From Foresight to Reactive Traffic Worlds: A Survey of Driving World Models for Closed-Loop Simulation

An Xu, Zekai Jin, Hanrong Zhang, Xiaoning Sun, Chengbo Zhang, and Yunfei Yin

***Abstract*—Every driving log records a single realized future. Once a candidate policy brakes, merges, or yields differently, the recording no longer specifies how the encounter will unfold. Learned driving environments provide models of this unobserved continuation, yet a visually plausible rollout is not necessarily informative about policy performance. We organize the literature by the intervention dependencies demonstrated in each evaluated setting. Replay-Constrained settings alter the returned observation while retaining a prescribed behavioral future. Ego-Responsive settings establish a relation between an ego command and its realized motion, with relevant non-ego behavior held fixed. Traffic-Reactive settings establish that surrounding road users respond to a counterfactual ego trajectory, whether that trajectory is generated from a command or prescribed for the test. Recurrence is recorded separately according to whether an updated state enters a subsequent policy decision. We then relate these capabilities to evidence for observation fidelity, ego controllability, traffic response, policy transfer, and transportation-scale behavior. This view distinguishes the interactions a simulator can reproduce from the conclusions its evaluation can support.**

***Index Terms*—modeling and simulation, data-based approaches, autonomous driving, traffic networks, driving world models, transportation-scale validity**

I. Introduction

EVERY driving log records one realized future. Policy evaluation concerns the futures that did not occur: a collision avoided by earlier braking, a gap opened by a more assertive merge, or a traffic response induced by a different plan. These counterfactuals cannot be tested safely or repeatedly on public roads [1], [2]. Once a candidate policy selects an action different from the recorded one, the log no longer specifies the interaction that follows [3], [4]. Closed-loop evaluation must reconstruct this missing branch while preserving the physical and behavioral dependencies that make the resulting interaction meaningful.

Simulation provides a controlled setting for reconstructing these branches. A simulator can reset scenarios, vary conditions, and compare policies from matched initial states, including encounters too hazardous or too rare for routine road testing [1]. Platforms such as CARLA combine configurable roads, sensors, weather, vehicle dynamics, and background traffic into repeatable policy experiments [5], [6]. In a simulator, once the ego action changes, the modeled responses affect the evidence used to reward, rank, or reject the policy. Different background-traffic mechanisms, from fixed replay trajectories to reactive traffic models, can therefore produce different policy rankings from the same initial scene [3], [7]–[9].

Learned environments extend the range of counterfactuals that can be reconstructed from driving data. Conventional simulators represent roads, vehicles, sensors, and behavior through explicit components whose fidelity and scalability depend on their coupling [10], [11]. Data-driven systems reconstruct scenes and traffic behavior from demonstrations [7], [10]; neural rendering and generative models synthesize sensor observations and scene evolution beyond the recorded viewpoint. The resulting environments can turn driving corpora into experimental settings, provided that the intervention dependencies needed by the experiment are actually represented.

Architecture and output modality do not determine what a learned simulator can establish in a policy experiment. Neural scene representations render camera or light detection and ranging (LiDAR) observations from novel viewpoints [12]–[14]; action-conditioned video models generate alternative visual continuations under candidate ego trajectories [15]–[17]; and learned traffic simulators update surrounding actors from an evolving joint state [18], [19]. A photorealistic continuation may preserve the recorded traffic response, whereas a low-dimensional simulator may represent reciprocal interaction.

We therefore organize these systems by the relations demonstrated after an ego intervention. A queried viewpoint changes the observation returned to the policy. An ego command may change the realized ego motion. A changed ego trajectory may alter the behavior of relevant road users. These relations are empirically distinct, and none implies that a downstream policy is queried again. Recurrence is recorded separately when the post-intervention state or observation enters another policy decision [7], [20], [21].

Fig. 1 separates three intervention relations. Replay-Constrained settings answer an observation query against a prescribed behavioral future. Ego-Responsive settings establish command-to-motion execution but do not demonstrate that relevant traffic changes in response. Traffic-Reactive settings establish such a response to a counterfactual ego trajectory, regardless of how that trajectory is obtained. Classification therefore applies to the relation demonstrated in an evaluated configuration, not permanently to a model architecture.

Existing surveys organize driving world models by representation, predicted modality, generative architecture, latent dynamics, planning use, or autonomous-driving application [22]–

An Xu and Yunfei Yin are with the School of Transportation Science and Engineering, Harbin Institute of Technology, Harbin 150090, China (e-mail: AnX.RTL2324@tum-asia.edu.sg; yunfeiyin@hit.edu.cn). Corresponding author: Yunfei Yin.

Zekai Jin, Xiaoning Sun, and Chengbo Zhang are with the Department of Civil Engineering, McGill University, Montreal, QC H3A 0C3, Canada (e-mail: zekai.jin@mail.mcgill.ca; xiaoning.sun@mail.mcgill.ca; chengbo.zhang@mail.mcgill.ca).

Hanrong Zhang is with the Department of Computer Science, University of Illinois Chicago, Chicago, IL 60607, USA (e-mail: hzhan135@uic.edu).

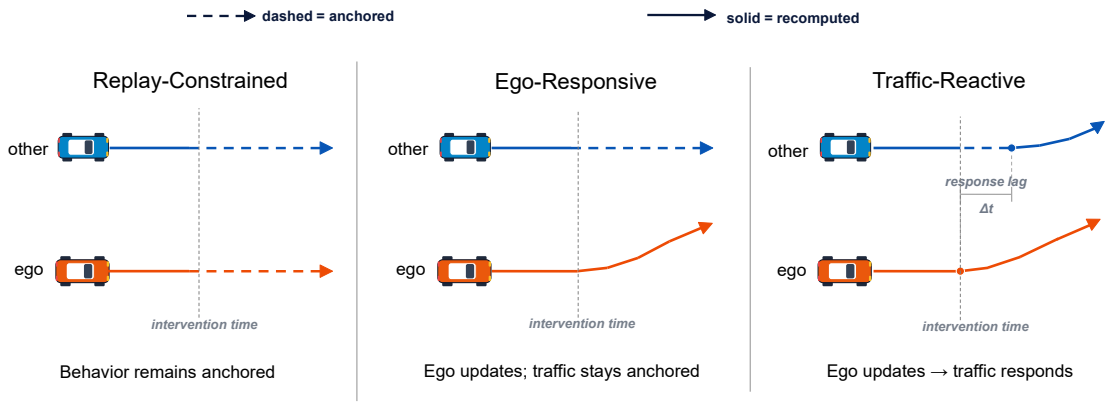

Fig. 1. Intervention relations from the same initial encounter. Replay-Constrained settings preserve a prescribed behavioral future while changing the queried observation. Ego-Responsive settings establish that an ego command produces the intended realized motion, without a demonstrated traffic response. Traffic-Reactive settings establish that ego-relevant road users respond when the counterfactual ego trajectory changes; the trajectory may be generated from a command or prescribed for the test. Dashed paths remain prescribed, solid paths depend on the intervention, and Δt denotes response lag. Regimes describe evaluated settings rather than permanent model identities; policy recurrence is recorded separately.

