Agentic World Modeling: Foundations, Capabilities, Laws, and Beyond

Anonymous authors
Paper under double-blind review

Abstract

As AI systems move from generating text to accomplishing goals through sustained interaction, the ability to model environment dynamics becomes a central bottleneck. Agents that manipulate objects, navigate software, coordinate with others, or design experiments require predictive environment models, yet the term *world model* carries different meanings across research communities. We introduce a "levels × laws" taxonomy organized along two axes. The first defines three capability levels: **L1 Predictor**, which learns one-step local transition operators; **L2 Simulator**, which composes them into multi-step, action-conditioned rollouts that respect domain laws; and **L3 Evolver**, which autonomously revises its own model when predictions fail against new evidence. The second identifies four governing-law regimes (physical, digital, social, and scientific) that determine what constraints a world model must satisfy and where it is most likely to fail. Using this framework, we synthesize over 400 works and summarize more than 100 representative systems spanning model-based reinforcement learning, video generation, web and GUI agents, multi-agent social simulation, and AI-driven scientific discovery. We analyze methods, failure modes, and evaluation practices across level–regime pairs, propose decision-centric evaluation principles and a minimal reproducible evaluation package, and outline architectural guidance, open problems, and governance challenges. The resulting roadmap connects previously isolated communities and charts a path from passive next-step prediction toward world models that can simulate, and ultimately reshape, the environments in which agents operate.

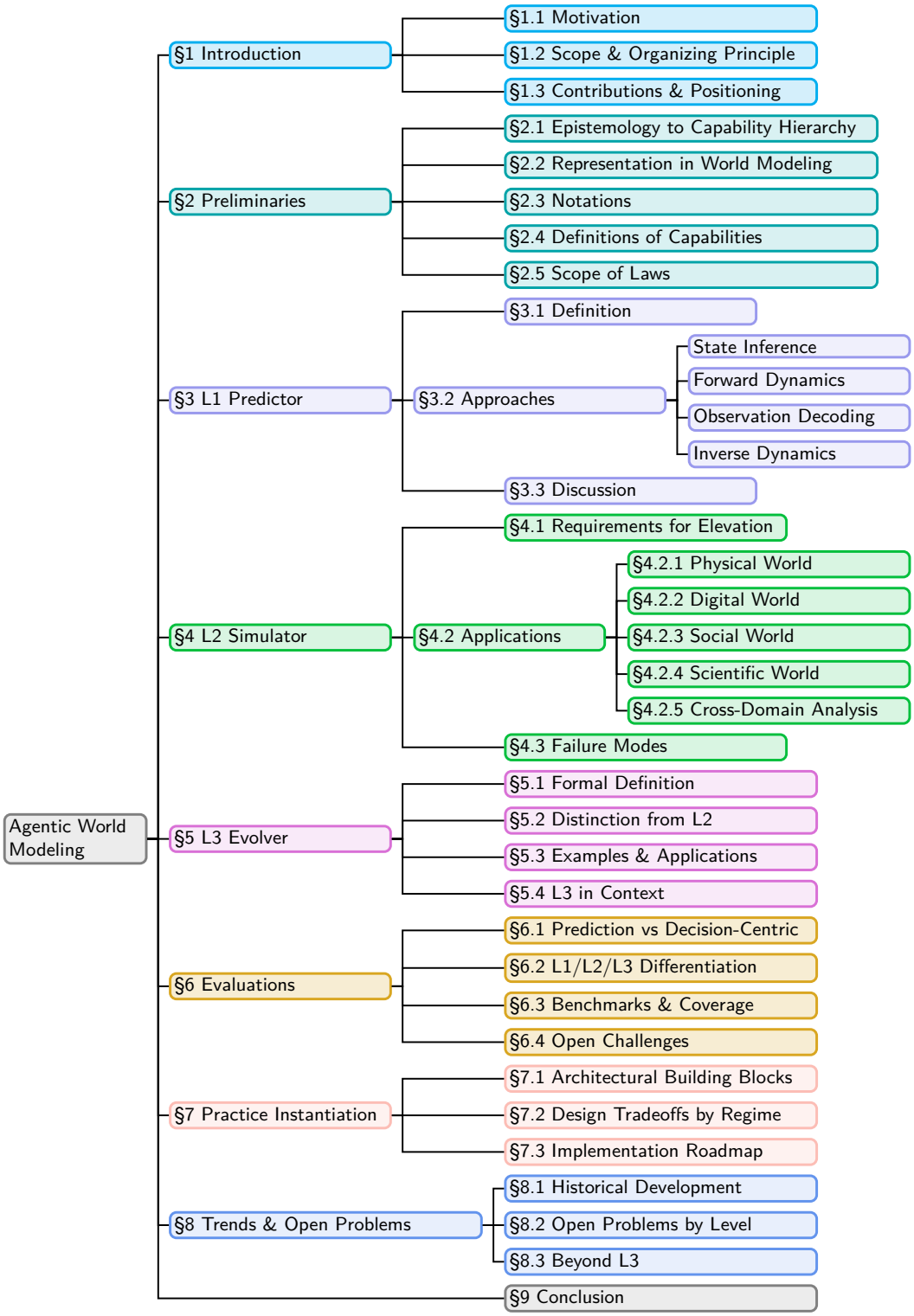

Figure 1: **Organizational structure of this survey.** The paper is organized around three capability levels (L1 Predictor, L2 Simulator, L3 Evolver) and four governing-law regimes (physical, digital, social, scientific worlds), with supporting sections on evaluation, implementation, and open problems.

1 Introduction

One may say the eternal mystery of the world is its comprehensibility.

Einstein (1936)

The ambition to build internal models of reality has a long intellectual history, appearing in philosophical accounts of mental models (Craik, 1943; Johnson-Laird, 1983) and in modern machine learning as learned latent dynamics that support prediction, control, simulation, and scientific reasoning (Ha & Schmidhuber, 2018; Hafner et al., 2020; Karniadakis et al., 2021). The phrase *world model* is now widely used across research communities, but its precise technical meaning varies considerably (Ding et al., 2025; Zhu et al., 2024). In reinforcement learning, agents learn transition structure to imagine futures before acting (Sutton, 1991; Ha & Schmidhuber, 2018; Hafner et al., 2020; Schrittwieser et al., 2020). In computer vision, *world models* often denote video or 3D generators that maintain visual dynamics and temporal coherence (Brooks et al., 2024; Bruce et al., 2024; Agarwal et al., 2025; Liang et al., 2026a;b; Bian et al., 2025; Kong et al., 2025). In language modeling and agent systems, the term can refer to text-grounded simulation for planning, web interaction, and social environments (Wang et al., 2024c; Gu et al., 2025b; Park et al., 2023; Zhang et al., 2026e;d). In robotics, learned dynamics serve safe planning, data-efficient policy learning, and sim-to-real transfer (Wu et al., 2023a; Yang et al., 2024b; Min et al., 2024). For science, systems pair surrogate models with hypothesis-driven experimentation (Karniadakis et al., 2021; Lu et al., 2024).

From a complementary perspective, world models and agents are closely coupled. At its core, a world model learns the state-transition dynamics of an environment: given a current state and an action, it predicts the resulting next state. An agent, conversely, selects actions given a task objective and its current observations. These two components are mutually supportive. Agents rely on world models to anticipate the consequences of candidate actions, enabling look-ahead planning and sample-efficient learning (Hafner et al., 2025; Schrittwieser et al., 2020; Dong et al., 2026; 2025). Conversely, world models benefit from agent-generated experience, which provides targeted, task-relevant trajectories that improve the model's accuracy in decision-critical regions of the state space (Sutton, 1991). This close coupling motivates the capability-based perspective adopted in this survey: while world models serve many purposes, we operationally define their value by the quality of decisions they enable for downstream agents.

Because world models constitute a foundational component whose value extends beyond any single agent architecture, their growing importance makes conceptual clarity all the more urgent. Yet the diversity outlined above also creates conceptual fragmentation: a vision researcher may evaluate a world model by the visual fidelity of its generated frames, while a reinforcement learning practitioner judges the same term by whether it improves task performance. As a result, papers may report strong progress under one interpretation of *world model* while remaining incomparable under another. This paper addresses that fragmentation by providing a common language that can align communities without erasing domain-specific differences.

1.1 Motivation

1. **Current survey landscape.** Several recent surveys have attempted to organize this rapidly growing literature. Ding et al. (2025) propose a dual taxonomy of *understanding* versus *predicting*, mapping world models onto application domains such as autonomous driving, robotics, and social simulacra. Zhu et al. (2024) focus on the generative capabilities catalyzed by Sora, surveying world models for video generation, autonomous driving, and autonomous agents. Yue et al. (2025) provide a roadmap for 2D visual world modeling with a four-generation capability taxonomy (G1–G4) applied to robotics, autonomous driving, and gaming. Their G1–G4 taxonomy is useful for distinguishing increasingly interactive visual generation systems; our L1–L3 hierarchy is complementary rather than competing, because it abstracts away from the visual modality and asks whether a system supports local prediction, decision-usable simulation, or evidence-driven revision across physical, digital, social, and scientific regimes. Roughly, early G-levels emphasize appearance and action-conditioned prediction, whereas our L2/L3 boundary is determined by constraint-valid rollout and persistent model update. Domain-specific surveys have also proliferated: Li et al. (2025e) provide a three-axis framework (functionality, temporal modeling, spatial representation)

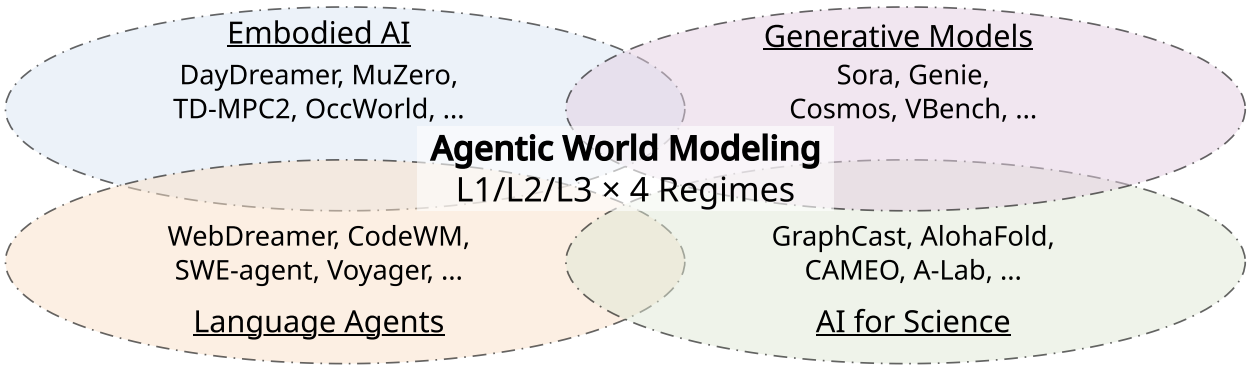

Figure 2: **Positioning of this survey relative to existing world model and agent surveys.** Four clusters, Embodied World Models, Generative World Models, Language Agents, and AI for Science, each cover subsets of the field. Our survey (center) integrates cross domain coverage with a capability based taxonomy (L1/L2/L3 × four regimes), bridging largely isolated communities.

specifically for embodied AI; Feng et al. (2025b) and Tu et al. (2025) survey world models for autonomous driving; Kong et al. (2025) examine 3D and 4D world modeling; Zhang et al. (2025d) survey world models for robotic manipulation; and a growing number of position papers question what it means for a learned model to "understand" physics (LeCun, 2022; Kang et al., 2025a). In AI for science, Wei et al. (2025b) survey autonomous scientific discovery across life sciences, chemistry, materials, and physics, unifying process-oriented, autonomy-oriented, and mechanism-oriented perspectives. A parallel line of surveys addresses agent planning and reasoning: Wei et al. (2025a) survey LLM planning capabilities across plan generation and verification, Huang et al. (2024a) taxonomize planning mechanisms into decomposition, selection, and reflection, Cao et al. (2025) provide a systematic comparison of fine-tuning versus search-based planning methods, Zhao et al. (2025) organize agentic reasoning into single-agent, tool-based, and multi-agent frameworks, and Arunkumar et al. (2026) propose a unified agent taxonomy spanning perception, planning, action, and collaboration. These surveys complement ours: they focus on how agents *decide and act*, whereas we focus on the predictive substrate (the world model) that makes those decisions informed. Despite their valuable contributions, existing surveys share a common organizational principle that we argue is fundamentally limiting: they partition the field by **modality** or by **application domain**. Our work differs by organizing the field through a capability-based taxonomy that cuts across modalities, covering decision-making domains from embodied manipulation and autonomous driving to web agents, multi-agent coordination, and scientific discovery pipelines.

Gaps in existing surveys. The modality-centric and domain-centric taxonomies leave two critical gaps. First, they fail to capture the *capability progression* that cuts across modalities. A key example is model-based reinforcement learning, where latent-space "imagination" rollouts can match or exceed model-free baselines across diverse domains such as Atari, continuous control, and Minecraft (Hafner et al., 2025; Schrittwieser et al., 2020; Hafner et al., 2020). We formalize this progression as a three-level capability hierarchy: one-step prediction, long-horizon simulation, and evidence-driven model revision. A second motivation for our framework is the intensifying debate over whether large-scale generative models are merely plausible generators or genuine world simulators. Existing surveys have surfaced this tension (Brooks et al., 2024; Bruce et al., 2024; Kang et al., 2025a; Ding et al., 2025), but a capability-based taxonomy helps state the question more precisely in terms of rollout, intervention sensitivity, and constraint consistency. We identify four progressively stronger capabilities, namely rollout, intervention sensitivity, constraint consistency, and closed-loop use, that characterize world models and go beyond generic predictors (formalized in Section 2). Moreover, existing surveys underrepresent the role of world modeling in agentic AI applications, including web agents, tool-use agents, and multi-agent systems, where learned environment dynamics are essential for planning and action selection (Gu et al., 2025b; Zhang et al., 2025b; Wang et al., 2024c; Park et al., 2023). The goal of this paper is to establish a capability-based taxonomy with clear boundary conditions, and to use it to connect research communities that currently evaluate world modeling systems with different assumptions, objectives, and metrics.

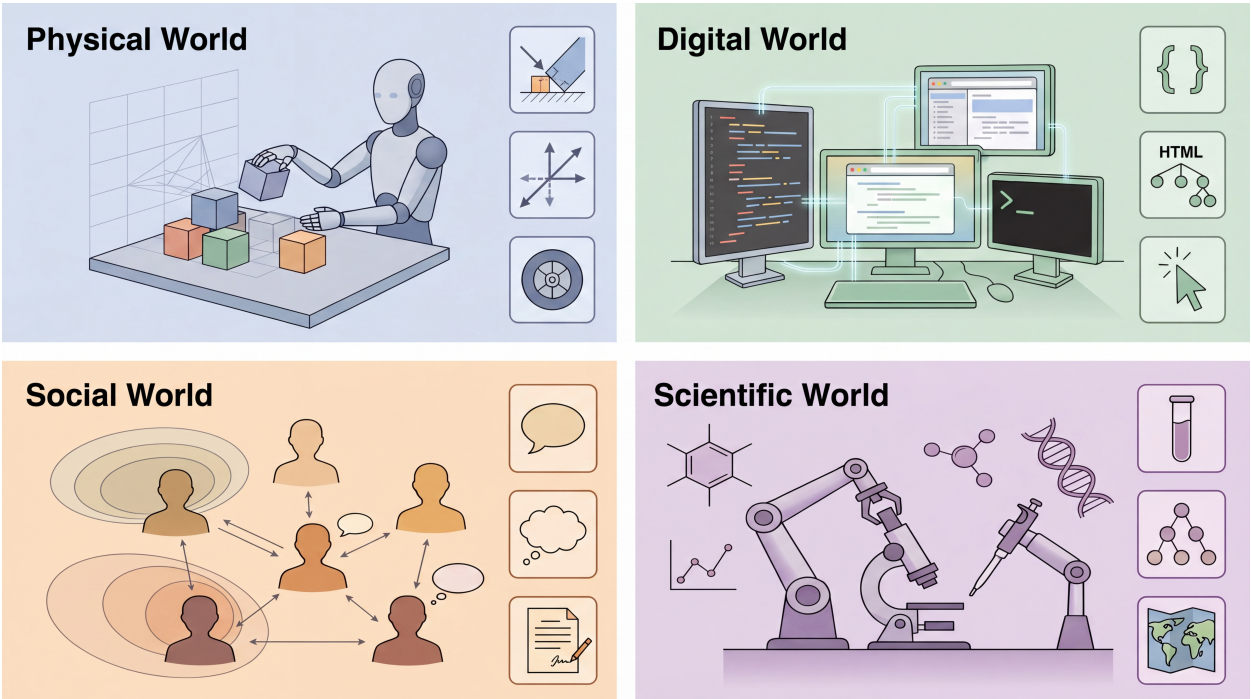

Figure 3: **Schematic illustrations of the four governing-law regimes.** Representative scenes for each regime: a humanoid agent manipulating blocks (Physical World), code and UI surfaces (Digital World), a network of interacting agents with speech acts (Social World), and instrumented experimentation with robotic microscope and pipette (Scientific World). Each regime's formal constraints are discussed in Section 2.5.

Figure 2 positions this survey relative to existing work along two axes: scope (domain-specific to cross-domain) and organizing principle (modality-centric to capability-centric). Figure 1 shows the organizational structure of the paper at a glance, grouping sections by the three capability levels (L1 Predictor, L2 Simulator, L3 Evolver) and the four governing-law regimes (physical, digital, social, and scientific worlds).

1.2 Scope and Organizing Principle

Governing principles across domains. We organize the paper along two orthogonal axes: (i) **capability level** (L1/L2/L3, defined formally in Section 2), and (ii) **governing-law regime**, the constraints that legitimate transitions must satisfy in a domain. These levels are stages of world-modeling capability rather than mutually exclusive model classes: the same system may invoke different levels at different moments depending on task demand. Figure 3 provides a schematic overview of these four regimes.

- **Laws of the Physical World**: perception; physical interaction; robotic manipulation, navigation, autonomous driving, egocentric video prediction, action-conditioned video modeling, 3D world modeling..
- **Laws of the Digital World**: program semantics; web navigation, software tool use, GUI environments.
- **Laws of the Social World**: beliefs; goals; norms; social coordination, dialogue, multi-agent settings.
- **Laws of the Scientific World**: latent mechanisms; experimental observables; causal structure; scientific discovery pipelines, measurement-coupled prediction, hypothesis-driven experimentation.

In particular, the physical and scientific regimes are separated by how constraints are accessed: physical-world systems often admit analytic or simulator-based verification of transitions, whereas scientific-world systems typically require empirical validation because the governing mechanisms are only partially known.

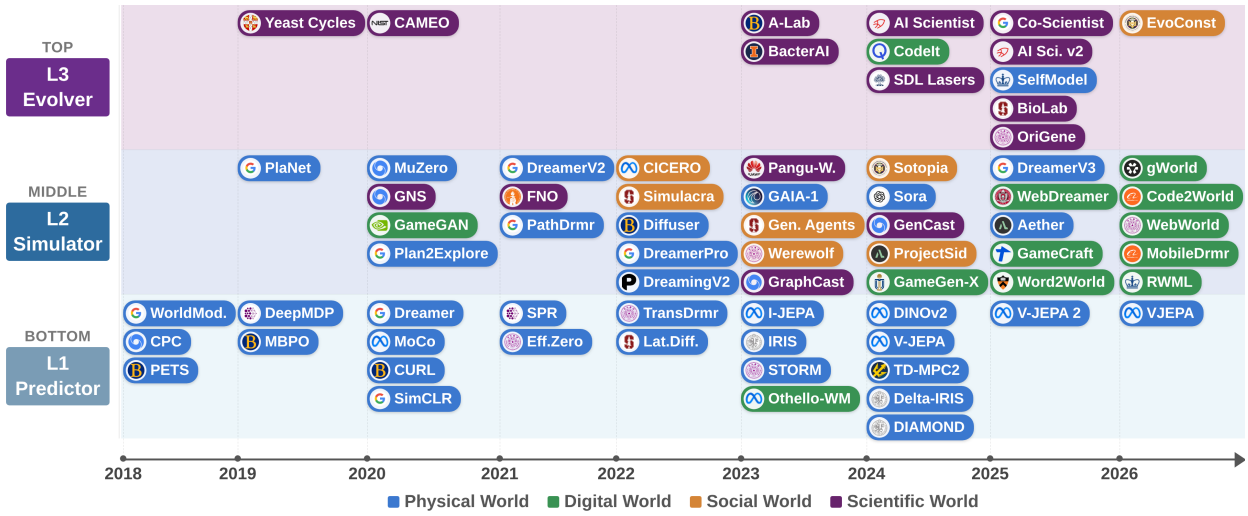

Figure 4: **Timeline of representative world modeling systems (2018–2026) organized by capability level.** The roadmap shows 70 survey anchors, capped at five systems per year–level cell for readability. L1 Predictor denotes one-step dynamics, L2 Simulator denotes decision-usable multi-step rollout, and L3 Evolver denotes full evidence-driven model revision; partial L3 loops remain in Table 8. Each pill is colored by governing-law regime: **Physical** (blue), **Digital** (green), **Social** (orange), and **Scientific** (purple).

Regimes are not "orthogonal modalities": real systems mix them. The value of the taxonomy is diagnostic; it clarifies *which* invariants a method tries to preserve and *which queries* it can answer reliably.

More generally, a world model can predict transitions along any organizing dimension, such as spatial scales, frequency bands, or causal depth, provided it maintains the capability criteria along that axis. Throughout, we use *world model* to denote learned (or hybrid) operators that support intervention-aware transition queries, and *world modeling* to denote the staged process of strengthening those operators.

How an agent uses the three levels at runtime. The L1/L2/L3 taxonomy is not a static classification of systems but a description of the capability an agent invokes at any given moment. A single deployed system can operate at different levels depending on the task demand:

1. **L1 (Predictor).** The agent executes fast, reactive one-step predictions (such as perception, low-level motor control, or token-by-token generation) without maintaining a multi-step plan.
2. **L2 (Simulator).** The agent upgrades to this level when the task requires comparing candidate action sequences, reasoning counterfactually about alternative futures, or verifying that a planned trajectory respects governing-law constraints; here the agent rolls out a multi-step simulation before committing.
3. **L3 (Evolver).** The agent escalates to this level when its current model produces systematic prediction failures that cannot be resolved by re-planning within the existing model structure, that is, when the model itself must be revised, assets distilled, and updates validated before the next deployment.

This runtime dispatch view clarifies why L3 is not a replacement for L1/L2 but a governance layer that improves the stack when evidence demands it. Within a full agentic stack, world models are only one component: tool use determines how the agent acts on the environment, memory determines what evidence persists across episodes, multi-agent coordination shapes the effective transition dynamics in social settings, and reflection determines when failures trigger revision rather than mere re-planning. Our focus is the world-model substrate, but its role is always in service of these broader agentic loops.

1.3 Contributions and Positioning

Key Contributions

This paper makes three principal contributions (Figure 4):

1. **Capability-based roadmap for world modeling in agentic AI (L1→L2→L3).** We propose a three-level capability hierarchy with testable boundary conditions: L1 **Predict World** (one-step prediction), L2 **Simulate World** (long-horizon, action-conditioned rollout with constraint satisfaction), and L3 **Modify World** (evidence-driven model growth through autonomous data collection and dynamics revision). These are stages of capability, not types of models.
2. **Cross-domain synthesis via governing laws.** We unify computer vision, language modeling, model-based RL and robotics, and AI for science into a single capability coordinate system. Different governing laws (Section 2) define the types or partitions of world models, **partially independent** of the L1→L2→L3 capability axis. This two-dimensional organization (capability level × law regime) reveals shared principles across communities that have developed in isolation, while clarifying domain-specific challenges that make direct transfer non-trivial.
3. **L3 as a distinct capability level.** Evidence-driven model growth, where a system autonomously collects new evidence and revises its own dynamics model, has appeared in scattered forms across scientific discovery (Lu et al., 2024), autonomous experimentation, and online adaptation. We argue this capability is qualitatively different from L2 rollout and formalize it as a distinct level, and identifying the open problems that must be resolved to realize this capability at scale.

Positioning. We present this paper as a *position-driven survey proposing a capability taxonomy for world modeling*. It advances a specific conceptual framework, namely the L1/L2/L3 capability hierarchy paired with a governing-law regime taxonomy, and argues for its adoption across the world modeling community. Unlike a pure survey, it proposes testable boundary conditions and uses them to re-examine how existing systems are classified. Unlike a pure position paper, it substantiates each argument with a comprehensive literature review spanning computer vision, reinforcement learning, robotics, natural language processing, and AI for science. This paper does not introduce a new benchmark or leaderboard; instead, it offers a unifying conceptual framework for interpreting and comparing existing systems and evaluations.

Outline. Section 2 establishes the conceptual and notational foundations: it motivates the three capability stages from epistemological intuition, gives each a formal definition with testable boundary conditions, and clarifies the distinctions between world modeling and generic prediction, world models and planners, and world modeling and commonsense. Sections 3–5 present the three capability levels in detail with representative methods and cross-domain analysis. Section 6 discusses evaluation methodology, Section 7 addresses architectural and computational considerations, and Section 8 identifies emerging trends and open problems. Section 9 concludes. We note that L3 is not a terminal stage; Section 8 introduces *meta-world modeling*, in which the governing laws themselves become learnable, and identifies the open problems this entails.

2 Preliminaries

This section establishes the conceptual and notational foundations used throughout the paper. **(1) From epistemology to a capability hierarchy** draws on philosophical traditions to propose a three-level decomposition of world modeling capability (L1 Predictor, L2 Simulator, L3 Evolver) and to motivate why the boundaries fall where they do. **(2) Notation and formal definitions** fixes a unified symbol system and uses it to give each stage (L1, L2, L3) a precise definition with testable boundary conditions. **(3) Conceptual boundaries** clarifies the distinctions between world modeling and generic prediction, between world models and planners, and relates world modeling to the broader notion of commonsense reasoning that underwrites the reliable everyday action that agents must exhibit beyond narrow predictive tasks.

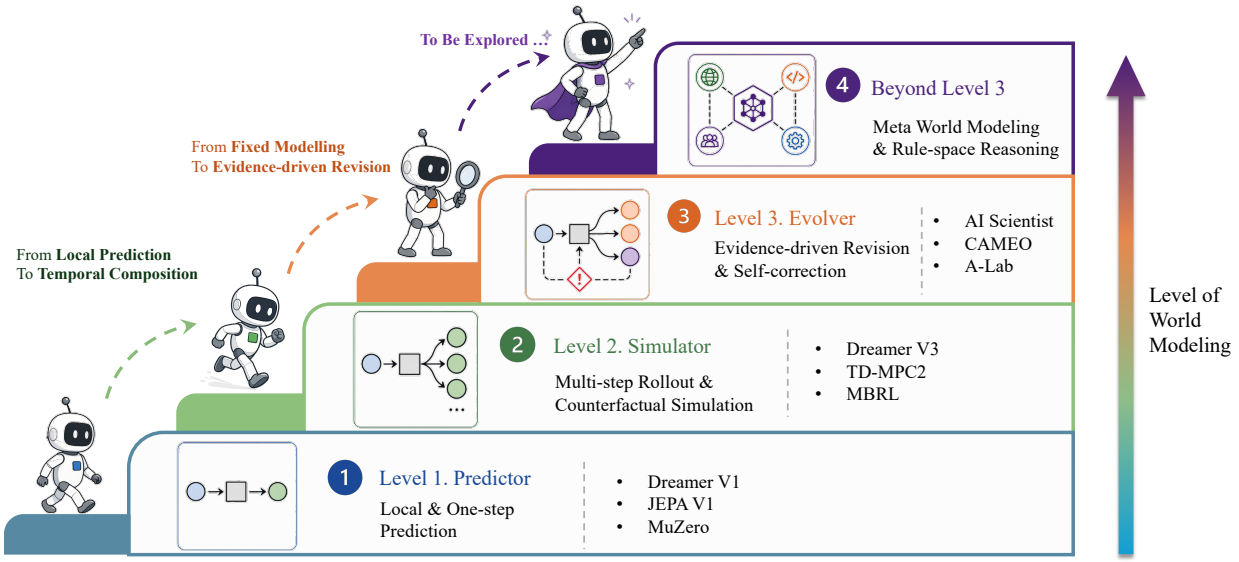

Figure 5: **From local prediction to evidence-driven revision: a hierarchical view of world modeling.** Level 1 models empirical regularities for prediction, Level 2 supports possible-world semantics and counterfactual simulation, and Level 3 introduces evidence-driven revision through continual interaction with the environment. This hierarchy frames world modeling as an ascending process from pattern recognition, to temporal rollout, to adaptive model evolution in real-world practice.

2.1 Philosophical Motivations

A natural question for any world-modeling survey is: *what stages of understanding does a system pass through as it moves from pattern matching to genuine modeling?* Epistemology, the study of what counts as knowledge and how knowledge grows, offers a useful lens. Different philosophical traditions identify qualitatively different kinds of epistemic achievement; we draw on these traditions to propose a three-level capability hierarchy for world models. These philosophical analogies are heuristic rather than historical or one-to-one. We do not claim that ML systems implement philosophical programmes, but that philosophical distinctions help us see why certain capability boundaries recur across domains and what design questions each stage foregrounds (Figure 5).

L1 Predictor: from pattern to one-step forecast. The simplest epistemic achievement is learning patterns from data: given past observations, predict the next one. In philosophy, this is the terrain of Hume's *constant conjunction* (Hume, 1739); an agent records statistical co-occurrences without certifying why they hold. When a model learns one-step latent transitions from trajectories, it occupies exactly this epistemic position: it extracts succession from data and bets that the pattern persists. This view aligns with predictive coding framework in cognitive science (Rao & Ballard, 1999; Friston, 2010) and the "Bayesian brain" hypothesis that perception is probabilistic inference (Clark, 2015), motivating one-step latent forecasting as a computational primitive (Lake et al., 2017).

We call this stage **L1 (Predictor)**. This Humean stance has inherent fragility. The i.i.d. assumption underlying most ML is effectively Hume's *Uniformity Principle* (the premise that the future will resemble the past), so when the distribution shifts, L1 models that rely on learned regularities fail to generalize. Nevertheless, this provides the most basic inductive bias, which is the foundation of modeling.

L2 Simulator: rollout and counterfactual. Pattern matching alone does not answer *what would happen if we acted differently*. The next stage adds intervention and counterfactual reasoning: the ability to roll out coherent futures under chosen actions or hypothetical initial conditions and use the results for decision-making. David Lewis's theory of *closest possible worlds* (Lewis, 1973) captures this jump: effective counterfactual reasoning explores worlds maximally similar to our own, where only a minimal intervention

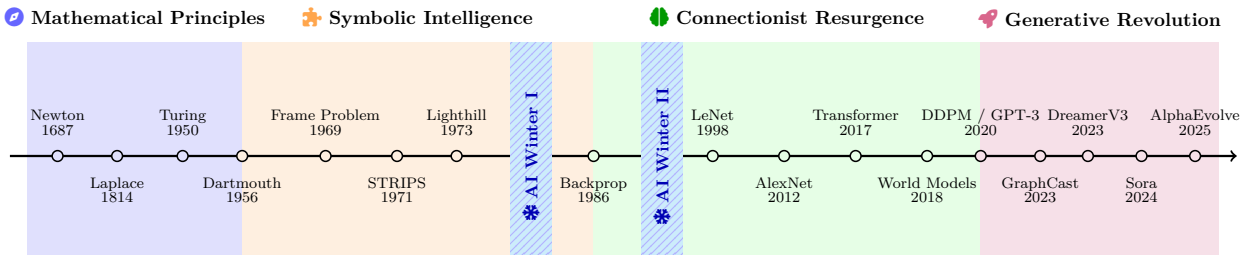

Figure 6: **Historical development of world modeling across four eras:** Mathematical Principles (–1956), Symbolic Intelligence (1956–1986), Connectionist Resurgence (1986–2020), and Generative Revolution (2020–present). Two AI winters (1974–1980, 1987–1993) mark transitions between paradigms. See discussions in Section 8.1. This argues that *a good representation of world model should be instantiation-agnostic*.

distinguishes actual from counterfactual outcomes, providing a principled basis for reasoning about what would have happened under alternative actions taken by the agent at decision points.

We call this stage **L2 (Simulator)**. Because L2 rollouts are model-relative, their reliability depends on the learned model's own transition structure rather than on direct access to ground-truth dynamics. They risk epistemic drift, which produces internally coherent trajectories for the training manifold. Plato's Allegory of the Cave (Plato, 1992) offers a vivid metaphor: a simulator excelling at predicting shadows on a wall may remain fundamentally bounded by the wall's dimensions, unable to access the fire casting those shadows.

L3 Evolver: model revision from evidence. Even a powerful simulator eventually encounters situations where its predictions systematically fail, not because of parameter error but because the model class itself is too narrow. Epistemology offers a rich vocabulary for this transition. Lakatos's distinction between a *hard core* (architecture, inductive biases) and a *protective belt* (learned parameters) (Lakatos, 1978) provides a useful parallel. Gradient steps mostly adjust the belt, while persistent structured errors may require changes to the core, such as new modules, parsers, constraints, or simulator hooks.

