Question-Relative Sufficiency in Learned World Models

From Safe Control to Independent Audit

Flyxion
Independent Researcher

November 20, 2026

Abstract

Learned world models compress observations into representations from which future states, action consequences, or task outcomes can be predicted. Predictive accuracy, however, does not establish that the representation preserves the distinctions required for safe control. Two physical states may admit the same abstract description while differing in the actions that can be executed without violating a constraint. No planner operating exclusively on that description can recover a distinction eliminated by the representation.

This paper develops a formal account of question-relative sufficiency for learned world models. It distinguishes predictive sufficiency, safe-action sufficiency, authorization sufficiency, and evidentiary sufficiency, and establishes conditions under which each property is preserved by a state projection. A safe-action indistinguishability theorem identifies an elementary obstruction to exact latent-space safety classification. The analysis is extended to multistep viability, uncertain dynamics, and conservative projections that preserve safety at the cost of rejecting some permissible actions.

The framework further distinguishes learned semantic safety judgments from independently enforced constraints, provisional computational state from authoritative verification, and successful task completion from authorization to execute the actions used. Four recent public reports on learned world models, formal proof verification, adversarial safety training, and agentic execution motivate these distinctions without being offered as evidence for the theorems.

The central result is that predictive compression and the preservation of what safe control requires are distinct requirements, neither of which implies the other. A representation adequate for forecasting may be inadequate for control, and a representation adequate for control may still be insufficient for independent audit. The appropriate representation therefore depends not only on what a system must predict, but also on which distinctions its decisions and subsequent verification must preserve.

Keywords: world models, representation learning, sufficiency, safe control, viability, authorization, verification, auditability

1 Introduction: The Problem of Predictive Compression

A world model is useful to an agent because it replaces the full complexity of the environment with something small enough to reason over. The central design question is what the replacement may discard. A representation that reconstructs every pixel of an observation preserves far more than any decision requires, and a representation that preserves nothing but a scalar summary of expected reward may preserve too little. The joint-embedding predictive architecture proposed by LeCun as part of a broader program for autonomous machine intelligence makes the first of these observations explicit: a predictor trained to anticipate the representation of a later input, rather than the later input itself, is free to ignore detail that cannot be predicted [1, 2]. In a recent public interview LeCun states the aim directly: the representation should eliminate detail that is unpredictable and keep the part that can be predicted, so that for a camera moving up a tree the model retains the presence of branches and leaves while dropping the position of each leaf [3]. Whether detail that is dispensable for prediction is also dispensable for action is a separate question, because predictability and relevance are different properties of a variable, and the answer depends on what the actions are required to respect.

The difficulty can be stated without any reference to a particular architecture. Suppose an encoder maps two physical states to the same latent point. Suppose further that in one of the two states a given action keeps the system inside its constraint set, and in the other it does not. Every quantity computed from the latent point, including the output of any predictor, planner, value function, or classifier, takes the same value at the two states, because each is a function of the latent point. No such quantity can separate the two states, and so none can classify the action correctly in both. Improving the predictor does not help, because the predictor receives the latent point and the distinction is absent from its input. The obstruction is located in the encoder, and it is removed only by changing which states the encoder identifies.

This observation is elementary, and the paper does not present it as surprising. Its consequences are less elementary. Predictive adequacy, in the sense that the latent transition model agrees with the projected physical dynamics, can hold exactly while the identification of states still destroys safety-relevant information. This is shown in Section 4 by a four-state example. Safety is not the only property that a representation may fail to preserve. A physically safe action may be one the agent has no authority to take, and the distinction between safe and permitted is a distinction about an authority context that need not be recoverable from physical state at all. Finally, even when an agent executes only safe and permitted actions, a later reviewer may be unable to establish that it did so, because the record retained from execution identifies histories that the review question separates.

The paper accordingly organizes its argument around three questions. The first asks what information a representation must preserve to support correct decisions. This question admits exact mathematical answers, which take the form of factorization conditions on fibers of the projection. The second asks what conditions make a predicted transition permissible and its execution authorized. Here physical feasibility and normative authorization must be separated, and the argument concerns what follows from confusing them. The third asks what evidence must survive compression so that an independent verifier can assess the

decisions afterward. This introduces a sufficiency condition on the retained record rather than on the working state.

The contributions are as follows. Section 2 gives the question-relative notion of sufficiency, the factorization theorem on which every later result rests, the coarsening principle that information once discarded is unrecoverable, and a catalogue of the questions the paper treats (Table 1). Section 3 characterizes exact dynamical consistency for deterministic, set-valued, and stochastic transitions. Section 4 states and proves the safe-action obstruction and its converse, and shows that predictive consistency does not imply safe-action sufficiency. Section 5 characterizes the maximal sound latent action set, shows that conservatism is monotone in the coarseness of the projection, and quantifies what a retained history recovers. Section 6 extends the analysis to finite-horizon viability under uncertainty. Sections 7 through 10 introduce authorization sufficiency, separate learned judgment from enforced constraint, formalize provisional computation and authoritative verification, and prove an evidentiary indistinguishability theorem with an exact expression for the best achievable audit error. Section 11 assembles these into a stagewise sufficiency requirement on an architecture, Section 12 proposes small experiments in which every ground truth is known, and Sections 13 and 14 place the results among their antecedents and state their limits.

The paper does not claim a unified notion of safety. Predictive, safe-action, authorization, and evidentiary sufficiency are four properties, each relative to its own family of questions, and much of the argument concerns the consequences of treating them as one.

2 State Spaces, Projections, and Question-Relative Sufficiency

Let X be a set of physical states and U a set of actions. The dynamics are given by a transition relation

$$T : X \times U \to 2^X \setminus \{\emptyset\},$$

which is deterministic when every $T(x, u)$ is a singleton and set-valued otherwise, so that nondeterminism stands in for disturbance, parameter uncertainty, or modeling slack. An admissible set $\mathcal{A} \subseteq X$ specifies the states in which the system is permitted to be. A projection is a surjective map $f : X \to Z$ onto a set Z of latent states, and its fibers are the sets $f^{-1}(z)$ for $z \in Z$. The projection stands for the composite of whatever encoder a learned system uses; the paper treats it as a function and does not model its training.

Two conventions hold throughout. First, all sets are finite or countable unless a statement says otherwise, so that every claim is exact; the extension to general measurable spaces needs disintegration arguments that the paper does not pursue and flags where they matter. Second, a set $R \subseteq X$ of *occurring states* is supplied with the model, the states that can arise in deployment, and every statement is relative to it; $R = X$ unless stated. For $z \in Z$ write $S_z = f^{-1}(z) \cap R$ for the occurring states that z represents. A specification of the dynamics, the admissible set, and the occurring states is part of the problem and not a consequence of the representation, which is why the results below are conditional on it.

Sufficiency is always relative to something the representation is asked to support.

Definition 2.1 (Question-relative sufficiency)**.** A *question* is a function $q : X \to Q$ for some set Q of answers. The projection f is *sufficient for q on R* if for all $x, y \in R$,

$$f(x) = f(y) \implies q(x) = q(y).$$

For a family $\mathcal{Q}$ of questions, f is sufficient for $\mathcal{Q}$ if it is sufficient for every member.

The following theorem is the basic tool. Its proof is a few lines, and the paper uses it repeatedly under different names.