TABLE I

CLOSEST SYNTHESES ADJACENT TO THE PRESENT SCOPE. ROWS REPORT EACH WORK'S CENTRAL QUESTION, ANALYTICAL UNIT, TREATMENT OF FEEDBACK, AND VALIDATION EMPHASIS. PUBLICATION STATUS IS STATED BECAUSE SEVERAL 2025–2026 SYNTHESES WERE PUBLIC PREPRINTS AT THE SURVEY CUTOFF.

Prior synthesis	Central question	Primary unit	Treatment of feedback	Validation emphasis
Initial DWM survey (journal) [22]	How world models support autonomous driving	World-model method	Future prediction is connected to driving applications and decision making	Foundations, applications, limitations, and research directions
Generation–planning survey (preprint) [23]	How world models span future generation and agent planning	Representation and method family	Organizes future generation, behavior planning, and prediction–planning interaction	Training paradigms, task evaluation, and deployment challenges
Modality-centered DWM survey (preprint) [24]	What driving scenes world models predict	Predicted scene modality	Scene evolution and interaction are reviewed within modality-defined approaches	Datasets, task-specific metrics, limitations, and future directions
Latent DWM survey (preprint) [25]	How latent structure supports prediction, planning, and reasoning	Latent world, action, and generator	Closed-loop use is related to latent dynamics and internal model mechanics	Closed-loop metrics, robustness, deployability, and resource-aware evaluation
Model predictive control (MPC)-oriented DWM survey (preprint) [26]	How predicted futures support receding-horizon decisions	Predictive transition interface	Rollout, constraint reasoning, cost evaluation, and replanning form the organizing view	Open–closed-loop evidence gap, planning utility, uncertainty, and scalability
Autonomous vehicle (AV) interaction-model survey (journal) [27]	How road users interact under competing system objectives	Road-user interaction model	Rule-based, learning-based, and hybrid mechanisms describe reciprocal behavior	Safety, efficiency, sustainability, simulation, and real-world validation
World-model admissibility framework (preprint) [28]	When a conclusion drawn from a world model is credible enough to use	Simulation conclusion	Action following and robustness are tested before accepting closed-loop conclusions	Staged admissibility from visual fidelity to policy-level assurance

[27]. Related reviews also cover scenario generation, simulation platforms, and generative testing [2], [6], [11], [29], [30], while an admissibility framework considers when conclusions drawn from a world model are credible enough for use [28]. However, these studies generally do not use the relation between an ego intervention and the resulting simulation response as the main basis for organizing the literature. Table I highlights this difference. Our focus is therefore on what changes after an off-log intervention, whether the changed state enters a later policy decision, and what level of policy or transportation claim is supported by the available evidence.

Feedback relation and transportation scale answer different questions. As illustrated in Fig. 2, a credible response in an isolated encounter cannot determine corridor- or network-level effects. Local braking, yielding, and merging can alter shockwave propagation, bottleneck throughput, and network stability [31]–[33], but a single merge does not reveal platoon stability or travel-time distributions across an urban network. Transportation scale is therefore an independent validation axis: the spatial extent, time horizon, and traffic population represented in the evidence must match the intended claim.

This survey makes three primary contributions:

1) A feedback-centered organization of learned driving environments unifies neural sensor simulation, action-conditioned world models, and data-driven traffic simulators under three regimes: Replay-Constrained, Ego-Responsive, and Traffic-Reactive.
2) A functional decomposition distinguishes future generation, observation rendering, candidate evaluation, and recurrent policy interaction, clarifying when post-intervention state must persist into a later policy decision.
3) A claim–evidence map relates observation fidelity, ego controllability, traffic-response validity, and policy transfer to encounter, corridor, and network scales, separating simulator capability from supported claim scope.

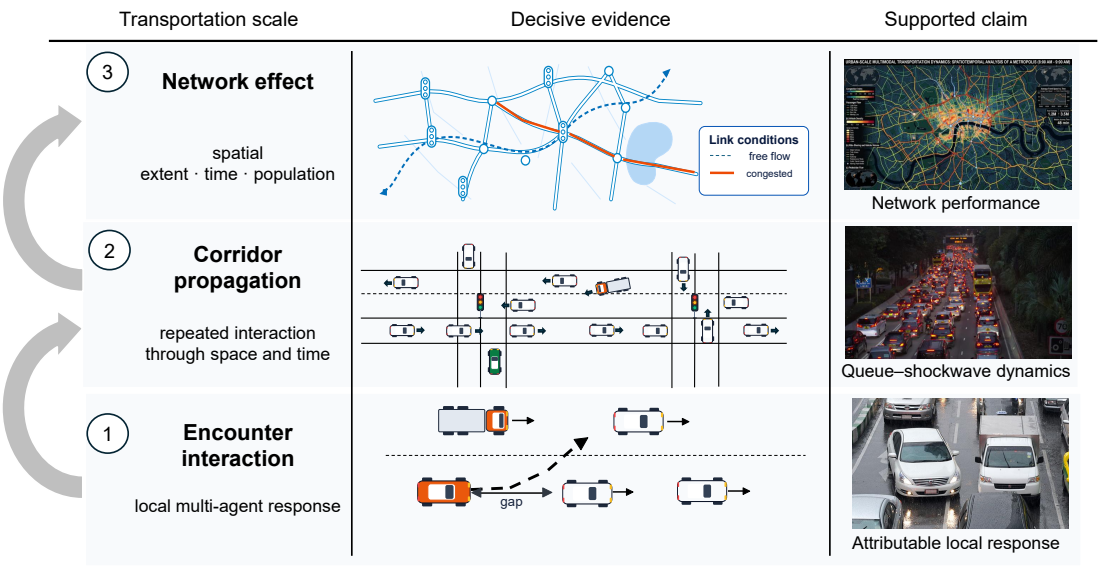

Fig. 2. Transportation-scale validity from encounter interaction to corridor propagation and network effects. The supported claim scale expands jointly across spatial extent, time, and population. Paired interventions support attribution within an encounter; an independent microscopic reference tests whether repeated responses produce credible corridor propagation; and traffic- or network-scale references test delay, throughput, queue dynamics, and conflict exposure. Larger scenes alone do not extend the supported claim scale.

