 (1965).

- [5] S. Shapiro, *Philosophy of Mathematics: Structure and Ontology* (Oxford University Press, New York, 1997).
- [6] S. Mac Lane, *Categories for the Working Mathematician*, Graduate Texts in Mathematics, Vol. 5 (Springer, New York, 1971).
- [7] P. C. Eklof and G. Sabbagh, Model-completions and modules, *Annals of Mathematical Logic* **2**, 251 (1971).
- [8] W. Hodges, *Model Theory*, Encyclopedia of Mathematics and its Applications, Vol. 42 (Cambridge University Press, Cambridge, 1993).
- [9] J. Harrison, *Theorem Proving with the Real Numbers* (Springer, London, 1998).
- [10] The Mathlib Community, The Lean mathematical library, in *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020)* (2020) pp. 367–381.
- [11] L. de Moura, S. Kong, J. Avigad, F. van Doorn, and J. von Raumer, The Lean theorem prover (system description), in *Automated Deduction – CADE-25*, Lecture Notes in Computer Science, Vol. 9195 (2015) pp. 378–388.
- [12] A. Baanen, Use and abuse of instance parameters in the Lean mathematical library, *Journal of Automated Reasoning* 10.1007/s10817-024-09712-7 (2024).
- [13] D. B. Lenat, *AM: An artificial intelligence approach to discovery in mathematics as heuristic search*, Ph.D. thesis, Stanford University (1976).
- [14] A. Davies, P. Veličković, L. Buesing, *et al.*, Advancing mathematics by guiding human intuition with AI, *Nature* **600**, 70 (2021).
- [15] G. Raayoni, S. Gottlieb, Y. Manor, *et al.*, Generating conjectures on fundamental constants with the Ramanujan Machine, *Nature* **590**, 67 (2021).
- [16] B. Romera-Paredes, M. Barekatin, A. Novikov, *et al.*, Mathematical discoveries from program search with large language models, *Nature* **625**, 468 (2024).