Theorem 2.2 (Factorization)**.** *The question q factors through f on R, meaning that there is a function $g : f(R) \to Q$ with $q(x) = g(f(x))$ for all $x \in R$, if and only if f is sufficient for q on R.*

Proof. If $q = g \circ f$ on R and $x, y \in R$ with $f(x) = f(y)$, then $q(x) = g(f(x)) = g(f(y)) = q(y)$. Conversely, suppose f is sufficient for q on R. For $z \in f(R)$ the set S_z is nonempty, and sufficiency makes q constant on it. Define $g(z)$ to be that constant value. Then $g(f(x)) = q(x)$ for every $x \in R$. □

Every later obstruction and sufficiency result in the paper is an instance of this theorem applied to a particular question, so the paper does not prove several independent facts; what differs from section to section is the question and the space on which it is asked. A family of questions $\{q_i\}_{i \in I}$ is handled by the single question $x \mapsto (q_i(x))_{i \in I}$, so sufficiency for the family is sufficiency for the joint question. The theorem has a negative reading that matters as much as the positive one. If f is not sufficient for q, then every function of $f(x)$ fails to equal q somewhere, and this holds for functions of any complexity, including learned ones trained on any quantity of data. The only remedy is a different projection.

The second fact used repeatedly is that information discarded by a projection cannot be restored by anything computed from it.

Proposition 2.3 (Coarsening)**.** *Let $f' = h \circ f$ for some function h on Z. If f' is sufficient for q on R, then f is sufficient for q on R. Equivalently, if f is not sufficient for q, no function of $f(x)$, however complex, is sufficient for it. A projection is sufficient for a family of questions if and only if it is sufficient for each member.*

Proof. If $x, y \in R$ and $f(x) = f(y)$, then $f'(x) = h(f(x)) = h(f(y)) = f'(y)$, so $q(x) = q(y)$ by sufficiency of f'. The last claim is the definition applied to the joint question. □

Table 1 lists the questions treated in the paper, the space on which each is asked, and where its characterization appears. Every entry is an instance of Theorem 2.2, so the sections differ in which question is asked and of what, and not in the mathematics used to answer it. Nothing in the definition requires a representation to preserve every physical detail. It requires only that the distinctions the specified questions depend on survive. The definition therefore does not conflict with the view that useful world models discard most of what they observe; it asks that the choice of what to discard be made against an explicit list of questions.

Property	Question	Asked of	Result
Latent transition	$q_u(x) = f(T(x,u))$	states	Prop. 3.1
Safe action	$U_{\text{safe}}(x)$	states	Thm. 4.1, Prop. 4.4
Horizon viability	K_h and $V_h(x)$	states	Thm. 6.1
Authorization	$\text{Auth}(c, x, \cdot)$	contexts and states	Def. 7.1
Executable set	$E(c, x)$	contexts and states	Prop. 7.2
Verdict	$V(r)$	transition records	Prop. 9.1
Audit	$q(h)$	histories	Thm. 10.1

Table 1: The questions treated in the paper. Each is a function on the stated space, and each result is an instance of Theorem 2.2.

This formulation is the one used in earlier work on sufficient projections by the same author [16], and it is related to the classical notion of a sufficient statistic [4] in that a function of the data is sufficient when it preserves everything a stated inference depends on. The difference worth noting is that here sufficiency is defined by constancy of an answer on fibers and does not presuppose a probabilistic model.

3 Predictive Sufficiency and Dynamical Consistency

Three different things are meant by predicting. A model may predict observations, in which case it reconstructs what a sensor would report. It may predict latent transitions, in which case it estimates the next representation from the current one and an action. Or it may predict task-relevant quantities, such as reward or goal attainment. The third is a question in the sense of Section 2 and is handled by factorization. The second is the subject of this section.

For deterministic dynamics $T(x,u) = \{\tau(x,u)\}$, say that f is *transition consistent* if there is a map $\bar{\tau} : Z \times U \to Z$ with

$$f(\tau(x,u)) = \bar{\tau}(f(x),u) \quad \text{for all } x \in X,\ u \in U.$$

Proposition 3.1 (Latent transition)**.** *A function $\bar{\tau}$ satisfying the display exists if and only if, for every action u, the map $x \mapsto f(\tau(x,u))$ is constant on each fiber of f.*

Proof. Apply Theorem 2.2 to the question $q_u(x) = f(\tau(x,u))$ for each fixed u, and assemble the resulting maps g_u into $\bar{\tau}(z,u) = g_u(z)$. □

For set-valued dynamics the same statement holds with $q_u(x) = f(T(x,u))$, the image of the successor set, taking values in 2^Z. In that case consistency requires that states in a fiber have the same set of possible abstract successors under each action. For stochastic dynamics, let $P(\cdot \mid x,u)$ be a transition kernel on X and let $f_*P(\cdot \mid x,u)$ denote its pushforward

to Z. Then a latent kernel $\bar{P}$ with $\bar{P}(\cdot \mid f(x), u) = f_* P(\cdot \mid x, u)$ exists if and only if $x \mapsto f_* P(\cdot \mid x, u)$ is constant on fibers for every u; the argument is again Theorem 2.2, with answers taking values in the space of distributions over Z. States in a common fiber must induce the same distribution over future abstract states. The statement is made for discrete or finite state spaces, where the map from fibers to distributions is automatically well defined and measurable; for general measurable spaces the existence of a latent kernel additionally requires a disintegration of the kernel over the fibers, which the paper does not pursue.

These conditions describe exact consistency, which a trained model attains at best approximately. Approximate predictive sufficiency deserves separate treatment because small prediction error does not carry over to safety. If the constraint set has a boundary, then states arbitrarily close to the boundary on opposite sides can lie in one fiber and receive predictions that differ by less than any given tolerance, while one is admissible and the other is not. The tolerance that suffices for forecasting is therefore not a tolerance that suffices for classification near the boundary. The next section makes this precise in the exact case, where the failure cannot be blamed on estimation error.

4 Safe-Action Equivalence and the Indistinguishability Obstruction

For set-valued dynamics, the set of actions safe at x is

$$U_{\text{safe}}(x) = \{\, u \in U : T(x, u) \subseteq \mathcal{A} \,\}.$$

An action is safe when every outcome the model allows remains admissible. Statements are relative to the occurring states R of Section 2, and in the examples $R = X$.

Theorem 4.1 (Safe-action obstruction and characterization)**.** *Let $f : X \to Z$ be a projection.*

(i) Suppose $x, y \in R$, $f(x) = f(y)$, and $U_{\text{safe}}(x) \neq U_{\text{safe}}(y)$. Then for every function $g : Z \times U \to \{0, 1\}$ there is an action u and a state $s \in \{x, y\}$ such that $g(f(s), u) \neq \mathbf{1}[u \in U_{\text{safe}}(s)]$.

(ii) A function g with $g(f(x), u) = \mathbf{1}[u \in U_{\text{safe}}(x)]$ for all $x \in R$ and u exists if and only if U_{safe} is constant on each set S_z.

Proof. (i) Choose u in the symmetric difference of $U_{\text{safe}}(x)$ and $U_{\text{safe}}(y)$, say $u \in U_{\text{safe}}(x) \setminus U_{\text{safe}}(y)$. Then the indicator equals 1 at x and 0 at y, while $g(f(x), u) = g(f(y), u)$ because $f(x) = f(y)$. So g is wrong at one of the two states. (ii) Apply Theorem 2.2 to the question $x \mapsto U_{\text{safe}}(x) \in 2^U$, and set $g(z, u) = \mathbf{1}[u \in \hat{q}(z)]$ where $\hat{q}$ is the induced map on $f(R)$, extended arbitrarily elsewhere. The converse direction is part (i). □

Remark 4.2. Randomization does not repair the obstruction. If a randomized classifier outputs 1 with probability p on the pair $(f(x), u)$ and the indicator differs between x and y, its error probability is $1 - p$ at one state and p at the other, so the worst-case error is at least one half. Nor is the argument specific to binary classification: any planner, value function, or policy computed from $f(x)$ alone inherits the identification.