We call this stage **L3 (Evolver)**: the capacity to rebuild the laboratory when evidence demands it. This extends the full design–execute–observe–reflect loop: the system not only simulates but actively designs experiments, executes them, observes outcomes, and reflects to revise its model stack. Duhem–Quine holism (Duhem, 1954; Quine, 1951) explains why blame-assignment is non-trivial. Errors redistribute across modules until diagnostics isolate the brittle component. Proposed revisions should yield measurable improvements on held-out probes, regression suites, or experimental outcomes, rather than post-hoc adjustments that preserve the existing model despite contrary evidence from the environment.

2.2 Representation in World Modeling: Lessons from Scientific Theories

The capability hierarchy in Section 2.1 addresses *what a world model can do*, but leaves open a prior question: *in what form should the world model actually be represented?* This question should no just solely treated as an implementation detail, yet it determines whether the capabilities defined above, especially L3 revision, are realizable in practice across the diverse application domains covered in later sections.

Historically, symbolic approaches to machine intelligence struggled to scale (see Section 8.1), leading modern systems to adopt latent, implicit representations. Here scientific theories offer a telling contrast. Newton's laws, Maxwell's equations, and the Standard Model are instances of world models expressed in compact *symbolic* form, and arguably represent the most successful human instances of L3 systems: explicit, revisable, and composable. This contrast forces a question the field has largely avoided: is the endpoint of world modeling symbolic discovery, with neural latents as a scaffold, or are latent dynamics themselves the goal?

In scientific discovery, model updates arise at multiple scales: small anomalies trigger local modifications, while persistent discrepancies such as the "two dark clouds" (Kelvin, 1901) in late 19th-century physics expose epistemic gaps that force revisions to a theory's *invariance structure*. The shift from Newtonian

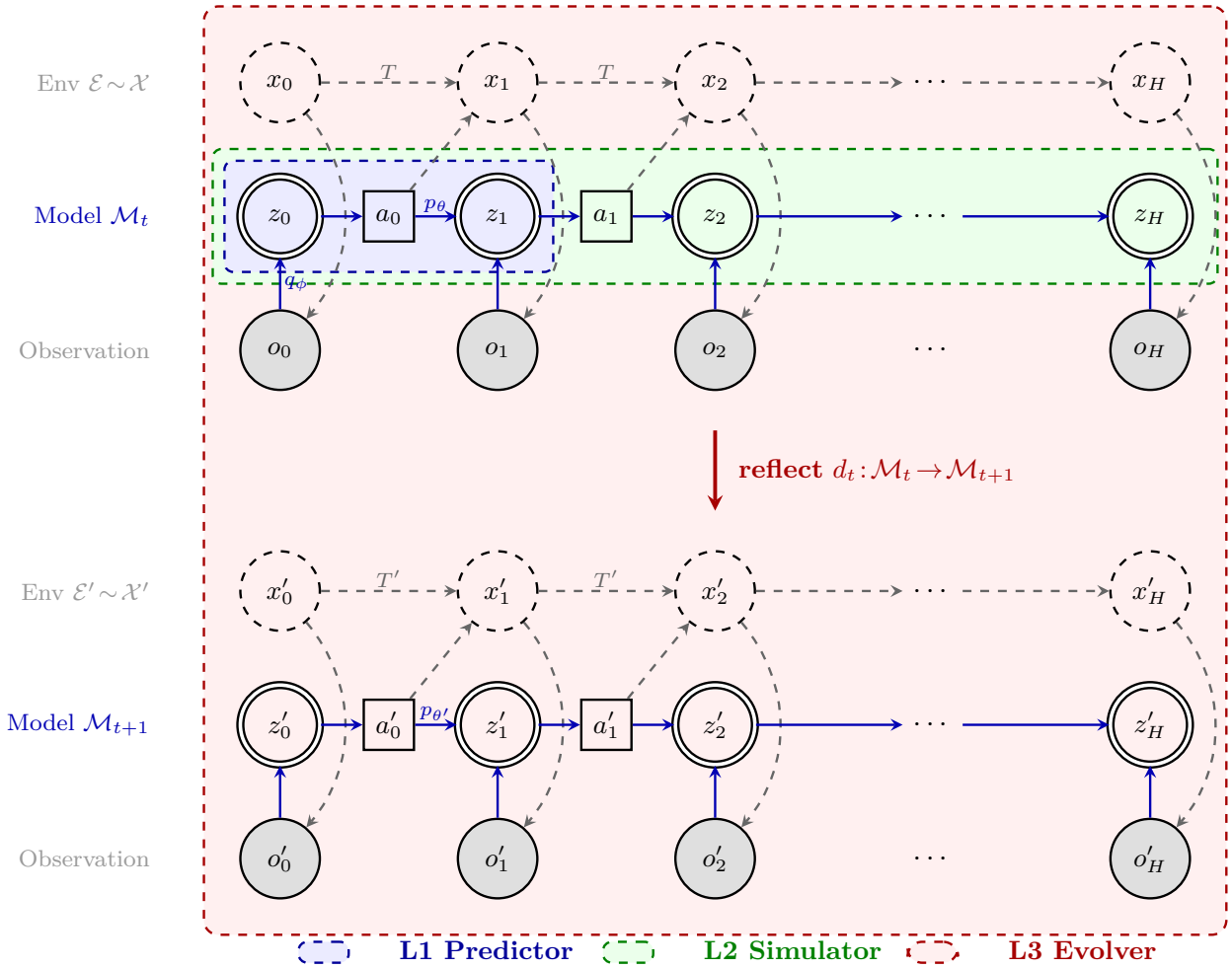

Figure 7: **Unified POMDP graphical model of L1-L3.** Dashed circles denote hidden environment states x; double circles denote learned latent states z; shaded circles denote observations o; squares denote actions a. Blue solid arrows denote the learned model (inference q_ϕ and dynamics p_θ); dashed gray arrows denote the environment transition T and observation emission. The top block shows the agent's POMDP under the current environment $\mathcal{E} \sim \mathcal{X}$ with model $\mathcal{M}_t$; the bottom block shows the same structure under a revised environment $\mathcal{E}' \sim \mathcal{X}'$ with model $\mathcal{M}_{t+1}$, obtained via the red **reflect** arrow. Colored dashed boxes mark each level's scope: L1 covers a single-step latent transition $p_\theta(z_t \mid z_{t-1}, a_{t-1})$; L2 covers the full trajectory rollout $\hat{p}(\tau \mid z_0, a_{1:H}, c)$ under a fixed model; L3 covers evidence-driven model revision $\mathcal{M}_t \to \mathcal{M}_{t+1}$, which corresponds to moving from $\mathcal{X}$ to a revised environment $\mathcal{X}'$ when the current model systematically fails.

to relativistic mechanics, for instance, replaced Galilean invariance with Lorentz invariance. Modern ML systems also encode invariances, such as translation equivariance in convolutions and shape bias in attention-based models (Geirhos et al., 2018), but do so *implicitly*, through architecture and training, rather than as explicitly modifiable structures. This suits L1 prediction and L2 simulation under a fixed model, but at L3 (where the task is to revise the model structure itself) it becomes a liability. Symbolic representations, by contrast, expose governing principles as first-class objects that can be directly inspected and modified.

We therefore take representation to be a foundational question about what a world model *is*, not a choice among interchangeable designs. Latent dynamics are indispensable as a scaffold for L1 and L2, but the endpoint of L3, namely genuine revision of governing laws, requires a symbolic substrate. On this view, L1→L2→L3 is a progression not only in rollout depth, but in how laws are discovered, composed, and

Table 1: **Notation summary used in this paper.**

Symbol	Definition
	Environment
$\mathcal{E} = (\mathcal{X}, \mathcal{A}, \Omega, T, O, R, \gamma)$	POMDP environment tuple
x_t	Hidden environment state at time t
o_t	Observation at time t (pixels, tokens, audio, etc.)
a_t	Action at time t
$T(x_{t+1} \mid x_t, a_t)$	Environment transition kernel
$O(o_t \mid x_t)$	Environment observation (emission) model
$R,\ \gamma$	Reward function and discount factor
	Learned World-Model Components
z_t	Learned latent / internal state
$q_\phi(z_t \mid o_{\leq t}, a_{\leq t-1})$	State inference (encoder / filter); parameters ϕ
$p_\theta(z_t \mid z_{t-1}, a_t)$	Forward dynamics (one-step latent transition); parameters θ
$p_\psi(o_t \mid z_t)$	Observation decoder; parameters ψ
$\pi_\eta(a_t \mid z_{t-1}, z_t)$	Inverse dynamics model; parameters η
$\hat{p}(\cdot)$	Trajectory-level (composed) distribution; hat marks approximate object
	Trajectories and Planning
$a_{1:H} = (a_1, \ldots, a_H)$	Action sequence of horizon length H
$\tau = (z_1, \ldots, z_H)$	Future latent segment (anchored at z_0)
$\hat{p}(\tau \mid z_0, a_{1:H}, c)$	L2 rollout query: trajectory distribution conditioned on anchor, actions, and constraints c
b_t; $\mathrm{Bel}(b_t, a_t, o_{t+1})$	Classical belief state and Bayesian belief update
π	Policy (consumes world-model queries; not part of the world-model factorization)
	L3 Model Revision
$\mathcal{M}_t$	World-modeling stack at revision step t
d_t	Deployment evidence (trajectories, errors, tests)
$\mathcal{H}$	Hypothesis space for model revision

revised. Practical instantiations or implementations across regimes are surveyed in Section 7. In the next Section 2.4, we introduce a foundational formalism that is instantiation-agnostic.

2.3 Notations

The preceding section proposed three capability stages from epistemological intuition. We now fix a unified symbol system, and Section 2.4 uses it to give each stage a precise definition. To cover model-based RL, predictive representation learning, video/world simulation, and generative modeling, we ground the notation in a Partially Observable Markov Decision Process (POMDP) (Kaelbling et al., 1998; Puterman, 1994). Figure 7 places this POMDP structure at the heart of the three-level taxonomy: each capability stage is visualized as a highlighted scope on the same graphical model. The environment is denoted by the tuple

$$\mathcal{E} = (\mathcal{X}, \mathcal{A}, \Omega, T, O, R, \gamma),$$

where $\mathcal{X}$ is the (unobserved) state space, $\mathcal{A}$ the action space, and Ω the observation space (pixels, tokens, audio, etc.). Transitions and observations follow

$$x_{t+1} \sim T(x_{t+1} \mid x_t, a_t), \qquad o_t \sim O(o_t \mid x_t).$$

Under partial observability, agents maintain a belief b_t or a learned latent state z_t. Classical belief updates are written $b_{t+1} = \mathrm{Bel}(b_t, a_t, o_{t+1})$; we reserve the symbol τ for latent trajectories below. Learned systems infer latents from history:

$$z_t = f_\phi(o_{\leq t}, a_{\leq t-1}) \quad \text{or} \quad q_\phi(z_t \mid o_{\leq t}, a_{\leq t-1}).$$

- T, O: environment transition and observation mechanisms.
- $q_\phi(\cdot)$: inference (history $\to$ latent).

- $p_\theta(\cdot)$: learned local predictive or generative factors (one-step dynamics, decoders, etc.), with parameters θ (and analogously ϕ, ψ for inference and rendering).
- $\hat{p}(\cdot)$: trajectory-level (or otherwise composed) distributions; the hat marks an explicit approximate object, e.g. the rollout marginal induced by repeated application of p_θ.
- π, R, γ: planner / policy, reward, and discount. These consume world-model queries but are not part of the world-model factorization $(q_\phi, p_\theta, p_\psi)$.

Convention: $\hat{p}$ is reserved for composed objects such as $\hat{p}(\tau \mid z_0, a_{1:H}, c)$; ordinary one-step dynamics are always written $p_\theta(z_t \mid z_{t-1}, a_t)$. Table 1 provides a concise reference for the symbols used in this paper.

$a_{1:H} = (a_1, \ldots, a_H)$ denotes an action sequence of length H applied starting immediately after an anchor state z_0. The future segment is

$$\tau \;=\; (z_1, z_2, \ldots, z_H),$$

so that $\hat{p}(\tau \mid z_0, a_{1:H}, c)$ matches the L2 formalism in Section 4. From an arbitrary time index t, the same convention applies after a trivial shift: anchor at z_t, condition on $a_{t+1:t+H}$.

2.4 Definitions of Capabilities

With the symbol system established in Section 2.3, we now give each capability stage a precise definition with testable boundary conditions.

L1 Predictor

An L1 world model provides local predictive operators that factorize into up to four components:

$$\begin{aligned}
\text{Inference / filtering:} \quad & q_\phi(z_t \mid o_{\leq t}, a_{\leq t-1}), && (1)\\
\text{Forward dynamics:} \quad & p_\theta(z_t \mid z_{t-1}, a_t) \quad \text{or, without actions, } p_\theta(z_t \mid z_{t-1}), && (2)\\
\text{Observation decoder:} \quad & p_\psi(o_t \mid z_t), && (3)\\
\text{Inverse dynamics:} \quad & \pi_\eta(a_t \mid z_{t-1}, z_t). && (4)
\end{aligned}$$

These operators target **one-step** (or short-horizon) accuracy under the training distribution; no guarantee is made about the coherence of multi-step composition. Section 3 presents representative methods in detail.

L2 Simulator

An L2 world model extends L1 from local operators to **decision-usable multi-step simulation**. It must support trajectory-level queries of the form

$$\hat{p}(\tau \mid z_0, a_{1:H}, c), \qquad \tau = (z_1, \ldots, z_H),$$

subject to three boundary conditions that collectively mark L1 $\rightarrow$ L2:

1. **Long-horizon coherence:** rollouts remain usable over H steps rather than degrading immediately via compounding error.
2. **Intervention sensitivity:** counterfactual edits (action or premise changes) induce stable and directionally meaningful trajectory changes.
3. **Constraint consistency:** generated futures respect the governing laws of the target regime (the physical, digital, social, or scientific world).

The key difference from L1 is not one-step quality but **rollout fidelity under composition**.

The three L2 boundary conditions are complementary rather than redundant. Long-horizon coherence concerns whether rollout quality survives composition over time; intervention sensitivity concerns whether

changes in actions or premises induce stable and directionally meaningful changes in the predicted future; and constraint consistency concerns whether the resulting trajectories remain valid under the governing laws of the target regime. None of these implies the others in general: a model may generate coherent but action-insensitive rollouts, or action-sensitive rollouts that still violate domain constraints. In practice they can also trade off against one another, for example when aggressive constraint enforcement stabilizes trajectories at the cost of reduced responsiveness to interventions.

A fourth capability, **closed-loop use** (supporting planning, acting, and self-improvement through interaction with the modeled environment), further separates world modeling from generic prediction but is orthogonal to L1/L2/L3: a weather emulator can be an L2 world model with no embedded planner. We reserve "closed-loop" for two different senses that must not be conflated: using a world model inside a control or planning loop is an orthogonal deployment property, whereas revising the world-model stack itself from deployment evidence is the defining hallmark of L3.

L3 Evolver

An L3 world model extends L2 from rollout over a fixed scaffold to **evidence-driven model revision**. In addition to simulation queries, an L3 system maintains an explicit update loop over model assets:

$$(\mathcal{M}_t,\ d_t) \xrightarrow{\text{diagnose} + \text{distill} + \text{validate}} \mathcal{M}_{t+1},$$

where $\mathcal{M}_t$ is the current world-modeling stack at revision step t and d_t is new deployment evidence (trajectories, errors, counterexamples, tests). Three boundary conditions mark L2 $\rightarrow$ L3:

1. **Evidence-grounded diagnosis:** failures are attributed to actionable causes using replayable evidence.
2. **Persistent asset update:** fixes are promoted as reusable assets (skills, rules, parsers, tests), not only ephemeral in-context patches.
3. **Governed validation:** updates pass regression and robustness gates (including rollback and canary policies) before default enablement.

The key difference from L2 is that the **model itself becomes an object of revision**, not merely a fixed scaffold to be queried (Lu et al., 2024; Boiko et al., 2023). Recapping the scopes in Figure 7: **L1 (Predictor)** is a single-step transition $p_\theta(z_t \mid z_{t-1}, a_{t-1})$ with its supporting inference and decoding operators, acting locally on one edge of the latent chain; **L2 (Simulator)** composes those local operators into a trajectory $\hat{p}(\tau \mid z_0, a_{1:H}, c)$ under a fixed model $\mathcal{M}_t$ and governing-law constraint c; and **L3 (Evolver)** revises the model stack $\mathcal{M}_t \rightarrow \mathcal{M}_{t+1}$ from distilled evidence d_t, yielding a different latent graph (the bottom block of the figure) whose effective environment $\mathcal{E}' \sim \mathcal{X}'$ may differ from the original, whether because the world itself has shifted, because the agent has uncovered previously unmodeled structure, or because the hypothesis space has been expanded. The three levels form a containment hierarchy: L2 invokes L1 at each step, and L3 invokes L2 each time it probes the world for evidence before committing to a model update.

Agent-centered view: state, action, and task. The formal components above describe an agent whose decisions are determined by three elements: the state it believes the world to be in, the action it can execute, and the task (or constraint c) it must satisfy. This triple, not a flat observation-to-action mapping, defines the interface between world model and planner. Building a useful z_t involves two orthogonal challenges that structure Section 3: (i) *spatial representation*: compressing a high-dimensional observation o_t into a compact latent that retains decision-relevant structure (geometry, semantics, affordances), and (ii) *temporal fusion*: integrating history $(o_{\leq t}, a_{\leq t-1})$ so that z_t approximates a Markov belief even in partially observable settings. Actions are not flat variables: they can emerge from representation learning rather than being pre-defined, with the core dynamics captured by the latent representation and everything else serving as a decoder (LeCun, 2022). Real agent behavior decomposes across temporal scales and abstraction levels, including low-level motor primitives, mid-level skills, and high-level task plans. The world model must predict transitions at

the granularity that matches the planner's query horizon. This action hierarchy interacts directly with the L1→L2 boundary: local dynamics suffice for primitive-level prediction (Sun et al., 2025a), but skill- and task-level rollouts require the multi-step coherence that defines L2. At the L3 level, the agent must not only predict transitions across temporal scales but also decide when its own transition model is inadequate and initiate model revision. L3 treats the world-modeling stack itself as an object of action. Diagnostic probes, architecture modifications, and regression tests become "meta-actions" that operate on the model rather than on the environment itself, reshaping how the system learns rather than merely how it acts.

2.5 Scope of Laws

As introduced in Section 1.2, we organize the survey along two orthogonal axes: capability level (L1/L2/L3) and governing-law regime. This subsection elaborates the four regimes and the constraints each imposes on the learned transition function. We distinguish **Laws of the Physical World** (governing agents that perceive and act in physical environments), **Laws of the Digital World** (governing deterministic program semantics: code, APIs, and state machines), **Laws of the Social World** (governing the dynamics of minds and institutions: beliefs, goals, and norms), and **Laws of the Scientific World** (governing systems that exist independently of human design, whose dynamics must be discovered from empirical observation). These four regimes are representative, not exhaustive. Real-world systems often operate under multiple regimes simultaneously. For example, autonomous driving involves both physical dynamics and social norms, while drug design couples natural mechanisms with digital simulation pipelines.

Laws of the Physical World constrain transitions through the physical dynamics that embodied agents must respect: contact mechanics, collision response, gravitational acceleration, friction, and kinematic feasibility. In robotics manipulation, autonomous driving, and interactive 3D simulation, the learned transition $p_\theta(z_t \mid z_{t-1}, a_t)$ must encode these physical interactions faithfully (Todorov et al., 2012; Hu et al., 2023; Wang et al., 2024e). This regime is distinguished by analytically characterizable governing equations. A physics engine or analytic model can verify whether a predicted transition is consistent with rigid-body constraints and Newtonian mechanics. Constraint violations appear as objects passing through each other, gravity reversing mid-rollout, or physically impossible deformations. Such failures are immediately detectable because the ground-truth dynamics admit closed-form or numerically exact reference solutions.

Laws of the Digital World constrain transitions through deterministic program semantics, including API contracts, UI state machines, file-system logic, and network protocols. In web navigation, code generation, and software testing, the transition function $p_\theta(z_t \mid z_{t-1}, a_t)$ is largely *deterministic* but branches heavily through error codes, permission checks, and edge cases (Gu et al., 2025b; Yao et al., 2022). This regime is defined by transitions that are both *specifiable* and *verifiable*. The program can be executed and its output compared against the model's prediction. Constraint violations appear as producing an API call that does not exist, ignoring returned error codes, or violating type constraints. Because the underlying system is a formal artifact, such errors are mechanically checkable.

Laws of the Social World constrain transitions through beliefs, goals, norms, social contracts, and institutional rules. In social simulation, dialogue systems, and multi-agent interaction, $p_\theta(z_t \mid z_{t-1}, a_t)$ maps joint actions and mental states to new mental states and social outcomes (Park et al., 2023; Zhou et al., 2025). Two properties set this regime apart. Transitions are *reflexive*, meaning that agents' beliefs about the state actively change the state itself. They are also *normative*, governed not only by what will happen but by what should happen according to shared conventions. Constraint violations appear as breaking a promise without consequence, forgetting a prior commitment, or ignoring established social norms. Such failures undermine coherence because social outcomes depend on mutual expectation.

Laws of the Scientific World constrain transitions through latent causal mechanisms that must be *discovered* from empirical observation rather than specified *a priori*. In weather prediction, molecular dynamics, protein folding, and drug design, $p_\theta(z_t \mid z_{t-1}, a_t)$ encodes atmospheric dynamics, chemical kinetics, or biological processes whose exact functional forms are unknown or too complex to write analytically (Karniadakis et al., 2021; Lam et al., 2023; Abramson et al., 2024). This regime differs in that the governing equations are *not available in closed form*. The world model must learn them from data and be validated against experimental measurement. Constraint violations appear as predicting physically impossible molecular configurations,

violating conservation laws that hold empirically, or ignoring known causal dependencies. Detection typically requires comparison with laboratory or observational data rather than symbolic verification.

With these foundations in place, the following sections instantiate each capability level in turn: Section 3 surveys L1 methods, Section 4 addresses L2 simulation, and Section 5 examines L3 model revision.

3 L1 Predictor: Local Markov Prediction

The hierarchical structure begins with **L1**, which assesses a world model's local predictive ability by requiring it to sustain a meaningful internal state and use *local* predictive mechanisms to anticipate the next state, including potential observations or actions. In the unified graphical model of Figure 7, L1 is the scope of a single edge $z_{t-1} \to z_t$ conditioned on action a_{t-1}; everything in this section elaborates the operators that populate this one-step transition and examines how they are realized in contemporary world-model systems.

3.1 Definition

L1 concerns the local predictive ability of a world model for an agent acting in an environment to accomplish a task or goal. More precisely, an agent is a system that, given observations, makes decisions and takes actions in order to satisfy an objective. In this paper, the role of an L1 world model is therefore not merely to predict the next signal, but to provide local predictive operators that support such decision-making at the granularity of one step (or a short fixed horizon). This epistemic stance aligns with Hume's constant conjunction: regularities are extracted from observed data without claiming causal necessity (Section 2.1).

The POMDP formulation that underlies L1 originates from the reinforcement learning literature, where an agent must select actions under partial observability to maximize cumulative reward (Kaelbling et al., 1998; Puterman, 1994). In this setting, the agent maintains an internal belief over hidden states and crafts a policy $\pi(a_t \mid b_t)$ that maps beliefs to actions. This formulation constitutes the prototypical agent–environment loop (Sutton, 1991). For an agent that interacts with the environment to accomplish a task, the POMDP decomposes into four local operators: state inference, forward dynamics, observation decoding, and inverse dynamics. Together these describe the foundational learning problems for world models at the L1 level.

Following this formulation (Section 2), **L1** is characterized by **local predictive operators** operating on a learned internal state z_t (resembling a belief state), where the central modeling concept centers on a **one-step** (or short fixed-horizon) transition operator. In practical terms, z_t is deduced from observations and actions and functions as a learned approximation to the latent environmental state and/or belief (Hafner et al., 2025; Schrittwieser et al., 2020). The concept of learning such latent dynamics can be traced back to locally linear latent models for control (Watter et al., 2015) and Gaussian-process dynamics (Deisenroth & Rasmussen, 2011), and has been enhanced by contemporary deep learning architectures (Ha & Schmidhuber, 2018; Hafner et al., 2020). The term "Markov" in **L1** denotes the *Markovian property in the learned internal state* z_t, indicating that z_t is adequate (or nearly adequate) for predicting the subsequent local step, rather than the direct observability of the environmental state (Hafner et al., 2019; 2025; Gelada et al., 2019).

At the model level, L1 factorizes into four local operators over z_t (Table 2). The *core* operator is latent dynamics ($z_{t-1} \to z_t$); the others are common supporting operators:

- **State inference** (observation → state, Eq. equation 1): $z_t = f_\phi(o_{\leq t}, a_{\leq t-1})$ or $q_\phi(z_t \mid o_{\leq t}, a_{\leq t-1})$. The learned belief-like state summarizes relevant history for prediction (Hafner et al., 2019; Lesort et al., 2018).
- **Forward dynamics** (state → next state; core L1 operator, Eq. equation 2): $z_t \sim p_\theta(z_t \mid z_{t-1}, a_t)$ (action-conditioned) or $z_t \sim p_\theta(z_t \mid z_{t-1})$ (action-free).
- **Observation decoding** (state → observation, Eq. equation 3): $p_\psi(o_t \mid z_t)$, mapping latent state back to observation space (Kingma & Welling, 2014; Rezende et al., 2014).
- **Inverse dynamics** (Eq. equation 4): $\pi_\eta(a_t \mid z_{t-1}, z_t)$, used as an auxiliary objective or for representation shaping (Pathak et al., 2017; Hafner et al., 2020).

Table 2: **L1 component factorization.** The four local operators form the building blocks of L1 world models. The core operator is forward dynamics; the others are supporting operators.

Operator	Mapping	Formal Definition	Role
State Inference	$o_t \to z_t$	$z_t = f_\phi(o_{\leq t}, a_{\leq t-1})$ or $q_\phi(z_t \mid o_{\leq t}, a_{\leq t-1})$	Compress observations into latent belief state
Forward Dynamics	$z_{t-1} \to z_t$	$p_\theta(z_t \mid z_{t-1}, a_t)$	Predict next latent state given action
Observation Decoding	$z_t \to o_t$	$p_\psi(o_t \mid z_t)$	Reconstruct observations as training signals
Inverse Dynamics	$(z_{t-1}, z_t) \to a_t$	$\pi_\eta(a_t \mid z_{t-1}, z_t)$	Infer actions; representation shaping

Table 3: **Representative L1 methods.** Columns indicate which local operators each method instantiates: state inference (SI), forward dynamics (FD), observation decoding (OD), and inverse dynamics (ID).

Method	Links	SI	FD	OD	ID	Architecture
Representation Learning						
VAE (Kingma & Welling, 2014)	Paper –	✔	✗	✔	✗	MLP encoder–decoder
β-VAE (Higgins et al., 2017)	Paper –	✔	✗	✔	✗	MLP encoder–decoder
VQ-VAE (van den Oord et al., 2017)	Paper Code	✔	✗	✔	✗	CNN + discrete codebook
CPC (Oord et al., 2018)	Paper –	✔	✗	✗	✗	CNN + autoregressive
SimCLR (Chen et al., 2020)	Paper Code	✔	✗	✗	✗	ResNet + projection head
MoCo (He et al., 2020)	Paper Code	✔	✗	✗	✗	Momentum encoder
CURL (Laskin et al., 2020)	Paper Code	✔	✗	✗	✗	CNN+momentum encoder
SPR (Schwarzer et al., 2021)	Paper Code	✔	✗	✗	✗	CNN + prediction MLP
I-JEPA (Assran et al., 2023)	Paper Code	✔	✗	✗	✗	ViT + predictor
V-JEPA (Bardes et al., 2024)	Paper Code	✔	✗	✗	✗	ViT + predictor
DINOv2 (Oquab et al., 2024)	Paper Code	✔	✗	✗	✗	ViT self-distillation
Model-Based RL						
PILCO (Deisenroth & Rasmussen, 2011)	Paper Code	✗	✔	✗	✗	Gaussian process
E2C (Watter et al., 2015)	Paper –	✔	✔	✔	✗	Locally linear latent
PETS (Chua et al., 2018)	Paper Code	✗	✔	✗	✗	Ensemble of NNs
World Models (Ha & Schmidhuber, 2018)	Paper Code	✔	✔	✔	✗	VAE + MDN-RNN
Dreamer (Hafner et al., 2020)	Paper Code	✔	✔	✔	✗	RSSM (GRU + stoch.)
DreamerV2 (Hafner et al., 2021)	Paper Code	✔	✔	✔	✗	RSSM (discrete stoch.)
DreamerV3 (Hafner et al., 2025)	Paper Code	✔	✔	✔	✗	RSSM + symlog
MuZero (Schrittwieser et al., 2020)	Paper –	✔	✔	✗	✗	MLP dynamics + MCTS
EfficientZero (Ye et al., 2021)	Paper Code	✔	✔	✗	✗	MuZero + self-sup.
TD-MPC2 (Hansen et al., 2024)	Paper Code	✔	✔	✗	✗	MLP latent dynamics
DeepMDP (Gelada et al., 2019)	Paper –	✔	✔	✗	✗	Bellman-aligned latent
MBPO (Janner et al., 2019)	Paper Code	✗	✔	✗	✗	Ensemble of NNs
Token / Diffusion-Based						
IRIS (Micheli et al., 2023)	Paper Code	✔	✔	✔	✗	VQ-VAE + Transformer
TransDreamer (Chen et al., 2022)	Paper Code	✔	✔	✔	✗	Transformer-XL + stoch.
Latent Diffusion (Rombach et al., 2022)	Paper Code	✔	✗	✔	✗	Latent-space diffusion
STORM (Zhang et al., 2023)	Paper Code	✔	✔	✔	✗	Transformer + VAE
DIAMOND (Alonso et al., 2024)	Paper Code	✗	✔	✗	✗	Pixel-space diffusion
Delta-IRIS (Micheli et al., 2024)	Paper Code	✔	✔	✔	✗	VQ-VAE + delta coding

3.2 Approaches

We categorize notable **L1** techniques based on the four local operators delineated earlier: state inference (concerned with deriving z_t from observations and historical data), forward dynamics (the fundamental

transition model), observation decoding (associating z_t with o_t), and inverse dynamics (deducing actions from successive states) (Ding et al., 2025; Moerland et al., 2023). Table 3 summarizes representative methods and their key innovations. We devote the most space to forward dynamics because it is the operator that most directly determines whether an L1 system can later be elevated into an L2 simulator; the other components are still essential, but their role is primarily to make the latent state usable for that transition.

3.2.1 State Inference

State inference condenses high-dimensional observations into a compact latent representation z_t that preserves crucial decision-making information, and it integrates temporal context to ensure that z_t approximates a Markovian belief in partially observable scenarios (Lesort et al., 2018).

Contrastive Predictive Coding (CPC; Oord et al. 2018) trains an encoder to maximize mutual information between present and future embeddings via the InfoNCE loss, which contrasts temporally adjacent positive pairs against negatives drawn from the same batch. SimCLR (Chen et al., 2020) and MoCo (He et al., 2020) established general-purpose contrastive frameworks through augmentation-based positive pairs and momentum-updated encoders, respectively, providing pretrained visual backbones that downstream world models build upon. However, general visual representations do not guarantee that z_t preserves control-relevant information. CURL (Laskin et al., 2020) addressed this by extending contrastive learning to RL with temporal adjacency between consecutive frames as positive pairs, achieving sample efficiency comparable to model-based methods in Atari and continuous control. Self-Predictive Representations (SPR; Schwarzer et al. 2021) trains an encoder to forecast its own future representations, incorporating temporal structure and decision-relevant dynamics. Both methods show strong sample efficiency on Atari 100k, confirming that world model representations benefit from being tailored to decision-making objectives.

Rather than contrasting pairs, another family of methods predicts the embedding of masked regions directly in latent space. JEPA and its variants, I-JEPA (Assran et al., 2023) and V-JEPA (Bardes et al., 2024), forecast hidden-region embeddings without decoding back to pixels (LeCun, 2022). This approach encourages the encoder to grasp semantic and structural consistencies without being bound to intricate reconstruction at the pixel level. On another front, DINOv2 (Oquab et al., 2024) from the foundation-model domain generates versatile visual features through self-distillation, establishing robust state encoders for subsequent tasks. A complementary direction makes the inferred state explicitly object-centric and programmatic rather than purely continuous. Thinking with Blueprints converts an image into a JSON-style blueprint that records the positions, sizes, and attributes of question-relevant objects, and then reasons over this structured representation to answer spatial queries (Ma et al., 2026). Although proposed for VLM spatial reasoning rather than sequential control, it is highly relevant to L1 state inference because it shows that useful internal state can take the form of a decision-oriented scene description, not only a dense latent embedding.