A. Scope, Comparison Axes, and Organization

The survey covers representative work on learned driving simulation publicly available from 2017 through August 2026, together with foundational studies that establish core simulation and validation concepts. The comparison uses six axes: (1) the joint state that persists after an intervention, (2) how the ego action changes that state, (3) how ego-relevant traffic is recomputed, (4) the observation returned to the policy, (5) whether the updated state supports successive policy decisions, and (6) the transportation scale supported by the empirical evidence. The supplementary material documents the literature-search procedure and detailed system comparisons.

Section II formalizes the policy–environment loop, functional roles, feedback regimes, and transportation scales. Sections III, IV, and V examine Replay-Constrained, Ego-Responsive, and Traffic-Reactive worlds, respectively. Section VI relates these regimes to policy evaluation, training, and stress testing. Section VII examines the evidence required at each validity layer and transportation scale. Section VIII identifies open research problems, and Section IX concludes the survey.

II. Foundations: Learned World Models as Simulators

Closed-loop policy simulation requires more than predicting a future from a fixed history. A policy issues a command, the environment realizes its consequence, and a new observation may be returned for another decision. Systems implement parts of this interaction through learned traffic behavior, vehicle dynamics, latent transitions, or neural sensor rendering [7], [12], [19], [34]. The learned component need not replace the full simulator; its scientific role depends on which relation it supplies and which relation the reported experiment exercises.

To avoid conflating model structure with simulation capability, we separate the functional operators of the policy–environment loop from the intervention dependencies they demonstrate. Below, we formalize this interactive loop, examine what changes after an ego intervention, define the resulting feedback regimes, and decouple these empirical relations from architectural forms and system roles.

A. The Policy–Environment Loop

Consider a driving policy π acting at discrete time step t from a history $\mathcal{H}_t$ of past observations and ego commands. Let s_t denote the simulator state, u_t^{ego} the ego command issued by the policy, τ_t^{ego} the realized ego motion over the current step, and $a_t^{-\text{ego}}$ the actions of surrounding road users. A general interactive driving environment can be written as

$$\begin{aligned} u_t^{\text{ego}} &= \pi(\mathcal{H}_t), \\ \tau_t^{\text{ego}} &= \mathcal{D}(s_t, u_t^{\text{ego}}), \\ a_t^{-\text{ego}} &\sim \mathcal{B}_\phi(s_t, \tau_t^{\text{ego}}), \\ s_{t+1} &\sim \mathcal{T}_\theta(s_t, \tau_t^{\text{ego}}, a_t^{-\text{ego}}), \\ o_{t+1} &\sim \mathcal{R}_\psi(s_{t+1}), \\ \mathcal{H}_{t+1} &= (\mathcal{H}_t, u_t^{\text{ego}}, o_{t+1}). \end{aligned} \tag{1}$$

Here, $\mathcal{H}_t$ contains the information available to the policy, whereas s_t represents the simulator state and may include variables not directly observed by the policy, such as actor intentions, interaction memory, or scene dynamics. The dynamics operator $\mathcal{D}$ maps the requested ego command to physically feasible motion. The traffic-response model $\mathcal{B}_\phi$ produces surrounding-road-user actions conditioned on the current joint state and realized ego motion. The transition function $\mathcal{T}_\theta$ advances the joint state, and $\mathcal{R}_\psi$ generates the next policy observation o_{t+1}.

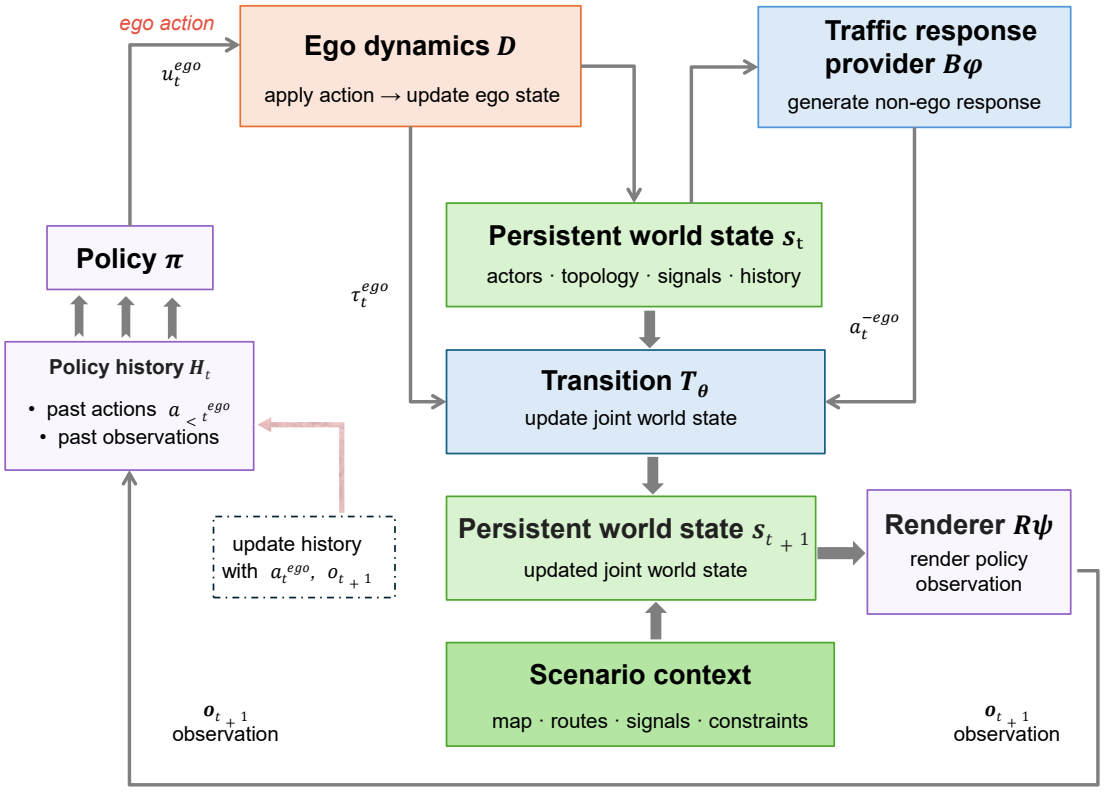

Fig. 3. Functional policy–environment loop. Ego dynamics map a policy command to realized ego motion, the traffic-response model updates surrounding road users, the transition advances the joint state, and the observation model returns the next policy input. The three feedback regimes differ in how far an ego intervention propagates through these relations. Recurrence additionally requires the updated state and observation to support the next policy decision.