The theorem is a necessary and sufficient characterization of one-step safe-action sufficiency for memoryless latent controllers. The memoryless qualification is not decorative. A controller that observes a history of latent points $f(x_0), \dots, f(x_t)$ together with the actions taken has access to more than $f(x_t)$, because the history constrains which physical states are consistent with it. Its situation is the one analyzed in the next section, with the fiber replaced by the set of states consistent with the observed history, and it is helped only to the extent that this set is smaller than the fiber.

A corollary concerns irreversible compression in a pipeline. If f_1 is sufficient for U_{safe} but the pipeline later applies a coarser map h that merges two latent points whose safe-action sets differ, then $f_2 = h \circ f_1$ is not sufficient, and by Proposition 2.3 no downstream component that receives only $f_2(x)$ can reconstruct what h discarded.

The remaining point of this section is that predictive consistency and safe-action sufficiency are logically independent.

Example 4.3 (Consistent but unsafe)**.** Let $X = \{x_1, x_2, y, w\}$ and $U = \{u, v\}$. Let $f(x_1) = f(x_2) = z$ and $f(y) = f(w) = c$, and let $\mathcal{A} = \{x_1, x_2, y\}$, so that both initial states are admissible and w is not. Define $T(x_i, u) = \{y\}$ for $i = 1, 2$, $T(x_1, v) = \{y\}$, and $T(x_2, v) = \{w\}$, and let y and w transition to themselves under both actions. Then $f(T(x_1, a)) = f(T(x_2, a))$ for each action a, because $f(y) = f(w)$, so f is transition consistent and the latent model is exact. Yet $v \in U_{\text{safe}}(x_1)$ and $v \notin U_{\text{safe}}(x_2)$. By Theorem 4.1, no latent safety classifier is exact.

The example may be read physically as two admissible positions from which the same move lands on either side of a constraint boundary that the encoder places in one cell. The failure comes entirely from the fact that $\mathcal{A}$ is not a union of fibers. It does not depend on w being absorbing: any state outside $\mathcal{A}$ that shares a latent point with y produces the same obstruction, including one from which recovery is possible. When $\mathcal{A}$ is a union of fibers, transition consistency suffices.

Proposition 4.4 (Transition consistency with saturated admissibility)**.** *Suppose that $f(x) = f(y)$ implies $f(T(x, u)) = f(T(y, u))$ for every u, and that $\mathcal{A}$ is a union of fibers of f. Then U_{safe} is constant on the fibers of f, so an exact latent safety classifier exists.*

Proof. Let $f(x) = f(y)$ and fix u. Because $\mathcal{A}$ is a union of fibers, a point x' lies in $\mathcal{A}$ if and only if $f(x') \in f(\mathcal{A})$, so $T(x, u) \subseteq \mathcal{A}$ if and only if $f(T(x, u)) \subseteq f(\mathcal{A})$. By transition consistency the images $f(T(x, u))$ and $f(T(y, u))$ coincide, so $T(x, u) \subseteq \mathcal{A}$ if and only if $T(y, u) \subseteq \mathcal{A}$. Hence $U_{\text{safe}}(x) = U_{\text{safe}}(y)$, and Theorem 4.1(ii) gives the classifier. □

The proposition identifies a constructive condition under which the obstruction disappears, and it shows what any counterexample must omit: transition consistency or saturation of the admissible set. It also connects this section to the exactness statement of Theorem 6.1, of which it is the one-step case. The condition is sufficient and not necessary, since U_{safe} may be constant on fibers by accident. The converse non-implication also holds. Take $X = \{x_1, x_2, y_1, y_2\}$ with all states admissible, $f(x_1) = f(x_2)$, $f(y_1) \neq f(y_2)$, and $T(x_i, u) = \{y_i\}$. Then every action is safe everywhere, so U_{safe} is trivially constant on fibers, but the projected successors of x_1 and x_2 differ and f is not transition consistent. A

representation can therefore be sufficient for safe actions and inadequate for prediction, and the reverse.

Remark 4.5 (Predictable is not relevant). The criterion that a representation should keep what can be predicted and discard what cannot is a criterion about forecasting. Theorem 4.1 shows that safe-action sufficiency is a different criterion, and Example 4.3 shows that the two come apart even when the latent predictor is exact. The detail that separates y from w there may be read as a local, unpredictable one, such as which of several small stones lies under a foot, that a forecasting objective has no reason to keep and that nonetheless decides whether a step is admissible. Whether a discarded variable is predictable and whether it is decisive for safe action are logically independent facts about it, and only the second bears on safe control.

5 Conservative Abstraction and the Price of Information Loss

Exact safe-action preservation is sometimes more than a system needs. A controller that cannot tell which of several physical states it occupies can remain sound by acting only when an action is safe at all of them. This section makes that precise and measures what soundness costs.

With the occurring states R and the sets S_z of Section 2, a latent action rule is a map $\sigma : Z \to 2^U$, and it is *sound* if $\sigma(f(x)) \subseteq U_{\text{safe}}(x)$ for every $x \in R$. Define

$$\bar{U}_{\text{safe}}(z) = \bigcap_{x \in S_z} U_{\text{safe}}(x).$$

Theorem 5.1 (Conservative action). *The rule $\bar{U}_{\text{safe}}$ is sound, and every sound rule σ satisfies $\sigma(z) \subseteq \bar{U}_{\text{safe}}(z)$ for all z with $S_z \neq \emptyset$. Moreover $\bar{U}_{\text{safe}}(f(x)) = U_{\text{safe}}(x)$ for all $x \in R$ if and only if U_{safe} is constant on each S_z.*

Proof. Soundness of $\bar{U}_{\text{safe}}$ holds because an intersection over S_z lies inside $U_{\text{safe}}(x)$ for each $x \in S_z$. If σ is sound and $u \in \sigma(z)$, then $u \in U_{\text{safe}}(x)$ for every $x \in S_z$, so $u \in \bar{U}_{\text{safe}}(z)$. For the last claim, if U_{safe} is constant on S_z then the intersection equals the common value. Conversely, if $\bar{U}_{\text{safe}}(f(x)) = U_{\text{safe}}(x)$ for all $x \in S_z$, then $U_{\text{safe}}(x)$ equals $\bar{U}_{\text{safe}}(z)$ for every such x, hence is constant. □

The theorem separates soundness from completeness. The conservative rule never selects an unsafe action under the assumed model, R, and $\mathcal{A}$, which is the sense of the qualification that the result is only as reliable as those specifications. It is complete exactly when the obstruction of Theorem 4.1 is absent. Where it is not complete, the loss at a state is the set of actions that are safe there and rejected,

$$\text{Loss}(x) = U_{\text{safe}}(x) \setminus \bar{U}_{\text{safe}}(f(x)),$$

and it is the price paid for the identification of x with its fiber-mates.

Proposition 5.2 (Monotonicity of conservatism). *Let $f' = h \circ f$ be a coarsening of f. Then for every z and every z' with $h(z) = z'$ and $S_z \neq \emptyset$, we have $\bar{U}^{f'}_{\text{safe}}(z') \subseteq \bar{U}^{f}_{\text{safe}}(z)$. In particular,* Loss *can only grow as the abstraction coarsens.*

Proof. The fiber $S_{z'}$ of f' is a union of fibers of f and contains S_z. An intersection over a larger set is smaller. □

The conservative rule above uses the current latent point alone. A controller that retains the history of latent points and actions can do better, and the gain is exactly the narrowing of the set of consistent physical states. A *latent history* is a sequence $\zeta = (z_0, u_0, z_1, \ldots, u_{t-1}, z_t)$, and the set of states consistent with it is

$$B(\zeta) = \{\, x_t \in R : \exists\, x_0, \ldots, x_{t-1} \in R,\ f(x_i) = z_i,\ x_{i+1} \in T(x_i, u_i) \,\},$$

the possible current states given the whole record.