A third line of work shapes z_t through control-oriented auxiliary objectives such as reward anticipation, inverse-model losses (Pathak et al., 2017), and value-function consistency, as formalized in DeepMDP (Gelada et al., 2019). This framework articulates the necessity for the latent Markov chain to approximately adhere to the Bellman equations. Embed to Control (E2C; Watter et al. 2015) acquired locally linear latent dynamics simultaneously with a VAE encoder-decoder for LQR-based planning within the latent space.

When a single observation is insufficient, the model must aggregate past information into z_t. The Recurrent State Space Model (RSSM) of Hafner et al. (2019) splits the latent into a deterministic recurrent pathway $h_t = f(h_{t-1}, z_{t-1}, a_{t-1})$ and a stochastic component $z_t \sim q_\phi(z_t \mid h_t, o_t)$, compressing arbitrary-length histories while preserving stochastic uncertainty. This recurrent belief state (h_t, z_t) serves as the internal state for all downstream prediction and control in the Dreamer family (Hafner et al., 2020; 2021; 2025).

Scientific applications illustrate how the same state-inference principle operates when raw observations are high-dimensional and the scientifically meaningful state is latent. In structural biology, protein structure prediction can be cast as L1 state inference: mapping an amino acid sequence (observation) to a dominant 3D coordinate state. The AlphaFold lineage progressed from learned distance-based potentials (Senior et al., 2020) to end-to-end Evoformer architectures with near-experimental accuracy (Jumper et al., 2021) to diffusion-based prediction of joint biomolecular complex structures (Abramson et al., 2024). Parallel efforts showed that strong structure prediction is also achievable through three-track networks (Baek et al.,

2021) and protein language models enabling single-sequence inference (Lin et al., 2023). In neuroscience, HMM (Baker et al., 2014), RNN (Gohil et al., 2022) and Transformer (Khan et al., 2023) are used to map electrophysiological recordings to a set of latent network modes, following a state-inference paradigm conceptually similar to Ha & Schmidhuber (2018). Analysis of the learned interpretable latent representations reveals various findings: cortical activity at rest can be described by transient, intermittently recurring events (Vidaurre et al., 2018) organized into cycles on 300–1,000 ms timescales (van Es et al., 2025).

3.2.2 Forward Dynamics: The Core L1 Operator

These approaches directly establish $p_\theta(z_t \mid z_{t-1}, a_t)$ and form the *core* of L1. The precision of the dynamics network is crucial for producing valuable one-step forecasts and must be expressive enough to be aggregated over numerous steps, a critical requirement that is more stringent at L2 (Moerland et al., 2023).

In model-based Reinforcement Learning (RL), the action-conditioned latent transition model plays a pivotal role. PILCO (Deisenroth & Rasmussen, 2011) employed Gaussian-process dynamics coupled with analytical uncertainty propagation for efficient continuous control. MuZero (Schrittwieser et al., 2020) adopts a deterministic dynamics function $z_t = f_\theta(z_{t-1}, a_t)$ trained end-to-end for value prediction and Monte Carlo Tree Search without observation reconstruction. EfficientZero (Ye et al., 2021) further enhanced this approach by incorporating self-supervised consistency losses, achieving superhuman performance in Atari games with just two hours of experience. Conversely, the Recurrent State Space Model (RSSM) in Dreamer (Hafner et al., 2020; 2021; 2025) leverages stochastic dynamics to facilitate uncertainty-aware rollouts. PETS (Chua et al., 2018) illustrated that an *ensemble* of dynamics models offers reliable epistemic uncertainty estimation crucial for robust planning. TD-MPC (Hansen et al., 2022) learns latent dynamics through temporal difference objectives, directly aligning the dynamics model with value estimation; TD-MPC2 (Hansen et al., 2024) scales up to a single 317 million-parameter agent proficient in mastering 104 tasks across various domains.

A recent trend is the shift from continuous latent dynamics to discrete-token or diffusion-based transitions. For instance, IRIS (Micheli et al., 2023) tokenizes observations using a VQ-VAE codebook (van den Oord et al., 2017) and models the resulting sequence with an autoregressive Transformer. Meanwhile, Trans-Dreamer (Chen et al., 2022) swaps the GRU of RSSM with Transformer-XL to enhance long-range attention. Zhang et al. (2023) combine Transformer sequence modeling with stochastic VAE dynamics, whereas Micheli et al. (2024) encode stochastic deltas between frames instead of entire frames. On the diffusion front, DIAMOND (Alonso et al., 2024) employs diffusion denoising as the one-step transition operator to preserve visual intricacies that may be overlooked by low-capacity latent dynamics.

Building on the principles of predictive coding (Rao & Ballard, 1999; Friston, 2010), methodologies such as CPC (Oord et al., 2018), SPR (Schwarzer et al., 2021), and JEPA (Bardes et al., 2024; Assran et al., 2023) forecast the *latent embedding* of the forthcoming observation rather than the observation itself. While SimPLe (Kaiser et al., 2020) showcased the viability of pixel-level video prediction as a world model for efficient Atari RL, the discrepancy in prediction fidelity between pixel-space and latent-space underscores the progression towards abstract dynamics and the transition from L1 to L2.

Beyond using dreaming for policy optimization, one-step dynamics models also function as generators of experience: MBPO (Janner et al., 2019) integrates short model rollouts into a replay buffer to enhance sample efficiency; Nagabandi et al. (2018) demonstrated that model-based pre-training combined with model-free fine-tuning leverages the strengths of both approaches. DayDreamer (Wu et al., 2023a) transfers latent imagination to physical robots, Wang et al. (2024b) extend this to transfer reinforcement learning (RL) knowledge across visual domains, and Hao et al. (2025) propel world models towards acquiring long-horizon physical skills through neural motion simulation. These applications underscore that L1 dynamics serve as a data catalyst, a planning foundation (Schrittwieser et al., 2020; Hafner et al., 2025; Hansen et al., 2024), and a mechanism for compressing raw interactions into high-level behaviors (Moerland et al., 2023).

3.2.3 Observation Decoding

The decoder implements $p_\psi(o_t \mid z_t)$ and has three key functions. It provides a training signal to ensure that z_t preserves sufficient information, serves as a diagnostic interface to examine the model's learned

representations, and acts as a rendering engine for generating envisioned observations during dreaming (Ha & Schmidhuber, 2018; Hafner et al., 2020).

The Variational Autoencoder (VAE; Kingma & Welling 2014, independently Rezende et al. 2014) offers a standard probabilistic framework: the encoder $q_\phi(z_t \mid o_t)$ maps observations to a latent posterior, while the decoder $p_\psi(o_t \mid z_t)$ reconstructs observations from latents, trained simultaneously through the ELBO. In world-modeling workflows, the VAE compresses raw pixel inputs into a concise code transmitted to a dynamics model (Ha & Schmidhuber, 2018; Hafner et al., 2019). β-VAE (Higgins et al., 2017) amplifies the KL divergence term to encourage disentangled factors, while VQ-VAE (van den Oord et al., 2017) replaces continuous latents with a discrete codebook, foundational for token-based world models (Micheli et al., 2023). The *World Models* concept by Ha & Schmidhuber (2018) introduced the fusion of a VAE encoder, an LSTM dynamics model (MDN-RNN), and a distinct controller. Subsequently, the Dreamer lineage (Hafner et al., 2020; 2021; 2025) expanded this paradigm into an end-to-end latent imagination framework, where policy and value functions are trained solely on imagined latent trajectories, with the decoder acting mainly as a regularizer and auxiliary fidelity check rather than as a generative objective in its own right.

Large-scale video generation models like Sora (Brooks et al., 2024) leverage high-capacity observation decoders to produce photorealistic frames from latent trajectories. Latent Diffusion Models (Rombach et al., 2022) compress images into a lower-dimensional latent space and apply diffusion processes more efficiently in that space, while the Diffusion Transformer (DiT; Peebles & Xie 2023) enhances scalability by replacing the U-Net backbone with a standard Transformer. Recent image-generation backbones also explore information-adaptive tokenization and generation, such as Dynamic Generative Image Transformer (Mao et al., 2026b), which suggests a useful design direction for observation-level decoders even though it is not itself a multi-step world model. These models illustrate the feasibility of achieving high-quality $p_\psi(o_t \mid z_t)$ at scale. However, the quality of the *latent dynamics* that steer the decoder, and the ability of z_t to facilitate coherent multi-step prediction, remain challenges that motivate the transition to the L2 discussion.

3.2.4 Inverse Dynamics

The inverse dynamics operator $\pi_\eta(a_t \mid z_{t-1}, z_t)$ deduces the action taken between two consecutive latent states. This operator serves multiple roles in modern world-model systems. Pathak et al. (2017) utilized inverse dynamics as a curiosity-driven exploration strategy, refining the representation to capture only the controllable aspects of the environment. By training the encoder to predict actions between consecutive states, the inverse model filters out exogenous visual noise (e.g., moving clouds, flickering backgrounds) that is irrelevant to the agent's decisions. In a broader context, inverse dynamics acts as an additional training signal that prompts z_t to preserve action-relevant characteristics, complementing forward dynamics and reconstruction as a mechanism for learning decision-useful representations (Lesort et al., 2018).

A particularly impactful application of inverse dynamics is retrospective action labeling for large-scale imitation learning. Baker et al. (2022) trained an inverse dynamics model on a small corpus of action-labeled Minecraft gameplay and then applied it to label a vastly larger set of unlabeled internet videos, enabling Video PreTraining (VPT) to learn complex behaviors such as diamond mining from passive observation alone. This pipeline demonstrates that inverse dynamics can bridge the gap between abundant unlabeled video and the action annotations required for behavioral cloning, effectively transforming observation-only data into a usable training signal for policy learning. Inverse dynamics also underpins goal-conditioned policy architectures. Given a current state z_{t-1} and a desired goal state z_g, the inverse model predicts the action that would transition toward the goal, providing a natural interface for hierarchical planning where a high-level planner selects subgoals and a low-level inverse model executes them (Ghosh et al., 2021). Agrawal et al. (2016) demonstrated that a robot can learn intuitive physics by "poking" objects and training an inverse model to predict the poke parameters from observed state changes, illustrating how inverse dynamics can ground physical understanding through interaction.

Many effective world models exclude inverse dynamics entirely, while others use it as a light regularization technique. A practical but under-discussed issue is action-label quality: when actions are inferred retrospectively rather than logged directly, inverse-model errors accumulate precisely at distribution edges where world models most need reliable supervision. Furthermore, the inverse operator assumes a unique or near-

unique action between state pairs, an assumption that breaks down in stochastic environments or when multiple actions lead to the same outcome, limiting its reliability in such settings.

3.3 Discussion

Although this paper focuses on temporal local prediction indexed by time t, the same local-operator view can apply along non-temporal axes such as diffusion steps, refinement steps, or hierarchical update stages. We treat these as edge cases of L1 rather than as a distinct capability level, because the key property is still local transition prediction rather than decision-usable multi-step rollout.

L1 alone does not ensure coherent behavior over long horizons. Challenges such as compounding one-step errors (Janner et al., 2019; Chua et al., 2018), maintaining consistency across numerous steps, and lacking methods for intervention or counterfactual reasoning highlight the need for L2. Likewise, neither L1 nor L2 inherently adapt the model based on new evidence; this capability is the focus of L3. The fundamental distinction between L1 and L2 lies in whether the system is formulated and assessed for *multi-step rollout accuracy and constraint adherence*, rather than solely focusing on one-step prediction precision (Hafner et al., 2025; Ding et al., 2025; Moerland et al., 2023). The limitation of L1 is not that one-step prediction is unimportant, but that local predictive quality alone does not guarantee decision-usable behavior under composition. The practical question is thus when short-horizon operators stop meeting the planner's needs.

4 L2 Simulator: Decision-Usable Multi-Step Simulation

Where L1 answers *"what is the next local state given the current state and action?"*, L2 answers a decision-relevant question: *"if the agent executes a candidate action sequence under task constraints, what future trajectory is likely to unfold?"* This elevation turns one-step operators into a simulator that an agent can query before committing to action, thus providing an *imagination* of the future without requiring real-environment interaction. Model-based planning exploits exactly this capability: by rolling out candidate plans inside a learned model, the agent compares outcomes and selects the most promising course of action (Sutton, 1991; Hafner et al., 2025; Schrittwieser et al., 2020). An important corollary is that any system used to generate synthetic training data for an agent implicitly serves as a world model, since it must produce state transitions realistic enough to support policy improvement (Gu et al., 2025b; Fang et al., 2025). It is worth noting that, decision-usable simulation focuses on plausible dynamics, which holds its distinction between L1 where state changes are arbitrary. For example, a cup passing through a solid table, a car drifting through lane boundaries without consequence, or a social commitment silently vanishing each represent failures to preserve the governing invariants of the target regime.

Table 4 maps the three L2 boundary conditions to concrete instantiations in each governing-law regime. More precisely, an L2 system supports trajectory-level queries of the form

$$\hat{p}(\tau \mid z_0, a_{1:H}, c), \quad \tau = (z_1, \ldots, z_H),$$

where $a_{1:H}$ denotes an action sequence and c denotes optional constraints imposed by the governing-law regime. Intervention-structured rollouts align with the interventional rung of Pearl's causal hierarchy (Section 2.1). What separates L2 from L1 is not one-step predictive quality alone, but **coherent multi-step rollout under the governing laws**. L2 thus stitches per-edge L1 operators into a full trajectory $z_0 \to z_1 \to \cdots \to z_H$ (top block of Figure 7).

4.1 Requirements for Elevation

Composing L1's local operators over multiple steps does not automatically yield a decision-usable simulator: compounding errors, action-insensitive rollouts, and violated domain invariants can each render the resulting trajectories misleading for planning. This echoes the classical frame problem (McCarthy & Hayes, 1969; Shanahan, 1997), in particular, local transition rules alone do not specify which properties should remain invariant under action, while the concern here is operational rather than logical. The interface between a planner and an L2 world model is the *query*: given an action sequence $a_{1:H}$ from state z_0 under constraints

Table 4: **L2 boundary conditions instantiated by governing-law regime.** Each cell specifies what the abstract condition means concretely in that domain.

	Physical World	**Digital World**	**Social World**	**Scientific World**
Coherence	Object persistence and stable contacts over H-step manipulation sequences	DOM/file-system consistency across multi-step UI/code interactions	Commitment and relationship stability across multi-turn dialogue	Causal chain validity across experimental sequences
Sensitivity	Force/placement perturbation alters grasp outcome proportionally	UI failure injection (pop-ups, timeouts) causes appropriate replan	Changing one agent's strategy shifts negotiation outcome	Parameter change produces directionally correct measurement shift
Consistency	No interpenetration, energy conservation, kinematic feasibility	API contract adherence, type constraints, state-machine validity	Norm compliance, belief consistency, reflexive social dynamics	Conservation laws, causal graph consistency, evidence-chain validity

c, the model returns rollouts that the planner uses to compare candidates and select the one maximizing an objective. We treat closed-loop use (planning, acting, or control in interaction with an environment) as an orthogonal deployment property: the level boundary is determined by the depth and reliability of the world-model query, not by whether the system operates in a feedback loop. We use three boundary conditions to mark the elevation from L1 to L2:

1. **Long-horizon coherence:** rollouts remain usable over multiple steps, rather than degrading immediately through compounding error.
2. **Intervention sensitivity:** counterfactual edits, for example, changing actions, premises, or controllable inputs. These induce stable and directionally meaningful trajectory changes.
3. **Constraint consistency:** generated futures respect the governing-law constraints of the target regime, whether physical, digital, social, or scientific.

These are not merely conceptual distinctions; together they induce a practical test for whether a system deserves to be called an L2 simulator. A candidate system should be evaluated not only on one-step prediction quality, but also on whether performance remains decision-usable as rollout horizon increases, whether counterfactual interventions produce coherent and policy-relevant divergence, and whether generated trajectories continue to satisfy regime-specific validity constraints. A model that predicts the next step accurately yet collapses under composition, ignores action edits, or violates domain rules remains better understood as L1 with strong local prediction, rather than as a full L2 simulator.

From L1 to L2. At L1, composing one-step operators yields a trajectory distribution that factorizes as $\hat{p}(\tau \mid z_0, a_{1:H}) = \prod_{t=1}^{H} p_\theta(z_t \mid z_{t-1}, a_t)$, with each step optimized independently; the trajectory is an unregulated byproduct. At L2, the governing-law constraint c couples steps together: conceptually,

$$\hat{p}(\tau \mid z_0, a_{1:H}, c) \propto \prod_{t=1}^{H} p_\theta(z_t \mid z_{t-1}, a_t)\, \phi_c(\tau),$$

where $\phi_c(\tau)$ is a governing-law compatibility term over the full rollout. The hard-indicator case $\mathbf{1}[c(\tau)]$ is a special case of $\phi_c(\tau)$ when violations are treated as strictly inadmissible. Because $\phi_c(\tau)$ depends on the entire trajectory, the L2 distribution does not factorize into independent per-step terms.

Each requirement maps to a diagnostic signal and a mitigation strategy. *Long-horizon fidelity* is diagnosed by a success cliff at a specific horizon H. The primary mitigation of this is task segmentation with frequent replanning. *Action controllability* is diagnosed by action insensitivity across rollouts (changing a_t produces no meaningful trajectory change), where mitigation requires explicit action-consistency evaluation. *Constraint consistency* is measured by the constraint-violation rate. In such cases, mitigations include hard constraint layers and verification gates. A fourth property, *calibration*, requires that confidence aligns with actual accuracy under distribution shift; in such cases, overconfident wrong predictions signal failure, and distribution-shift detection is the main remedy.

Residual frame-problem manifestations. Modern neural world models sidestep the classical frame problem's representational burden by learning implicitly from data what persists and what changes (Goodfellow et al., 2016; Hafner et al., 2025), enabling scalable model-based RL (Hafner et al., 2020; Schrittwieser et al., 2020; Moerland et al., 2023) and video prediction (Babaeizadeh et al., 2018; Brooks et al., 2024) without explicit frame axioms. Yet the problem resurfaces at rollout time: context-window limits and hallucination cause models to lose track of relevant past information, violating long-horizon coherence, while rare preconditions under-represented in training data undermine constraint consistency (Ding et al., 2025; Shanahan, 1997). These failure modes motivate the techniques surveyed below.

4.2 Applications

In this section, we categorize L2 systems into four governing-law regimes. Tables 5 and 6 provide anchor systems for cross-regime comparison and summarize how each domain instantiates the boundary conditions.

4.2.1 Laws of the Physical World

In the physical domain, L2 models should respect geometry, kinematics, and conservation laws. The governing constraints are contact, reachability, stability, and energy conservation; violations of any of these will mislead a planner into proposing actions that fail catastrophically in real execution.

Physics simulation. *Rigid-body control simulators.* Classical physics simulators remain the foundation layer for executable transition validity in embodied world modeling. MuJoCo provides articulated rigid-body dynamics and contact-rich control, with dm_control packaging these capabilities into a standardized continuous-control suite (Todorov et al., 2012; Tassa et al., 2020). Brax pushes differentiable rigid-body simulation toward accelerator-scale throughput (Freeman et al., 2021), while Isaac Gym and Isaac Lab emphasize massive GPU-parallel robotics simulation (Makoviychuk et al., 2021; Mittal et al., 2025).

Scalable and general-purpose simulation platforms. Genesis positions itself as a generative and universal physics engine (Genesis Authors, 2024), reflecting the broader trend toward higher-throughput simulators that can jointly support both control and large-scale synthetic-data generation.

Interaction-centric embodied simulators. At the graphics-and-robotics interface, SAPIEN provides part-aware, interaction-centric simulation, and ManiSkill3 scales GPU-parallel rendering for generalizable embodied AI (Xiang et al., 2020; Tao et al., 2024). These systems are not learned simulators; they are explicit law executors whose value lies in precise contact handling, articulated constraints, and reproducible rollouts.

Video generation models. *Appearance-first long-horizon video generation.* A scalable route to physical-world simulation is the *video interface*: given current observations and optional actions, the model returns imagined future frames. This line begins with appearance-first rollout, where systems such as Sora, Lumiere, and VideoPoet demonstrate coherent visual dynamics over extended horizons (Brooks et al., 2024; Bar-Tal et al., 2024; Kondratyuk et al., 2024), with geometry-aware structure increasingly emerging beyond pixel-level realism (Li et al., 2024b). FramePack (Zhang et al., 2025c) and Self-Forcing (Huang et al., 2025d) reduce long-horizon drift through frame-context packing.

Action-conditioned and interactive video worlds. A second direction moves from passive continuation toward intervention-aware generation. Genie learns latent action spaces from unlabeled Internet video (Bruce et al., 2024), while GAIA-1 conditions future generation on explicit control signals for counterfactual evaluation (Hu et al., 2023). More recent systems push this line toward real-time, long-horizon, and streaming interaction: Oasis explores open-ended interactive generation in a unified transformer world (Decart et al., 2024); WorldPlay emphasizes long-term geometric consistency for real-time interactive world modeling (Sun et al., 2025b); Matrix-Game 3.0 extends interactive generation to streaming settings with explicit long-horizon memory (Wang et al., 2026c); Yume-1.5 studies text-controlled interactive world generation (Mao et al., 2025); and LongLive targets real-time interactive long video generation (Yang et al., 2025b). Taken together, these systems mark a shift from passive video prediction toward controllable, intervention-aware, and temporally persistent video worlds.

Table 5: **Representative L2 anchor systems:** Physical and Digital Worlds. Columns indicate long-horizon coherence (**LH**), intervention sensitivity (**IS**), and constraint consistency (**CC**). The table is a compact comparison set, not an exhaustive inventory of all systems discussed in the prose.

Method	Links	LH	IS	CC	Architecture
Physical World					
MuZero (Schrittwieser et al., 2020)	Paper —	✔	✔	✔	MLP dynamics + MCTS
Plan2Explore (Sekar et al., 2020)	Paper Code	✔	✔	✗	Dreamer + self-supervised exploration
PathDreamer (Koh et al., 2021)	Paper Code	✔	✗	✗	Autoregressive visual VLN
DreamerPro (Deng et al., 2022)	Paper Code	✔	✔	✗	RSSM + prototypical representations
DreamingV2 (Okada & Taniguchi, 2022)	Paper —	✔	✔	✗	DreamerV2 + reconstruction-free
Diffuser (Janner et al., 2022)	Paper Code	✔	✗	✗	Diffusion trajectory planning
DreamerV3 (Hafner et al., 2025)	Paper Code	✔	✔	✔	RSSM + symlog loss
DayDreamer (Wu et al., 2023a)	Paper Code	✔	✔	✔	RSSM real-world robots
GAIA-1 (Hu et al., 2023)	Paper —	✔	✔	✗	Transformer video generation
DIAMOND (Alonso et al., 2024)	Paper Code	✔	✔	✔	U-Net diffusion
Sora (Brooks et al., 2024)	Paper —	✔	✗	✗	DiT video diffusion
Genie (Bruce et al., 2024)	Paper —	✔	✔	✗	ST-transformer + VQ actions
iVideoGPT (Wu et al., 2024b)	Paper Code	✔	✔	✗	Transformer + VQ-VAE
OccWorld (Zheng et al., 2024)	Paper Code	✔	✔	✔	GPT 3D occupancy prediction
Vista (Gao et al., 2024)	Paper Code	✔	✔	✗	Diffusion driving generation
DriveDreamer (Wang et al., 2024e)	Paper Code	✔	✔	✗	Diffusion AD generation
Copilot4D (Zhang et al., 2024)	Paper —	✔	✔	✔	VQ-VAE + point diffusion
LWM (Liu et al., 2025)	Paper Code	✔	✗	✗	RingAttention long-context LLM
DreMa (Barcellona et al., 2024)	Paper Code	✔	✔	✗	Compositional 3DGS twins
Cosmos (Agarwal et al., 2025)	Paper Code	✔	✔	✗	Autoregressive + diffusion hybrid
Aether (Zhu et al., 2025)	Paper Code	✔	✔	✔	CogVideoX geometry fine-tune
PIN-WM (Li et al., 2025d)	Paper Code	✔	✔	✔	Differentiable rigid-body + 3DGS
Yume (Mao et al., 2025)	Paper Code	✔	✔	✗	Video diffusion world generation
GAIA-2 (Russell et al., 2025)	Paper —	✔	✔	✗	Latent diffusion multi-view AD
RoboScape (Shang et al., 2025)	Paper Code	✔	✔	✗	Physics-informed robot video
BridgeV2W (Chen et al., 2026)	Paper —	✔	✔	✗	Action-conditioned embodied video
HWM (Zhang et al., 2026c)	Paper Code	✔	✔	✗	Hierarchical latent + MCTS
Digital World					
GameGAN (Kim et al., 2020)	Paper Code	✔	✔	✗	GAN neural game engine
WebDreamer (Gu et al., 2025b)	Paper Code	✔	✔	✗	LLM web state simulation
CodeWM (Dainese et al., 2024)	Paper Code	✔	✔	✔	LLM + MCTS code generation
WorldCoder (Tang et al., 2024)	Paper Code	✔	✔	✔	LLM incremental code synthesis
GameGen-X (Che et al., 2024)	Paper Code	✔	✔	✗	Diffusion transformer, InstructNet control
GameNGen (Valevski et al., 2025)	Paper —	✔	✔	✗	U-Net diffusion
WMA (Chae et al., 2025)	Paper Code	✔	✔	✗	LLM web transition prediction
WebSynthesis (Gao et al., 2025)	Paper Code	✔	✔	✗	LLM + MCTS planning
NeuralOS (Rivard et al., 2025)	Paper Code	✔	✔	✗	RNN + pixel diffusion
GameFactory (Yu et al., 2025)	Paper Code	✔	✔	✗	Action-controlled video generation
GameCraft (Li et al., 2025b)	Paper Code	✔	✔	✗	Diffusion game video generation
MobileDreamer (Cao et al., 2026)	Paper —	✔	✔	✗	LLM GUI sketch prediction
Word2World (Li et al., 2025f)	Paper Code	✔	✔	✗	LLM text-based WM
Code2World (Zheng et al., 2026)	Paper Code	✔	✔	✔	VLM code rendering
gWorld (Koh et al., 2026)	Paper Code	✔	✔	✔	VLM code rendering
WebWorld (Xiao et al., 2026)	Paper —	✔	✔	✗	Fine-tuned VLM web simulator
RWML (Yu et al., 2026)	Paper —	✔	✔	✗	LLM + RL sim-to-real

Decision-oriented video world models. In model-based RL, SimPLe (Kaiser et al., 2020) and DIAMOND (Alonso et al., 2024) make the decision-theoretic role of video world models explicit. In robotics, DreamZero (Ye et al., 2026b) and DreamDojo (Gao et al., 2026) demonstrate zero-shot and generalist policy learning via video world models, while FutureVLA (Xu et al., 2026) couples visuomotor prediction directly with Vision-Language-Action policies to unify perception and control.

Table 6: **Representative L2 anchor systems (continued):** Social and Scientific Worlds. Columns indicate long-horizon coherence (**LH**), intervention sensitivity (**IS**), and constraint consistency (**CC**). Links are provided where available; the table is a compact comparison set, not an exhaustive inventory of all systems discussed in the prose.

Method	Links	LH	IS	CC	Architecture
Social World					
Deal or No Deal (Lewis et al., 2017)	Paper Code	✔	✔	✔	RNN + RL self-play
Social Simulacra (Park et al., 2022)	Paper —	✔	✔	✗	GPT prompt-chain community simulation
CICERO (Bakhtin et al., 2022)	Paper Code	✔	✔	✔	LLM + strategic planning
Generative Agents (Park et al., 2023)	Paper Code	✔	✔	✔	LLM reflective memory
Sotopia (Zhou et al., 2024)	Paper Code	✔	✔	✔	LLM social evaluation
AvalonBench (Light et al., 2023)	Paper Code	✔	✔	✔	LLM deductive reasoning
Werewolf (Xu et al., 2023)	Paper Code	✔	✔	✔	LLM + RL strategic policy
ProjectSid (AL et al., 2024)	Paper Code	✔	✔	✔	LLM multi-agent civilization simulation
OASIS (Yang et al., 2024c)	Paper Code	✔	✔	✔	LLM social simulation
MASim (Zhang et al., 2025f)	Paper —	✔	✔	✗	Multilingual agent simulation
SWM-AP (Zhang et al., 2025e)	Paper —	✔	✔	✗	Social WM mechanism design
AIvilization (Fan et al., 2026)	Paper —	✔	✔	✔	Sandbox economy simulation
PolicySim (Huang et al., 2026a)	Paper Code	✔	✔	✗	LLM platform policy sandbox
Scientific World					
GNS (Sanchez-Gonzalez et al., 2020)	Paper Code	✔	✗	✔	GNN message passing
ChemBO (Korovina et al., 2020)	Paper Code	✔	✗	✔	GP + synthesis-graph BO
P3BO (Angermueller et al., 2020)	Paper —	✔	✔	✗	Adaptive population-based optimization
FNO (Li et al., 2021)	Paper Code	✔	✗	✔	Fourier neural operator
Pangu-Weather (Bi et al., 2023)	Paper Code	✔	✗	✔	3D Earth transformer
ClimaX (Nguyen et al., 2023)	Paper Code	✔	✗	✔	ViT climate foundation
GraphCast (Lam et al., 2023)	Paper Code	✔	✗	✔	GNN autoregressive weather
GenCast (Price et al., 2024)	Paper Code	✔	✗	✔	Spherical ensemble diffusion
NeuralGCM (Kochkov et al., 2024)	Paper Code	✔	✗	✔	Hybrid physics–NN core
BAX (Chitturi et al., 2024)	Paper Code	✔	✔	✔	GP + user-directed acquisition
Aurora (Bodnar et al., 2025)	Paper Code	✔	✗	✔	3D Swin weather foundation
Lingshu-Cell (Zhang et al., 2026a)	Paper —	✔	✗	✔	Masked diffusion cellular WM

Evaluation and limitations. Within our L2 framing, however, visual plausibility does not equal decision-usability. Intervention sensitivity remains fragile, long-horizon coherence is easily overstated when judged by perceptual quality alone (Guo et al., 2025c), and constraint consistency is difficult to verify from rendered frames. Standard metrics such as FVD (Unterthiner et al., 2018) capture distributional realism; VBench-style suites (Huang et al., 2024b; 2025f) better decompose controllability; VBench-2.0 (Zheng et al., 2025a) extends evaluation to physics consistency and commonsense reasoning; and VChain (Huang et al., 2025e) introduces visual chain-of-thought for causal coherence. Video interfaces are the most scalable observation-layer entry point, but planner-critical structure remains implicit in pixels.

Robotics and sim-to-real transfer. *World models transferred to real robots.* DayDreamer (Wu et al., 2023a) showed that Dreamer-family world models can transfer from simulation to physical robots while handling sensor noise, contact dynamics, and actuation delays. DreamZero (Ye et al., 2026b) achieves zero-shot policy learning via world action models that predict both next states and actions, and FutureVLA (Xu et al., 2026) embeds visuomotor prediction within Vision-Language-Action models to improve action grounding.

Physics-grounded bridges for sim-to-real robustness. PIN-WM (Li et al., 2025d) integrates differentiable physics with learned visual world modeling, creating "digital cousins" via physics-aware randomization.

Spatial Reasoning. Wang et al. (2025e) shows that agents can construct spatial mental models from limited views; Zhang et al. (2026b) asks whether agents can *actively* explore to construct spatial beliefs rather than reasoning passively over given views.

Representation requirement. Across these systems, the key question is not whether richer representations are possible, but what is the weakest representation that still preserves planner-critical structure, such as object persistence, free space, contact onset, support relations, and action-conditioned change over useful horizons.

4.2.2 Laws of the Digital World

The Laws of the Digital World govern transitions in systems defined by formal specifications, from finite automata (UI state machines) and context-free grammars (structured data formats) to Turing-complete programs (general software). Unlike the Laws of the Physical World or the Laws of the Social World, these constraints are *explicitly specified and mechanically verifiable*: a transition either satisfies the program's semantics or it does not. Because software transitions approximate deterministic state machines and failures are loggable (error codes, popups, permission denials, timeouts), the core challenge for a Simulator in code worlds is *structured state prediction* (DOM trees, program state, game state) rather than visual fidelity.

Coding agents. An emerging paradigm represents world models as executable programs rather than neural networks. CodeWM (Dainese et al., 2024) uses LLMs guided by Monte Carlo Tree Search to generate Python programs that serve as explicit, interpretable world models for reinforcement learning across 18 environments. WorldCoder (Tang et al., 2024) takes a complementary approach, with a model-based LLM agent building a Python world model through environment interaction, gaining sample efficiency and transferring across environments by editing its code. WKM (Qiao et al., 2024) provides both global task knowledge and dynamic state knowledge to guide LLM agent planning, while CWM (Copet et al., 2025), a 32B open-weights LLM released to advance research on code generation with world models, achieves 65.8% on SWE-bench Verified with test-time scaling. A conceptually distinct variant pushes further: rather than using an LLM to *generate* a code world model, the world model *is* a running software system. Web World Models (Feng et al., 2025a) implement world state as ordinary web code with latent state exposed as typed web interfaces, delegating logical consistency to deterministic execution of the web stack while LLMs generate context and high-level decisions. In a similar spirit, SWE-World (Sun et al., 2026) replaces containerized execution with a learned surrogate that predicts intermediate execution outcomes and final test feedback, preserving the standard agent-environment loop without running real environments and lifting a 32B coder from 6.2% to 52.0% on SWE-bench Verified. These code-based approaches yield interpretable, composable, and verifiable world models that neural dynamics can only approximate.