Equation (1) specifies functional dependencies rather than a required implementation architecture, as illustrated in Fig. 3. Reaction delays, sensing latencies, and multi-rate execution can be represented within the state or history so that actions and observations remain aligned with simulated time. The central question for policy evaluation is not whether an environment is generative, agent-based, or hybrid, but which of these dependencies are recalculated when a candidate policy departs from the recorded log.

B. What Changes After an Intervention

A policy command is not yet a realized consequence. The command u_t^{ego} specifies an intended maneuver through inputs that range from control signals (steering, throttle, braking) to waypoints, target lanes, or continuous trajectories. The realized motion $\tau_t^{\text{ego}} = \mathcal{D}(s_t, u_t^{\text{ego}})$ is the trajectory executed under vehicle dynamics, actuator limits, and scene boundaries. An action-conditioned model may accept an off-log command without faithfully producing the corresponding motion. For example, a model conditioned on a left-turn command may continue straight when strong training priors overwhelm the conditioning token [35]. Disentangling u_t^{ego} from τ_t^{ego} is therefore necessary to distinguish commanded actions from physically realized interventions. Underlying state and exposed observation likewise play distinct roles. The simulator state s_t tracks the authoritative variables required to advance the world, including road topology, traffic light phases, bounding boxes, velocities, and interaction histories. The observation $o_t \sim \mathcal{R}_\psi(s_t)$ is the sensor representation delivered to the policy, such as surround camera images, LiDAR point clouds, or occupancy representations. Neural sensor simulators highlight this distinction: novel-view synthesizers render realistic camera frames from queried viewpoints, but they require non-ego trajectories and actor kinematics to be provided by an external engine [12]. Establishing high-fidelity observation generation is thus separate from demonstrating dynamic evolution of the underlying traffic state. Finally, surrounding road users may or may not adjust their behavior when the ego vehicle follows an unrecorded path. Traffic reactivity is captured by the conditional distribution $\mathcal{B}_\phi(s_t, \tau_t^{\text{ego}})$. In an unreactive setting, surrounding trajectories remain fixed to log replay or pre-scripted paths regardless of ego intervention. In a reactive setting, non-ego agents adjust their speed, yield, or change lanes in response to τ_t^{ego}. Whether an environment updates $a_t^{-\text{ego}}$ is a question of causal dependence, not model class: rule-based models can yield reactively, while large generative world models can remain entirely non-reactive if background behavior is passively baked into log rollouts.

C. Feedback Relations: The Three Simulation Regimes

The functional dependencies exercised after an off-log ego intervention define the environment's feedback regime. Rather than classifying simulators by their training loss or network topology, we categorize evaluated simulation settings by the deepest intervention relation they close in Eq. (1): In a **Replay-Constrained** regime, the behavioral future of surrounding road

users remains anchored to the recorded log or to pre-scripted trajectories independently of the candidate policy's commands, even if an observation model synthesizes novel viewpoints or edited sensor artifacts. By contrast, an **Ego-Responsive** regime closes the ego control loop ($u_t^{\text{ego}} \rightarrow \tau_t^{\text{ego}}$), ensuring that realized ego motion and the resulting viewpoint faithfully respond to policy actions, yet surrounding traffic remains unreactive and anchored. Finally, a **Traffic-Reactive** regime closes the multi-agent interaction loop ($\tau_t^{\text{ego}} \rightarrow a_t^{-\text{ego}}$), wherein surrounding road users adapt their maneuvers—yielding, braking, or changing lanes—in response to counterfactual ego trajectories, exhibiting an empirical response lag Δt.

These three regimes characterize evaluated simulation settings rather than permanent properties of model architectures. A single model may operate in different regimes depending on its test configuration. Having defined the feedback relations that an environment can close, we next decouple them from model architectures and functional software roles.

D. Architectural Realizations and System Roles

Model architecture, software role, and feedback capability are frequently conflated. Having a recurrent neural network, an action-conditioning branch, or a multi-agent transformer does not automatically place a system into the Traffic-Reactive or even Ego-Responsive regime. Architecture determines how the functions in Eq. (1) are parameterized; the evaluation protocol determines which dependencies are actually exercised. In practice, learned driving models parameterize the operators in Eq. (1) across diverse architectural families. Video and sensor-space generative models map past tokens and action prompts directly to future video frames or LiDAR ranges, often absorbing $\mathcal{D}$, $\mathcal{T}_\theta$, and $\mathcal{R}_\psi$ into an implicit end-to-end forward pass [15]–[17]. Neural scene renderers implement $\mathcal{R}_\psi$ via NeRF or 3D Gaussian Splatting, rendering sensor observations from explicit 3D actor poses and geometry [12]–[14], [36]. Agent-based and bird's-eye-view (BEV) simulators implement $\mathcal{B}_\phi$ and $\mathcal{T}_\theta$ in structured bird's-eye-view or vectorized state spaces, predicting actor trajectories or occupancy grids without full sensor synthesis [7], [18], [19], [37]. Finally, latent dynamics models abstract scene evolution into continuous or discrete latent spaces, propagating hidden states through learned transitions [25], [34]. Depending on how these architectures are embedded in an evaluation pipeline, learned driving models occupy four distinct operational roles:

1) **One-Shot Future Generator:** Predicts an open-loop rollout of finite horizon T from an initial prompt, without accepting intervening actions or maintaining a persistent interactive state.
2) **Observation Renderer:** Translates externally supplied vehicle poses and scene geometry into high-dimensional sensor streams while an external simulator advances the world.
3) **Offline Candidate Evaluator:** Evaluates and scores multiple proposed ego trajectories in parallel prior to execution, serving as a predictive safety filter or value function without hosting rollouts.
4) **Recurrent Environment:** Maintains an authoritative persistent world state s_t, receives the policy's real-time action u_t^{ego}, updates traffic and ego states, and returns the successor observation o_{t+1} for the next decision.