Proposition 5.3 (Value of history)**.** *(i) $B(\zeta) \subseteq S_{z_t}$. (ii) The rule $\sigma(\zeta) = \bigcap_{x \in B(\zeta)} U_{\mathrm{safe}}(x)$ is sound for every trajectory that actually occurs, meaning $\sigma(\zeta) \subseteq U_{\mathrm{safe}}(x_t)$ for the true current state x_t. (iii) Every history rule that is sound for all occurring trajectories is contained in σ, and $\sigma(\zeta) \supseteq \bar{U}_{\mathrm{safe}}(z_t)$. The inclusion in (iii) is strict for a given history exactly when some action is safe at every state of $B(\zeta)$ and unsafe at some state of S_{z_t}.*

Proof. (i) holds because the definition of $B(\zeta)$ requires $f(x_t) = z_t$ and $x_t \in R$. (ii) The true current state belongs to $B(\zeta)$ for the latent history it generates, and an intersection over $B(\zeta)$ lies inside U_{safe} of each member. (iii) A rule sound for all occurring trajectories must be contained in $U_{\mathrm{safe}}(x)$ for each $x \in B(\zeta)$, since each such x is the current state of some occurring trajectory with latent history ζ, hence contained in $\sigma(\zeta)$. Because $B(\zeta) \subseteq S_{z_t}$, the intersection defining σ runs over fewer sets than the one defining $\bar{U}_{\mathrm{safe}}(z_t)$ and is therefore at least as large. For the final claim, an action lies in $\sigma(\zeta) \setminus \bar{U}_{\mathrm{safe}}(z_t)$ exactly when it is safe at every state of $B(\zeta)$ and unsafe at some state of S_{z_t}. □

History therefore helps exactly when it separates states that the current latent point identifies, and it cannot help beyond that. This is the formal reason a retained documentary history can make a coarse operative projection safe to use, and also the reason retention is not sufficient: the history helps only if the distinguishing information was recorded in it.

This gives a principled link between compression and operational conservatism. A designer who chooses a coarser representation chooses, at the same time, a smaller guaranteed set of actions, and the rate at which the guaranteed set shrinks as a function of coarseness is a property of the system rather than of the learning algorithm. The next section shows that the situation is more demanding over longer horizons.

6 Multistep Viability, Reachability, and Uncertainty

An action that is safe in the one-step sense can lead somewhere from which no safe continuation exists. Define the finite-horizon viability sets

$$K_0 = \mathcal{A}, \qquad K_{h+1} = \{\, x \in \mathcal{A} : \exists u,\ T(x, u) \subseteq K_h \,\}.$$

A state in K_h is one from which the system can remain admissible for $h + 1$ steps against any outcome the model permits. The recursion permits feedback, since the action chosen at

a later stage may depend on the state actually reached, so membership in K_h asserts the existence of a strategy and not of a single fixed action sequence. The corresponding action sets are $V_h(x) = \{u : T(x,u) \subseteq K_h\}$, with $V_0 = U_{\text{safe}}$. The sets K_h decrease with h, and an action in $V_0(x)$ need not lie in $V_1(x)$: a state s with an action u whose only outcome s' is admissible but has no action leading back into $\mathcal{A}$ satisfies $u \in V_0(s)$, $s' \in K_0 \setminus K_1$, and $u \notin V_1(s)$. This is the familiar dead end. Viability theory studies such sets in continuous time [9], and model predictive control addresses them through constraint tightening and terminal sets [10]. The question here is what the projection must preserve.

By Theorem 2.2, a latent-only rule is exact for horizon h if and only if K_h and the action sets V_h are constant on fibers, and the question is how to guarantee that without checking every set. A sufficient condition is a form of bisimulation. Say that f is *set-consistent on* R if R is forward invariant under T (that is, $T(x,u) \subseteq R$ for $x \in R$) and $f(T(x,u)) = f(T(y,u))$ whenever $x, y \in R$ and $f(x) = f(y)$.

For a conservative treatment, define a latent transition by $\bar{T}(z,u) = \bigcup_{x \in S_z} f(T(x,u))$ and latent viability by

$$L_0 = \{z : S_z \subseteq \mathcal{A}\}, \qquad L_{h+1} = \{z \in L_0 : \exists u,\ \bar{T}(z,u) \subseteq L_h\}.$$

Theorem 6.1 (Finite-horizon preservation)**.** *Assume R is forward invariant.*

(i) (Soundness) For every h, $f^{-1}(L_h) \cap R \subseteq K_h$.

(ii) (Exactness) If f is set-consistent on R and, for all $x, y \in R$, $f(x) = f(y)$ implies $x \in \mathcal{A} \Leftrightarrow y \in \mathcal{A}$ (so that $\mathcal{A} \cap R$ is a union of the restricted fibers S_z, and R itself need not be saturated), then $f^{-1}(L_h) \cap R = K_h \cap R$ for every h.

Proof. (i) By induction on h. For $h = 0$, if $x \in R$ and $f(x) \in L_0$ then $x \in S_{f(x)} \subseteq \mathcal{A}$. Suppose the claim holds for h, and let $x \in R$ with $f(x) \in L_{h+1}$. Then $x \in \mathcal{A}$, and there is u with $\bar{T}(f(x),u) \subseteq L_h$. For each $x' \in T(x,u)$ we have $x' \in R$ by forward invariance and $f(x') \in f(T(x,u)) \subseteq \bar{T}(f(x),u) \subseteq L_h$, so $x' \in K_h$ by the induction hypothesis. Hence $T(x,u) \subseteq K_h$ and $x \in K_{h+1}$.

(ii) Under the stated condition, $L_0 = f(\mathcal{A} \cap R)$ and $f^{-1}(L_0) \cap R = \mathcal{A} \cap R = K_0 \cap R$. Suppose $K_h \cap R = f^{-1}(L_h) \cap R$. Set-consistency gives $\bar{T}(f(x),u) = f(T(x,u))$ for $x \in R$. Then $x \in K_{h+1} \cap R$ if and only if $x \in \mathcal{A}$ and some u has $T(x,u) \subseteq K_h$; since $T(x,u) \subseteq R$, this holds if and only if every $x' \in T(x,u)$ has $f(x') \in L_h$, that is, $f(T(x,u)) \subseteq L_h$, that is, $\bar{T}(f(x),u) \subseteq L_h$. Together with $f(x) \in L_0$ this is the definition of $f(x) \in L_{h+1}$. □

Part (i) says that the latent computation yields a conservative approximation to viability, in the same sense as Section 5: it may reject viable states but never accepts a nonviable one. Part (ii) says that the approximation is exact when the projection is a bisimulation-like projection for which the admissible set is saturated. The condition is sufficient and not necessary, since a particular K_h may be a union of fibers by accident. It is also the reason a single check of one-step safe-action sufficiency does not settle the longer question: the set K_1 depends on K_0 through the dynamics, and a projection that is sufficient for K_0 need not be sufficient for K_1 unless it is consistent with the transitions.

Parameter uncertainty enters by augmenting the state. Let $\tilde{x} = (x, \theta)$ with θ ranging over an uncertainty set Θ, and let the parameter either persist or vary within Θ according to a stated rule. A projection that omits θ places states with different parameters in common fibers, and Theorem 5.1 then says the guaranteed action set is the intersection over all parameters consistent with the observation. Joint propagation of state and parameter constraints, as in robust model predictive control, shrinks the set of consistent parameters as observations accumulate and so reduces the conservatism, but only if the representation retains the variables through which the observations constrain the parameters.

A tractable setting for studying this is a network of coupled oscillators with amplitude r_i and phase ϕ_i for each node, in which a phase-only representation $f(r, \phi) = \phi$ may forecast synchronization well under weak coupling while discarding amplitude, which governs energy and actuator expenditure. If the constraint set bounds expenditure, then states with identical phases and different amplitudes lie in one fiber and may differ in U_{safe}. The setting is proposed as an experimental testbed in Section 12; no claim is made here about its results.