Web agents. Web agents usually browse websites; therefore, modeling and simulating state transitions within a website is crucial for building effective web world models. WebDreamer (Gu et al., 2025b) introduced the idea of using an LLM as an implicit world model of the internet, but subsequent work showed that off-the-shelf LLMs are insufficient: dedicated training with transition-focused abstraction is needed (Chae et al., 2025). A growing body of work addresses the co-evolution of agent and world model. WebEvolver (Fang et al., 2025) tightly couples the two in a mutual improvement loop, while DreamGym (Chen et al., 2025f) builds experience models with chain-of-thought reasoning, achieving over 30% improvement on WebArena. At larger scale, WebSynthesis (Gao et al., 2025) combines world models with MCTS-based planning using entirely synthetic data, and WebWorld (Xiao et al., 2026) trains an open-web simulator on over one million trajectories supporting 30+ step simulation. Early Experience (Zhang et al., 2025b) proposes a paradigm that leverages agent-generated trajectories as reward-free supervision, enabling implicit world modeling and self-reflection while bridging imitation learning and reinforcement learning. AUI (Lin et al., 2025a) takes a different approach, employing a Coder to optimize websites by leveraging feedback from a Computer-Use Agent in an iterative collaboration loop. Orthogonal design choices concern what data to learn from and when to query the model: generating training trajectories from tool specifications alone (Simia; Li et al. 2025g), adding a metacognitive layer that decides *whether* to consult the world model at each step (WAC; Shen et al. 2026), and collecting on-policy agent trajectories to cover the out-of-distribution behaviors that statically harvested corpora tend to miss, so the learned simulator keeps pace with the agent's shifting behavior over training.

GUI agents. GUI agents (Qin et al., 2025; Lin et al., 2025b; Xu et al., 2024) typically execute actions in real environments. However, in scenarios where actions may be dangerous or lead to undesired outcomes,

it is beneficial to estimate them beforehand. A GUI world model can simulate and evaluate these actions, thereby providing a more reliable assessment. Therefore, MobileDreamer (Cao et al., 2026) transforms GUI images into task-related sketches for structured state prediction, while MobileWorldBench (Li et al., 2025c) provides systematic evaluation with 1.4 million (state, action, future state) triplets. Complementary to explicit GUI world models, UI-AGILE shows that effective reinforcement learning and precise inference-time grounding remain equally important for strong downstream GUI-agent performance (Lian et al., 2025). A central design question is the output representation: ViMo (Luo et al., 2025) generates future observations as images using symbolic text representation, while gWorld (Koh et al., 2026) generates renderable web code as the predicted next state, suggesting that generating the code that renders the GUI can be more faithful than generating pixels directly. At the OS level, NeuralOS (Rivard et al., 2025) simulates desktop GUIs by predicting screen frames from user inputs, while CUWM (Guan et al., 2026) targets desktop software where persistent document state must be preserved across long-horizon workflows. Code2World (Zheng et al., 2026) further extends this line by treating code as a renderable world, where generated programs directly produce visual states (e.g., HTML) upon execution. This enables modeling environment dynamics as executable code generation, tightly coupling perception, action, and state transition in interactive domains such as GUIs.

Game agents. Game agents must act over long horizons in open-ended environments whose rules and win conditions are formally checkable, so a world model can serve both as a reliable surrogate environment for training and as a learned predictor the agent consults to roll out and evaluate candidate moves before committing to them. The most explicit realization treats the world model as a verifiable rollout engine: Code World Models for General Game Playing (Lehrach et al., 2025) prompts an LLM to translate natural-language rules and observed trajectories into executable Python implementing state transitions, legal-move enumeration, and termination checks, and a planner such as MCTS then queries this program to search only over genuinely legal continuations, matching or surpassing Gemini 2.5 Pro on nine of ten games. MineWorld (Guo et al., 2025a) instead learns environment dynamics directly, training a visual-action autoregressive Transformer over interleaved image and action tokens whose parallel decoding sustains real-time interaction at four to seven frames per second, introducing action-following metrics beyond visual quality and significantly outperforming state-of-the-art open-source diffusion world models. A complementary line internalizes the world model inside the policy: Game-TARS (Wang et al., 2025g) pretrains a generalist multimodal game agent that roughly doubles the success rate of the prior state of the art on open-world Minecraft. RAGEN (Wang et al., 2025f) frames agent learning as multi-turn reinforcement learning over interactive tasks, establishing that it works for self-evolving LLM agents while characterizing the training-stability failure modes that arise. VAGEN (Wang et al., 2025d) extends this paradigm to VLM agents and shows that explicitly reinforcing world-model reasoning, such as state estimation and transition prediction within the reasoning trace, yields further gains. RAGEN-2 (Wang et al., 2026b) continues in this direction by diagnosing a reasoning-collapse failure mode in agentic RL and proposing a fix. Across these systems the recurring theme is structured prediction over program and game state, where transitions are explicitly specified and, by construction, mechanically verifiable rather than merely visually plausible.

Tool-calling agents. Tool-calling agents invoke external functions such as APIs, databases, and Model Context Protocol (MCP) servers, where each call mutates hidden environment state that the agent cannot observe directly, so a world model is most useful as a verifiable rollout engine that the agent queries to predict the outcome of a call before committing to it, enabling look-ahead planning, self-verification, and cheap training environments. Because tool transitions are explicitly specified by function signatures and database schemas and their failures are mechanically loggable as error codes or rejected parameters, the simulator's task here is structured prediction of program and database state rather than visual fidelity. GTM (Ren et al., 2025) instantiates this as a universal tool simulator that, from a prompt-level configuration, generates outputs faithfully mimicking real execution, with training data synthesized across more than 20,000 tools, and in agent reinforcement learning it runs at significantly faster simulation speed than real tools while preserving comparable output quality. AWM (Wang et al., 2026a) instead grounds the simulator in executable code, generating code-driven, database-backed environments whose state transitions are more reliable and consistent than purely LLM-simulated feedback (e.g., Simia (Li et al., 2025g)), and whose accessible database states supply reliable reward functions for large-scale RL of multi-turn tool-use agents. MCP-Cosmos (Ganapavarapu & Patel, 2026) pushes the simulator into the MCP ecosystem under a "Bring

Your Own World Model" strategy, letting agents simulate state transitions and refine plans in latent space before execution and improving tool success rate and parameter accuracy in experiments spanning more than twenty MCP-Bench tasks. At a finer granularity, DyMo (Guo et al., 2025b) equips the model with state prediction alongside function calling during post-training, so it predicts the future state of its actions through an internal environment model; on the Berkeley Function Calling Leaderboard V2 this raises success rates and significantly reduces hallucinations, and when wrapped in self-verification sampling it lets the model refuse unreliable outputs. Yet a foresight study (Qian et al., 2026) cautions that simply exposing such simulators is not enough, since agents rarely invoke simulation, frequently misuse predicted rollouts, and can even degrade in performance, leaving when to simulate and how to interpret and integrate foresight into downstream reasoning as the central open challenges for reliable anticipatory tool use.

4.2.3 Laws of the Social World

Societal world models extend L2 to human interaction, where governing laws are beliefs, desires, intentions, norms, and institutions rather than physics. Social worlds exhibit three distinctive properties, in particular, *opacity* (agents cannot directly observe each other's mental states), *reflexivity* (beliefs about social state create feedback loops), and *normativity* (transitions are governed partly by shared norms). Such traits make the transition function partially constituted by collective agreement rather than natural law (Zheng et al., 2025b). A usable social simulator separates surface language from underlying social state: dialogue can vary, but core states (goals, beliefs, relations, norms) must remain consistent and yield interpretable transitions, as formalized by the Rational Speech Acts framework (Goodman & Frank, 2016; Degen, 2023). Concretely, a social compatibility term $\phi_c(\tau)$ can encode commitment consistency: if agent i promises action b at time t, later states receive low compatibility when i violates b without explanation, renegotiation, or sanction. Similar terms can score norm compliance, role consistency, or belief-state coherence over the trajectory.

Theory of mind as social state. The computational foundation was laid by Bayesian ToM (BToM), which formalizes mental state inference as probabilistic inverse planning over rational agents (Baker et al., 2011). Neural approaches began with ToMnet (Rabinowitz et al., 2018), whose character, mental state, and prediction networks jointly infer traits and beliefs, and recent work such as LaBToM (Ying et al., 2025) bridges Bayesian inverse planning with formal epistemic language. However, current models lack robust mental state reasoning: FANToM (Kim et al., 2023) reveals "illusory ToM" across all state-of-the-art LLMs, and ExploreToM (Sclar et al., 2024) achieves accuracy as low as 9% for GPT-4o (Chen et al., 2025c). A complementary challenge is the *dual-structure* problem: a social agent must simultaneously model others' mental states (theory of mind) and maintain its own persistent internal state across long interactions, in particular, goals, persona, memory, and knowledge. Cognitive Architectures for Language Agents (CoALA) (Sumers et al., 2024) formalizes this dual structure as separate memory and action spaces that must remain mutually consistent, and provides a principled framework for understanding how current LLM agents do and do not achieve stable self-representation.

Strategic interaction. CICERO (Bakhtin et al., 2022) integrates a language model with piKL planning for Diplomacy, jointly optimizing game actions and dialogue while modeling second-order beliefs, achieving more than 2× the average human score. Deal or No Deal (Lewis et al., 2017) pioneered dialogue rollouts for forward simulation of negotiation dynamics. Werewolf and Avalon games serve as concentrated testbeds for deception, trust, and belief manipulation (Xu et al., 2023; Light et al., 2023), revealing that deceivers consistently prevail by exploiting cognitive limitations.

Sandbox simulation. Generative Agents demonstrated emergent social dynamics: a 25-agent simulation (Park et al., 2023) used memory-based state tracking and periodic reflection, while Sotopia (Zhou et al., 2024) formalized social simulation evaluation across seven dimensions. Scale has increased dramatically: Project Sid (AL et al., 2024) deployed 1,000 agents exhibiting emergent specialization and governance, and OASIS (Yang et al., 2024c) scaled to one million agents reproducing information spreading and group polarization. At the individual level, Argyle et al. (2023) demonstrate "silicon sampling", which conditions LLMs on specific demographic profiles to simulate survey responses from targeted subpopulations and shows strong alignment with American National Election Studies data, opening a path toward individual social

world modeling. Generative Social Choice (Fish et al., 2024) extends this to democratic aggregation, using LLMs to generate representative statements from diverse synthetic participants and enabling deliberation.

Challenges and design principles. Social simulation remains premature: LLMs degrade sharply beyond second-order belief reasoning (Wu et al., 2023b), agents suffer from role drift and goal forgetting (Park et al., 2023; Zhou et al., 2024), and formal commitment tracking (Telang et al., 2021) remains unintegrated into any LLM architecture. A practical design pattern separates a compact social state representation (commitments, constraints, relations), a dialogue generator, and a transition updater that enforces consistency and makes state transitions loggable and replayable. Flexible persona generation is essential for populating social simulators with diverse, controllable agents; PersonaGym (Samuel et al., 2025) provides a benchmark for evaluating how faithfully LLMs enact specified personas across complex social tasks, revealing systematic failures in maintaining persona consistency under adversarial probing. For personalization at the individual level, LaMP (Salemi et al., 2024) introduces a benchmark of seven tasks requiring LLMs to generate outputs consistent with a specific user's history, and shows that retrieval-augmented approaches significantly close the gap.

4.2.4 Laws of the Scientific World

In AI for Science, the transition from L1 to L2 shifts the focus from modeling local states or structures to simulating dynamics over multiple steps. These dynamics arise along two axes. The first concerns the temporal evolution of a system, where the model predicts how a natural system unfolds over time under given conditions or interventions. The second concerns the scientific research itself, where the model simulates sequences of hypotheses, experiments, and outcomes to support reasoning and action. These two forms define the corresponding forms of simulation in scientific world models: forward simulation of system dynamics, and decision simulation based on surrogate evaluation of candidate experiments.

Forward simulation. World models approximate the evolution of scientific systems by replacing expensive numerical solvers with learned transition operators. GNS (Sanchez-Gonzalez et al., 2020) showed that message passing on particle graphs can simulate fluids, rigid bodies, and deformable materials with generalizable dynamics. The Fourier Neural Operator (Li et al., 2021) established resolution-invariant operator learning via spectral convolutions, achieving 1000× speedup over traditional solvers and underpinning subsequent weather and fluid surrogates. At planetary scale, Pangu-Weather (Bi et al., 2023) and GraphCast (Lam et al., 2023) outperform the ECMWF operational system on 90% of verification targets. GenCast (Price et al., 2024) extends these to probabilistic forecasting via a diffusion architecture, outperforming the ensemble system on 97.2% of targets. NeuralGCM (Kochkov et al., 2024) integrates learned parameterizations within a differentiable general circulation model, producing emergent phenomena such as tropical cyclones and illustrating the value of coupling mechanistic structure with learned components. Aurora (Bodnar et al., 2025) further scales this paradigm to a foundation model of the Earth system, achieving strong performance across multiple forecasting tasks at substantially reduced computational cost. In molecular science, neural network potentials pioneered by Behler & Parrinello (2007) enabled orders-of-magnitude speedup over density functional theory for molecular dynamics, establishing the foundation for all subsequent ML fields Deng et al. (2023); Chen et al. (2025b; 2023); Huang et al. (2025c).

Decision simulation. World models reduce the cost of scientific discovery by simulating the experimental decision loop in-silico. Representative systems span molecular design (ChemBO; Korovina et al. 2020), biological sequence optimization with population-based model ensembles and meta-level search reallocation (P3BO; Angermueller et al. 2020), and materials discovery guided by user-defined algorithmic objectives (BAX; Chitturi et al. 2024). Across these systems, the model simulates not only individual outcomes but the sequential process of experiment selection, maintaining and updating beliefs over candidates while identifying inconsistencies during optimization. However, these capabilities remain confined to a fixed data regime: the model cannot actively design and execute experiments to acquire *new information* that challenges its current assumptions. As a result, while such systems can correct optimization errors, they cannot resolve uncertainty arising from incomplete knowledge, leading to accumulated bias over long horizons. L3 world models (Section 5) overcome this by actively gathering evidence to revise the model.

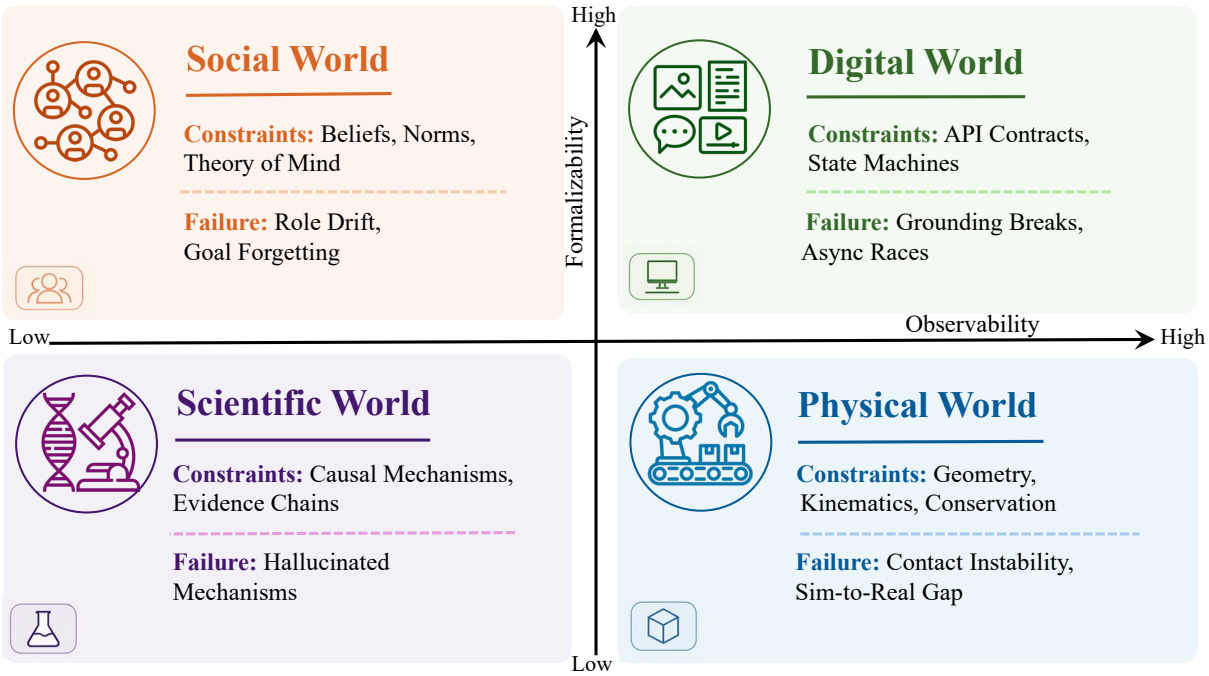

Figure 8: **Diagnostic map of the four governing-law regimes.** The axes are schematic rather than metric: the horizontal axis reflects how formally specifiable and mechanically verifiable the transition rules are, while the vertical axis reflects how directly the relevant state and constraints are observable. The purpose of the figure is comparative rather than classificatory: it highlights why different regimes demand different forms of rollout validation even when all are instances of L2 simulation. Real systems are often mixed-regime and may sit between regions rather than inside a single box.

4.2.5 Cross-Domain Analysis

Table 7: Cross-domain comparison of L2 simulators. Each governing-law regime imposes different constraints, failure modes, and evaluation priorities.

Domain	Governing Laws	State Type	Common Failures	Evaluation Focus
Physical	Geometry, kinematics	Continuous	Contact instability, drift	Stability; failure clustering
Social	Beliefs, norms, ToM	Goals, relations	Role drift, goal forgetting	Counterfactual sensitivity
Digital	API contracts, UIs	DOM, permissions	Grounding breaks, races	Error-branch coverage
Scientific	Mechanisms, evidence	Hypotheses	Hallucinated mechanisms	Evidence-chain repair

Figure 8 positions the four regimes along two diagnostic axes: formalizability and observability of the governing constraints. Across all four regimes, a recurring pattern emerges: a good Simulator does not have to look more like the world; it must look more like the constraints. Physics uses geometry/contact constraints; software uses state machines and structured feedback channels; social worlds use role/norm consistency; science uses evidence chains and falsifiability. Making constraints explicit (loggable, replayable, regressable) often improves long-horizon stability more than increasing perceptual fidelity. Table 7 summarizes the governing laws, state types, common failure modes, and evaluation focus for each regime.

Cross-regime systems. Many real-world deployments do not fall neatly into a single governing-law regime; instead, they require an L2 simulator to maintain coherent rollouts across *multiple* constraint families simultaneously. When regimes interact, a violation in one domain can cascade into another: a physically

implausible vehicle maneuver may render a social-intent prediction meaningless, or a software bug may invalidate an otherwise sound experimental plan. Designing and evaluating cross-regime systems therefore demands joint constraint satisfaction rather than per-regime evaluation in isolation.

- **Autonomous driving:** physical (vehicle dynamics, contact mechanics) + social (pedestrian intent prediction, traffic norm compliance) (Hu et al., 2023; Wang et al., 2024e).
- **Minecraft agents (Voyager):** physical (3D navigation, combat dynamics) + digital (crafting recipes, inventory management, game-state logic) (Wang et al., 2024a).
- **Diplomacy (CICERO):** social (negotiation, trust modeling, alliance formation) + digital (game-state management, rule enforcement) (Bakhtin et al., 2022).
- **Autonomous laboratories (A-Lab):** scientific (experiment design, hypothesis evaluation) + physical (sample manipulation, instrument constraints) (Szymanski et al., 2023; Huang et al., 2025b; Huang et al.; Fei et al., 2026; Boiko et al., 2023).

4.3 Failure Modes

Across all four domains, five recurring failure modes constrain L2 systems:

1. **Compounding error.** Small per-step deviations are amplified over time, pushing imagined trajectories into branches increasingly unrelated to reality. The most effective mitigations are not making one-step predictions look better, but shortening effective planning windows (decomposing long tasks into verifiable short segments and replanning frequently with real feedback), using multi-timescale structure (Shaj et al., 2023), and baking evidence-gathering actions into policies.
2. **State aliasing and drift.** In complex environments, distinct real states can look highly similar (two UI pages, slightly different kitchen layouts, or a one-word change in social tone). When representations collapse these states, agents can take irreversible wrong actions. Effective practices include explicit verification at key nodes, memory and retrieval augmentation, and explicit failure attribution labels (Xie et al., 2024; Yang et al., 2025a; Nasiriany et al., 2024).
3. **Controllability failure.** A visually rich model that is weakly action-conditioned is less useful for planning than a rough model that responds to actions. When the model is action-insensitive, comparing "do A vs. do B" becomes meaningless (Wu et al., 2024b; Liu et al., 2025; Brooks et al., 2024; Ball et al., 2025).
4. **Exploitability and simulator escape.** If a simulator or evaluation harness has loopholes, search/-planning will exploit them systematically. This is especially common in software worlds and automated evaluation (Xie et al., 2024; Yang et al., 2025a; Zheng et al., 2025b).
5. **Calibration failure under distribution shift.** Environment changes (UI versions, layouts, accents, object properties) often trigger overconfident wrong predictions. In practice, confident but wrong should be treated as a strong signal for evolution (Xie et al., 2024; Yang et al., 2025a; Nasiriany et al., 2024).

These failures are not merely model shortcomings; they are system-level pathologies produced by the interaction between representation, rollout horizon, control procedure, and evidence quality. The takeaway is that improving average-case predictions is insufficient unless systems can (i) localize failures via evidence and (ii) change behavior under shift and exploit pressure.

This constraints-first lens is also a useful guide for choosing what to log and what to regress-test. If the core constraint is violated (e.g., impossible action succeeds, or a structured feedback channel disappears), the agent will learn the wrong lessons. Conversely, when constraints are explicit and stable, even simple agents can improve reliably via Evolver-style asset distillation (Xie et al., 2024; Yang et al., 2025a; Nasiriany et al., 2024; Zheng et al., 2025b; Ghugare et al., 2025).

5 L3 Evolver: Evidence-Driven Model Revision

Scientific discovery is naturally organized as a loop: a researcher *designs* experiments, *executes* them, *observes* outcomes, and *reflects* to guide the next step. Recent systems realize one or more components of this process, but many operate without a fully autonomous loop. Decision simulators (Section 4.2.4) operate within the *design* step: they simulate experimental outcomes and update beliefs, but all updates occur within a fixed information regime without active data collection. At the other end, RL-based self-reflection systems such as

VL-Rethinker (Wang et al., 2025c) realize the *reflect* step through explicit verify-and-rethink behavior, but do not maintain a persistent world-model stack. Many automated research pipelines (Yang et al., 2024d; Li et al., 2024a) chain *design* and *reflect* (hypothesis generation and literature synthesis) but lack the *execute* and *observe* components that would close the full loop.

The key distinction of L3 lies in how new information is acquired and used. Rather than passively fitting incoming data or exploiting a fixed model for planning, an L3 system actively designs interventions to reduce uncertainty in its own world model. In the unified view of Figure 7, this revision is the vertical *reflect* arrow that connects the top block (model $\mathcal{M}_t$ operating over environment $\mathcal{X}$) to the bottom block (revised model $\mathcal{M}_{t+1}$ operating over an effective environment $\mathcal{X}'$): each iteration of the loop modifies the latent graph that L2 rolls out on. Each iteration of the loop targets discrepancies between prediction and observation, using them to refine parameters, extend model structure, or revise underlying assumptions. In this sense, L3 is driven not by reward maximization alone, but by the systematic reduction of model uncertainty through evidence accumulated across many iterations of design, execution, and reflection.

The meta-learning paradigm, notably "learning to learn" (Andrychowicz et al., 2016), which casts optimizer design itself as a learning problem, foreshadows this self-revision capability: a system improves its own learning procedure rather than merely fitting data. L3 extends this principle from parameter optimization to world-model revision. The result is a paradigm for the next stage of world modeling: systems that not only simulate but also evolve by themselves, acquiring continual learning and self-revision capabilities. In an agent context, this means that the agent's world model is no longer a static artifact consumed at inference time; instead, it becomes a curious living component that diagnoses its own failures, designs targeted experiments to resolve ambiguities, and distills the resulting evidence into persistent model updates. Such self-evolving agents represent a qualitative shift from L2, where the model is a fixed tool for planning, to L3, where the model itself is the object of continual improvement driven by its own deployment experience.

5.1 Formal Definition

L3 extends L2 from simulation within a fixed information regime to **closed-loop, evidence-driven model revision**. In summary, L3 systems are defined by realizing the entire design–execute–observe–reflect loop, in which new evidence is actively acquired to challenge and revise the model across iterations.

Formally, an L3 system maintains and updates a world-modeling stack:

$$\mathcal{M}_t \xrightarrow{\text{design}} a_t \xrightarrow{\text{execute}} o_t \xrightarrow{\text{observe}} d_t \xrightarrow{\text{reflect}} \mathcal{M}_{t+1}$$

where $\mathcal{M}_t$ denotes the current world-modeling stack at iteration t, a_t the designed experiment or action, o_t the raw outcome, and d_t the distilled evidence used to update the stack (Figure 9).

Importantly, the presence of this loop alone is not sufficient to establish L3 capability. What distinguishes L3 systems is that evidence is translated into *persistent, reusable model updates* that are validated under regression checks, rather than remaining as transient, in-context adjustments. The model itself becomes the object of improvement, not merely a fixed substrate for planning.

This formulation closely mirrors the structure of scientific practice. A scientific community can be viewed as an L3 system operating over a shared model $\mathcal{M}_t$, consisting of established theories together with known anomalies. Under "normal science" (Kuhn, 1962), the community continuously updates $\mathcal{M}_t \to \mathcal{M}_{t+1}$ through incremental refinements that preserve the underlying model class. When accumulated anomalies exceed the explanatory capacity of the current model, the same update process produces a more substantial transition, in which the structure of $\mathcal{M}_t$ itself is revised, corresponding to a paradigm shift. In this view, both gradual refinement and paradigm shifts are instances of the same evidence-driven update, differing only in scale.

In practice, the bottleneck of an L3 system is typically not generating candidate fixes but validating them safely. Multimodal critic models (Zhang et al., 2025a) and regression-gated update pipelines (Ren et al., 2026; Jimenez et al., 2024; Yang et al., 2024a) provide practical infrastructure for the observe and reflect stages. We cite these systems not as full L3 exemplars, but as components of an evolver design space once persistent update and validation mechanisms are in place.

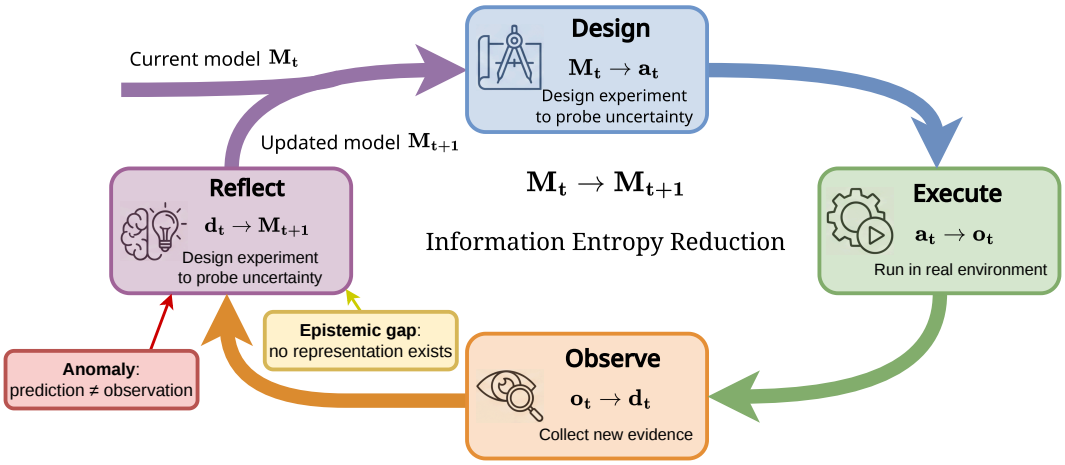

Figure 9: **The L3 evolution loop.** A full cycle proceeds through four stages: design, execute, observe, and reflect, producing a revised world-modeling stack $\mathcal{M}_{t+1}$.

Revision triggers and evolution policy. The reflect stage is responsible for deciding *when* and *how* the world model should be revised, in particular distinguishing between incremental improvement and structural change. In practice, this decision is driven by two types of signals. An *anomaly* denotes a mismatch between prediction and observation. While many anomalies can be absorbed through local adjustments, persistent anomalies that resist resolution within the current model class reveal an *epistemic gap*, indicating that the underlying representation or hypothesis space is insufficient. Resolving such gaps typically requires structural changes to the model, corresponding to a paradigm shift. In the philosophical framing of Section 2.1, anomalies that can be absorbed correspond to adjustments within Lakatos's "protective belt" (learned parameters), while persistent anomalies that expose epistemic gaps demand changes to the "hard core" (architecture, inductive biases). In operational terms, this induces a hierarchy of responses: small anomalies are handled via online adaptation within an episode, persistent anomalies trigger parameter updates that are distilled into the model, and epistemic gaps require structural modifications such as introducing new modules or expanding the hypothesis space. Duhem–Quine holism (Section 2.1) highlights that this attribution is inherently ambiguous: a mismatch between prediction and observation can often be explained by multiple components of the model, including representation, dynamics, or auxiliary assumptions. As a result, it is non-trivial to determine whether an anomaly can be resolved through local updates within the current model class or reflects a deeper epistemic gap that requires structural change to the underlying representation.

This difficulty is further shaped by the choice of representation. Epistemic gaps often involve missing or incorrect inductive biases or invariances in the model. While such gaps can, in principle, be addressed through structural changes in learned models, these changes are typically indirect: modifying architecture or training procedures does not guarantee a predictable change in the underlying invariances the model captures. In contrast, symbolic representations allow these invariances to be expressed and manipulated explicitly, as seen in scientific laws and principles. This suggests that while latent representations provide a flexible substrate for absorbing anomalies through parameter updates, resolving epistemic gaps may require representations that expose and manipulate invariances explicitly, as is standard in symbolic scientific models. We return to this tension between latent and symbolic representations in Section 2.2.

5.2 Distinction from L2

We use three boundary conditions to mark L2 → L3, each corresponding to a transition in the design–execute–observe–reflect loop:

1. **Active information expansion ($\mathcal{M}_t \rightarrow e_t$)**: the system designs experiments that actively probe uncertainty or challenge its current belief, rather than only optimizing within existing knowledge.

2. **Autonomous execution and observation** ($e_t \to d_t$): the system carries out experiments and acquires evidence through interaction, rather than relying on simulated or pre-existing data.
3. **Belief revision under challenge** ($d_t \to \mathcal{M}_{t+1}$): observations are used to reflect on and revise the model, including updating parameters, structure, or assets, enabling correction of prior assumptions.

The boundary conditions above can be unified as a single principle: whether the world model remains fixed or becomes plastic during deployment. This transition from L2 to L3 manifests in three aspects: whether the model can update its parameters and structure after deployment, how it accumulates new capabilities over time, and whether it passively consumes data or actively generates it through experimentation.

Fixed vs. adaptive. An L2 simulator is typically fixed post-training. It can generate infinite rollouts based on its training data, but its core transition function $p_\theta(z_t \mid z_{t-1}, a_t)$ does not evolve; it explores the implications of its frozen knowledge. In contrast, an L3 system is adaptive post-deployment: it treats its own parameters or structure as a hypothesis to be updated, i.e. $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t + \text{Evidence}$.

Modes of growth. L3 growth goes beyond simple data buffering and encompasses three different modes:

- **Parameter update**: modifying weights via gradient descent or Bayesian updates on new evidence, e.g., online learning, continual RL fine-tuning, and Bayesian model updates.
- **Architecture update**: dynamically adding new modules, experts, or capacity to handle complexity, for example, expanding the context window or allocating new memory slots.
- **Hypothesis-space expansion**: extending the model class to represent explanations that were previously inexpressible. This corresponds to introducing new variables, mechanisms, or abstractions, shifting from "I don't know which of these k options is true" to "the correct explanation is not among the current k options." This is the most challenging mode and is closely tied to abduction and genuine scientific discovery.

Passive vs. active. While L2 systems may support passive online learning (updating weights on a stream of incoming data) or decision simulating (Section 4.2.4), L3 is characterized by *active trial-and-error loop*. It does not just wait for data; it acts to generate data that maximizes information gain regarding a specific hypothesis or area of uncertainty. This active stance transforms the agent from a consumer of experience to a designer of experiments, a qualitative shift that connects directly to the philosophy of abduction and scientific method (Section 2.1). L3 should not be defined by closed-loop use in the generic planning sense; rather, it is defined by closing the evidence-to-revision loop, so that deployment outcomes are used to diagnose, update, and validate the world-modeling stack itself over successive iterations of use.