Functional roles and feedback regimes are orthogonal. For example, a neural renderer acting as an Observation Renderer operates in the Replay-Constrained regime when paired with logged actor trajectories, but becomes part of a Traffic-Reactive pipeline when coupled to an interactive traffic engine such as Waymax or nuPlan [20], [38]. Conversely, a complex end-to-end world model operating as a Recurrent Environment remains functionally Replay-Constrained if its background vehicles blindly execute recorded paths regardless of ego braking or swerving. Classifying a simulator therefore requires identifying its specific operational configuration and demonstrated intervention dependencies (Supplementary Table S2).

E. Recurrence, Evidence, and Transportation Scale

Recurrence concerns persistence across policy decisions rather than either feedback relation. A recurrent environment advances a successor state, generates a new observation, and lets the policy act again from that consequence [7], [34]. Recursive generation without a downstream policy query is temporal continuation, not policy recurrence. Recurrence permits multi-step interaction, but does not establish the validity of the resulting rollout.

Three properties are therefore tracked separately:

1) **Feedback Regime:** The intervention-dependent relation demonstrated by the environment.
2) **Validation Evidence:** The empirical evidence supporting observation fidelity, ego controllability, traffic-response validity, long-horizon behavior, and policy transfer.
3) **Transportation Scale:** The encounter, corridor, or network scale directly supported by evidence over the corresponding spatial extent, time horizon, and traffic population.

These properties describe different aspects of simulator credibility and should be evaluated separately. A Traffic-Reactive simulator can produce unrealistic responses, and a recurrent observation loop can support sensor-level policy testing without reciprocal traffic interaction. Likewise, causal responsiveness demonstrated in a single merge does not establish credible shockwave propagation or network delay [31], [39]. Increasing scene size does not by itself extend the scope of a supported conclusion. A simulation claim is justified only over the feedback relations, validity properties, and transportation scales directly supported by evidence.

III. Replay-Constrained Worlds: Counterfactual Observation from Recorded Drives

A recorded drive provides sensor observations along one realized trajectory. Replay-Constrained environments extend this coverage by synthesizing observations at altered ego poses or edited scene states while non-ego behavior remains prescribed. They support counterfactual observation, but not attribution of traffic response to the ego intervention.

A. Counterfactual Views and Scene Edits

Early data-driven simulators increased observation diversity by compositing recorded backgrounds with inserted vehicle models, as demonstrated by AADS [40]. Neural scene reconstruction extends this idea by recovering editable 3D representations from logged driving data. UniSim and NeuRAD synthesize both camera and LiDAR observations from novel viewpoints [12], [14]. MARS focuses on image rendering in editable NeRF-based driving scenes [13], while DrivingGaussian targets photorealistic surround-view image synthesis and uses LiDAR primarily as a geometric prior for scene reconstruction [41]. Across these systems, controlled changes to ego pose, actor placement, or scene content allow sensor consequences of altered viewpoints, occlusions, and obstacle configurations to be studied without recollecting the scene.

The rendered observation remains conditional on a supplied scene state. A renderer can generate sensor data for specified ego and actor poses, but it does not determine how those poses should evolve after an intervention. Actor trajectories may still come from the recorded log, manual edits, or an external motion model. Large viewpoint shifts can also expose regions that are weakly constrained or absent in the source observations, reducing reconstruction fidelity [12]. Scene editability therefore extends observation coverage; it does not establish an intervention-dependent behavioral response from surrounding traffic.

B. Evaluation near Recorded Trajectories

When successive policy decisions alter the ego pose, a neural renderer can participate in a recurrent ego–observation loop: the new pose determines the next sensor observation, which is returned to the policy for another decision. UniSim demonstrates this form of sensor-level closed-loop evaluation [12]. Cam2Sim reconstructs real driving recordings as playable CARLA scenarios and augments camera observations with Gaussian-Splatting-based rendering, allowing a system under test to operate repeatedly within the reconstructed scene [42]. In both cases, observation generation can respond to changes in ego pose without requiring the renderer itself to determine surrounding-traffic behavior.

These environments support evaluation close to recorded trajectories. Held-out views and sensor recordings test geometric, photometric, and temporal fidelity, while downstream perception or planning models reveal whether rendering errors alter task performance [43], [44]. NAVSIM provides a complementary setting for efficient planner screening from logged scene representations without modeling reciprocal traffic response [45]. Such evaluations can expose perception failures, viewpoint sensitivity, and accumulated tracking error, but they do not test how surrounding road users respond to an off-log ego maneuver.

C. Boundary of Replay-Constrained Feedback

Replay-Constrained environments can return observations that differ from the recorded sensor stream, including observations rendered from an externally specified off-log ego pose. Supplying a new pose to a renderer does not show that a policy command produced that pose or that surrounding actors respond to it. If the updated pose is used for another rendering query, the observation loop may be recurrent while the feedback regime remains Replay-Constrained.

The behavioral boundary is equally strict. If surrounding actors continue along trajectories specified independently of the ego intervention, their subsequent actions cannot be attributed to that intervention. An aggressive merge, for example, cannot test whether a following vehicle yields when that vehicle remains committed to its recorded trajectory.

Ego-Responsive feedback begins when different ego commands are shown to produce corresponding differences in realized ego motion. Persistence into another policy decision is assessed separately as recurrence.

IV. Ego-Responsive Worlds: From Commands to Realized Motion

Action-conditioned world models can represent a relation absent from novel-view rendering: different ego commands can produce different realized ego motions from the same driving history. Ego-Responsive feedback requires the requested command u_t^{ego} to produce the intended change in realized ego motion τ_t^{ego}; supplying a pose or trajectory as a rendering query does not test this relation. Whether the resulting state enters a later policy decision is a separate recurrence property, so a finite action-conditioned branch can be Ego-Responsive without forming a recurrent environment.