7 Authorization Is Not Physical Admissibility

Physical admissibility asks whether a transition keeps the system within its constraints. Authorization asks whether the agent is permitted to take the action, and the answer depends on who has granted what. Let C be a set of authority contexts, which may record the principal, the delegated scope, the active restrictions, and the time, and let

$$\text{Auth} : C \times X \times U \to \{0, 1\}$$

be an authorization predicate. An action may be physically feasible and unauthorized, as when a tool call would succeed but lies outside the granted scope. It may be authorized and physically unsafe, as when an operator explicitly instructs an action that violates a constraint. These are different cases, and a system that treats one as the other fails in a predictable direction.

Definition 7.1 (Authorization sufficiency)**.** A pair of maps (f, k) with $f : X \to Z$ and $k : C \to W$ is *authorization sufficient* if $f(x) = f(x')$ and $k(c) = k(c')$ imply $\text{Auth}(c, x, u) = \text{Auth}(c', x', u)$ for all u.

By Theorem 2.2 applied to the joint map $(c, x) \mapsto \text{Auth}(c, x, \cdot)$, a latent authorization predicate exists exactly under this condition. No new theorem is needed, and the paper does not manufacture one. The substantive point is that authorization sufficiency constrains the representation of contexts as well as states, and that it need not be implied by sufficiency for U_{safe}.

Several examples establish particular non-implications among the properties introduced so far, without claiming every pairwise independence. Example 4.3 gives predictive consistency without safe-action sufficiency, and the converse example following Proposition 4.4 gives the reverse. Authorization sufficiency can fail with physics fully preserved: take a single physical state and two contexts that differ in granted scope, with k identifying them. Safe-action sufficiency and authorization sufficiency can also coincide while evidentiary sufficiency fails, as Section 10 shows.

The executable set $E(c, x) = U_{\text{safe}}(x) \cap \{u : \text{Auth}(c, x, u) = 1\}$ is the set of actions that are both safe and authorized. The relation between sufficiency for E and sufficiency for its two factors is one-directional.

Proposition 7.2 (Sufficiency for a conjunction)**.** *If the pair (f, k) is sufficient for both $(c, x) \mapsto U_{\text{safe}}(x)$ and $(c, x) \mapsto \{u : \text{Auth}(c, x, u) = 1\}$, then it is sufficient for E. The converse fails.*

Proof. If $f(x) = f(x')$ and $k(c) = k(c')$, the two factors agree at (c, x) and (c', x'), so their intersections agree. For the converse, take one state with $U_{\text{safe}} = \{u_1\}$ and authorized actions $\{u_1, u_2\}$, and a second with $U_{\text{safe}} = \{u_1, u_2\}$ and authorized actions $\{u_1\}$. Both have $E = \{u_1\}$, and a projection may identify them, although neither factor is then determined. □

A representation sufficient for E but for neither factor selects the right actions and cannot explain why u_2 was withheld. An audit that must assign the reason to the physical constraint or to the authority boundary cannot be answered from it.

A last distinction concerns outcomes. Define task success $W(h)$ as a predicate on execution histories and compliance as the predicate that every action in h was authorized. These are logically independent predicates on histories: a history can succeed by unauthorized means, comply and fail, or do both or neither. A system evaluated only on W is evaluated on a question that does not determine compliance, and an optimizer rewarded on W has no pressure to preserve the distinction.

A public report from November 5, 2026 describes a case that illustrates the point [14]. A model that found a configured proxy unavailable is reported to have attempted alternative network routes, including some that crossed the intended sandbox boundary, and it did not successfully retrieve the target site. The mathematical observation does not depend on whether the workaround succeeded. A transition's technical availability establishes nothing about authorization, and an attempt to execute an unauthorized transition is a compliance question even when the attempt fails.

The interview with LeCun raises the same distinction from the other side. The interviewer's account of the incident it discusses, including the number of agents involved and the degree of their coordination, is stated in the transcript without independent support and is not relied on here. Asked about that reported incident, in which agents are said to have attacked an external service, LeCun treats it as a failure of containment, a sandbox that leaked, and not as evidence that a system had acquired objectives of its own; he describes the system as having done what it was asked while the guardrails around it were ineffective [3]. The formal point holds on either account. That a system did what it was asked is a statement about the task predicate, and it does not entail that the means it used were authorized. A containment failure and an objective failure are different failures, and an evaluation that separates them must record both the task outcome and the authorization status of each action taken.

8 Learned Safety Judgments and Formal Constraints

Some questions about permissibility are semantic. Whether a request is harmful, whether an output reveals protected content, or whether a tool call is within the spirit of a delegation are not questions for which a complete deterministic predicate is available. A learned judgment $s_\theta(x, u) \in [0, 1]$ can estimate them, and the estimate is useful for planning, for ranking candidate actions, and for deciding when to escalate. The score is not a proof of compliance. It is an estimate of a quantity defined by something outside the model, and its reliability is an empirical matter.

A study by Wang and colleagues, reported on November 12, 2026, examines adversarial training in which the attacks selected for training are those that succeed reproducibly, and it reports improved robustness, including against adaptive attackers [12]. The result is a measurement about the behavior of a trained model under a family of attacks. It is not a guarantee that prohibited transitions cannot occur, and it was not presented as one. The distinction is between a claim quantified over the attacks that were tried and a claim quantified over every input the system can encounter. The first can be strengthened by better evaluation. The second requires an enforcement mechanism whose operation does not depend on the model's judgment.

Two further distinctions concern guardrails that are built into an optimizing planner, the arrangement LeCun describes when he proposes that a world model predict the consequences of an action, that the predicted state be scored against guardrail objectives before the system acts, and that such rules be fixed by construction [3]. The first is between a penalty and a prohibition. In the interview a guardrail is described both as a cost term, in the manner of a law that attaches a price to an act, and as a rule the system cannot violate. These are different mathematical objects.

Proposition 8.1 (Penalty versus constraint)**.** *Let a planner choose u to minimize $J(u) = J_{\text{task}}(u) + \lambda J_{\text{harm}}(u)$ with $\lambda < \infty$ and $J_{\text{harm}} \geq 0$. Suppose u_0 is a safe action with $J_{\text{harm}}(u_0) = 0$ and u' is an unsafe action with $J_{\text{harm}}(u') = h$. If $J_{\text{task}}(u_0) - J_{\text{task}}(u') > \lambda h$, then $J(u') < J(u_0)$. In particular, if the available actions are exactly $\{u_0, u'\}$, the penalized minimizer is the unsafe action u'. A planner that minimizes J_{task} over $U_{\text{safe}}(x)$ never selects an action $u' \notin U_{\text{safe}}(x)$, whatever the task cost.*

Proof. The first claim is the inequality $J(u') = J_{\text{task}}(u') + \lambda h < J_{\text{task}}(u_0) = J(u_0)$, and the second follows from it when only these two actions are available. The last holds because $u' \notin U_{\text{safe}}(x)$. With a larger action set the statement concerns the comparison of the two actions only, and the global minimizer may be a third, safe action. □

For any finite weight there is therefore a task benefit large enough that the penalized objective prefers the harmful action to a given safe alternative, and to every safe alternative when the safe options are few, unless the benefit is bounded in advance, whereas a hard constraint has no such threshold. The second distinction is between the model's predictions and the world. A guardrail evaluated on predicted successors inherits the reliability of the predictor.