5.3 Examples and Applications

L3 is most tractable in domains that are highly instrumented, offer rapid feedback, and provide well-defined evaluation criteria. Empirical support for L3 is uneven across domains: autonomous science and other highly instrumented settings provide the clearest demonstrations, whereas social, code, and embodied environments remain partly empirical and partly prospective design space. We illustrate this landscape, together with the characteristic evidence signals and failure modes in each, across four governing-law regimes in Figure 10.

Physical intelligence. In embodied settings, L3 manifests as adaptive probing to infer and update dynamics models. When a robot encounters unexpected contact dynamics, such as a slippery surface or a deformable object, the system can actively execute diagnostic actions (small perturbations designed to disambiguate between hypotheses about the contact model) and use the resulting evidence to update its dynamics model. The anomaly signals in this regime are inherently physical: force/torque deviations, unexpected contact events, and discrepancies between predicted and observed end-effector trajectories provide quantitative evidence for model updates. Recent work demonstrates that robots can autonomously detect physical damage and re-train persistent self-models: Hu et al. (2025) show that an egocentric visual self-model detects morphology changes via prediction-versus-observation mismatch and re-trains to recover locomotion. AdaptSim (Ren et al., 2023) meta-learns an adaptation policy that iteratively revises simulation parameters

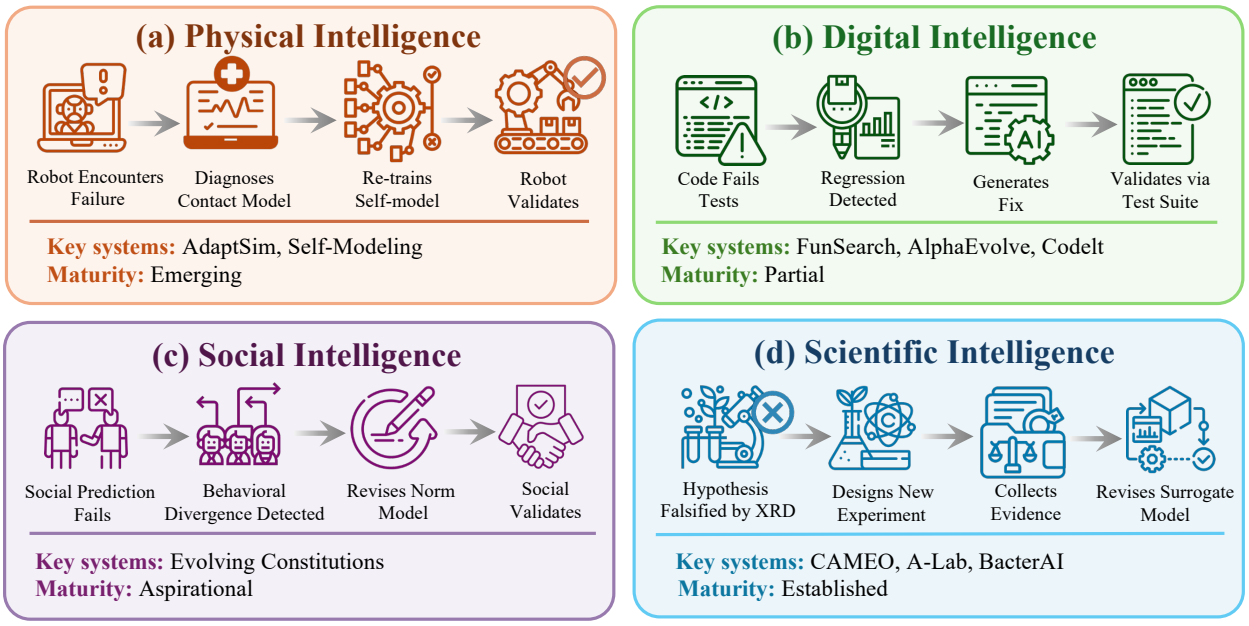

Figure 10: **L3 evolution across four governing-law regimes.** Each panel illustrates the design–execute–observe–reflect loop in a representative domain: (a) Physical intelligence—adaptive probing revises contact dynamics; (b) Social intelligence—norm drift triggers social-model revision; (c) Digital intelligence—evaluator-driven program search with regression gates; (d) Scientific intelligence—closed-loop autonomous experimentation at a synchrotron beamline.

from small amounts of real-world task performance data, closing the sim-to-real gap through evidence-driven simulation revision rather than fixed domain randomization, with each real-world deployment informing the next round of simulation updates.

Digital intelligence. Software and web environments are naturally suited to L3 because state is fully observable, actions are deterministically replayable, and regression testing provides a built-in validation gate. Evaluator-driven discovery loops exemplify this regime. Romera-Paredes et al. (2024) pair a pretrained LLM with an automated evaluator in an evolutionary loop: the LLM generates candidate programs, the evaluator scores them against a formal specification, and high-scoring solutions are fed back for further refinement. This loop discovered new constructions for the cap set problem (a long-standing open problem in combinatorics) and new bin-packing heuristics that outperform known baselines. The evaluator serves as an automated regression gate, a key L3 property, although the system realizes only the design and observe components (program generation and automated scoring) without active information expansion or persistent model revision. Novikov et al. (2025) extend this evolutionary coding paradigm: by pairing LLM-generated program mutations with automated correctness evaluators, the system improved on Strassen's matrix multiplication algorithm after 56 years and solved 20% of open mathematical problems beyond the prior state of the art, illustrating the power of formal verification as an L3 gatekeeper in algorithmic domains. CodeIt (Butt et al., 2024) closes a tighter loop: the LLM is fine-tuned from its own search trajectories via prioritized hindsight replay, so that the generative model itself (serving as an implicit world model of program space) persistently improves across tasks. The AI Scientist-v2 (Yamada et al., 2025) pushes further into computational experiments by employing agentic tree search for experiment selection: the system autonomously formulates hypotheses, designs and executes experiments, analyzes results, and writes complete manuscripts. A VLM feedback loop iteratively refines figures and content. In 2025, this system produced an entirely AI-generated paper that passed peer review at an ICLR workshop. However, the system's experiments are computational (running ML training jobs), and its revision loop operates on paper quality rather than mechanistic understanding, illustrating the gap between L3 in well-instrumented computational domains and the harder challenge of genuine scientific discovery. In AUI (Lin et al., 2025a), a Coder–Computer-Use Agent loop instantiates this principle in website: the Coder iteratively revises website

implementations, while the CUA acts as an automated evaluator by executing task trajectories and verifying functional correctness (e.g., navigation success and task completion). The resulting feedback, grounded in executable interactions rather than static inspection, serves as a regression signal that guides subsequent code updates, forming a closed-loop optimization process aligned with L3 properties.

Social intelligence. L3 in social domains requires revising the agent's social model when predicted behavior of other agents deviates from observed behavior, for example, when Theory-of-Mind predictions fail systematically or when social norms drift over time. This is currently the hardest regime for L3 because attribution is inherently ambiguous (a failed social prediction may reflect incorrect beliefs about the other agent's goals, an outdated norm model, or stochastic behavior) and because social experiments are ethically constrained. Early work on norm emergence and convention formation in multi-agent populations (Section 4.2.3) represents a preliminary step toward social L3, but persistent, validated revision of social world models from deployment evidence remains largely open. A preliminary step toward social L3 is the evolutionary synthesis of multi-agent governance rules: Kumar et al. (2026) use LLM-driven genetic programming to evolve interpretable constitutions from societal stability scores, surpassing human-designed rules by 123%.

Scientific intelligence. The most complete current examples of L3 come from autonomous science, where the full design–execute–observe–reflect loop is closed by instrumentation. The paradigm of autonomous closed-loop scientific discovery was established by Robot Scientist Adam (Sparkes et al., 2010), the first machine to autonomously design experiments about gene function, execute them, observe the outcomes, and revise its model. Its successor system demonstrated closed-loop cycles of experiment design, execution, and model revision in yeast systems biology, accelerating biological model development (Coutant et al., 2019). CAMEO (Kusne et al., 2020) implements closed-loop materials discovery via Bayesian active learning at a synchrotron beamline: the system predicts which phase a candidate composition will form, synthesizes it, characterizes the product via X-ray diffraction, updates its Bayesian belief model, and actively selects the next experiment to maximize information gain. Each experimental cycle takes seconds to minutes, and the system discovered a novel phase-change memory material without additional human training. A-Lab (Szymanski et al., 2023) extends this to fully autonomous synthesis: three robotic arms automate powder dosing, heating, and XRD characterization, with an active-learning algorithm generating improved recipes when targets fail. In 17 days of closed-loop operation, A-Lab performed 353 experiments and realized 36 compounds from 57 targets. Crucially, analysis of failed syntheses provided structured evidence to refine future synthesis strategies; the failures were not discarded but distilled into persistent knowledge. Strieth-Kalthoff et al. (2024) extend the self-driving laboratory paradigm to distributed, multi-site operation: a delocalized SDL autonomously discovers novel organic laser emitters by iteratively updating a Bayesian surrogate from synthesis and characterization data across geographically separated facilities. BacterAI (Dama et al., 2023) demonstrates that L3 can operate with zero prior biological knowledge: the system iteratively designs and executes experiments to map microbial amino acid requirements, revising its metabolic model purely from experimental evidence. In computational chemistry, MOOSE-Chem (Yang et al., 2025d) demonstrates that an LLM-based framework can rediscover chemistry hypotheses published in Nature and Science in 2024 using only pre-2024 literature, providing evidence that the hypothesis-generation component of the L3 loop is already feasible for natural-science domains. Its successor, MOOSE-Chem2 (Yang et al., 2025c), introduces hierarchical search over fine-grained hypothesis components to improve both precision and novelty of generated discoveries. CASCADE (Huang et al., 2025b) provides a self-evolving multi-agent framework for materials science and chemistry that cumulatively acquires reusable executable skills via continuous learning and self-reflection meta-skills, complemented by memory systems, benchmarked across hundreds of functions in over 20 databases, packages, and software tools, with applications in hypothesis-driven discovery and autonomous laboratory workflows. Broader agentic systems are pushing the L3 loop further into biomedicine. Biomni (Huang et al., 2025a) provides a general-purpose biomedical AI agent that integrates over 100 tools and 59 databases spanning 25 subfields, enabling autonomous execution of tasks from causal gene prioritization to drug repurposing. BioLab (Jin et al., 2025) extends this to end-to-end autonomous life-sciences research via a multi-agent system built on biological foundation models. OriGene (Zhang et al., 2025h) demonstrates a self-evolving virtual disease biologist that autonomously discovers therapeutic targets through iterative hypothesis refinement. The AI co-scientist system (Gottweis et al., 2025) employs a generate–debate–evolve approach to hypothesis generation, with multi-agent tournament processes that

Table 8: **Representative L3 systems by governing-law regime.** Loop steps indicate which stages of the design, execute, observe, and reflect cycle each system realizes.

System	Links	Design	Execute	Observe	Reflect
Physical World					
AdaptSim (Ren et al., 2023)	Paper Code	✔	✔	✔	✗
Self-Modeling (Hu et al., 2025)	Paper Code	✔	✔	✔	✔
Digital World					
FunSearch (Romera-Paredes et al., 2024)	Paper Code	✔	✔	✔	✗
CodeIt (Butt et al., 2024)	Paper Code	✔	✔	✔	✔
SWE-agent (Yang et al., 2024a)	Paper Code	✔	✔	✔	✗
AUI (Lin et al., 2025a)	Paper Code	✔	✔	✔	✗
AlphaEvolve (Novikov et al., 2025)	Paper Code	✔	✔	✔	✗
Social World					
Evolving Const. (Kumar et al., 2026)	Paper —	✔	✔	✔	✔
AgentSociety (Piao et al., 2025)	Paper Code	✔	✔	✔	✗
Scientific World					
Robot Scientist (Sparkes et al., 2010)	Paper —	✔	✔	✔	✔
CAMEO (Kusne et al., 2020)	Paper Code	✔	✔	✔	✔
Yeast Cycles (Coutant et al., 2019)	Paper —	✔	✔	✔	✔
BacterAI (Dama et al., 2023)	Paper Code	✔	✔	✔	✔
A-Lab (Szymanski et al., 2023)	Paper —	✔	✔	✔	✔
SDL Lasers (Strieth-Kalthoff et al., 2024)	Paper Code	✔	✔	✔	✔
AI Scientist (Lu et al., 2024)	Paper Code	✔	✔	✔	✔
CASCADE (Huang et al., 2025b)	Paper Code	✔	✔	✔	✔
Biomni (Huang et al., 2025a)	Paper Code	✔	✔	✔	✗
BioLab (Jin et al., 2025)	Paper —	✔	✔	✔	✔
OriGene (Zhang et al., 2025h)	Paper Code	✔	✔	✔	✔
Co-Scientist (Gottweis et al., 2025)	Paper —	✔	✔	✔	✔
AI Scientist v2 (Yamada et al., 2025)	Paper Code	✔	✔	✔	✔

have been validated in drug repurposing and epigenetic target discovery. Complementing these systems, Yang et al. (2026) introduce a dynamic benchmark revealing that current LLMs still fall short on genuine biological knowledge derivation, underscoring the persistent gap between literature retrieval and true L3 revision that actually updates the underlying model.

Table 8 summarizes representative L3 systems across the four governing-law regimes, indicating which stages of the design–execute–observe–reflect loop each system realizes.

Evidence quality and falsifiability. The quality of evolution depends on the quality of evidence. Table 9 organizes the revision signals that trigger L3 model updates in each governing-law regime: what the agent detects, why it indicates the current model is wrong, and how falsifiable the signal is. A useful principle is to prefer falsifiable evidence (Section 2.1). A screenshot combined with a DOM snapshot, error code, and action sequence is reproducible and refutable; "I think the page didn't load" is not. Human feedback should not be treated as a single falsifiability class: subjective or preference feedback is weakly falsifiable, whereas expert diagnostic feedback can be strongly falsifiable when its claims are subsequently checked by tests, experiments, or structured evaluation. Evolver's progress depends on making lessons verifiable, and reversible when wrong. This requirement connects directly to the anomaly and epistemic-gap triggers defined in Section 5.1: an anomaly is actionable only when the deviation between prediction and observation can be quantified from recorded evidence, and an epistemic gap is recognizable only when the system can demonstrate that no existing hypothesis adequately accounts for the observation. In large-scale deployments, evidence must also be compressible and indexable. Practical systems maintain multi-resolution evidence: a compact error category combined with a state fingerprint and diff summary for fast retrieval, together with pointers to heavier artifacts (screenshots, DOM snapshots, full logs) for deep audits. Evidence quality is also tightly

Table 9: **Revision signals for L3 evolution by governing-law regime.** Row color encodes *within-domain* falsifiability (not comparable across regimes): High, Medium, Low.

Revision Signal	Trigger Condition
Physical World	
Kinematic infeasibility (Ren et al., 2023)	Trajectory violates joint limits or collision bounds
Contact dynamics mismatch (Hu et al., 2025)	Force/torque deviates from predicted contact model
Morphology change (Hu et al., 2025)	Visual self-model diverges from observed body state
Social World	
Interventional inconsistency (Piao et al., 2025)	Policy change fails to elicit proportional behavioral shift
Global behavioral drift (Kumar et al., 2026)	Parameter perturbation yields inconsistent aggregate response
Individual faithfulness violation (Taubenfeld et al., 2024)	Agent behavior deviates from demographic priors
Digital World	
Regression detection (Romera-Paredes et al., 2024)	Previously passing test fails post-update
Execution outcome mismatch (Lin et al., 2025a)	Predicted state differs from actual execution result
Task completion failure (Butt et al., 2024)	Action sequence fails to achieve specified goal
Scientific World	
Hypothesis falsification (Szymanski et al., 2023)	Experiment contradicts predicted outcome
Prediction–measurement gap (Kusne et al., 2020)	Surrogate output diverges from measurement
Epistemic gap detection (Dama et al., 2023)	Observation falls outside model's representational scope

coupled to privacy and safety constraints: an Evolver pipeline must separate what is stored persistently (sanitized logs, hashed fingerprints) from what is kept transient or behind access controls, protecting sensitive data while retaining an audit trail (Xie et al., 2024; Yang et al., 2025a).

Continuous self-improvement also introduces governance challenges, including benchmark overfitting, knowledge contamination, and misattribution of failures to wrong components. These risks and the practical measures to mitigate them (versioning, rollback, regression gates) are discussed as open problems in Section 8.

5.4 L3 in Context: Maturity, Governance, and Outlook

Having established the L3 evolution loop, its domain instantiations, and the role of evidence quality, we now examine its practical status and implications. This subsection addresses two complementary questions: *maturity*, i.e., where L3 systems have been successfully realized across governing-law regimes; and *governance*, i.e., what risks arise from persistent, automated model revision. Together, these perspectives characterize L3 both as a modeling paradigm and as a deployed system that must evolve reliably under real-world constraints.

Maturity across different domains. We summarize maturity across the four governing-law regimes:

1. **Scientific (Established).** The most mature regime, offering fast, structured feedback, unambiguous anomaly signals (hypothesis falsification), and well-defined revision targets (surrogate model parameters, synthesis recipes) (Kusne et al., 2020; Szymanski et al., 2023; Sparkes et al., 2010; Dama et al., 2023). Primary bottleneck: instrument access and real-data budget.

2. **Digital (Partial).** Regression testing provides an automated validation gate, but many systems still lack the active information-expansion boundary condition (Romera-Paredes et al., 2024; Novikov et al., 2025; Butt et al., 2024). Primary bottleneck: active experiment design is often absent.

3. **Physical (Emerging).** Promising but limited by attribution difficulty: a failed manipulation can stem from perception, dynamics, actuation, or environmental change, and isolating the brittle component requires careful experimental design (Ren et al., 2023; Hu et al., 2025). Primary bottleneck: failure attribution across perception, dynamics, and actuation.
4. **Social (Aspirational).** Social experiments are ethically constrained, attribution is inherently ambiguous, and behavioral ground truth is noisy (Kumar et al., 2026). Primary bottleneck: attribution ambiguity and ethical constraints on social experimentation.

Governance challenges. Three governance risks arise specifically from persistent, automated model revision. *Benchmark overfitting* occurs when the regression gate is too close to the training distribution; the system learns to pass tests rather than improve genuinely. *Knowledge contamination* occurs when the revision loop incorporates evidence that is itself biased or adversarially constructed, silently degrading the model on OOD inputs. *Misattribution cascades* occur when a fix for one failure mode inadvertently degrades another component; without comprehensive regression suites, the net effect of an update can be negative. Mitigations include held-out probe sets that are refreshed independently of training data, canary deployment that surfaces regressions before full rollout, and causal ablations that isolate the contribution of each update.

Relationship to Sections 6 and 7. From an evaluation perspective (Section 6), assessing L3 requires protocols that go beyond single-episode accuracy: the key metric is whether the system improves across revision cycles k without regressing on held-out probes. From an implementation perspective (Section 7), L3 places the heaviest demands on the system stack (persistent storage, replay infrastructure, regression harnesses, and rollback mechanisms) that are often underspecified in current architectures. Building toward L3 therefore means investing in evaluation infrastructure as much as in model capacity.

6 Evaluations

Evaluating world models for agentic AI requires moving beyond standard generative metrics toward decision-centric protocols organized around three boundary conditions: long-horizon coherence, intervention sensitivity, and constraint consistency. This section first motivates this shift (Section 6.1) and then shows how the *same* benchmark can test L1, L2, or L3 depending on the evaluation protocol (Section 6.2). World-model-specific evaluations show that even frontier models still suffer from substantial capability gaps, while no single benchmark fully captures the space of interest.

6.1 From prediction-centric to decision-centric evaluation

While standard generative metrics such as Fréchet Inception Distance (FID), Fréchet Video Distance (FVD), SSIM, and per-pixel reconstruction loss capture perceptual quality, they are at best weak indicators (Brooks et al., 2024; Ball et al., 2025) of agentic capability and offer limited predictive power for the downstream decisions an agent must ultimately make once embedded in the real-world environment.

As a result, a world model can generate visually convincing rollouts while still breaking down during planning because of hallucinated object dynamics, action-insensitive transitions, or subtle physics violations. These errors are often invisible to distribution-level metrics but devastating for downstream decision-making.

The root cause is a mismatch between what is measured and what matters. The object of evaluation should not be a single-step prediction $p_\theta(z_t \mid z_{t-1}, a_t)$ in isolation, but the **trajectory-level rollout**

$$\hat{p}(\tau \mid z_0, a_{1:H}, c), \qquad \tau = (z_1, \ldots, z_H),$$

and specifically whether this rollout is reliable enough for a planner to act on (Section 4). Aggregate measures such as mean success rates further obscure the picture by masking high variance across task instances (Agarwal et al., 2021; Henderson et al., 2018).

We therefore organize evaluation around the three **boundary conditions** that mark L1→L2 (Section 4.1):

1. **Long-horizon coherence:** rollouts remain decision-usable over H steps rather than degrading via error.

2. **Intervention sensitivity:** counterfactual edits (action or premise changes) induce stable and directionally meaningful trajectory changes.

3. **Constraint consistency:** generated futures respect the governing laws of the target regime (Section 4).

These three conditions hold across the four governing-law regimes introduced in Section 4, and together they give a common framework within which we can organize the evaluation protocols, benchmark analyses, and reporting standards described in the remainder of this section.

World-model evaluation is ultimately meaningful only insofar as it reflects downstream decision quality. Benchmarks that capture long-horizon coherence, intervention sensitivity, or constraint consistency are valuable not merely as diagnostics, but because these properties should translate into better plan selection, fewer costly invalid actions, and greater task success under distribution shift. The relevant bridge is therefore not "Does the model look realistic?" but "Does improved model validity change what the agent chooses, and does that shift in choice in turn improve real-world task outcomes?"

Two aggregate metrics operationalize these conditions for downstream decision-making. The **Action Success Rate** (ASR) measures how often a planner that uses the world model's rollouts to select actions achieves the task goal in the real environment:

$$\mathrm{ASR} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\big[\text{task}_i \text{ succeeds under policy derived from } \hat{p}\big].$$

The **Counterfactual Outcome Deviation** (COD) measures intervention sensitivity by comparing rollout outcomes under two policies $a^{(1)}_{1:H}$ and $a^{(2)}_{1:H}$ that differ at a single intervention step k:

$$\mathrm{COD}(k) = \mathbb{E}\big[d\big(\hat{z}^{(1)}_H,\, \hat{z}^{(2)}_H\big)\big],$$

where d is a task-relevant distance (e.g., goal-state distance in physical tasks, edit distance in software tasks). When COD is low, a world model is largely unresponsive to changes in action, which makes it uninformative for counterfactual planning. Together, ASR and COD provide a more direct link between world-model quality and downstream agentic performance: ASR assesses whether the model supports good decisions, whereas COD assesses whether the model responds in a meaningful way to action-level interventions.

6.2 Differentiating L1, L2, and L3 via evaluation protocol

Importantly, the *same* benchmark can evaluate different capability levels depending on protocol. The level is fixed not by the benchmark but by what the protocol demands: L1 tests local prediction, L2 tests decision-usable simulation under the three boundary conditions, and L3 tests whether the system can revise itself from evidence (Section 5.1). The examples below make this concrete, one per governing-law regime.

RoboCasa (Physical World). At L1, the benchmark reduces to predicting the next end-effector position given current state and action, measured by single-step position error. Elevating the protocol to L2 requires executing a full kitchen task (e.g., a pick-place-heat sequence) under mid-task perturbations such as object displacement or drawer obstruction; the relevant metrics shift to long-horizon success rate, catastrophic action fraction, and recovery rate after perturbation. An L3 protocol would further demand that the agent, after repeated failures on a novel kitchen layout, distills a persistent grasp strategy (e.g., "this handle requires a top-down approach") whose benefit carries over to subsequent trials. In practice, most robotic manipulation systems still report L1-style single-task success rates, and perturbation injection remains nonstandard.

OSWorld and SWE-bench (Digital World). These benchmarks span the L1–L2 boundary. Single-step click prediction (OSWorld) or single-line code completion (SWE-bench) constitutes L1 evaluation. L2 demands cross-file issue resolution under injected failures (network timeouts, unexpected pop-ups, out-of-distribution states), tracked via long-horizon consistency and catastrophic action fraction (Xie et al., 2024; Jimenez et al., 2024). The leap to L3 would require the system to generate durable artifacts: a reproduction

script that becomes a regression test (SWE-bench), or a reusable installation procedure distilled from a failed attempt (OSWorld). Today, leaderboard systems primarily operate at L1–L2; L3-style asset generation is reported anecdotally across isolated case studies but not yet evaluated systematically or at scale.

Sotopia (Social World). Social simulation introduces a distinctive challenge: the "perturbation" is another agent's strategy shift rather than a physics disturbance. An L1 evaluation measures next-turn prediction accuracy or perplexity. For L2, one agent's strategy is changed mid-conversation (counterfactual injection), and the protocol tracks goal completion under perturbation, commitment consistency, and whether social outcomes shift appropriately (Zhou et al., 2024). L3 evaluation would require the agent to distill, after repeated negotiation failures, a new social strategy or norm-handling rule that persists and transfers to structurally similar scenarios. Existing social benchmarks rarely inject such counterfactual perturbations, leaving L3-style social strategy evolution largely unexplored in current negotiation and role-playing settings.

ScienceWorld and DiscoveryBench (Scientific World). Of the four regimes, autonomous science is closest to a genuine L3 evaluation paradigm. At L1, the task is predicting the outcome of a single experimental action (e.g., "what happens when acid is added to base?"). L2 protocols require designing and executing a multi-step experimental sequence while maintaining causal coherence, measured by sequence validity, hypothesis-consistent action rate, and robustness to misleading observations (Wang et al., 2022; Majumder et al., 2025). The critical L3 step is hypothesis revision: when experimental evidence falsifies a prediction, the system must update its belief structure and avoid previously falsified paths. Closed-loop systems such as CAMEO already demonstrate this kind of evidence-driven model revision in laboratory settings.

Evaluation gaps and coverage. The vast majority of current systems are evaluated only at L1 (single-step accuracy or end-to-end success rate without perturbation injection). L2-style evaluation protocols (counterfactual injection, degradation curves, constraint-violation detection) exist in principle and are demonstrated by individual benchmarks, but are not yet standard practice across the field. L3 evaluation infrastructure (regression suites, asset validation gates, cross-episode improvement tracking) is essentially nonexistent outside autonomous science. Closing this gap is a prerequisite for claims about world-modeling capability.

6.3 Benchmarks and coverage analysis

A growing line of work asks whether current systems genuinely learn world models or merely surface correlations. WorldSimBench (Qin et al., 2024) and WorldModelBench (Li et al., 2025a) reveal that perceptual realism and action-conditioned fidelity can diverge sharply, while Vafa et al. (2024) and Kang et al. (2025a) demonstrate fundamental coherence and consistency failures that persist at scale. No single benchmark covers all capabilities. We use four qualitative coverage labels in the capability matrix. **Strong (S)** means the benchmark directly and intentionally tests the capability through explicit task design and scoring. **Medium (M)** means the capability is exercised in a substantial but partial or indirect way. **Weak (W)** means the benchmark offers only incidental evidence about the capability. − means the capability is not meaningfully tested. These labels are judgment-based but are assigned according to task design, scoring visibility, and whether failure on the capability can be unambiguously detected from benchmark traces. Broader multimodal evaluation resources such as HSSBench can complement this landscape by probing humanities- and social-science reasoning, although they do not directly evaluate decision-usable social-state rollouts or other world-model-specific boundary conditions (Kang et al., 2025b). We also propose the Minimal Reproducible Evaluation Package (MREP), a community standard for version locking, trace logging, failure taxonomy, tail statistics, and boundary condition mapping. Seen this way, MREP is not only an evaluation proposal for individual papers but also the minimal evidence infrastructure required to sustain any credible L3 gatekeeping loop in practice.

6.4 Open Challenges in Evaluation

Evaluation challenges remain even when benchmark suites and logging infrastructure improve. Some are methodological, including benchmark saturation and evaluation gaming; others are infrastructural, including

the cost and variability of human evaluation and the lack of systematic meta-evaluation. *Benchmark saturation*: as top systems converge on near-ceiling performance, the discriminative power of existing benchmarks decreases. *Evaluation gaming*: systems can optimize for benchmark-specific artifacts rather than genuine capability. *Human evaluation*: for social-world and open-ended scenarios where automated metrics are unreliable, human judgment remains necessary. *Meta-evaluation*: whether an evaluation protocol itself is valid is a question that is rarely addressed systematically across current world-model benchmarks.

7 Architectural and Computational Considerations

The value of a taxonomy is not categorization for its own sake, but guiding system design. This section decomposes world-model implementations along three architectural axes, namely representation, dynamics, and control interface (Section 7.1), and examines how the governing-law regime constrains which combinations are viable in practice (Section 7.2). Deploying these systems raises cross-cutting engineering challenges: the choice between end-to-end and modular training, latency-compute tradeoffs, sim-to-real transfer, and graceful degradation under model uncertainty. A learned world model amortizes simulation cost into a fixed computation graph at inference time, whereas explicit simulation typically scales more directly with the number of entities, interactions, solver steps, or horizon length. This does not mean neural inference is literally $O(1)$ in every relevant variable: its cost still depends on model size, input resolution, sequence length, and rollout depth. The practical advantage is instead that learned dynamics can offer near-constant-cost approximations with respect to aspects of system complexity that would otherwise require increasingly expensive explicit simulation. Efficiency techniques matter here not as generic deployment tricks but because they interact differently with the three capability levels. For L1 systems, compression mainly trades off against one-step predictive accuracy. For L2 systems, memory and rollout efficiency directly affect achievable horizon, counterfactual branching, and thus long-horizon coherence. For L3 systems, the same efficiency choices affect whether regression-gated update loops are cheap enough to run continuously in deployment. Scaling further demands efficiency techniques: few-step distillation for real-time planning, quantization and pruning under the constraint that compounding errors amplify even minor per-step degradation, and KV cache compression for long-horizon autoregressive dynamics.

7.1 Architectural building blocks: representation, dynamics, and control

Building a world-model system requires choosing components along three axes (Table 10). Each choice carries distinct tradeoffs that determine which capability level (L1/L2/L3) the resulting system can reach and in which governing-law regime the resulting design will be most effective along each of these three axes.

Representation. At one extreme, symbolic or programmatic states (e.g., VirtualHome (Puig et al., 2018)) offer interpretability and enable hard constraint enforcement, but demand heavy manual engineering and cover only pre-specified state spaces; they are best evaluated by success rate and error-branch coverage. At the other extreme, latent continuous representations, such as the RSSM in DreamerV3 (Hafner et al., 2025) and V-JEPA2 (Assran et al., 2025), handle high-dimensional multimodal inputs with relatively little hand-designed structure. Their weakness is that, over long horizons, they are more susceptible to semantic drift and state aliasing, making long-horizon consistency and failure attribution especially important for evaluation. VL-JEPA Chen et al. (2025a) develop a joint embedding predictive architecture which predicts the continuous embeddings of the target text. VLog (Lin & Shou, 2025) use a learnable token to retrieve the narration then serve as video-centric vocabulary in the long video understanding. Between these two extremes lie structured 3D representations, including occupancy models such as RoboOccWorld (Zhang et al., 2025g) and point-flow models such as PointWorld (Huang et al., 2026b). They are appealing because they fit physical constraints more naturally, but this advantage often comes with reconstruction and computational bottlenecks. As a result, reachability and stability become particularly important in evaluation. Finally, discrete token representations (e.g., VQ-VAE codebooks in IRIS (Micheli et al., 2023)) enforce compositionality and enable exact likelihood training via cross-entropy, bridging continuous perception with autoregressive dynamics.

Dynamics. Stochastic latent dynamics, exemplified by DreamerV3 (Hafner et al., 2025), express uncertainty and multimodality through principled ELBO training and uncertainty-aware rollouts, but may de-

grade or become miscalibrated over long horizons. Where uncertainty modeling is less critical, deterministic value-aware dynamics (MuZero (Schrittwieser et al., 2020), TD-MPC2 (Hansen et al., 2024)) optimize the transition function directly for downstream value prediction, trading generative flexibility for tighter integration with the control objective. Autoregressive token dynamics (iVideoGPT (Wu et al., 2024b), LWM (Liu et al., 2025)) offer a unified scalable interface that handles multiple modalities through a shared vocabulary, though long-horizon logical consistency remains a weak point. Diffusion-based dynamics (the Sora technical line (Brooks et al., 2024), DIAMOND (Alonso et al., 2024), and interactive environments such as Genie (Ball et al., 2025)) deliver photorealistic observation-level transitions, but the multi-step denoising they require at inference time often comes with weak action controllability.

Control interface. Online MPC-style approaches (TD-MPC2 (Hansen et al., 2024), PETS (Chua et al., 2018)) replan at every step using short-horizon rollouts, providing fast correction at the cost of compute and latency pressure. Tree search and expansion (MuZero (Schrittwieser et al., 2020), EfficientZero (Ye et al., 2021)) enable counterfactual branching and systematic look-ahead, though they amplify model errors and can exploit benchmark loopholes. Rather than planning in the environment at all, imagined-rollout policy optimization (the Dreamer family (Hafner et al., 2020; 2021; 2025)) trains a policy entirely on model-generated trajectories, avoiding real interaction during learning but requiring highly accurate dynamics. At the deployment end, offline policy distillation (GR-1 (Wu et al., 2024a)) enables cheap inference yet is fragile under distribution shift, motivating OOD stress tests. A distinct strategy altogether, replayable-environment interfaces (OSWorld (Xie et al., 2024), SWE-agent (Yang et al., 2024a)) sidestep learned dynamics entirely, treating the real environment as its own simulator and relying on receipt parsing and state fingerprinting. More broadly, part of the control problem is deciding when external computation should be invoked at all, rather than treating tool use as either mandatory or absent; adaptive tool-integration work provides a useful planner-side example of this distinction (Wang et al., 2025b).