A. From Conditioning Signals to Realized Ego Motion

Driving world models condition future generation on ego intent at different levels of abstraction. GAIA-1 combines video, text, and vehicle-action inputs to control generated driving futures [15]. Vista supports conditions ranging from high-level commands and goal points to trajectories, steering angles, and speed [17]. DriveDreamer combines structured traffic conditions with action prediction [46]; ADriver-I models interleaved vision–action sequences for control and future-frame prediction [47]; and Drive-WM generates alternative multi-view futures under candidate maneuvers for trajectory selection [16].

These conditioning signals impose different requirements on the mapping from command to motion. Steering and acceleration require temporal integration through vehicle dynamics, whereas waypoints and trajectory targets specify desired geometry more directly; both still require feasibility checks against vehicle dynamics and scene constraints. An action condition therefore does not establish ego responsiveness by itself. The relevant relation is

$$u_t^{\mathrm{ego}} \rightarrow \tau_t^{\mathrm{ego}},$$

and the realized motion must follow the requested maneuver over the claimed operating range.

ACT-Bench examines this command–motion gap by comparing conditioning trajectories with motion recovered from generated futures and reports substantial action-fidelity errors among evaluated driving world models [35]. ReSim addresses

a complementary coverage problem by introducing non-expert and hazardous trajectories that extend beyond the predominantly expert motions observed in ordinary driving logs [48]. These evaluations expose the distinction between conditional sensitivity and calibrated adherence outside ordinary expert trajectories. Ego responsiveness therefore requires command-specific trajectory error and physical-feasibility evidence over the maneuver range being claimed.

B. Finite Branches and Policy Recurrence

Action-conditioned generation can support ego-dependent branching without recurrent policy interaction. In offline candidate evaluation, several ego commands are applied to the same history to generate alternative finite futures before a maneuver is selected. Drive-WM exemplifies this use by generating futures under distinct driving maneuvers and evaluating candidate trajectories from the resulting visual predictions [16]. Such branching can establish that the generated ego consequence depends on the command even when the selected branch is not carried forward as the state of an ongoing simulation.

Policy recurrence asks a different question: whether the consequence of one policy decision persists into the state or history used for the next decision. ADriver-I recursively incorporates generated vision–action pairs into subsequent prediction, while Vista conditions longer continuations on previously generated content and historical observations [17], [47]. These mechanisms support temporal continuation, but continuation alone does not establish recurrent policy interaction. The relevant evidence is whether the realized consequence of one command becomes part of the world from which the next policy decision is evaluated, rather than being regenerated from the original recorded history.

Paired commands provide a direct test of intervention sensitivity. Starting from the same history, hard braking and lane changing should produce different ego motions in the corresponding directions. Such separation is necessary but not sufficient for action adherence: two commands can produce distinct trajectories while both deviate from their requested motions. Command-specific trajectory error and physical feasibility determine whether the requested maneuver is faithfully realized. If recurrence is claimed, the resulting ego state must additionally persist into subsequent policy decisions rather than collapse toward the same recorded future [35], [48].

C. Boundary of Ego-Responsive Feedback

Ego-Responsive feedback establishes that an ego command changes the realized ego motion. It does not establish that surrounding traffic changes because of that intervention. Non-ego actors may remain on recorded trajectories, follow prespecified future layouts, or exhibit plausible motion without a demonstrated dependence on the altered ego trajectory.

Temporal co-occurrence is not sufficient evidence of reciprocal response. A following vehicle that appears to brake after an ego cut-in may already have been assigned that motion by the recorded or supplied future. An aggressive merge tests yielding only when the follower's behavior changes as a function of the intervened ego trajectory.

A distinct question concerns intervention-dependent non-ego behavior. Traffic-Reactive feedback is established when relevant surrounding-road-user behavior changes as a function of the counterfactual ego motion. The ego motion may be generated from a policy command or prescribed directly for the intervention test; Traffic-Reactive classification therefore provides no independent evidence of command-to-motion fidelity. The response may be demonstrated within a finite branch or a recurrent environment, with recurrence recorded separately.

V. Traffic-Reactive Worlds: Intervention-Dependent Traffic Response

Traffic-Reactive feedback is established when relevant surrounding road users change their behavior because the counterfactual ego motion changes the interaction. A highway merge, abrupt brake, or assertion of priority may alter the actions of nearby vehicles when it changes their spatial or temporal conflict. Demonstrating this feedback therefore requires evidence that the non-ego change is attributable to the ego intervention. Behavioral plausibility and the transfer of policy conclusions remain separate questions.

The counterfactual ego trajectory need not be generated from a policy command. It may instead be prescribed directly for an intervention test. Traffic-Reactive classification therefore does not establish the command-to-motion relation $u_t^{\mathrm{ego}} \rightarrow \tau_t^{\mathrm{ego}}$. Nor does it require recurrence: intervention-dependent traffic response may be demonstrated within a finite branch or a recurrent environment. Actors unlikely to be affected within the evaluated horizon provide a useful negative control and should remain stable up to ordinary stochastic variation.

A. From Forecasting to Counterfactual Traffic Response

Motion forecasting and traffic simulation may use the same driving data but support different claims. A motion forecaster estimates plausible multi-agent futures from a shared history. Traffic-Reactive simulation asks a counterfactual question: whether relevant non-ego behavior changes when the ego trajectory is changed while the preceding scene is held fixed. A plausible joint forecast does not answer this question unless the surrounding traffic is shown to depend on the intervened ego motion.

Data-driven traffic simulators provide one route to such dependence by repeatedly updating agent states from the evolving scene. TrafficSim, SimNet, BITS, and TrafficBots model multi-agent behavior over successive simulation steps rather than replaying a single recorded future [18], [19], [49], [50]. Their relevance to Traffic-Reactive evaluation depends on the reported simulation setting: the ego trajectory must be allowed to depart from the reference future, and the behavior of relevant surrounding actors must change in response to the resulting scene.

Autoregressive sequence models provide another mechanism for generating extended multi-agent futures. BehaviorGPT and SMART feed generated motion back into subsequent prediction [51], [52]. Recursive generation supports temporal continuation, but it does not by itself establish Traffic-Reactive feedback. A coherent joint rollout may still fail to change surrounding behavior when the ego trajectory is perturbed.