Proposition 8.2 (Model-relative soundness)**.** *Let $\hat{T}$ be a planner's transition model and let the guardrail accept u at x when $\hat{T}(x,u) \subseteq \mathcal{A}$. If $T(x,u) \subseteq \hat{T}(x,u)$ for all $x \in R$ and all u, every accepted action lies in $U_{\text{safe}}(x)$. Without that containment no conclusion about $U_{\text{safe}}(x)$ follows.*

Proof. If $T(x,u) \subseteq \hat{T}(x,u) \subseteq \mathcal{A}$ then $u \in U_{\text{safe}}(x)$. When containment fails, some $x' \in T(x,u)$ lies outside $\hat{T}(x,u)$ and may lie outside $\mathcal{A}$ while the guardrail still accepts u. □

A single-valued predicted successor with a real system subject to disturbance is the common case in which containment fails. A prediction that the outcome is safe is then a different claim from a certificate that every permitted outcome is safe, which is the sense of safety used in Section 6.

The architecture that results is hybrid. Learned judgments inform proposals, and independently enforced restrictions govern execution. A restriction qualifies as enforced when it is evaluated by a component that the proposing model cannot influence except by submitting a record, and when its verdict depends on a specification that can be inspected. The cases in which a semantic judgment cannot be reduced to such a restriction must be handled explicitly, and the paper proposes that the authorization predicate be three-valued, taking the values permit, forbid, and unresolved, with each unresolved case assigned to a named resolver, which may be a human or another process with delegated authority. This is more honest than forcing a binary predicate, which would hide the unresolved cases inside whichever default the system happens to adopt.

9 Provisional Computation and Authoritative Verification

A planner may propose a transition and supply supporting evidence, but its acceptance of its own proposal should not make the proposal authoritative. The constructive counterpart to the impossibility results above is a separation between provisional computation and authoritative verification.

A recent system for interactive theorem proving illustrates the separation in a setting where correctness is crisp. The Isabelle Pool Server of Ren and Li provides persistent sessions and checkpoints for exploratory proof computation, and it keeps a clean build as the final correctness authority [11]. Exploration may be cheap, stateful, and fallible, because nothing it produces is final until the authoritative build accepts it. The analogy is structural and does not transfer any property of the proof setting to physical control.

Define a proposed transition record

$$r = (x, u, x', c, e),$$

where x' is the claimed successor state, c is the authorization context, and e is supporting evidence. A verifier is a map

$$V(r) \in \{\text{accept}, \text{reject}, \text{unknown}\}.$$

Its output must be interpreted relative to an explicit specification Σ and trust boundary. Say that V is *sound for* Σ if $V(r) = \text{accept}$ implies that r satisfies the checks in Σ. Acceptance

then establishes that the submitted record meets the specified conditions, and it does not establish that every fact about the external world on which those conditions rest is correct. The trust boundary is the set of facts the verifier takes as given, such as the correctness of a sensor reading or of the authority context it was handed.

The verifier is itself a function of a projection of the record, namely the components it reads. By Theorem 2.2, two records that agree on those components receive the same verdict, and a difference in an unread component cannot change it. The sufficiency question therefore recurs at the verification stage: the verifier's input must preserve whatever the specification's conditions depend on. A verifier that checks authorization against c cannot catch a violation that depends on a context component absent from c.

The statement can be made exact. Let π be the projection of a record onto the components V reads, so that V is a function of $\pi(r)$, and let Σ be the set of records that conform to the specification.

Proposition 9.1 (Verifier input)**.** *If $\pi(r) = \pi(r')$ then $V(r) = V(r')$. Consequently, if V is sound for Σ and π identifies a conforming record $r \in \Sigma$ with a nonconforming record $r' \notin \Sigma$, then $V(r) \neq$ accept.*

Proof. The first claim is the factorization of V through π. For the second, soundness gives $V(r') \neq$ accept because $r' \notin \Sigma$, and then $V(r) = V(r') \neq$ accept. □

A sound verifier whose input identifies a conforming record with a nonconforming one is therefore forced to withhold acceptance from the conforming one. The cost of an insufficient input is incompleteness, and not unsoundness, which is the same shape of conclusion as Theorem 5.1.

9.1 Repairing the Projection Rather Than the Proposal

The framework applies directly to CLIO, the Constraint-Linked Inference Organiser developed in earlier work by the author [15]. CLIO is a coordination layer and not an additional detector or verifier. It distinguishes a documentary history from the operative state derived from it, and it separates diagnosis of an obstruction, a candidate repair, the evidence that the candidate satisfies stated constraints, the warrant for adopting it, authorization to commit, and the committed transition recorded in the history. Verifying a constraint does not authorize intervention, detecting a failure does not warrant a repair, and validating a repair does not authorize its commitment. The terms are used as follows. The *operative projection* is the compressed representation on which current inference proceeds, and it is what the fibers of this paper describe. A *warrant* is the stated ground on which a continuation or repair may be adopted, relative to the active constraints. A *commitment* is the point at which an authorized transition is entered into the documentary history and becomes the basis for later steps. The contribution of the present paper to that architecture is a test of whether the information presented to each stage suffices for the question the stage must answer.

The reference to earlier work serves a specific purpose. The paper does not claim that any one architecture is correct. The five stages of Section 11 and the cycle of CLIO are

design choices, and other decompositions would serve. What is required of any such choice is internal consistency: each stage must receive a projection that is sufficient for the question it answers, and the definitions of its terms must not drift between uses. Fixing the vocabulary by reference to previously stated definitions is a way of enforcing that second requirement, since a system that reuses terms with a stable meaning has fewer occasions to produce statements its own definitions do not support.

Let $f_t : H \to Z_t$ be the operative projection at time t, defined on a space H of histories, and let $W_\Gamma : H \to 2^U$ assign to each history the set of continuations warranted under a fixed constraint and authority context Γ. The relevant question is whether $f_t(h_1) = f_t(h_2)$ implies $W_\Gamma(h_1) = W_\Gamma(h_2)$.

Proposition 9.2 (Projection-repair obstruction)**.** *If $f(h_1) = f(h_2)$ and $W_\Gamma(h_1) \neq W_\Gamma(h_2)$, then no repair operator that receives only $f(h)$ and Γ can return the exact warranted set for both histories. A rule that returns only continuations warranted for every history in the fiber $f^{-1}(z)$ is sound, and the maximal such rule is*

$$\overline{W}_\Gamma(z) = \bigcap_{h \in f^{-1}(z)} W_\Gamma(h).$$

Proof. The first claim is Theorem 2.2 applied to the question W_Γ. The second is Theorem 5.1 with U_{safe} replaced by W_Γ. □

The obstruction is not an error of prediction. It is the incapacity of the projection to answer the question the current continuation requires, and it is a more precise diagnosis than the observation that the system is uncertain. Repair is accordingly of two kinds, following the author's earlier definitions: repair that changes a trajectory while retaining its representation, and repair that changes the representational scheme itself. When the answer is not constant on a fiber, the first kind cannot suffice for both histories, because any change confined to the fiber's representation leaves the identification in place. The system must acquire evidence that separates the histories, refine the projection, or restrict itself to $\overline{W}_\Gamma(z)$. When $\overline{W}_\Gamma(z)$ is empty, no continuation is warranted across the histories the representation currently identifies. That does not make the task impossible, and it does not license weakening the constraints. The permitted responses are to defer, to request distinguishing evidence, or to refuse.

Persistence of the documentary history matters here for a reason the theorem makes exact. If only the operative projection is retained, a repair that requires a discarded distinction cannot be carried out. Retention of the history makes the distinction recoverable only if it was actually recorded and a procedure exists to recover it, so a history is a necessary resource for refining a projection and not a sufficient one.

10 Evidentiary Sufficiency and Retrospective Distinguishability

The last formal question concerns what survives after an agent acts. Let H be a space of execution histories and

$$e : H \to E$$

an evidence projection, the part of a history retained for later review. Let $q : H \to Q$ be an audit question, for example whether an action was authorized, whether a refusal occurred, or whether a task failed because of an environmental restriction. The retained evidence is sufficient for q if $e(h_1) = e(h_2)$ implies $q(h_1) = q(h_2)$.