7.2 Design tradeoffs across governing-law regimes

The building blocks above are not interchangeable; the governing-law regime determines which combinations are viable and which failure modes dominate. Table 11 summarizes how deployment-regime latency budgets constrain the viable dynamics model classes and their control interfaces.

Physical-world systems. Everything hinges on contact, reachability, and stability under continuous actions. Representations must preserve geometry and contact relations; dynamics must be stable over short-to-medium horizons; and the control interface must be fast enough for closed-loop correction. Latent or structured 3D representations paired with MPC or imagined-rollout policies dominate this regime (Hafner et al., 2025; Hansen et al., 2024; Huang et al., 2026b). Short-horizon rollouts reduce compounding error (Janner et al., 2019), and MPC provides an online correction mechanism. The main pitfalls are the preassumed existence of de facto 3D scenes, degraded 3D reconstruction capabilities, semantic drift in latent space, constraint violations that remain plausible in the learned representation, and the sim-to-real gap for contact-rich interactions. It is useful to distinguish at least three transfer curves in practice: transfer across input modalities, transfer across sensor suites, and transfer across environments, since each exposes a distinct failure mode of the learned dynamics and demands its own diagnostic instrumentation.

Digital-world systems. State-machine and branch consistency, rather than learned dynamics, are the primary bottleneck. Symbolic or DOM-based states paired with replayable environments are the dominant design in this setting (Xie et al., 2024; Jimenez et al., 2024; Yang et al., 2024a). Because they expose explicit state machines and support strong evidence logging, they make failures easier to trace and thus support Evolver-style asset distillation. This transparency, however, is not without cost: grounding may break under UI changes, loading variability and race conditions introduce non-deterministic noise, and benchmark artifacts remain vulnerable to reward gaming and to subtle shifts in the underlying software stack.

Social-world systems. The dominant bottleneck is maintaining coherent agent identity and relational state across extended interactions. Persona state must persist over hundreds of turns without drift, yet Theory-of-Mind (ToM) inference, which updates beliefs about other agents' goals, knowledge, and intentions,

Table 10: **Architectural building blocks for world models.** Three design axes are cross-referenced with concrete options, representative systems, strengths, and dominant failure modes.

Design Axis	Options, Systems, Strengths, and Failure Modes
	Representation
Representation	☐ **Symbolic / Programmatic:** VirtualHome. Interpretable; hard constraint enforcement. Failure: heavy manual engineering; limited state space. ☐ **Latent Continuous:** DreamerV3 (RSSM); V-JEPA 2. Scalable; absorbs high-dim multi-modal input. Failure: semantic drift; state aliasing over long horizons. ☐ **Structured 3D:** RoboOccWorld; PointWorld. Natural physical-constraint alignment. Failure: reconstruction bottleneck; high compute cost. ☐ **Discrete Tokens:** IRIS (VQ-VAE codebook). Compositional; exact cross-entropy training. Failure: codebook collapse; lossy quantization.
	Dynamics
Dynamics	☐ **Stochastic Latent:** DreamerV3. Principled uncertainty via ELBO; multimodal. Failure: miscalibration over long horizons. ☐ **Deterministic Value-Aware:** MuZero; TD-MPC2. Tight value integration; planning-optimized. Failure: no explicit uncertainty; brittle under novelty. ☐ **Autoregressive Token:** iVideoGPT; LWM. Unified multi-modal interface; scalable. Failure: weak long-horizon logical consistency. ☐ **Diffusion-Based:** Sora; DIAMOND; Genie 2. Photorealistic observation-level transitions. Failure: multi-step denoising latency; weak action control.
	Control Interface
Control Interface	☐ **Online MPC:** TD-MPC2; PETS. Fast closed-loop correction; reactive. Failure: high per-step compute; latency pressure. ☐ **Tree Search:** MuZero; EfficientZero. Counterfactual branching; systematic look-ahead. Failure: amplifies model errors; benchmark exploitation. ☐ **Imagined-Rollout Policy:** Dreamer family. No real interaction during training. Failure: requires highly accurate dynamics. ☐ **Offline Distillation:** GR-1. Cheap and fast deployment. Failure: distribution shift. ☐ **Replayable Environment:** OSWorld; SWE-agent. Real env as simulator; attributable failures. Failure: grounding breaks under UI/API changes.

imposes per-step costs that grow with the number of modeled agents. Multi-agent communication compounds the problem: n-agent interactions generate $O(n^2)$ pairwise belief updates per step, making naïve scaling infeasible for the 10,000+-agent simulations now appearing in the literature (Piao et al., 2025). Norm-consistency checking adds a further constraint: valid social rollouts must respect evolving norms (politeness conventions, negotiation protocols, institutional rules), and violations must be detectable at rollout time rather than post hoc (Zhou et al., 2024). The overarching challenge is that agent identity is not a fixed state vector but an emergent property of interaction history; maintaining stable identity under multi-turn dynamics while still allowing genuine belief revision remains an open architectural problem that current LLM-based agents address only superficially through system-prompt conditioning.

Generative simulation systems. The central tension is between visual fidelity and action controllability. High-fidelity diffusion or autoregressive models (Brooks et al., 2024; Bruce et al., 2024; Alonso et al., 2024) excel at producing photorealistic outputs useful for demonstration and synthetic data generation, but action-conditioning is often unstable and long-horizon consistency is difficult. A system can be mistakenly treated as planning-ready when it is not decision-usable; evaluation should prioritize action-response consistency and long-horizon stability over raw perceptual realism (Wu et al., 2024b; Liu et al., 2025).

Scientific-world systems. Evidence-chain validity and falsifiability matter more than perceptual quality in this regime (cf. the Popperian reading in Section 2.1). Representations must be interpretable and traceable to experimental evidence; dynamics must respect known mechanism boundaries; and the control interface should support experiment selection and belief updates rather than action execution (Wang et al., 2022). The

distinctive risks are hallucinated mechanisms that appear plausible but lack grounding, correlation mistaken for causation, and negative results that are silently discarded rather than propagated through the model.

VLA vs. native world models. A crosscutting architectural question is whether to embed world-model capabilities inside a Vision-Language-Action (VLA) model or to build a dedicated world-model module. VLAs inherit the scaling infrastructure and pretraining data of large language models, but their world-modeling capacity is implicit and difficult to isolate or evaluate. Recent efforts to make this capacity more explicit include spatially guided training that injects geometric structure into VLA policy learning (Chen et al., 2025d), aiming to bridge the gap between implicit visual knowledge and the explicit physical state awareness that world models require. Related work makes this implicit capacity more procedural than geometric: Pixel Reasoner equips VLMs with explicit visual operations such as zoom-in and select-frame for curiosity-driven evidence gathering, while Visual Rationale Learning treats such visual actions as core reasoning primitives rather than optional tools, together highlighting a broader shift toward explicit perceptual control inside VLM-like agents even when no standalone transition model is exposed (Su et al., 2025; Wang et al., 2025a). Native world models expose an explicit transition function that can be queried, composed, and stress-tested independently. Competition between these paradigms is partly a sociotechnical question: the massive investment in LLM infrastructure creates path dependencies that favor VLA-style integration even when a dedicated module might be technically superior (Hooker, 2021). From an evaluation standpoint, the litmus test is whether the system's predictions can be decoupled from its language generation and tested against the three boundaries (Section 4.1). Some architectural choices are also sociotechnical rather than purely algorithmic: whether the field converges on native world models or VLA-style surrogates may partly depend on tool ecosystems, available datasets, and hardware compatibility, besides intrinsic modeling power.

These regimes are not mutually exclusive. In practice, mature systems often stack multiple design patterns: symbolic or workflow planning at the top for high-level task decomposition, replayable environments in the middle for receipt validation and failure attribution, and short-horizon continuous control at the bottom for real-time correction (Moerland et al., 2023). This suggests that **the relevant unit of analysis is the composed system, not any single module in isolation**. Representation, dynamics, and control should therefore be evaluated together, in light of the constraints they impose and the evidence they make available. Many apparent disagreements in the literature then look less like fundamental disputes about whether world models work, and more like differences in where systems land along these design axes.

7.3 Implementation Roadmap

Table 12 distills the architectural guidance from the preceding sections into a concise roadmap organized by capability level and governing-law regime. For each cell, we list the representation format that best preserves the regime's planner-critical structure, the dynamics model class that is most tractable at that capability level, and the single most important engineering bottleneck that must be addressed to reach the next level.

Three cross-cutting engineering principles hold across all cells. First, **separate what is learned from what is enforced**: hard constraint layers (collision checkers, state-machine validators, regression gates) should be applied at inference time rather than learned implicitly, because soft enforcement through training loss cannot guarantee zero-violation rollouts. Second, **instrument before you iterate**: logging, replay, and failure attribution infrastructure should be built into the system from the start; without replay, L3 revision becomes anecdotal and ungovernable. Third, **match the representation to the planner's query**: a representation that looks realistic but does not expose the variables the planner needs (free space, permission state, reaction rate) is worse than a lower-fidelity representation that does.

8 Trends & Open Problems

The preceding sections have established L1–L3 as a capability ladder for world models. We now situate this ladder historically (Figure 6), survey the research frontier that is pushing each rung forward, and catalog the open problems whose resolution will determine whether world models mature from impressive demonstrations into reliable scientific and engineering tools.

Table 11: **Deployment latency budgets and engineering bottlenecks by regime.** Inference latency budgets range from sub-100 ms for real-time robotics to minutes for offline scientific planning; the table maps each regime to viable dynamics model classes and primary engineering bottlenecks. These are deployment budget ranges rather than measured benchmark results; empirical throughput depends on model size, hardware, batching, simulator implementation, and verification overhead.

Regime	**Latency, Dynamics Class, and Bottleneck**
	Physical World
Real-time robotics	☐ **Latency:** <100 ms.
	☐ **Dynamics:** Latent dynamics + MPC; lightweight RSSM; neural ODE.
	☐ **Bottleneck:** Per-step inference latency; compounding error within control loop.
Autonomous driving	☐ **Latency:** <200 ms.
	☐ **Dynamics:** Occupancy flow; latent diffusion; BEV prediction.
	☐ **Bottleneck:** Sensor-fusion latency; safety-critical constraint verification.
Embodied navigation	☐ **Latency:** 100–500 ms.
	☐ **Dynamics:** RSSM; object-centric GNN; point-cloud dynamics.
	☐ **Bottleneck:** 3D reconstruction cost; memory for large-scale maps.
	Digital World
Web / GUI agents	☐ **Latency:** ∼1–5 s.
	☐ **Dynamics:** LLM-as-world-model; DOM-based prediction; state-machine rollout.
	☐ **Bottleneck:** LLM inference cost; UI non-determinism and race conditions.
Software engineering	☐ **Latency:** ∼5–30 s.
	☐ **Dynamics:** LLM + MCTS rollout; code-graph traversal.
	☐ **Bottleneck:** Context window limits; cross-file dependency resolution.
Game AI (real-time)	☐ **Latency:** <50 ms.
	☐ **Dynamics:** Tree search (MCTS); value-aware latent dynamics; EfficientZero.
	☐ **Bottleneck:** Branching factor; search-depth vs. latency tradeoff.
	Social World
Social / multi-agent	☐ **Latency:** ∼1–10 s.
	☐ **Dynamics:** ToM network; multi-agent rollout; commitment-graph update.
	☐ **Bottleneck:** $O(n^2)$ pairwise belief updates; persona-state drift.
	Scientific World
Scientific planning	☐ **Latency:** Minutes to hours.
	☐ **Dynamics:** Full diffusion ensemble; Bayesian surrogate; PINN; active learning.
	☐ **Bottleneck:** Experiment budget; surrogate calibration; data scarcity.

8.1 Historical Development

Mathematical Principles (–1956). The impulse to build predictive models of reality long predates artificial intelligence. Newton's *Principia* (Newton, 1687) provided the first unified mathematical world model: given initial positions and velocities, his laws of motion and gravitation could, in principle, predict arbitrary future states of a mechanical system. Laplace (1814) distilled this ambition into the thought experiment now known as "Laplace's demon," an intelligence that, given complete knowledge of the present, could compute the entire future of the universe. Turing (1950) then posed the question of whether machines could think, establishing the conceptual bridge from mathematical modeling to artificial intelligence. These developments established the core tension that persists today: the tradeoff between model fidelity and tractable horizon. The philosophical foundations of this progression, from Hume's empiricism through Kant's structural priors to Lakatos's governed revision, are discussed in Section 2.1.

Table 12: **Design roadmap across governing-law regimes.** For each regime, we summarize the representation, dynamics, and bottleneck at L1–L3.

	Representation	Dynamics	Bottleneck
		Physical	
L1	Latent state, point-cloud input	RSSM, latent transitions	Long-horizon prediction error
L2	3D, object-centric state	Latent MBRL, neural ODE rollout	Contact instability, constraints
L3	Physics prior, residual model	Hybrid sim-to-real adaptation	Failure attribution across modules
		Digital	
L1	DOM tree, UI state	LLM-based state prediction	Grounding on unseen layouts
L2	State-machine abstraction	LLM rollout, MCTS planning	Exploits, race conditions
L3	Versioned tests, execution traces	Regression-gated updates	Safe deployment, rollback
		Social	
L1	Belief state, dialogue history	ToM, recurrent updates	Hidden mental states
L2	Commitment graph, norm state	Multi-agent rollout	Role drift, forgetting
L3	Social model, update gates	Bayesian revision	Attribution ambiguity, ethics
		Scientific	
L1	Molecular graph, field state	GNN surrogate, FNO dynamics	OOD generalization
L2	Hypothesis-evidence chain	Bayesian surrogate, PINN rollout	Hallucinated calibration
L3	Protocol, surrogate model	Active Bayesian learning	Data budget, instruments

Symbolic Intelligence (1956–1986). Early AI attempted to hand-code world models as logical rules and constraints. STRIPS (Fikes & Nilsson, 1971) introduced the first action-schema representation for robotic planning, but the Frame Problem (McCarthy & Hayes, 1969) (see Section 4.1) revealed that every action requires explicit axioms specifying what does *not* change, a burden that grows combinatorially. The Lighthill Report (Lighthill, 1973) catalyzed the first AI winter (1974–1980) by exposing the gap between laboratory demonstrations and real-world competence. The second winter (1987–1993) followed the brittleness of expert systems and the collapse of the Lisp-machine market: hand-crafted knowledge bases such as CYC could not gracefully handle uncertainty and commonsense exceptions (Lenat, 1995). The overarching lesson was clear: purely symbolic world models do not scale to open-world domains.

Connectionist Resurgence (1986–2020). The revival of neural networks, from backpropagation (Rumelhart et al., 1986) through deep convolutional networks (LeCun et al., 1998; Krizhevsky et al., 2012) and Transformers (Vaswani et al., 2017), shifted the paradigm from hand-coded rules to learned representations. World models re-emerged in model-based reinforcement learning, from latent dynamics models to general pixel-based control (see Section 3 for details).

Generative Revolution (2020–present). Diffusion models (Ho et al., 2020) and large-scale language models such as GPT-3 (Brown et al., 2020) have catalyzed a qualitative shift, building on the Transformer backbone established in the preceding era. Video generation models (Brooks et al., 2024; Agarwal et al., 2025; Bruce et al., 2024) and LLM-based agents (Hao et al., 2023; Wang et al., 2024a) are blurring the boundary between prediction and simulation, though systematic physics violations persist (Gu et al., 2025a) (Sections 3–4). More broadly, the field is converging toward a *neuro-symbolic* frontier (Marra et al., 2024; Zhao et al., 2026) that combines neural dynamics modules for learning transition functions (L1/L2) with symbolic components for constraint enforcement and hypothesis-space expansion (L3).

Across all four eras, representation learning serves as shared infrastructure: the quality of the learned state z_t determines the ceiling for prediction (L1), simulation (L2), and revision (L3) alike. Whether the representation is a latent vector, a discrete token sequence, a 3D point cloud, or a program, the governing-law regime determines which invariants the representation must preserve.

This historical arc suggests a consistent lesson: progress in world modeling has come not from scale alone, but from changing what is represented, what is compositional over horizon, and what can be revised from evidence. The open problems below are organized around remaining bottlenecks at L1, L2, and L3.

8.2 Open Problems by Capability Level

The preceding sections reveal a clear trajectory: world models are progressing from isolated one-step predictors toward integrated, agent-facing simulators that must respect domain-specific governing laws over extended horizons. Across all four regimes, this progression exposes a common pattern. In embodied domains, visual plausibility is outpacing physical faithfulness: models generate convincing video but violate conservation laws and object permanence under rollout, with the best systems achieving only 0.262 success rate on physical-consistency tests (Gu et al., 2025a; Li et al., 2025e). In social domains, large-scale agent simulations reproduce emergent phenomena such as opinion polarization and governance formation (Piao et al., 2025; Dai et al., 2024), but LLM agents exhibit systematic biases toward consensus that diverge from human behavioral patterns (Taubenfeld et al., 2024; Chuang et al., 2024). In code domains, agents treat software as deterministic state machines while real systems are partially observable, asynchronous, and multi-tenant (Kara et al., 2025). In scientific domains, neural surrogates trained on simulation data degrade when applied to real experimental measurements, exposing a surrogate-to-reality gap analogous to sim-to-real in robotics (Minami et al., 2025). The overarching theme is that the bottleneck has shifted from generating plausible futures to ensuring those futures are *decision-usable*: faithful to governing constraints, responsive to interventions, and calibrated against real-world evidence.

We organize ten concrete open problems by the capability level at which they most directly arise.

Representation and Local Prediction.

1. **Physical faithfulness beyond visual plausibility.** Current video and 3D world models achieve perceptual realism but fail physical-consistency tests: PhyWorldBench (Gu et al., 2025a) reports that the best of twelve frontier models attains only a 0.262 success rate on conservation-law and object-permanence probes, with long-horizon error accumulation as the core structural weakness. Closing this gap requires physically grounded representations that enforce constraints under counterfactual rollout, not merely pixel fidelity, where representation choices discussed in Section 2.2 may offer one possible direction. Spatially guided training strategies that inject geometric supervision into vision-language-action models (Ye et al., 2026a) offer one promising direction.

2. **Metric-aware video world modeling.** Extending geometry-grounded editing from image pairs to temporally coherent video demands four coupled abilities: metric estimation across time, temporal composition of short-step predictions, identity and appearance preservation across frames, and instruction grounding that aligns predicted motion with semantic specifications (Ho et al., 2022; Xing et al., 2024). Moving to video provides denser temporal supervision and stronger identity constraints than image-pair approaches. Subject-faithful controllable editing methods such as RealCustom++ (Mao et al., 2026a) may supply useful interface components. Evaluation must measure metric controllability directly, not just perceptual quality (Huang et al., 2024b; Ge et al., 2024).

3. **Programmable visual representation.** Current visual world models represent state as raw pixels or latent embeddings, neither of which is compositional or precisely editable. Code offers a structured alternative: VCode (Lin et al., 2025c) reconstructs images as SVG programs preserving symbolic semantics over pixel fidelity, Code2Video (Chen et al., 2025e) shows that executable Manim scripts outperform pixel-generation models on structured content by making every spatial and temporal element directly addressable, and VIGA (Yin et al., 2026) extends the paradigm to 3D by reconstructing scenes and simulating physical interactions through generated Blender code. The open problem is unifying these code-based representations into a single world-model interface for both 2D and 3D compositional editing.

Simulation Fidelity and Intervention.

4. **Partially observable software as a POMDP.** No existing code world model maintains belief distributions over hidden backend state (server sessions, database rows, in-flight requests, and background processes), nor reasons about asynchronous transitions with variable latency. Injecting realistic asynchronous failures into standard benchmarks causes significant drops in task success across all state-of-the-

art agents (Kara et al., 2025). Solving this requires temporal-belief architectures that jointly model what has happened, what is in progress, and what the agent cannot yet observe.

5. **Concurrent multi-user state.** Real software is multi-tenant; world models must predict state under a Dec-POMDP where concurrent users' actions are unobservable (Wang et al., 2024d). Conflict-free replicated data types (Kleppmann & Beresford, 2017) provide the formal substrate for merging concurrent updates, but no current world model integrates distributed-systems semantics with learned belief tracking over hidden users and pending writes. This problem sits at the intersection of the Digital World and Social World regimes, requiring joint reasoning about software state and multi-agent intent.

6. **Agent-human behavioral alignment at scale.** LLM agents exhibit systematic biases toward moderation and consensus (Taubenfeld et al., 2024; Chuang et al., 2024), producing two failure modes: mode collapse, where diverse simulated populations converge to homogeneous behavior, and calibration inadequacy, where single-turn persona alignment fails under multi-turn dynamics. Language and cultural priors can inject diversity, but this effect diminishes as cultural distance between populations shrinks. Systematic methods for grounding simulated behavior in real human behavioral distributions is lacking.

Evidence-Driven Revision and Self-Evolution. Existing autonomous science flagships such as CAMEO (Kusne et al., 2020) and A-Lab (Szymanski et al., 2023) demonstrate that closed-loop model revision is feasible in highly instrumented domains, while evaluator-guided algorithmic discovery systems such as FunSearch (Romera-Paredes et al., 2024) and AlphaEvolve (Novikov et al., 2025) demonstrate partial L3 loops with strong validation (Section 5). Several open problems must be solved before L3 generalizes.

7. **Continual learning of societal transition functions.** Large-scale simulations with 10,000+ agents across millions of interactions reproduce emergent phenomena such as opinion polarization and governance formation (Piao et al., 2025; Dai et al., 2024), yet cannot autonomously detect when social dynamics have shifted. The core challenge is to identify an outdated transition model, acquire corrective evidence, and revise without catastrophic forgetting of stable patterns (van de Ven et al., 2024). This problem connects to L3, where model revision must be triggered by distributional evidence rather than supervised labels.

8. **Closing the surrogate-to-reality gap.** Scientific surrogates validated on simulation data degrade on real measurements; prediction error decreases as a power law with computational data but plateaus without real-data calibration, mirroring the sim-to-real gap in robotics (Minami et al., 2025). In the scientific regime, this surrogate-to-reality gap is the direct analogue of sim-to-real transfer in robotics; the implementation-side mitigations discussed in Section 7.2 therefore provide a useful template, even though the measurement bottlenecks and evidence budgets differ. Notably, L2 scientific simulators such as GraphCast, NeuralGCM, and Aurora (Section 4) provide the prediction substrate on which L3 revision operates; their fidelity sets the ceiling for downstream evidence-driven diagnosis. The OPAL-surrogate framework (Singh et al., 2024) provides hierarchical Bayesian credibility gates that formalize when a surrogate is trustworthy. The central open question is how to allocate scarce real experimental observations optimally between model calibration and scientific discovery.

9. **Modeling laws that themselves evolve.** In biology, ecology, and climate, governing dynamics are non-stationary: viral fitness landscapes shift (Lassig et al., 2017), climate forcing alters atmospheric dynamics (Beucler et al., 2024), and evolutionary pressures create tipping points (Evangelou et al., 2024). World models must learn second-order meta-transition operators governing how p_θ itself drifts, together with revision triggers that detect law change from observational evidence. Causal discovery under non-stationarity (Huang et al., 2020; Song et al., 2023) provides identifiability results but treats change as variation within a fixed meta-model rather than as structural law replacement.

10. **Harness designs for agentic world modeling.** Agent performance has evolved through three successive abstractions: prompt engineering optimizes what the model is told, context engineering Anthropic (2025) curates the information state across turns, and harness engineering designs the executable environment surrounding the model: tools, memory, feedback loops, and inter-agent topology (Rajasekaran, 2026; Pan et al., 2026). This progression implies that agent behavior is governed not by the model alone but by transition dynamics of its execution environment, making harness design a form of world modeling

for software agents. The problem is how to learn and synthesize harnesses from interaction data, treating the execution environment itself as the object of modeling rather than a fixed engineering assumption.

Cross-regime shared challenges. Despite the diversity of governing-law regimes, three open problems recur across all four domains and constitute the deepest bottlenecks for world modeling in agentic AI. *Deployment shift*: world models trained on offline data or simulation systematically underperform when the environment drifts. UI layouts change, physical contact properties shift, social norms evolve, and scientific instruments recalibrate. Robust world modeling requires online mechanisms that detect distribution shift early and trigger targeted revision rather than waiting for catastrophic failure. *Constraint enforcement*: all four regimes have governing laws that valid trajectories must satisfy (contact stability, state-machine consistency, norm compliance, evidence-chain validity), yet current models enforce these constraints only softly through training objectives; hard enforcement at inference time, via symbolic layers, constrained rollout, or verification gates, remains an open architectural problem. *Persistent update governance*: L3 systems that revise themselves from evidence face a trilemma of stability (avoid regressing on past capabilities), plasticity (incorporate new evidence quickly), and auditability (trace every update to its evidence source); no current system resolves all three, and the governance infrastructure (versioning, canary deployment, rollback policies, and regression harnesses) is underspecified in most published architectures. The MREP framework proposed in Section 6 offers a starting point for standardizing evaluation across these shared challenges by providing version-locked, reproducible evaluation packages that make cross-regime comparison tractable.

8.3 Beyond L3

The L1→L2→L3 hierarchy (Sections 3–5) assumes that the world operates under a fixed set of governing laws: L1 learns local regularities, L2 composes them into constraint-consistent rollouts, and L3 revises the model when evidence contradicts predictions. At L3, the system can update the laws governing its world model, but these updates remain grounded in explaining a single underlying reality.

A natural extension is **meta-world modeling**: systems that reason not only about a particular transition function, but about the space of possible transition functions itself. Rather than refining a model of the observed world, such systems would explore alternative rule systems that define different possible environments, for example by varying, extending, or constructing new assumptions, constraints, or governing principles.

As discussed in Section 2.2, the ability to explicitly represent and manipulate such principles becomes increasingly important in this setting. In particular, symbolic representations may provide a more natural interface for meta-world modeling, as they allow governing rules to be directly modified, composed, and compared across alternative worlds.

What forms this capability might take, whether through program synthesis, open-ended evolution, procedural world generation, or other mechanisms, remains an open question. More broadly, this raises the question of whether the endpoint of world modeling lies in increasingly accurate models of a world, or in systems that can systematically explore and reason over multiple worlds defined by different governing laws. We expect different research communities to arrive at different formulations of this capability, depending on whether the focus lies on predictive performance, scientific understanding, or generative modeling, and we leave the question of what constitutes the ultimate world model as an open invitation to the broader community.

9 Conclusion

This paper has proposed a capability-based taxonomy for world modeling organized along two axes: three capability levels (Predictor, Simulator, Evolver) and four perspectives (physical, digital, social, scientific).

An **L1 Predictor** learns local transition operators $p_\theta(z_t \mid z_{t-1}, a_t)$ whose quality is measured by one-step calibration, robustness, and identifiability (Section 3). An **L2 Simulator** composes these operators into long-horizon, action-conditioned rollouts that must satisfy three boundary conditions (long-horizon coherence, intervention sensitivity, and constraint consistency) under the governing laws of the target domain (Section 4). An **L3 Evolver** closes the loop by autonomously designing experiments, collecting evidence, and revising its dynamics model when predictions fail (Section 5). Current systems such as CAMEO (Kusne

et al., 2020) and A-Lab (Szymanski et al., 2023) in autonomous science provide the strongest evidence that closed-loop model revision is already feasible in well-instrumented domains, while evaluator-guided algorithmic discovery systems such as FunSearch (Romera-Paredes et al., 2024) and AlphaEvolve (Novikov et al., 2025) show how automated scoring and regression gates can support partial L3 loops.

Organizing domains by **governing-law** regime rather than by modality has revealed both shared principles and irreducible differences (Section 4). *Physical*-world systems benefit from geometric and conservation-law priors; *digital*-world systems exploit deterministic program semantics; *social*-world systems demand Theory-of-Mind representations; and *scientific*-world systems couple models to experimental evidence streams. The L1→L2→L3 taxonomy applies uniformly across these regimes, but the content of each level (what constitutes a valid rollout, what counts as a law violation, what evidence is available) varies fundamentally.

To make capability claims testable, we proposed decision-centric evaluation principles and a minimal reproducible evaluation package (Section 6), and provided an architectural roadmap mapping representation, dynamics, and control choices to each capability level and deployment regime (Section 7). The open problems in Section 8 trace a research agenda: causal representation learning at L1, law-consistent rollout and compositional generalization at L2, and safe autonomous experiment design and model revision at L3.

A cross-cutting theme emerging from this survey is the question of representation substrate. The L1→L2→L3 progression describes what a world model can do, but leaves open what form it should take. Latent continuous representations have proven indispensable for learning transition operators at scale, yet the history of scientific discovery suggests that law revision, the hallmark of L3, has typically relied on symbolic substrates: Newton's laws, Maxwell's equations, and the Standard Model are all world models whose governing principles are explicit, composable, and directly revisable. Current neural world models encode invariances implicitly through architecture and training, which suits L1 and L2 but becomes a liability at L3, where the task is to revise model structure itself. We therefore view the development of world models that can discover and manipulate symbolic governing laws from data, rather than merely absorbing them into latent representations, as one of the most important open problems in the field. How to extend this to physical, digital, social, and scientific worlds remains a fundamental challenge.

Key Takeaway

The future of agentic AI lies not in larger predictors, but in models that internalize the governing laws of the world, simulate its dynamics, and continuously evolve themselves through active trial-and-error loops, enabling them to navigate, interpret, and ultimately reshape the world.

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630:493–500, 2024.

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C. Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Pulkit Agrawal, Ashvin V. Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in Neural Information Processing Systems*, volume 29, pp. 5092–5100, 2016.

Altera AL, Andrew Ahn, Nic Becker, Stephanie Carroll, Nico Christie, Manuel Cortes, Arda Demirci, Melissa Du, Frankie Li, Shuying Luo, Peter Y Wang, Mathew Willows, Feitong Yang, and Guangyu Robert Yang. Project Sid: Many-agent simulations toward AI civilization. *arXiv preprint arXiv:2411.00114*, 2024.

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3988–3996, 2016.

Christof Angermueller, David Belanger, Andreea Gane, Zelda Mariet, David Dohan, Kevin Murphy, Lucy Colwell, and D Sculley. Population-based black-box optimization for biological sequence design. In *International Conference on Machine Learning*, pp. 324–334. PMLR, 2020.

Anthropic. Effective context engineering for AI agents. Anthropic Engineering Blog, 2025. URL `https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents`.

Lisa P. Argyle, Ethan C. Busby, Nancy Fulda, Joshua R. Gubler, Christopher Rytting, and David Wingate. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351, 2023.

V. Arunkumar, G. R. Gangadharan, and Rajkumar Buyya. Agentic artificial intelligence (AI): Architectures, taxonomies, and evaluation of large language model agents. *arXiv preprint arXiv:2601.12560*, 2026.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba, Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *International Conference on Learning Representations*, 2018.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N. Kinch, R. Dustin Schaeffer, Claudia Millán, Hahnbeom Park, Carson Adams, Caleb R. Glassman, Andy DeGiovanni, Jose H. Pereira, Andria V. Rodrigues, Alberdina A. van Dijk, Ana C. Ebrecht, Diederik J. Opperman, Theo Sagmeister, Christoph Buhlheller, Tea Pavkov-Keller, Manoj K. Rathinaswamy, Udit Dalwadi, Calvin K. Yip, John E. Burke, K. Christopher Garcia, Nick V. Grishin, Paul D. Adams, Randy J. Read, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

Adam P Baker, Matthew J Brookes, Iead A Rezek, Stephen M Smith, Timothy Behrens, Penny J Probert Smith, and Mark Woolrich. Fast transient networks in spontaneous human brain activity. *eLife*, 3:e01867, 2014.

Bowen Baker, Ilge Akkaya, Peter Zhokhov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video PreTraining (VPT): Learning to act by watching unlabeled online videos. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24639–24654, 2022.

Chris Baker, Rebecca Saxe, and Joshua Tenenbaum. Bayesian theory of mind: Modeling joint belief-desire attribution. In *Annual Meeting of the Cognitive Science Society*, volume 33, 2011.

Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, Mojtaba Komeili, Karthik Konath, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 378(6624): 1067–1074, 2022.

Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models, 2025. URL `https://deepmind.google/discover/blog/genie-3-a-new-frontier-for-world-models/`.

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, Yuanzhen Li, Michael Rubinstein, Tomer Michaeli, Oliver Wang, Deqing Sun, Tali Dekel, and Inbar Mosseri. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia*, pp. 1–11, 2024.

Leonardo Barcellona, Andrii Zadaianchuk, Davide Allegro, Samuele Papa, Stefano Ghidoni, and Efstratios Gavves. Dream to manipulate: Compositional world models empowering robot imitation learning with imagination. *arXiv preprint arXiv:2412.14957*, 2024.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael G. Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical Review Letters*, 98(14):146401, 2007.