Let $\mathcal{R}_t$ denote the set of surrounding actors whose interaction with the ego vehicle may be affected over a horizon H, and let $a_{\mathcal{R}_t}^{-\mathrm{ego}}$ denote their action sequence over that horizon. For two ego trajectories $\tau^{(0)}$ and $\tau^{(1)}$ applied from the same preceding state, Traffic-Reactive dependence requires

$$\begin{aligned} &\mathbb{P}\Big(a_{\mathcal{R}_t}^{-\mathrm{ego}} \mid s_t, \mathrm{do}(\tau^{\mathrm{ego}} = \tau^{(0)})\Big) \\ &\neq \mathbb{P}\Big(a_{\mathcal{R}_t}^{-\mathrm{ego}} \mid s_t, \mathrm{do}(\tau^{\mathrm{ego}} = \tau^{(1)})\Big), \end{aligned} \tag{2}$$

for at least one interaction-relevant pair of trajectories. Equation (2) is an operational intervention criterion rather than a claim that the learned simulator recovers the full causal structure of real traffic. The initial state and other experimental conditions are matched while the ego trajectory is varied. Actors outside $\mathcal{R}_t$ over that horizon provide a negative control rather than being expected to respond indiscriminately.

Table II compares representative systems across Sections III–V by learned function, source of world evolution, demonstrated counterfactual capability, and remaining validation question. Supplementary Table S2 provides detailed system comparisons and the evidence supporting each assignment.

B. Sources of Traffic Response

Traffic reactivity does not require a particular model family. Classical microscopic traffic models compute vehicle actions from quantities such as headway, relative speed, and lane utility. Car-following models including IDM and Gipps-type formulations, together with lane-changing models such as MOBIL, can therefore produce intervention-dependent responses when ego motion changes the local traffic state [55]–[58]. Learned multi-agent models estimate traffic behavior from data, while hybrid systems combine learned and rule-based components. The response mechanism affects behavioral validity, but not the definition of Traffic-Reactive feedback.

Explicit interaction models can make the source of a response easier to inspect. InterSim represents relations such as yielding and precedence through predicted conflict structure before updating trajectories [59]. Controllable traffic generators such as CTG++ and CtRL-Sim allow interaction styles or target behaviors to be specified externally [60], [61]. Such controllability is distinct from endogenous reactivity. Requesting a cut-in shows that a scenario can be steered toward that event; showing that a follower changes its braking because the cut-in occurred establishes an intervention-dependent traffic response.

C. Representative Simulation Settings

Waymax illustrates why feedback regime should be assigned to a reported simulation setting rather than permanently to a platform. Surrounding actors may follow logged trajectories, intelligent driver model (IDM) controllers, or learned simulation agents [7]. A setting that queries observations against logged actor playback is Replay-Constrained; if it executes an ego command while actors remain logged, it may instead establish Ego-Responsive feedback. Settings in which relevant actors change their behavior with the counterfactual ego trajectory can establish Traffic-Reactive feedback. The Waymo Open Sim Agents Challenge (WOSAC) complements such tests by evaluating multi-agent behavior over full scenario rollouts [62], although rollout realism does not replace intervention attribution.

Hybrid traffic–sensor environments separate traffic evolution from observation generation. Bench2Drive-R combines an interactive behavior model with generative sensor rendering so that changed traffic states are reflected in the observations returned to a vision-based driving system [20]. SimScale perturbs ego trajectories in a reactive simulation setting and renders corresponding off-log observations for policy learning [38]. HUGSIM also combines neural scene rendering with vehicle and actor updates; its reported settings include IDM-based traffic behavior and aggressive actors for safety-critical interactions [37]. These systems illustrate that photorealistic rendering and traffic response are separate functions even when they operate in the same simulation setting.

Generative sensor models are beginning to represent non-ego motion within the generated future. CausalDrive conditions on the initial scene and an ego trajectory without supplying future oracle layouts for surrounding actors [54]. This design admits paired tests of traffic-response dependence, but the reported mechanism should not be classified as Traffic-Reactive until varying the supplied ego trajectory produces an attributable change in relevant non-ego behavior. Because the ego trajectory is supplied directly, such a test would not establish command-to-motion fidelity.

Plausible joint motion, systematic response to an ego intervention, and agreement with credible human-response distributions are three distinct empirical achievements. A model may satisfy any one without satisfying the others. Traffic-Reactive classification concerns the second relation; behavioral validity additionally depends on the third.

OmniDreams illustrates a complementary boundary. Its sensor generation is conditioned on visual history and abstract simulation state that includes dynamic-agent information [21]. Recurrent or action-conditioned rendering therefore does not by itself establish Traffic-Reactive feedback. The relevant question is whether the traffic process that supplies dynamic-agent states changes surrounding behavior as a function of the counterfactual ego motion.

Across the configurations reviewed above, the available evidence is uneven across three linked questions. For action-conditioned generators, changing a command or trajectory can alter the generated continuation, but this sensitivity does not establish that the realized ego motion matches the requested magnitude, timing, or physical feasibility. ACT-Bench measures this command–motion gap directly, while ReSim extends controllability tests beyond the expert trajectories that dominate ordinary driving data [35], [48]. The distinction

TABLE II
REPRESENTATIVE DESIGN FAMILIES CONTRIBUTING TO LEARNED CLOSED-LOOP DRIVING SIMULATION. THE COLUMNS SEPARATE LEARNED FUNCTION, SOURCE OF WORLD EVOLUTION, ILLUSTRATIVE COUNTERFACTUAL CAPABILITIES WHERE DEMONSTRATED, AND REMAINING VALIDATION QUESTION. ENTRIES SUMMARIZE FAMILY-LEVEL PATTERNS RATHER THAN PROPERTIES ESTABLISHED FOR EVERY LISTED SYSTEM.