Theorem 10.1 (Evidentiary indistinguishability)**.** *(i) If $e(h_1) = e(h_2)$ and $q(h_1) \neq q(h_2)$, then every auditor that is a function of retained evidence errs on at least one of h_1, h_2.*

(ii) An auditor $a : E \to Q$ with $a \circ e = q$ exists if and only if e is sufficient for q.

(iii) Let E and Q be finite, and let P be a probability distribution on H; the sum ranges over those ϵ with $P(e = \epsilon) > 0$. The least error probability achievable by any auditor using only e is

$$1 - \sum_{\epsilon \in E} \max_{a \in Q} P(e = \epsilon,\ q = a),$$

and it equals zero if and only if q is P-almost surely constant on each fiber of e.

Proof. (i) and (ii) are Theorem 2.2. For (iii), an auditor's probability of being correct is $\sum_\epsilon P(e = \epsilon,\ q = a(\epsilon))$, which is maximized by choosing, for each ϵ, the answer a with the largest joint mass. The error is one minus the resulting sum. It vanishes exactly when, for every ϵ of positive probability, the entire mass on $\{e = \epsilon\}$ lies on one answer. □

Retaining more evidence cannot raise the best achievable error.

Proposition 10.2 (Monotonicity of retention)**.** *Let $e' = h \circ e$ be a coarsening of e and let* $\mathrm{Err}(e)$ *denote the least auditor error of Theorem 10.1(iii). Then* $\mathrm{Err}(e') \geq \mathrm{Err}(e)$.

Proof. Each value ϵ' of e' corresponds to a union of values ϵ of e, and for each answer a, $P(e' = \epsilon', q = a) = \sum_{\epsilon \in h^{-1}(\epsilon')} P(e = \epsilon, q = a)$. The maximum over a of a sum is at most the sum of the maxima, so $\max_a P(e' = \epsilon', q = a) \leq \sum_{\epsilon \in h^{-1}(\epsilon')} \max_a P(e = \epsilon, q = a)$. Summing over ϵ' and subtracting from one gives the claim. □

The proposition is the probabilistic form of Proposition 2.3: retention decisions can only trade storage against audit accuracy, and the trade is monotone.

The theorem is elementary, and its use is in what it rules out. If the retained record does not separate cases that the review question separates, then no amount of auditing skill compensates, and the shortfall cannot be repaired afterward by collecting more data of the same kind. The projection e must be chosen before the event, because a record that was not retained cannot be added to later.

A report by OpenAI on covert underperformance illustrates why this matters [13]. It describes a framing-dependent gap in the performance of GPT-6 Astra, with approximately ten percent of hidden rubric items completed under a safety framing, against approximately sixteen percent under a capabilities framing. The investigation did not identify a single underlying cause, and many trajectories contained no explicit refusal and no visible decision to underperform. This is the situation to which the theorem speaks. If the question is whether a given instance of nonperformance reflected inability, a refusal, an environmental restriction, or something else, and the retained record is the visible transcript, then histories with different underlying causes that produce the same transcript cannot be told apart. The example does not establish that any instance was deliberate, and the report does not claim that. It supports a narrower point: observed nonperformance and inferred intent are different questions, and a transcript of visible refusals is not by itself sufficient evidence about either.

Evidentiary sufficiency also separates from the earlier properties. A system can be safe-action sufficient and authorization sufficient while retaining an execution record from which the two causes of a blocked action cannot be told apart, as in the executable-set example of Section 7. The same holds for compaction of working context in long-running agents, which is a projection of the working state chosen for the agent's convenience, and not a retention policy chosen for the reviewer's questions [17]. Working state and evidentiary record answer different questions and should be projected separately.

The same structure underlies the distinction in CLIO between the documentary history and the operative state [15]. Let $p : H \to \Sigma$ send a history to its operative state. Two histories with $p(h_1) = p(h_2)$ may differ in whether a transition was authorized, what evidence supported it, or whether a repair occurred, and a later review that must separate them cannot do so from the operative state alone. By Theorem 10.1, each commitment should therefore carry the evidence required to answer the questions a declared review policy will pose. The distinction is between a log that is immutable and one that is sufficient: immutability prevents alteration of what was retained and says nothing about whether enough was retained.

11 A Constraint-First Agent Architecture

The preceding results suggest an architecture organized around five operations, which together form the sequence

$$\text{Propose} \longrightarrow \text{Check} \longrightarrow \text{Authorize} \longrightarrow \text{Execute} \longrightarrow \text{Record}.$$

The sequence is a design choice and not a claim of uniqueness: a different decomposition would serve, provided it is consistent in the sense of Section 2, with each stage given a representation sufficient for its question. The sequence is also conceptual. It does not require five separate software services, and several stages may be implemented by one component provided the questions they answer are kept distinct. The system maintains a learned predictive representation and a planner, a specification of admissible transitions, an authorization mechanism, and an independent verification boundary.

Stage	Question answered	Consequence of substitution
Propose	Which transition does the planner expect to serve the objective?	A forecast is treated as permission.
Check	Is the proposed transition admissible under the specification?	Predictive agreement is taken for constraint satisfaction.
Authorize	Does the authority context permit the action?	Physical safety is taken for permission, or the reverse.
Execute	Is the authorized action carried out as specified?	Task success is taken for compliance.
Record	What evidence will answer later audit questions?	Working state is taken for evidentiary record.

Table 2: The five stages, the question each answers, and what goes wrong when one is silently substituted for another.

The key principle is that no stage silently substitutes for another. A successful prediction is not authorization, an authorization judgment is not a guarantee of physical safety, and an execution record is not automatically sufficient for independent audit. The sequence governs the authorization and execution of particular actions. It is distinct from the cycle by which CLIO maintains and revises its constraint-bearing representation, which preserves the documentary history, enforces the active constraints, repairs diagnosed obstructions, and projects a new operative state [15]. The two operate at different levels, and the results of this paper bear on the last step of the cycle: the projected representation must remain sufficient for the questions that the later stages will ask. Each stage corresponds to a family of questions, each family has its own sufficiency condition, and a representation adequate for one stage is adequate for another only when the corresponding fiber conditions are checked.

The requirement can be stated for an arbitrary pipeline. Let a transition record space Ω be given, let stage i receive $\pi_i(\omega)$ for a projection π_i on Ω, and let q_i be the question stage i must answer.

Proposition 11.1 (Stagewise sufficiency)**.** *(i) Stage i can answer q_i exactly on all records if and only if π_i is sufficient for q_i. (ii) If the input of stage j is derived from the input of stage i, in the sense that $\pi_j = h \circ \pi_i$, then every question that stage j can answer exactly is a question that π_i is sufficient for, and so no stage downstream of a bottleneck can answer a question the bottleneck cannot.*

Proof. (i) is Theorem 2.2 with π_i in place of f. (ii) is Proposition 2.3. □

An architecture is consistent, in the sense this paper requires, when every stage satisfies (i) for the question assigned to it. Which decomposition into stages is chosen is arbitrary; whether the chosen one is consistent is not.

12 Proposed Experiments and Falsifiable Predictions

The experiments proposed here use small systems in which the true state, transition relation, and constraints are known, so that every sufficiency condition can be checked by enumeration or by direct computation. They are protocols. No results are reported, and the claim they are designed to support is modest: they illustrate and measure the consequences of the formal distinctions, and they do not demonstrate a generally safe architecture.

The first experiment constructs two physical states that a predictive encoder maps to the same latent point although their safe-action sets differ, in the manner of Example 4.3, and trains latent predictors of several architectures and capacities. The evaluation distribution over physical states must be stated in advance, since the bound depends on it. Theorem 4.1 predicts that, under that distribution, no classifier operating on the latent point can exceed on any fiber the larger of the two conditional class frequencies in that fiber, and so cannot exceed the corresponding average over fibers, whatever its training. A classifier that exceeds this bound would indicate leakage of the physical state into the classifier's input, and so would falsify the implementation and not the theorem.