Tom Beucler, Pierre Gentine, Janni Yuval, Ankitesh Gupta, Liran Peng, Jerry Lin, Sungduk Yu, Stephan Rasp, Fiaz Ahmed, Paul A. O'Gorman, J. David Neelin, Nicholas J. Lutsko, and Michael Pritchard. Climate-invariant machine learning. *Science Advances*, 10(6):eadj7250, 2024.

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3D neural networks. *Nature*, 619:533–538, 2023.

Hengwei Bian, Lingdong Kong, Haozhe Xie, Liang Pan, Yu Qiao, and Ziwei Liu. DynamicCity: Large-scale 4D occupancy generation from dynamic scenes. In *International Conference on Learning Representations*, 2025.

Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Anna Allen, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan A Weyn, Haiyu Dong, Jayesh K Gupta, Kit Thambiratnam, Alexander T Archibald, Chun-Chieh Wu, Elizabeth Heider, Max Welling, Richard E Turner, and Paris Perdikaris. A foundation model for the earth system. *Nature*, 641(8065):1180–1187, 2025.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. Technical report, OpenAI, 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *International Conference on Machine Learning*, pp. 4603–4623, 2024.

Natasha Butt, Blazej Manczak, Auke Wiggers, Corrado Rainone, David W. Zhang, Michaël Defferrard, and Taco Cohen. CodeIt: Self-improving language models with prioritized hindsight replay. In *International Conference on Machine Learning*, pp. 5013–5034. PMLR, 2024.

Pengfei Cao, Tianyi Men, Wencan Liu, Jingwen Zhang, Xuzhao Li, Xixun Lin, Dianbo Sui, Yanan Cao, Kang Liu, and Jun Zhao. Large language models for planning: A comprehensive and systematic survey. *arXiv preprint arXiv:2505.19683*, 2025.

Yilin Cao, Yufeng Zhong, Zhixiong Zeng, Liming Zheng, Jing Huang, Haibo Qiu, Peng Shi, Wenji Mao, and Guanglu Wan. MobileDreamer: Generative sketch world model for GUI agent. *arXiv preprint arXiv:2601.04035*, 2026.

Hyungjoo Chae, Namyoung Kim, Kai Tzu-iunn Ong, Minju Gwak, Gwanwoo Song, Jihoon Kim, Sunghwan Kim, Dongha Lee, and Jinyoung Yeo. Web agents with world models: Learning and leveraging environment dynamics in web navigation. In *International Conference on Learning Representations*, 2025.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. GameGen-X: Interactive open-world game video generation. *arXiv preprint arXiv:2411.00769*, 2024.

Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. TransDreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*, 2022.

Delong Chen, Mustafa Shukor, Theo Moutakanni, Willy Chung, Jade Yu, Tejaswi Kasarla, Yejin Bang, Allen Bolourchi, Yann LeCun, and Pascale Fung. VL-JEPA: Joint embedding predictive architecture for vision-language. *arXiv preprint arXiv:2512.10942*, 2025a.

Junwu Chen, Xu Huang, Cheng Hua, Yulian He, and Philippe Schwaller. Adsgt: Graph transformer for predicting global minimum adsorption energy. In *NeurIPS 2023 AI for Science Workshop*, 2023.

Junwu Chen, Xu Huang, Cheng Hua, Yulian He, and Philippe Schwaller. A multi-modal transformer for predicting global minimum adsorption energy. *Nature Communications*, 16(1):3232, 2025b.

Ruirui Chen, Weifeng Jiang, Chengwei Qin, and Cheston Tan. Theory of mind in large language models: Assessment and enhancement. In *Annual Meeting of the Association for Computational Linguistics*, pp. 31539–31558, 2025c.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Xinyi Chen, Yilun Chen, Yanwei Fu, Ning Gao, Jiaya Jia, Weiyang Jin, Hao Li, Yao Mu, Jiangmiao Pang, Yu Qiao, Yang Tian, Bin Wang, Bolun Wang, Fangjing Wang, Hanqing Wang, Tai Wang, Ziqin Wang, Xueyuan Wei, Chao Wu, Shuai Yang, Jinhui Ye, Junqiu Yu, Jia Zeng, Jingjing Zhang, Jinyu Zhang, Shi Zhang, Feng Zheng, Bowen Zhou, and Yangkun Zhu. InternVLA-M1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025d.

Yanzhe Chen, Kevin Qinghong Lin, and Mike Zheng Shou. Code2Video: A code-centric paradigm for educational video generation. *arXiv preprint arXiv:2510.01174*, 2025e.

Yixiang Chen, Peiyan Li, Jiabing Yang, Keji He, Xiangnan Wu, Yuan Xu, Kai Wang, Jing Liu, Nianfeng Liu, Yan Huang, and Liang Wang. BridgeV2W: Bridging video generation models to embodied world models via embodiment masks. *arXiv preprint arXiv:2602.03793*, 2026.

Zhaorun Chen, Zhuokai Zhao, Kai Zhang, Bo Liu, Qi Qi, Yifan Wu, Tarun Kalluri, Sara Cao, Yuanhao Xiong, Haibo Tong, Huaxiu Yao, Hengduo Li, Jiacheng Zhu, Xian Li, Dawn Song, Bo Li, Jason Weston, and Dat Huynh. Scaling agent learning via experience synthesis. *arXiv preprint arXiv:2511.03773*, 2025f.

Sathya R Chitturi, Akash Ramdas, Yue Wu, Brian Rohr, Stefano Ermon, Jennifer Dionne, Felipe H da Jornada, Mike Dunne, Christopher Tassone, Willie Neiswanger, and Daniel Ratner. Targeted materials discovery using bayesian algorithm execution. *NPJ Computational Materials*, 10(1):156, 2024.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, pp. 4759–4770, 2018.

Yun-Shiuan Chuang, Agam Goyal, Nikunj Harlalka, Siddharth Suresh, Robert Hawkins, Sijia Yang, Dhavan Shah, Junjie Hu, and Timothy Rogers. Simulating opinion dynamics with networks of llm-based agents. In *Findings of the association for computational linguistics: NAACL 2024*, pp. 3326–3346, 2024.

Andy Clark. *Surfing Uncertainty: Prediction, Action, and the Embodied Mind*. Oxford University Press, 2015.

Jade Copet, Quentin Carbonneaux, Gal Cohen, Jonas Gehring, Jacob Kahn, Jannik Kossen, Felix Kreuk, Emily McMilin, Michel Meyer, Yuxiang Wei, David Zhang, Kunhao Zheng, Jordi Armengol-Estapé, Pedram Bashiri, Maximilian Beck, et al. CWM: An open-weights LLM for research on code generation with world models. *arXiv preprint arXiv:2510.02387*, 2025.

Anthony Coutant, Katherine Roper, Daniel Trejo-Banos, Dominique Bouthinon, Martin Carpenter, Jacek Grzebyta, Guillaume Santini, Henry Soldano, Mohamed Elati, Jan Ramon, Céline Rouveirol, Larisa N. Soldatova, and Ross D. King. Closed-loop cycles of experiment design, execution, and learning accelerate systems biology model development in yeast. *Proceedings of the National Academy of Sciences*, 116(36): 18142–18147, 2019.

Kenneth James William Craik. *The Nature of Explanation*. Cambridge University Press, 1943.

Gordon Dai, Weijia Zhang, Jinhan Li, Siqi Yang, Chidera Onochie lbe, Srihas Rao, Arthur Caetano, and Misha Sra. Artificial leviathan: Exploring social evolution of llm agents through the lens of hobbesian social contract theory. *arXiv preprint arXiv:2406.14373*, 2024.

Nicola Dainese, Matteo Merler, Minttu Alakuijala, and Pekka Marttinen. Generating code world models with large language models guided by Monte Carlo tree search. In *Advances in Neural Information Processing Systems*, volume 37, pp. 60429–60474, 2024.

Adam C. Dama, Kevin S. Kim, Danielle M. Leyva, Annamarie P. Lunkes, Noah S. Schmid, Kenan Jijakli, and Paul A. Jensen. BacterAI maps microbial metabolism without prior knowledge. *Nature Microbiology*, 8:1018–1025, 2023.

Decart, Julian Quevedo, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A universe in a transformer. Blog post, 2024. URL `https://oasis-model.github.io`.

Judith Degen. The rational speech act framework. *Annual Review of Linguistics*, 9(1):519–540, 2023.

Marc Peter Deisenroth and Carl Edward Rasmussen. PILCO: A model-based and data-efficient approach to policy search. In *International Conference on Machine Learning*, pp. 465–472, 2011.

Bowen Deng, Peichen Zhong, KyuJung Jun, Janosh Riebesell, Kevin Han, Christopher J Bartel, and Gerbrand Ceder. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

Fei Deng, Ingook Jang, and Sungjin Ahn. DreamerPro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pp. 4956–4975. PMLR, 2022.

Jingtao Ding, Yunke Zhang, Yu Shang, Jie Feng, Yuheng Zhang, Zefang Zong, Yuan Yuan, Hongyuan Su, Nian Li, Jinghua Piao, Yucheng Deng, Nicholas Sukiennik, Chen Gao, Fengli Xu, and Yong Li. Understanding world or predicting future? A comprehensive survey of world models. *ACM Computing Surveys*, 2025.

Yifei Dong, Fengyi Wu, Guangyu Chen, Lingdong Kong, Xu Zhu, Qiyu Hu, Yuxuan Zhou, Jingdong Sun, Jun-Yan He, Qi Dai, Alexander G. Hauptmann, and Zhi-Qi Cheng. Towards unified world models for visual navigation via memory-augmented planning and foresight. *arXiv preprint arXiv:2510.08713*, 2025.

Yifei Dong, Fengyi Wu, Yilong Dai, Lingdong Kong, Guangyu Chen, Xu Zhu, Qiyu Hu, Tianyu Wang, Johnalbert Garnica, Feng Liu, Siyu Huang, Qi Dai, and Zhi-Qi Cheng. Language-conditioned world modeling for visual navigation. *arXiv preprint arXiv:2603.26741*, 2026.

Pierre Duhem. *The Aim and Structure of Physical Theory*. Princeton University Press, 1954.

Albert Einstein. Physics and reality. *Journal of the Franklin Institute*, 221(3):349–382, 1936.

Nikolaos Evangelou, Tianqi Cui, Juan M Bello-Rivas, Alexei Makeev, and Ioannis G Kevrekidis. Tipping points of evolving epidemiological networks: Machine learning-assisted, data-driven effective modeling. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 34(6), 2024.

Wenkai Fan, Shurui Zhang, Xiaolong Wang, Haowei Yang, Tsz Wai Chan, Xingyan Chen, Junquan Bi, Zirui Zhou, Jia Liu, and Kani Chen. AIvilization v0: Toward large-scale artificial social simulation with a unified agent architecture and adaptive agent profiles. *arXiv preprint arXiv:2602.10429*, 2026.

Tianqing Fang, Hongming Zhang, Zhisong Zhang, Kaixin Ma, Wenhao Yu, Haitao Mi, and Dong Yu. WebEvolver: Enhancing web agent self-improvement with coevolving world model. In *Conference on Empirical Methods in Natural Language Processing*, pp. 8959–8975, 2025.

Yuxing Fei, Bernardus Rendy, Xiaochen Yang, Junhee Woo, Xu Huang, Chang Li, Shilong Wang, David Milsted, Yan Zeng, and Gerbrand Ceder. Agentic llm reasoning in a self-driving laboratory for air-sensitive lithium halide spinel conductors. *arXiv preprint arXiv:2604.11957*, 2026.

Jichen Feng, Yifan Zhang, Chenggong Zhang, Yifu Lu, Shilong Liu, and Mengdi Wang. Web world models. *arXiv preprint arXiv:2512.23676*, 2025a.

Tuo Feng, Wenguan Wang, and Yi Yang. A survey of world models for autonomous driving. *arXiv preprint arXiv:2501.11260*, 2025b.

Richard E. Fikes and Nils J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3–4):189–208, 1971.

Sara Fish, Paul Gölz, David C. Parkes, Ariel D. Procaccia, Gili Rusak, Itai Shapira, and Manuel Wüthrich. Generative social choice. In *ACM Conference on Economics and Computation*, pp. 985–985, 2024.

C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax – a differentiable physics engine for large scale rigid body simulation. In *Advances in Neural Information Processing Systems*, 2021.

Karl Friston. The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2): 127–138, 2010.

Giridhar Ganapavarapu and Dhaval Patel. MCP-Cosmos: World model-augmented agents for complex task execution in MCP environments. *arXiv preprint arXiv:2605.09131*, 2026.

Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. In *Advances in Neural Information Processing Systems*, volume 37, pp. 91560–91596, 2024.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang, Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K. R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi Fan. DreamDojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

Yifei Gao, Junhong Ye, Jiaqi Wang, and Jitao Sang. WebSynthesis: World-model-guided MCTS for efficient WebUI-trajectory synthesis. *arXiv preprint arXiv:2507.04370*, 2025.

Songwei Ge, Aniruddha Mahapatra, Gaurav Parmar, Jun-Yan Zhu, and Jia-Bin Huang. On the content bias in Fréchet video distance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7277–7288, 2024.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2018.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G. Bellemare. DeepMDP: Learning continuous latent space models for representation learning. In *International Conference on Machine Learning*, pp. 2170–2179. PMLR, 2019.

Genesis Authors. Genesis: A generative and universal physics engine for robotics and beyond. GitHub Repository, 2024. URL `https://github.com/Genesis-Embodied-AI/Genesis`.

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning. In *International Conference on Learning Representations*, 2021.

Raj Ghugare, Roger Creus Castanyer, Catherine Ji, Kathryn Wantlin, Jin Schofield, Karthik Narasimhan, and Benjamin Eysenbach. Builderbench: The building blocks of intelligent agents. *arXiv preprint arXiv:2510.06288*, 2025.

Chetan Gohil, Evan Roberts, Ryan Timms, Alex Skates, Cameron Higgins, Andrew Quinn, Usama Pervaiz, Joost van Amersfoort, Pascal Notin, Yarin Gal, Stanislaw Adaszewski, and Mark Woolrich. Mixtures of large-scale dynamic functional brain network modes. *NeuroImage*, 263:119595, 2022. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2022.119595.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*, volume 1. MIT Press, 2016.

Noah D Goodman and Michael C Frank. Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, 20(11):818–829, 2016.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, Khaled Saab, Dan Popovici, Jacob Blum, Fan Zhang, Katherine Chou, Avinatan Hassidim, Burak Gokturk, Amin Vahdat, Pushmeet Kohli, Yossi Matias, Andrew Carroll, Kavita Kulkarni, Nenad Tomasev, Yuan Guan, Vikram Dhillon, Eeshit Dhaval Vaishnav, Byron Lee, Tiago R. D. Costa, José R. Penadés, Gary Peltz, Yunhan Xu, Annalisa Pawlosky, Alan Karthikesalingam, and Vivek Natarajan. Towards an AI co-scientist. *arXiv preprint arXiv:2502.18864*, 2025.

Jing Gu, Xian Liu, Yu Zeng, Ashwin Nagarajan, Fangrui Zhu, Daniel Hong, Yue Fan, Qianqi Yan, Kaiwen Zhou, Ming-Yu Liu, and Xin Eric Wang. PhyWorldBench: A comprehensive evaluation of physical realism in text-to-video models. *arXiv preprint arXiv:2507.13428*, 2025a.

Yu Gu, Kai Zhang, Yuting Ning, Boyuan Zheng, Boyu Gou, Tianci Xue, Cheng Chang, Sanjari Srivastava, Yanan Xie, Peng Qi, Huan Sun, and Yu Su. Is your LLM secretly a world model of the internet? Model-based planning for web agents. *Transactions on Machine Learning Research*, 2025b.

Yiming Guan, Rui Yu, John Zhang, Lu Wang, Chaoyun Zhang, Liqun Li, Bo Qiao, Si Qin, He Huang, Fangkai Yang, Pu Zhao, Lukas Wutschitz, Samuel Kessler, Huseyin A. Inan, Robert Sim, Saravan Rajmohan, Qingwei Lin, and Dongmei Zhang. Computer-using world model. *arXiv preprint arXiv:2602.17365*, 2026.

Junliang Guo, Yang Ye, Tianyu He, Haoyu Wu, Yushu Jiang, Tim Pearce, and Jiang Bian. MineWorld: a real-time and open-source interactive world model on minecraft. *arXiv preprint arXiv:2504.08388*, 2025a.

Shangmin Guo, Omar Darwiche Domingues, Raphaël Avalos, Aaron Courville, and Florian Strub. World modelling improves language model agents. *arXiv preprint arXiv:2506.02918*, 2025b.

Weiyu Guo, Ziyang Chen, Shaoguang Wang, Jianxiang He, Yijie Xu, Jinhui Ye, Ying Sun, and Hui Xiong. Logic-in-frames: Dynamic keyframe search via visual semantic-logical verification for long video understanding. *arXiv preprint arXiv:2503.13139*, 2025c.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, volume 31, pp. 2455–2467, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. In *International Conference on Learning Representations*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *International Conference on Machine Learning*, pp. 8387–8406. PMLR, 2022.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024.

Chenjie Hao, Weyl Lu, Yifan Xu, and Yubei Chen. Neural motion simulator: Pushing the limit of world models in reinforcement learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27608–27617, 2025.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Conference on Empirical Methods in Natural Language Processing*, pp. 8154–8173, 2023.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *AAAI Conference on Artificial Intelligence*, volume 32, pp. 3207–3214, 2018.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8633–8646, 2022.

Sara Hooker. The hardware lottery. *Communications of the ACM*, 64(12):58–65, 2021.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Yuhang Hu, Boyuan Chen, and Hod Lipson. Egocentric visual self-modeling for autonomous robot dynamics prediction and adaptation. *NPJ Robotics*, 3:14, 2025.

Biwei Huang, Kun Zhang, Jiji Zhang, Joseph Ramsey, Ruben Sanchez-Romero, Clark Glymour, and Bernhard Schölkopf. Causal discovery from heterogeneous/nonstationary data. *Journal of Machine Learning Research*, 21(89):1–53, 2020.

Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, Ryan Li, Lin Qiu, Gavin Li, Junze Zhang, Di Yin, Shruti Marwaha, Jennefer N. Carter, Xin Zhou, Matthew Wheeler, Jonathan A. Bernstein, Mengdi Wang, Peng He, Jingtian Zhou, Michael Snyder, Le Cong, Aviv Regev, and Jure Leskovec. Biomni: A general-purpose biomedical AI agent. bioRxiv 2025.05.30.656746, 2025a.

Renhong Huang, Ning Tang, Jiarong Xu, Yuxuan Cao, Qingqian Tu, Sheng Guo, Bo Zheng, Huiyuan Liu, and Yang Yang. PolicySim: An LLM-based agent social simulation sandbox for proactive policy optimization. *arXiv preprint arXiv:2603.19649*, 2026a.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. PointWorld: Scaling 3D world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026b.

Xu Huang, Junwu Chen, Philippe Schwaller, and Gerbrand Ceder. Skillpuzzler: A self-evolving agentic framework for materials and chemistry research with minimal reliance on predefined tools. In *NeurIPS 2025 AI for Science Workshop*.

Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. Understanding the planning of LLM agents: A survey. *arXiv preprint arXiv:2402.02716*, 2024a.

Xu Huang, Junwu Chen, Yuxing Fei, Zhuohan Li, Philippe Schwaller, and Gerbrand Ceder. Cascade: Cumulative agentic skill creation through autonomous development and evolution. *arXiv preprint arXiv:2512.23880*, 2025b.

Xu Huang, Bowen Deng, Peichen Zhong, Aaron D Kaplan, Kristin A Persson, and Gerbrand Ceder. Cross-functional transferability in foundation machine learning interatomic potentials. *npj Computational Materials*, 11(1):313, 2025c.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025d.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024b.

Ziqi Huang, Ning Yu, Gordon Chen, Haonan Qiu, Paul Debevec, and Ziwei Liu. VChain: Chain-of-visual-thought for reasoning in video generation. *arXiv preprint arXiv:2510.05094*, 2025e.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3):3268–3285, 2025f.

David Hume. *A Treatise of Human Nature*. Clarendon Press, Oxford, 1739.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, volume 32, pp. 12519–12530, 2019.

Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, pp. 9902–9915. PMLR, 2022.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-Bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, 2024.

Ruofan Jin, Yucheng Guo, Yuanhao Qu, Ming Yang, Chun Shang, Qirong Yang, Linlin Chao, Yi Zhou, Ruilai Xu, Ziyao Xu, Ruhong Zhou, Zaixi Zhang, Mengdi Wang, Xiaoming Zhang, and Le Cong. BioLab: End-to-end autonomous life sciences research with multi-agents system integrating biological foundation models. bioRxiv 2025.09.03.674085, 2025.

Philip N. Johnson-Laird. *Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness*. Harvard University Press, 1983.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596:583–589, 2021.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H. Campbell, K. Czechowski, D. Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Afroz Mohiuddin, Ryan Sepassi, G. Tucker, and Henryk Michalewski. Model-based reinforcement learning for Atari. In *International Conference on Learning Representations*, 2020.

Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. In *International Conference on Machine Learning*, pp. 28991–29017. PMLR, 2025a.

Zhaolu Kang, Junhao Gong, Jiaxu Yan, Wanke Xia, Yian Wang, Ziwen Wang, Huaxuan Ding, Zhuo Cheng, Wenhao Cao, Zhiyuan Feng, Siqi He, Shannan Yan, Junzhe Chen, Xiaomin He, Chaoya Jiang, Wei Ye, Kaidong Yu, and Xuelong Li. HSSBench: Benchmarking humanities and social sciences ability for multimodal large language models. *arXiv preprint arXiv:2506.03922*, 2025b.

Su Kara, Fazle E. Faisal, and Suman Nath. WAREX: Web agent reliability evaluation on existing benchmarks. *arXiv preprint arXiv:2510.03285*, 2025.

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

Lord Kelvin. I. nineteenth-century clouds over the dynamical theory of heat and light. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(7):1–40, 1901.

Abu Mohammad Shabbir Khan, Chetan Gohil, Pascal Notin, Joost van Amersfoort, Mark Woolrich, and Yarin Gal. DyNeMoC: A semi-supervised architecture for classifying time series brain data. In *International Conference on Learning Representations Workshop*, 2023.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Le Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. FANToM: A benchmark for stress-testing machine theory of mind in interactions. In *Conference on Empirical Methods in Natural Language Processing*, pp. 14397–14413, 2023.

Seung Wook Kim, Yuhao Zhou, Jonah Philion, Antonio Torralba, and Sanja Fidler. Learning to simulate dynamic environments with GameGAN. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1231–1240, 2020.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Martin Kleppmann and Alastair R. Beresford. A conflict-free replicated JSON datatype. *IEEE Transactions on Parallel and Distributed Systems*, 28(10):2733–2746, 2017.

Dmitrii Kochkov, Janni Yuval, Ian Langmore, Peter Norgaard, Jamie Smith, Griffin Mooers, Milan Klöwer, James Lottes, Stephan Rasp, Peter Düben, Sam Hatfield, Peter Battaglia, Alvaro Sanchez-Gonzalez, Matthew Willson, Michael P Brenner, and Stephan Hoyer. Neural general circulation models for weather and climate. *Nature*, 632:1060–1066, 2024.

Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. PathDreamer: A world model for indoor navigation. In *IEEE/CVF International Conference on Computer Vision*, pp. 14738–14748, 2021.

Woosung Koh, Sungjun Han, Segyu Lee, Se-Young Yun, and Jamin Shin. Generative visual code mobile world models. *arXiv preprint arXiv:2602.01576*, 2026.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, Jose Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Joshua V. Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan Yang, Irfan Essa, Huisheng Wang, David A. Ross, Bryan Seybold, and Lu Jiang. VideoPoet: A large language model for zero-shot video generation. In *International Conference on Machine Learning*, pp. 25105–25124. PMLR, 2024.

Lingdong Kong, Wesley Yang, Jianbiao Mei, Youquan Liu, Ao Liang, Dekai Zhu, Dongyue Lu, Wei Yin, Xiaotao Hu, Mingkai Jia, Junyuan Deng, Kaiwen Zhang, Yang Wu, Tianyi Yan, Shenyuan Gao, Song Wang, Linfeng Li, Liang Pan, Yong Liu, Jianke Zhu, Wei Tsang Ooi, Steven C. H. Hoi, and Ziwei Liu. 3D and 4D world modeling: A survey. *arXiv preprint arXiv:2509.07996*, 2025.

Ksenia Korovina, Sailun Xu, Kirthevasan Kandasamy, Willie Neiswanger, Barnabas Poczos, Jeff Schneider, and Eric Xing. ChemBO: Bayesian optimization of small organic molecules with synthesizable recommendations. In *International Conference on Artificial Intelligence and Statistics*, pp. 3393–3403, 2020.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25, pp. 1097–1105, 2012.

Thomas S. Kuhn. *The Structure of Scientific Revolutions*. University of Chicago Press, 4th edition, 1962.

Ujwal Kumar, Alice Saito, Hershraj Niranjani, Rayan Yessou, and Phan Xuan Tan. Evolving interpretable constitutions for multi-agent coordination. *arXiv preprint arXiv:2602.00755*, 2026.

A. Gilad Kusne, Heshan Yu, Changming Wu, Huairuo Zhang, Jason Hattrick-Simpers, Brian DeCost, Suchismita Sarker, Corey Oses, Cormac Toher, Stefano Curtarolo, Albert V. Davydov, Ritesh Agarwal, Leonid A. Bendersky, Mo Li, Apurva Mehta, and Ichiro Takeuchi. On-the-fly closed-loop materials discovery via Bayesian active learning. *Nature Communications*, 11(1):5966, 2020.

Imre Lakatos. *The Methodology of Scientific Research Programmes*, volume 1 of *Philosophical Papers*. Cambridge University Press, 1978.

Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023.

Pierre-Simon Laplace. *Essai philosophique sur les probabilités*. Courcier, Paris, 1814.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 5639–5650. PMLR, 2020.

Michael Lassig, Ville Mustonen, and Aleksandra M. Walczak. Predicting evolution. *Nature Ecology & Evolution*, 1(3):0077, 2017.

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview preprint*, 2022. URL `https://openreview.net/pdf?id=BZ5a1r-kVsf`.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Wolfgang Lehrach, Daniel Hennes, Miguel Lazaro-Gredilla, Xinghua Lou, Carter Wendelken, Zun Li, Antoine Dedieu, Jordi Grau-Moya, Marc Lanctot, Atil Iscen, John Schultz, Marcus Chiam, Ian Gemp, Piotr Zielinski, Satinder Singh, and Kevin P. Murphy. Code world models for general game playing. *arXiv preprint arXiv:2510.04542*, 2025.

Douglas B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.

Timothée Lesort, Natalia Diaz-Rodríguez, Jean-François Goudou, and David Filliat. State representation learning for control: An overview. *Neural Networks*, 108:379–392, 2018.

David Lewis. *Counterfactuals*. Harvard University Press, 1973.

Mike Lewis, Denis Yarats, Yann Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning of negotiation dialogues. In *Conference on Empirical Methods in Natural Language Processing*, pp. 2443–2453, 2017.

Dacheng Li, Yunhao Fang, Yukang Chen, Shuo Yang, Shiyi Cao, Justin Wong, Michael Luo, Xiaolong Wang, Hongxu Yin, Joseph E. Gonzalez, Ion Stoica, Song Han, and Yao Lu. WorldModelBench: Judging video generation models as world models. *arXiv preprint arXiv:2502.20694*, 2025a.

Jiaqi Li, Junshu Tang, Zhiyong Xu, Longhuang Wu, Yuan Zhou, Shuai Shao, Tianbao Yu, Zhiguo Cao, and Qinglin Lu. Hunyuan-GameCraft: High-dynamic interactive game video generation with hybrid history condition. *arXiv preprint arXiv:2506.17201*, 2025b.

Ruochen Li, Liqiang Jing, Chi Han, Jiawei Zhou, and Xinya Du. Learning to generate research idea with dynamic control. In *Advances in Neural Information Processing Systems Workshop*, 2024a.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, Kazuki Kozuka, and Aditya Grover. MobileWorldBench: Towards semantic world modeling for mobile agents. *arXiv preprint arXiv:2512.14014*, 2025c.

Wenxuan Li, Hang Zhao, Zhiyuan Yu, Yu Du, Qin Zou, Ruizhen Hu, and Kai Xu. PIN-WM: Learning physics-informed world models for non-prehensile manipulation. *arXiv preprint arXiv:2504.16693*, 2025d.

Xinqing Li, Xin He, Le Zhang, Min Wu, Xiaoli Li, and Yun Liu. A comprehensive survey on world models for embodied AI. *arXiv preprint arXiv:2510.16732*, 2025e.

Xuanyi Li, Daquan Zhou, Chenxu Zhang, Shaodong Wei, Qibin Hou, and Ming-Ming Cheng. Sora generates videos with stunning geometrical consistency. *arXiv preprint arXiv:2402.17403*, 2024b.

Yixia Li, Hongru Wang, Jiahao Qiu, Zhenfei Yin, Dongdong Zhang, Cheng Qian, Zeping Li, Pony Ma, Guanhua Chen, and Heng Ji. From word to world: Can large language models be implicit text-based world models? *arXiv preprint arXiv:2512.18832*, 2025f.

Yuetai Li, Huseyin A. Inan, Xiang Yue, Wei-Ning Chen, Lukas Wutschitz, Janardhan Kulkarni, Radha Poovendran, Robert Sim, and Saravan Rajmohan. Simulating environments with reasoning models for agent training. *arXiv preprint arXiv:2511.01824*, 2025g.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

Shuquan Lian, Yuhang Wu, Jia Ma, Yifan Ding, Zihan Song, Bingqi Chen, Xiawu Zheng, and Hui Li. UI-AGILE: Advancing GUI agents with effective reinforcement learning and precise inference-time grounding. *arXiv preprint arXiv:2507.22025*, 2025.

Ao Liang, Lingdong Kong, Tianyi Yan, Hongsi Liu, Wesley Yang, Ziqi Huang, Wei Yin, Jialong Zuo, Yixuan Hu, Dekai Zhu, Dongyue Lu, Youquan Liu, Guangfeng Jiang, Linfeng Li, Xiangtai Li, Long Zhuo, Lai Xing Ng, Benoit R. Cottereau, Changxin Gao, Liang Pan, Wei Tsang Ooi, and Ziwei Liu. WorldLens: Full-spectrum evaluations of driving world models in real world. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36385–36399, 2026a.

Ao Liang, Youquan Liu, Yu Yang, Dongyue Lu, Linfeng Li, Lingdong Kong, Huaici Zhao, and Wei Tsang Ooi. LiDARCrafter: Dynamic 4D world modeling from LiDAR sequences. In *AAAI Conference on Artificial Intelligence*, volume 40, pp. 18406–18414, 2026b.

Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. AvalonBench: Evaluating llms playing the game of avalon. *arXiv preprint arXiv:2310.05036*, 2023.

James Lighthill. Artificial intelligence: A general survey. Technical report, Science Research Council, 1973.

Kevin Qinghong Lin and Mike Zheng Shou. Vlog: Video-language models by generative retrieval of narration vocabulary. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3218–3228, 2025.

Kevin Qinghong Lin, Siyuan Hu, Linjie Li, Zhengyuan Yang, Lijuan Wang, Philip Torr, and Mike Zheng Shou. Computer-use agents as judges for generative user interface. *arXiv preprint arXiv:2511.15567*, 2025a.

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Weixian Lei, Lijuan Wang, and Mike Zheng Shou. ShowUI: One vision-language-action model for GUI visual agent. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19498–19508, 2025b.

Kevin Qinghong Lin, Yuhao Zheng, Hangyu Ran, Dantong Zhu, Dongxing Mao, Linjie Li, Philip Torr, and Alex Jinpeng Wang. VCode: a multimodal coding benchmark with SVG as symbolic visual representation. *arXiv preprint arXiv:2511.02778*, 2025c.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan Dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with blockwise ringattention. In *International Conference on Learning Representations*, 2025.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The AI scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

Dezhao Luo, Bohan Tang, Kang Li, Georgios Papoudakis, Jifei Song, Shaogang Gong, Jianye Hao, Jun Wang, and Kun Shao. ViMo: A generative visual GUI world model for app agents. *arXiv preprint arXiv:2504.13936*, 2025.

Weijian Ma, Shizhao Sun, Tianyu Yu, Ruiyu Wang, Tat-Seng Chua, and Jiang Bian. Thinking with blueprints: Assisting vision-language models in spatial reasoning via structured object representation. *arXiv preprint arXiv:2601.01984*, 2026.

Bodhisattwa Prasad Majumder, Harshit Surana, Dhruv Agarwal, Bhavana Dalvi Mishra, Abhijeetsingh Meena, Aryan Prakhar, Tirth Vora, Tushar Khot, Ashish Sabharwal, and Peter Clark. DiscoveryBench: Towards data-driven discovery with large language models. In *International Conference on Learning Representations*, 2025.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac Gym: High performance GPU-based physics simulation for robot learning. In *Advances in Neural Information Processing Systems*, 2021.