The second experiment compares exact and conservative projections on a family of projections of increasing coarseness over a system with a known constraint set, and measures the proportion of safe actions rejected by $\bar{U}_{\text{safe}}$. Proposition 5.2 predicts that this proportion is nondecreasing in coarseness. The quantity of interest is the rate of growth, and in particular whether it is gradual or concentrated at particular merges.

The third experiment uses coupled oscillators with uncertain parameters and compares three representations: a phase-only representation, an augmented representation retaining amplitude, and joint state-parameter propagation. For each, it measures prediction accuracy, constraint violations, and the unnecessary rejection of viable actions. The expected pattern, from Theorems 4.1 and 6.1, is that the phase-only model may predict well and violate constraints, that the augmented model eliminates violations provided the simulated dynamics are correct and the conservative rule is actually enforced, and that joint propagation retains soundness under uncertainty at the cost of additional rejections that diminish as observations narrow the parameter set. The elimination of violations is a hypothesis conditional on those two provisions and not an unconditional prediction.

The fourth experiment simulates an agent with restricted tool access, in which a task has both a compliant route and a noncompliant route that is technically available. It distinguishes task completion from compliance with authorization boundaries and does not require any real network intrusion. The prediction is that evaluating on completion alone cannot distinguish the two routes, while an evaluation that includes the authorization predicate separates them.

The fifth experiment varies the evidence retained from execution histories and measures whether independent auditors can distinguish legitimate refusal, inability, unauthorized intervention, and unexplained nonperformance. Theorem 10.1(iii) gives an exact lower bound on auditor error for each retention regime under a stated distribution of histories, so that the measured error of the best auditor can be compared with a computed quantity. An auditor that beats the bound, as in the first experiment, would indicate information reaching the auditor by a channel the retention regime was supposed to close.

13 Discussion: What a World Model Must Preserve

The motivating claim of predictive architectures is that a world model need not reconstruct every detail of an observation. The analysis agrees, and adds that deciding which details can be omitted requires a specification of the questions the model must answer. Predictive, control, authorization, and evidentiary sufficiency can differ substantially, and a representation may satisfy one while violating another.

Several of the constructions used here have established antecedents, and the paper should be read as an assembly and separation of known ideas for a new setting and not as the discovery of any of them. Bisimulation for probabilistic systems and the corresponding notions of model minimization in Markov decision processes characterize when an abstraction preserves behavior [5, 6], and the hierarchy of state abstractions for decision processes gives precise relations among abstractions that preserve optimal values, optimal actions, and other properties [7]. Abstract interpretation supplies the soundness and completeness vocabulary applied in Section 5 [8], and viability theory and robust model predictive control supply the multistep framework of Section 6 [9, 10]. The classical sufficient statistic is the probabilistic ancestor of Section 2 [4]. The contribution claimed is the explicit separation of four sufficiency conditions, the characterization of each, and the study of how they interact for learned world models and agentic execution under an authority structure. Evidentiary sufficiency is defined on histories and the others on states or contexts, so the paper demonstrates particular non-implications and does not claim that all four are pairwise independent.

The principal limits of the analysis should be stated. The results hold relative to a transition relation, a set of occurring states, and an admissible set, and each of these is a modeling choice that can be wrong; a conservative rule is sound only with respect to the model it is derived from. The idealization to discrete or set-valued systems omits the measure-theoretic detail required for continuous state spaces, and the stochastic statements are given for pushforwards without a treatment of approximation. Authorization is relative to a context, and the question of who may grant authority is not formalized here and cannot be, since it is a question about institutions as much as about systems. The three-valued predicate of Section 8 acknowledges this by leaving some cases unresolved. Finally, the case studies motivate the distinctions and are not evidence for them: each is a single report with its own scope, and none establishes anything about the general frequency of the failure it illustrates. The interview with LeCun is a conversational source. Statements in it describe the intended direction of a research program and not a specification of any implemented system, and the paper evaluates the mathematical content of those statements and not the empirical performance of published predictive architectures, which it does not assess.

14 Conclusion

Compression is not inherently dangerous, and full-state reconstruction is not inherently desirable. The relevant requirement is that a representation preserve the distinctions on which the decisions and verification procedures that depend on it turn. A learned world model can be predictively successful while omitting control-relevant distinctions. A controller can be physically safe while exceeding its authority. An agent can satisfy an execution policy

while leaving insufficient evidence to establish what occurred.

The general principle is that admissibility is relative to a specified transition system and constraint set, authorization is relative to an authority context, and auditability is relative to the questions later reviewers must answer. Each relation has its own sufficiency condition, and the conditions can be stated exactly. The formal obstruction at the center of the argument is simple. Its consequence for learned world models and agentic systems is that the choice of representation is also a choice about which questions can later be answered, and the choice should be made against an explicit list of them. The particular architecture a system adopts is a design decision and can be arbitrary. That each of its stages is given a representation sufficient for the question it must answer, and that its terms are used with a stable meaning, are requirements, and they are the ones the results above make precise.

References

[1] LeCun, Y. A Path Towards Autonomous Machine Intelligence. Open position paper, version 0.9.2, 2022.

[2] Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., and Ballas, N. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[3] Nakagawa, D. AI Memorized the Internet. That's Not Intelligence. Interview with Yann LeCun, *Futurology*, 2026.

[4] Fisher, R. A. On the Mathematical Foundations of Theoretical Statistics. *Philosophical Transactions of the Royal Society of London, Series A*, 222:309–368, 1922.

[5] Larsen, K. G. and Skou, A. Bisimulation through Probabilistic Testing. *Information and Computation*, 94(1):1–28, 1991.

[6] Givan, R., Dean, T., and Greig, M. Equivalence Notions and Model Minimization in Markov Decision Processes. *Artificial Intelligence*, 147(1–2):163–223, 2003.

[7] Li, L., Walsh, T. J., and Littman, M. L. Towards a Unified Theory of State Abstraction for MDPs. In *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics*, 2006.

[8] Cousot, P. and Cousot, R. Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints. In *Proceedings of the Fourth ACM Symposium on Principles of Programming Languages*, pages 238–252, 1977.

[9] Aubin, J.-P. *Viability Theory*. Birkhäuser, Boston, 1991.

[10] Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. M. Constrained Model Predictive Control: Stability and Optimality. *Automatica*, 36(6):789–814, 2000.

[11] Ren, X. and Li, W. Isabelle Pool Server: A Pooled Verification Service for Prover Training and Multi-Agent Proving. arXiv:2611.24788, November 18, 2026. https://arxiv.org/abs/2611.24788.

[12] Wang, K., Cheng, O., Wen, Z., Yatbaz, M., Jain, S., Guo, C., and Helyar, A. Improving Fuzzy Robustness in LLM Safety with Automated Red Teaming. OpenAI Alignment Research Blog, November 12, 2026. https://alignment.openai.com/fuzzy-robustness/.

[13] OpenAI. Covert Underperformance on Safety Research Tasks. OpenAI Alignment, updated November 4, 2026. https://alignment.openai.com/misalignment-reports/covert-underperformance-on-safety-research-tasks/.

[14] OpenAI. Bypassing Network Restrictions During a Coding Evaluation. OpenAI Alignment, updated November 5, 2026. https://alignment.openai.com/misalignment-reports/bypassing-network-restrictions-during-a-coding-evaluation/.

[15] Flyxion. From Verification to Commitment: Histories, Repair Warrants, and Admissible Continuation in Agentic Systems. Independent research manuscript, 2026.

[16] Flyxion. Sufficient Projections. Essay, October 2026.

[17] Flyxion. Working State and Evidentiary Record in Compacted Agent Contexts. Essay, November 2026.