Xiaofeng Mao, Zhen Li, Chuanhao Li, Xiaojie Xu, Kaining Ying, Tong He, Jiangmiao Pang, Yu Qiao, and Kaipeng Zhang. Yume-1.5: A text-controlled interactive world generation model. *arXiv preprint arXiv:2512.22096*, 2025.

Zhendong Mao, Mengqi Huang, Fei Ding, Mingcong Liu, Qian He, and Yongdong Zhang. RealCustom++: Representing images as real textual word for real-time customization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(2):2078–2095, 2026a.

Zhendong Mao, Mengqi Huang, Yijing Lin, Quan Wang, Lei Zhang, and Yongdong Zhang. Toward accurate image generation via dynamic generative image transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(5):5910–5927, 2026b.

Giuseppe Marra, Sebastijan Dumančić, Robin Manhaeve, and Luc De Raedt. From statistical relational to neuro-symbolic artificial intelligence: A survey. *Artificial Intelligence*, 328:104062, 2024.

John McCarthy and Patrick J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. In *Machine Intelligence*, volume 4, pp. 463–502. Edinburgh University Press, 1969.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *International Conference on Learning Representations*, 2023.

Vincent Micheli, Eloi Alonso, and François Fleuret. Efficient world models with context-aware tokenization. In *International Conference on Machine Learning*, pp. 35623–35638. PMLR, 2024.

Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, Liping Jing, Yiming Nie, and Bin Dai. DriveWorld: 4D pre-trained scene understanding via world models for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15522–15533, 2024.

Shunya Minami, Yoshihiro Hayashi, Stephen Wu, Kenji Fukumizu, Hiroki Sugisawa, Masashi Ishii, Isao Kuwajima, Kazuya Shiratori, and Ryo Yoshida. Scaling law of Sim2Real transfer learning in expanding computational materials databases for real-world predictions. *NPJ Computational Materials*, 11(146), 2025.

Mayank Mittal, Pascal Roth, James Tigue, Antoine Richard, Octi Zhang, Peter Du, Antonio Serrano-Muñoz, Xinjie Yao, René Zurbrügg, Nikita Rudin, Lukasz Wawrzyniak, Milad Rakhsha, Alain Denzler, Eric Heiden, Ales Borovicka, Ossama Ahmed, Iretiayo Akinola, Abrar Anwar, Mark T. Carlson, Ji Yuan Feng, Animesh Garg, Renato Gasoto, Lionel Gulich, Yijie Guo, M. Gussert, Alex Hansen, Mihir Kulkarni, Chenran Li, Wei Liu, Viktor Makoviychuk, Grzegorz Malczyk, Hammad Mazhar, Masoud Moghani, Adithyavairavan Murali, Michael Noseworthy, Alexander Poddubny, Nathan Ratliff, Welf Rehberg, Clemens Schwarke, Ritvik Singh, James Latham Smith, Bingjie Tang, Ruchik Thaker, Matthew Trepte, Karl Van Wyk, Fangzhou Yu, Alex Millane, Vikram Ramasamy, Remo Steiner, Sangeeta Subramanian, Clemens Volk, CY Chen, Neel Jawale, Ashwin Varghese Kuruttukulam, Michael A. Lin, Ajay Mandlekar, Karsten Patzwaldt, John Welsh, Huihua Zhao, Fatima Anes, Jean-Francois Lafleche, Nicolas Moënne-Loccoz, Soowan Park, Rob Stepinski, Dirk Van Gelder, Chris Amevor, Jan Carius, Jumyung Chang, Anka He Chen, Pablo de Heras Ciechomski, Gilles Daviet, Mohammad Mohajerani, Julia von Muralt, Viktor Reutskyy, Michael Sauter, Simon Schirm, Eric L. Shi, Pierre Terdiman, Kenny Vilella, Tobias Widmer, Gordon Yeoman, Tiffany Chen, Sergey Grizan, Cathy Li, Lotus Li, Connor Smith, Rafael Wiltz, Kostas Alexis, Yan Chang, David Chu, Linxi "Jim" Fan, Farbod Farshidian, Ankur Handa, Spencer Huang, Marco Hutter, Yashraj Narang, Soha Pouya, Shiwei Sheng, Yuke Zhu, Miles Macklin, Adam Moravanszky, Philipp Reist, Yunrong Guo, David Hoeller, and Gavriel State. Isaac Lab: A GPU-accelerated simulation framework for multi-modal robot learning. *arXiv preprint arXiv:2511.04831*, 2025.

Thomas M. Moerland, Joost Broekens, Aske Plaat, and Catholijn M. Jonker. Model-based reinforcement learning: A survey. *Foundations and Trends in Machine Learning*, 16(1):1–118, 2023.

Anusha Nagabandi, Gregory Kahn, Ronald S. Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *IEEE International Conference on Robotics and Automation*, pp. 7559–7566, 2018.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems*, 2024.

Isaac Newton. *Philosophiæ Naturalis Principia Mathematica*. Royal Society, London, 1687.

Tung Nguyen, Johannes Brandstetter, Ashish Kapoor, Jayesh K. Gupta, and Aditya Grover. ClimaX: A foundation model for weather and climate. In *International Conference on Machine Learning*, pp. 25904–25938. PMLR, 2023.

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

Masashi Okada and Tadahiro Taniguchi. DreamingV2: Reinforcement learning with discrete world models without reconstruction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 985–991, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Linyue Pan, Lexiao Zou, Shuo Guo, Jingchen Ni, and Hai-Tao Zheng. Natural-language agent harnesses. *arXiv preprint arXiv:2603.25723*, 2026.

Joon Sung Park, Lindsay Popowski, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Social simulacra: Creating populated prototypes for social computing systems. In *ACM Symposium on User Interface Software and Technology*, 2022.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Annual ACM Symposium on User Interface Software and Technology*, pp. 1–22, 2023.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, pp. 2778–2787. PMLR, 2017.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Jinghua Piao, Yuwei Yan, Jun Zhang, Nian Li, Junbo Yan, Xiaochong Lan, Zhihong Lu, Zhiheng Zheng, Jing Yi Wang, Di Zhou, Chen Gao, Fengli Xu, Fang Zhang, Ke Rong, Jun Su, and Yong Li. AgentSociety: Large-scale simulation of LLM-driven generative agents advances understanding of human behaviors and society. *arXiv preprint arXiv:2502.08691*, 2025.

Plato. *Republic*. Hackett Publishing, 1992.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R. Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, Remi Lam, and Matthew Willson. Probabilistic weather forecasting with machine learning. *Nature*, 637:84–90, 2024.

Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. VirtualHome: Simulating household activities via programs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8494–8502, 2018.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

Cheng Qian, Emre Can Acikgoz, Bingxuan Li, Xiusi Chen, Yuji Zhang, Bingxiang He, Qinyu Luo, Dilek Hakkani-Tür, Gokhan Tur, Yunzhu Li, and Heng Ji. Current agents fail to leverage world model as tool for foresight. *arXiv preprint arXiv:2601.03905*, 2026.

Shuofei Qiao, Runnan Fang, Ningyu Zhang, Yuqi Zhu, Xiang Chen, Shumin Deng, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Agent planning with world knowledge model. In *Advances in Neural Information Processing Systems*, volume 37, pp. 114843–114871, 2024.

Yiran Qin, Zhelun Shi, Jiwen Yu, Xijun Wang, Enshen Zhou, Lijun Li, Zhenfei Yin, Xihui Liu, Lu Sheng, Jing Shao, Lei Bai, Wanli Ouyang, and Ruimao Zhang. WorldSimBench: Towards video generation models as world simulators. *arXiv preprint arXiv:2410.18072*, 2024.

Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjun Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, Xiaojun Xiao, Kai Cai, Chuang Li, Yaowei Zheng, Chaolin Jin, Chen Li, Xiao Zhou, Minchao Wang, Haoli Chen, Zhaojian Li, Haihua Yang, Haifeng Liu, Feng Lin, Tao Peng, Xin Liu, and Guang Shi. UI-TARS: Pioneering automated gui interaction with native agents. *arXiv preprint arXiv:2501.12326*, 2025.

Willard Van Orman Quine. Two dogmas of empiricism. *The Philosophical Review*, 60(1):20–43, 1951.

Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, SM Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *International Conference on Machine Learning*, pp. 4218–4227. PMLR, 2018.

Prithvi Rajasekaran. Harness design for long-running application development. Anthropic Engineering Blog, 2026. URL `https://www.anthropic.com/engineering/harness-design-long-running-apps`.

Rajesh P. N. Rao and Dana H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87, 1999.

Allen Z. Ren, Hongkai Dai, Benjamin Burchfiel, and Anirudha Majumdar. AdaptSim: Task-driven simulation adaptation for sim-to-real transfer. In *Conference on Robot Learning*, 2023.

Baochang Ren, Yunzhi Yao, Rui Sun, Shuofei Qiao, Ningyu Zhang, and Huajun Chen. Aligning agentic world models via knowledgeable experience learning. *arXiv preprint arXiv:2601.13247*, 2026.

Zhenzhen Ren, Xinpeng Zhang, Zhenxing Qian, Yan Gao, Yu Shi, Shuxin Zheng, and Jiyan He. GTM: Simulating the world of tools for AI agents. *arXiv preprint arXiv:2512.04535*, 2025.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pp. 1278–1286. PMLR, 2014.

Luke Rivard, Sun Sun, Hongyu Guo, Wenhu Chen, and Yuntian Deng. NeuralOS: Towards simulating operating systems via neural generative models. *arXiv preprint arXiv:2507.08800*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

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

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. GAIA-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In *Annual Meeting of the Association for Computational Linguistics*, pp. 7370–7392, 2024.

Vinay Samuel, Henry Peng Zou, Yue Zhou, Shreyas Chaudhari, Ashwin Kalyan, Tanmay Rajpurohit, Ameet Deshpande, Karthik Narasimhan, and Vishvak Murahari. PersonaGym: Evaluating persona agents and LLMs. In *Conference on Empirical Methods in Natural Language Processing*, pp. 6999–7022, 2025.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International Conference on Machine Learning*, pp. 8459–8468. PMLR, 2020.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Max Schwarzer, Ankesh Anand, Rishab Goel, R. Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. In *International Conference on Learning Representations*, 2021.

Melanie Sclar, Jane Yu, Maryam Fazel-Zarandi, Yulia Tsvetkov, Yonatan Bisk, Yejin Choi, and Asli Celikyilmaz. Explore theory of mind: Program-guided adversarial data generation for theory of mind reasoning. *arXiv preprint arXiv:2412.12175*, 2024.

Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pp. 8583–8592. PMLR, 2020.

Andrew W. Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander W. R. Nelson, Alex Bridgland, Hugo Penedones, Stig Petersen, Karen Simonyan, Steve Crossan, Pushmeet Kohli, David T. Jones, David Silver, Koray Kavukcuoglu, and Demis Hassabis. Improved protein structure prediction using potentials from deep learning. *Nature*, 577:706–710, 2020.

Vaisakh Shaj, Saleh Gholam Zadeh, Ozan Demir, Luiz Ricardo Douat, and Gerhard Neumann. Multi time scale world models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 26764–26775, 2023.

Murray Shanahan. *Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia*. MIT Press, 1997.

Yu Shang, Xin Zhang, Yinzhou Tang, Lei Jin, Chen Gao, Wei Wu, and Yong Li. RoboScape: Physics-informed embodied world model. *arXiv preprint arXiv:2506.23135*, 2025.

Zhouzhou Shen, Xueyu Hu, Xiyun Li, Tianqing Fang, Juncheng Li, and Shengyu Zhang. World-model-augmented web agents with action correction. *arXiv preprint arXiv:2602.15384*, 2026.

Pratyush Kumar Singh, Kathryn A. Farrell-Maupin, and Danial Faghihi. A framework for strategic discovery of credible neural network surrogate models under uncertainty. *Computer Methods in Applied Mechanics and Engineering*, 427:117061, 2024.

Xiangchen Song, Weiran Yao, Yewen Fan, Xinshuai Dong, Guangyi Chen, Juan Carlos Niebles, Eric Xing, and Kun Zhang. Temporally disentangled representation learning under unknown nonstationarity. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8092–8113, 2023.

Andrew Sparkes, Wayne Aubrey, Emma Byrne, Amanda Clare, Muhammed N. Khan, Maria Liakata, Magdalena Markham, Jem Rowland, Larisa N. Soldatova, Kenneth E. Whelan, Michael Young, and Ross D. King. Towards robot scientists for autonomous scientific discovery. *Automated Experimentation*, 2(1):1, 2010.

Felix Strieth-Kalthoff, Han Hao, Vandana Rathore, Joshua Derasp, Théophile Gaudin, Nicholas H Angello, Martin Seifrid, Ekaterina Trushina, Mason Guy, Junliang Liu, Xun Tang, Masashi Mamada, et al. Delocalized, asynchronous, closed-loop discovery of organic laser emitters. *Science*, 384(6697):eadk9227, 2024.

Alex Su, Haozhe Wang, Weimin Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, 2024.

Qiao Sun, Liujia Yang, Wei Tang, Wei Huang, Kaixin Xu, Yongchao Chen, Mingyu Liu, Jiange Yang, Haoyi Zhu, Yating Wang, Tong He, Yilun Chen, Xili Dai, Nanyang Ye, and Qinying Gu. Learning primitive embodied world models: Towards scalable robotic learning. *arXiv preprint arXiv:2508.20840*, 2025a.

Shuang Sun, Huatong Song, Lisheng Huang, Jinhao Jiang, Ran Le, Zhihao Lv, Zongchao Chen, Yiwen Hu, Wenyang Luo, Wayne Xin Zhao, et al. Swe-world: Building software engineering agents in docker-free environments. *arXiv preprint arXiv:2602.03419*, 2026.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. WorldPlay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025b.

Richard S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, 1991.

Nathan J. Szymanski, Bernardus Rendy, Yuxing Fei, Rishi E. Kumar, Tanjin He, David Milsted, Matthew J. McDermott, Max Gallant, Ekin Dogus Cubuk, Amil Merchant, Haegyeom Kim, Anubhav Jain, Christopher J. Bartel, Kristin Persson, Yan Zeng, and Gerbrand Ceder. An autonomous laboratory for the accelerated synthesis of inorganic materials. *Nature*, 624:86–91, 2023.

Hao Tang, Darren Key, and Kevin Ellis. WorldCoder, a model-based LLM agent: Building world models by writing code and interacting with the environment. In *Advances in Neural Information Processing Systems*, volume 37, pp. 70148–70212, 2024.

Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, Chen Bao, Xinsong Lin, Yulin Liu, Tse kai Chan, Yuan Gao, Xuanlin Li, Tongzhou Mu, Nan Xiao, Arnav Gurha, Viswesh Nagaswamy Rajesh, Yong Woo Choi, Yen-Ru Chen, Zhiao Huang, Roberto Calandra, Rui Chen, Shan Luo, and Hao Su. ManiSkill3: GPU parallelized robotics simulation and rendering for generalizable embodied ai. *arXiv preprint arXiv:2410.00425*, 2024.

Yuval Tassa, Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Piotr Trochim, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, and Nicolas Heess. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020.

Amir Taubenfeld, Yaniv Dover, Roi Reichart, and Ariel Goldstein. Systematic biases in LLM simulations of debates. In *Conference on Empirical Methods in Natural Language Processing*, pp. 251–267, 2024.

Pankaj R. Telang, Munindar P. Singh, and N. Yorke-Smith. Maintenance of social commitments in multiagent systems. In *AAAI Conference on Artificial Intelligence*, volume 35, pp. 11369–11377, 2021.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

Sifan Tu, Xin Zhou, Dingkang Liang, Xingyu Jiang, Yumeng Zhang, Xiaofan Li, and Xiang Bai. The role of world models in shaping autonomous driving: A comprehensive survey. *arXiv preprint arXiv:2502.10498*, 2025.

Alan M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Keyon Vafa, Justin Y. Chen, Ashesh Rambachan, Jon Kleinberg, and Sendhil Mullainathan. Evaluating the world model implicit in a generative model. In *Advances in Neural Information Processing Systems*, volume 37, pp. 26941–26975, 2024.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations*, 2025.

Gido M. van de Ven, Nicholas Soures, and Dhireesha Kudithipudi. Continual learning and catastrophic forgetting. *arXiv preprint arXiv:2403.05175*, 2024.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, 2017.

Mats WJ van Es, Cameron Higgins, Chetan Gohil, Andrew J Quinn, Diego Vidaurre, and Mark W Woolrich. Large-scale cortical functional networks are organized in structured cycles. *Nature Neuroscience*, 28(10): 2118–2128, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Diego Vidaurre, Laurence T Hunt, Andrew J Quinn, Benjamin AE Hunt, Matthew J Brookes, Anna C Nobre, and Mark W Woolrich. Spontaneous cortical activity transiently organises into frequency specific phase-coupling networks. *Nature Communications*, 9(1):2987, 2018.

Changpeng Wang, Haozhe Wang, Xi Chen, Junhan Liu, Taofeng Xue, Chong Peng, Donglian Qi, Fangzhen Lin, and Yunfeng Yan. From illusion to intention: Visual rationale learning for vision-language reasoning. *arXiv preprint arXiv:2511.23031*, 2025a.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*, 2024a.

Haozhe Wang, Long Li, Chao Qu, Weidi Xu, Fengming Zhu, Wei Chu, and Fangzhen Lin. To code or not to code? adaptive tool integration for math language models via expectation-maximization. In *Annual Meeting of the Association for Computational Linguistics*, pp. 3060–3075, 2025b.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-Rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025c.

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Chi Wan, Yiping Lu, Zhengyuan Yang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. *arXiv preprint arXiv:2510.16907*, 2025d.

Qi Wang, Junming Yang, Yunbo Wang, Xin Jin, Wenjun Zeng, and Xiaokang Yang. Making offline RL online: Collaborative world models for offline visual reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 97203–97230, 2024b.

Qineng Wang, Baiqiao Yin, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu1 Ranjay Krishna, Saining Xie, et al. Spatial mental modeling from limited views. In *The Fourteenth International Conference on Learning Representations*, 2025e.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. ScienceWorld: Is your agent smarter than a 5th grader? In *Conference on Empirical Methods in Natural Language Processing*, pp. 11279–11298, 2022.

Ruoyao Wang, Graham Todd, Ziang Xiao, Xingdi Yuan, Marc-Alexandre Côté, Peter Clark, and Peter Jansen. Can language models serve as text-based world simulators? In *Annual Meeting of the Association for Computational Linguistics*, 2024c.

Tonghan Wang, Heng Dong, Yanchen Jiang, David C. Parkes, and Milind Tambe. On diffusion models for multi-agent partial observability: Shared attractors, error bounds, and composite flow. *arXiv preprint arXiv:2410.13953*, 2024d.

Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. DriveDreamer: Towards real-world-driven world models for autonomous driving. In *European Conference on Computer Vision*, pp. 55–72. Springer, 2024e.

Zhaoyang Wang, Canwen Xu, Boyi Liu, Yite Wang, Siwei Han, Zhewei Yao, Huaxiu Yao, and Yuxiong He. Agent world model: Infinity synthetic environments for agentic reinforcement learning. *arXiv preprint arXiv:2602.10090*, 2026a.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025f.

Zihan Wang, Chi Gui, Xing Jin, Qineng Wang, Licheng Liu, Kangrui Wang, Shiqi Chen, Linjie Li, Zhengyuan Yang, Pingyue Zhang, et al. Ragen-2: Reasoning collapse in agentic rl. *arXiv preprint arXiv:2604.06268*, 2026b.

Zihao Wang, Xujing Li, Yining Ye, Junjie Fang, Haoming Wang, Longxiang Liu, Shihao Liang, Junting Lu, Zhiyong Wu, Jiazhan Feng, et al. Game-tars: Pretrained foundation models for scalable generalist multimodal game agents. *arXiv preprint arXiv:2510.23691*, 2025g.

Zile Wang, Zexiang Liu, Jiaxing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, Yidan Xietian, Jiangbo Pei, Liang Hu, Boyi Jiang, Hua Xue, Zidong Wang, Haofeng Sun, Wei Li, Wanli Ouyang, Xianglong He, Yang Liu, Yangguang Li, and Yahui Zhou. Matrix-Game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026c.

Manuel Watter, Jost Tobias Springenberg, Joschka Boedecker, and Martin Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In *Advances in Neural Information Processing Systems*, volume 28, pp. 2746–2754, 2015.

Hui Wei, Zihao Zhang, Shenghua He, Tian Xia, Shijia Pan, and Fei Liu. PlanGenLLMs: A modern survey of LLM planning capabilities. *arXiv preprint arXiv:2502.11221*, 2025a.

Jiaqi Wei, Yuejin Yang, Xiang Zhang, Yuhan Chen, Xiang Zhuang, Zhangyang Gao, Dongzhan Zhou, Guangshuai Wang, Zhiqiang Gao, Juntai Cao, Zijie Qiu, Ming Hu, Chenglong Ma, Shixiang Tang, Junjun He, Chunfeng Song, Xuming He, Qiang Zhang, Chenyu You, Shuangjia Zheng, Ning Ding, Wanli Ouyang, Nanqing Dong, Yu Cheng, Siqi Sun, Lei Bai, and Bowen Zhou. From AI for science to agentic science: A survey on autonomous scientific discovery. *arXiv preprint arXiv:2508.14111*, 2025b.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, 2024a.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. iVideoGPT: Interactive VideoGPTs are scalable world models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 68082–68119, 2024b.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Ken Goldberg, and Pieter Abbeel. DayDreamer: World models for physical robot learning. In *Conference on Robot Learning*. PMLR, 2023a.

Yufan Wu, Yinghui He, Yilin Jia, Rada Mihalcea, Yulong Chen, and Naihao Deng. Hi-ToM: A benchmark for evaluating higher-order theory of mind reasoning in large language models. In *Conference on Empirical Methods in Natural Language Processing*, pp. 10691–10706, 2023b.

Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. SAPIEN: A simulated part-based interactive environment. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11097–11107, 2020.

Zikai Xiao, Jianhong Tu, Chuhang Zou, Yuxin Zuo, Zhi Li, Peng Wang, Bowen Yu, Fei Huang, Junyang Lin, and Zuozhu Liu. WebWorld: A large-scale world model for web agent training. *arXiv preprint arXiv:2602.14721*, 2026.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52040–52094, 2024.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Xintao Wang, Tien-Tsin Wong, and Ying Shan. DynamiCrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pp. 399–417. Springer, 2024.

Xiaoxu Xu, Hao Li, Jinhui Ye, Yilun Chen, Jia Zeng, Xinyi Chen, Linning Xu, Dahua Lin, Weixin Li, and Jiangmiao Pang. FutureVLA: Joint visuomotor prediction for vision-language-action model. *arXiv preprint arXiv:2603.10712*, 2026.

Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lu, Tianbao Xie, Amrita Saha, Doyen Sahoo, Tao Yu, and Caiming Xiong. Aguvis: Unified pure vision agents for autonomous GUI interaction. *arXiv preprint arXiv:2412.04454*, 2024.

Zelai Xu, Chao Yu, Fei Fang, Yu Wang, and Yi Wu. Language agents with reinforcement learning for strategic play in the werewolf game. *arXiv preprint arXiv:2310.18940*, 2023.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. The AI Scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *arXiv preprint arXiv:2504.08066*, 2025.

Chaoqun Yang, Xinyu Lin, Shulin Li, Wenjie Wang, Ruihan Guo, Fuli Feng, and Tat-Seng Chua. Can large language models derive new knowledge? A dynamic benchmark for biological knowledge discovery. *arXiv preprint arXiv:2603.03322*, 2026.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-Agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, volume 37, pp. 50528–50652, 2024a.

Pei Yang, Hai Ci, and Mike Zheng Shou. macOSWorld: A multilingual interactive benchmark for gui agents. *arXiv preprint arXiv:2506.04135*, 2025a.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024b.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. LongLive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025b.

Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, Prateek Gupta, Shuyue Hu, Zhenfei Yin, Guohao Li, Xu Jia, Lijun Wang, Bernard Ghanem, Huchuan Lu, Chaochao Lu, Wanli Ouyang, Yu Qiao, Philip Torr, and Jing Shao. Oasis: Open agent social interaction simulations with one million agents. *arXiv preprint arXiv:2411.11581*, 2024c.

Zonglin Yang, Xinya Du, Junxian Li, Jie Zheng, Soujanya Poria, and Erik Cambria. Large language models for automated open-domain scientific hypotheses discovery. In *Annual Meeting of the Association for Computational Linguistics*, pp. 13545–13565, 2024d.

Zonglin Yang, Wanhao Liu, Ben Gao, Yujie Liu, Wei Li, Tong Xie, Lidong Bing, Wanli Ouyang, Erik Cambria, and Dongzhan Zhou. MOOSE-chem2: Exploring LLM limits in fine-grained scientific hypothesis discovery via hierarchical search. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025c.

Zonglin Yang, Wanhao Liu, Ben Gao, Tong Xie, Yuqiang Li, Wanli Ouyang, Soujanya Poria, Erik Cambria, and Dongzhan Zhou. MOOSE-Chem: Large language models for rediscovering unseen chemistry scientific hypotheses. In *International Conference on Learning Representations*, 2025d.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. WebShop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems*, volume 35, pp. 20744–20757, 2022.

Jinhui Ye, Fangjing Wang, Ning Gao, Junqiu Yu, Yangkun Zhu, Bin Wang, Jinyu Zhang, Weiyang Jin, Yanwei Fu, Feng Zheng, Yilun Chen, and Jiangmiao Pang. ST4VLA: Spatially guided training for vision-language-action models. *arXiv preprint arXiv:2602.10109*, 2026a.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026b.

Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering atari games with limited data. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Shaofeng Yin, Jiaxin Ge, Zora Zhiruo Wang, Xiuyu Li, Michael J Black, Trevor Darrell, Angjoo Kanazawa, and Haiwen Feng. Vision-as-inverse-graphics agent via interleaved multimodal reasoning. *arXiv preprint arXiv:2601.11109*, 2026.

Lance Ying, Tan Zhi-Xuan, Lionel Wong, Vikash Mansinghka, and Joshua B Tenenbaum. Understanding epistemic language with a language-augmented bayesian theory of mind. *Transactions of the Association for Computational Linguistics*, 13:613–637, 2025.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. GameFactory: Creating new games with generative interactive videos. In *IEEE/CVF International Conference on Computer Vision*, pp. 11590–11599, 2025.

Xiao Yu, Baolin Peng, Ruize Xu, Yelong Shen, Pengcheng He, Suman Nath, Nikhil Singh, Jiangfeng Gao, and Zhou Yu. Reinforcement world model learning for LLM-based agents. *arXiv preprint arXiv:2602.05842*, 2026.

Jingtong Yue, Ziqi Huang, Zhaoxi Chen, Xintao Wang, Pengfei Wan, and Ziwei Liu. Simulating the visual world with artificial intelligence: A roadmap. *arXiv preprint arXiv:2511.08585*, 2025.

Di Zhang, Jingdi Lei, Junxian Li, Xunzhi Wang, Yujie Liu, Zonglin Yang, Jiatong Li, Weida Wang, Suorong Yang, Jianbo Wu, Peng Ye, Wanli Ouyang, and Dongzhan Zhou. Critic-V: VLM critics help catch vlm errors in multimodal reasoning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9050–9061, 2025a.

Han Zhang, Guo-Hua Yuan, Chaohao Yuan, Tingyang Xu, Tian Bian, Hong Cheng, Wenbing Huang, Deli Zhao, and Yu Rong. Lingshu-Cell: A generative cellular world model for transcriptome modeling toward virtual cells. *arXiv preprint arXiv:2603.25240*, 2026a.

Kai Zhang, Xiangchao Chen, Bo Liu, Tianci Xue, Zeyi Liao, Zhihan Liu, Xiyao Wang, Yuting Ning, Zhaorun Chen, Xiaohan Fu, et al. Agent learning via early experience. *arXiv preprint arXiv:2510.08558*, 2025b.

Lunjun Zhang, Yuwen Xiong, Ze Yang, Sergio Casas, Rui Hu, and Raquel Urtasun. Copilot4D: Learning unsupervised world models for autonomous driving via discrete diffusion. In *International Conference on Learning Representations*, 2024.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. *arXiv preprint arXiv:2504.12626*, 2025c.

Peng-Fei Zhang, Ying Cheng, Xiaofan Sun, Shijie Wang, Fengling Li, Lei Zhu, and Heng Tao Shen. A step toward world models: A survey on robotic manipulation. *arXiv preprint arXiv:2511.02097*, 2025d.

Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, et al. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *The Fourteenth International Conference on Learning Representations*, 2026b.

Wancong Zhang, Basile Terver, Artem Zholus, Soham Chitnis, Harsh Sutaria, Mido Assran, Amir Bar, Randall Balestriero, Adrien Bardes, Yann LeCun, and Nicolas Ballas. Hierarchical planning with latent world models. *arXiv preprint arXiv:2604.03208*, 2026c.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. STORM: Efficient stochastic transformer based world models for reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Xiaoyuan Zhang, Yizhe Huang, Chengdong Ma, Zhixun Chen, Long Ma, Yali Du, Song-Chun Zhu, Yaodong Yang, and Xue Feng. Social world model-augmented mechanism design policy learning. *arXiv preprint arXiv:2510.19270*, 2025e.

Xichen Zhang, Ziyi He, Yinghao Zhu, Sitong Wu, Shaozuo Yu, Meng Chu, Wenhu Zhang, Haoru Tan, and Jiaya Jia. SearchGym: Bootstrapping real-world search agents via cost-effective and high-fidelity environment simulation. *arXiv preprint arXiv:2601.14615*, 2026d.

Xichen Zhang, Sitong Wu, Yinghao Zhu, Haoru Tan, Shaozuo Yu, Ziyi He, and Jiaya Jia. Scaf-GRPO: Scaffolded group relative policy optimization for enhancing LLM reasoning. In *International Conference on Learning Representations*, 2026e.

Xuan Zhang, Wenxuan Zhang, Anxu Wang, See-Kiong Ng, and Yang Deng. MASim: Multilingual agent-based simulation for social science. *arXiv preprint arXiv:2512.07195*, 2025f.

Zhang Zhang, Qiang Zhang, Wei Cui, Shuai Shi, Yijie Guo, Gang Han, Wen Zhao, Jingkai Sun, Jiahang Cao, Jiaxu Wang, Hao Cheng, Xiaozhu Ju, Zhengping Che, Renjing Xu, and Jian Tang. Occupancy world model for robots. *arXiv preprint arXiv:2505.05512*, 2025g.

Zhongyue Zhang, Zijie Qiu, Yingcheng Wu, Shuya Li, Dingyan Wang, Zhuomin Zhou, Duo An, Yuhan Chen, Yu Li, Yongbo Wang, Chubin Ou, Zichen Wang, Jack Xiaoyu Chen, Bo Zhang, Yusong Hu, Wenxin Zhang, Zhijian Wei, Runze Ma, Qingwu Liu, Bo Dong, Yuexi He, Qiantai Feng, Lei Bai, Qiang Gao, Siqi Sun, and Shuangjia Zheng. OriGene: A self-evolving virtual disease biologist automating therapeutic target discovery. bioRxiv 2025.06.03.657658, 2025h.

Bingxi Zhao, Lin Geng Foo, Ping Hu, Christian Theobalt, Hossein Rahmani, and Jun Liu. LLM-based agentic reasoning frameworks: A survey from methods to scenarios. *arXiv preprint arXiv:2508.17692*, 2025.

Hongyu Zhao, Siyu Zhou, Haolin Yang, Zengyi Qin, and Tianyi Zhou. Neuro-symbolic synergy for interactive world modeling. *arXiv preprint arXiv:2602.10480*, 2026.

Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *arXiv preprint arXiv:2503.21755*, 2025a.

Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. OccWorld: Learning a 3D occupancy world model for autonomous driving. In *European Conference on Computer Vision*, pp. 55–72. Springer, 2024.

Xinyue Zheng, Haowei Lin, Kaichen He, Zihao Wang, Qiang Fu, Haobo Fu, Zilong Zheng, and Yitao Liang. MCU: An evaluation framework for open-ended game agents. In *International Conference on Machine Learning*, pp. 78221–78259. PMLR, 2025b.

Yuhao Zheng, Li'an Zhong, Yi Wang, Rui Dai, Kaikui Liu, Xiangxiang Chu, Linyuan Lv, Philip Torr, and Kevin Qinghong Lin. Code2World: A GUI world model via renderable code generation. *arXiv preprint arXiv:2602.09856*, 2026.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. SOTOPIA: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, 2024.

Xuhui Zhou, Jiarui Liu, Akhila Yerukola, Hyunwoo Kim, and Maarten Sap. Social world models. *arXiv preprint arXiv:2509.00559*, 2025.

Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *IEEE/CVF International Conference on Computer Vision*, pp. 8535–8546, 2025.

Zheng Zhu, Xiaofeng Wang, Wangbo Zhao, Chen Min, Bohan Li, Nianchen Deng, Min Dou, Yuqi Wang, Botian Shi, Kai Wang, Chi Zhang, Yang You, Zhaoxiang Zhang, Dawei Zhao, Liang Xiao, Jian Zhao, Jiwen Lu, and Guan Huang. Is Sora a world simulator? a comprehensive survey on general world models and beyond. *arXiv preprint arXiv:2405.03520*, 2024.