System design	Learned function	Source of world evolution	Illustrative capability, where demonstrated	Remaining validation question
Neural sensor worlds: UniSim, MARS, NeuRAD, SplatAD [12]–[14], [36]	Observation renderer over reconstructed camera/LiDAR scenes	Ego and actor states are logged, edited, or supplied by another simulator	Policy behavior under changed viewpoint, occlusion, actor placement, and sensor appearance	Rendering fidelity cannot identify who chose the next actor state; traffic response belongs to the external transition.
Action-conditioned future models: GAIA-1, Drive-WM, Vista, DrivingWorld [15]–[17], [53]	One-shot or chunked future generator; sometimes an offline candidate evaluator	A control, maneuver, or trajectory conditions a finite visual continuation	Ego-contingent observations and comparison among candidate maneuvers	Action following, persistent state, and non-ego dependence must be tested separately; plausible joint motion is not attributable response.
Structured multi-agent worlds: TrafficSim, TrafficBots, BehaviorGPT, SMART, Waymax [7], [18], [19], [51], [52]	Recurrent behavior and transition model over agent/map state	Learned, rule-based, or hybrid agents repeatedly update the joint scene	Long rollouts, interaction modes, replanning, and policy evaluation at high query rates	Sensor consequences may be absent; paired interventions and independent behavior references are needed to establish response attribution and plausibility.
Hybrid traffic–sensor environments: HUGSIM, Bench2Drive-R, SimScale [20], [37], [38]	Reactive traffic or vehicle dynamics coupled to a learned renderer	A structured state is updated before observations are generated for the policy	Clearer separation of behavior, dynamics, rendering, and end-to-end policy effect	Bias in rule-based or learned responses can determine the policy conclusion even when rendering and recurrent execution are strong.
End-to-end generative sensor environments: CausalDrive, OmniDreams [21], [54]	Action- or state-conditioned sensor generation; traffic motion may be generated internally or supplied by a coupled traffic process	Initial scene, ego input, and available traffic state produce an off-log sensor continuation within a recurrent setting	Sensor futures reflecting realized ego and non-ego state; traffic response depends on the process supplying or generating actor motion	Tight coupling increases the need for joint tests of action adherence, response attribution, behavioral plausibility, persistence, and policy transfer.

is consequential: conditioning sensitivity is evidence that an intervention affects the output, whereas command execution requires agreement with the requested motion over the operating range being claimed.

Traffic-response evaluations reveal a related limit of scope. ReactSim-Bench evaluates surrounding-agent behavior under prescribed off-log ego trajectories, and generative systems such as CausalDrive make comparable intervention tests possible without future oracle layouts [54], [63]. Such experiments can support claims about braking, yielding, or negotiation within the tested encounter families. They do not establish that the same local response mechanism composes across vehicles and time to reproduce corridor propagation or network performance. Larger scenes and longer rollouts expand the simulated context, but not automatically the transportation scale of the supported claim.

The consequences become visible at the policy level. In reactive nuPlan studies, changing the surrounding-traffic model alters benchmark scores and can change planner rankings [8], [9]. This result does not identify a single correct simulator. It shows that planner comparisons inherit assumptions from the simulated traffic process and that a policy conclusion is more robust when it persists across environments constructed from materially different behavioral assumptions. Together, these observations connect the feedback represented by an environment to the evidence needed for the resulting policy and transportation claims, without treating either as a model-maturity scale.

D. Evidence Boundary

Traffic-Reactive classification requires attributable dependence between the ego intervention and the response of relevant surrounding actors. It does not establish that the response distribution is realistic. A simulator in which every follower immediately yields to a cut-in can be strongly intervention-dependent while poorly representing human driving. Behavioral plausibility therefore requires separate evidence about response type, latency, severity, and variability.

Policy validity is separate from traffic reactivity: sensitivity to a traffic model does not determine whether the resulting ranking is credible. That conclusion requires an independent behavioral or policy reference. Section VII develops the corresponding attribution, plausibility, timing, and transfer evidence.

E. From Local Response to Transportation Effects

Traffic-Reactive evidence is usually established within individual encounters, but repeated local responses can affect larger traffic systems. Braking, merging, and gap acceptance can alter disturbance propagation, bottleneck capacity, and traffic stability. Ring-road experiments show that changing the longitudinal behavior of a single automated vehicle can suppress stop-and-go waves [31], while mixed-autonomy and cooperative-merging studies relate local vehicle behavior to string stability, throughput, and network dynamics [32], [33]. These results motivate a separate transportation-scale question: an attributable response in one encounter does not establish that the same response mechanism produces credible corridor or network dynamics.

VI. APPLICATIONS AND SIMULATION REQUIREMENTS

The simulator required for a driving experiment is determined by the conclusion sought from it. Reconstructed sensor views can reveal observation-level failures near recorded trajectories. Candidate-maneuver comparison requires faithful command-to-motion execution. Reciprocal-interaction studies require attributable changes in surrounding traffic, and claims about adaptation across decisions require policy recurrence.

Behavioral validity, transfer, and transportation-scale behavior remain separate empirical questions.

A. Policy Evaluation across Feedback Regimes

Recorded driving data provide an efficient baseline for planner evaluation. A planner can be initialized from a logged scene, generate a candidate trajectory, and be scored against recorded behavior or reconstructed scene geometry. NAVSIM uses logged scene context and fixed future traffic for efficient non-reactive planner evaluation [45]. Pseudo-Simulation instead approximates potential future ego states through synthetic observations without requiring sequential interactive simulation [64]. Both provide scalable alternatives to fully interactive evaluation, but neither establishes reciprocal traffic response to an off-log ego maneuver.

Neural sensor simulation extends evaluation beyond the recorded sensor stream. UniSim renders observations under altered ego poses and scene configurations, while NeuroNCAP uses reconstructed sensor-realistic scenes to expose end-to-end driving systems to safety-critical configurations [12], [65]. Such tests can reveal viewpoint sensitivity, perception failures, and tracking drift that trajectory-only evaluation may miss. Their supported conclusion concerns the policy's response to synthesized observations, not command execution or reciprocal traffic behavior.

Candidate-maneuver evaluation requires a different dependency. When several ego commands are compared through generated futures, the realized ego motion must respond to the requested maneuver. Drive-WM generates alternative multi-view futures under candidate driving maneuvers and evaluates the resulting trajectories for planning [16]. ReSim extends action-conditioned world simulation to non-expert and hazardous ego behaviors that are sparse in ordinary driving demonstrations [48]. These settings can support Ego-Responsive candidate comparison without requiring recurrent policy interaction.

Reciprocal-interaction claims require attributable traffic response. Reported reactive settings in Waymax and Bench2Drive-R, together with reactive nuPlan variants such as nuPlan-R, allow surrounding behavior to depend on the changed traffic state or counterfactual ego trajectory [7], [9], [20]. Collision, progress, and rule-compliance metrics in these settings are joint outcomes of the tested planner and the assumed traffic behavior model.

This dependence can materially affect planner comparison. Studies using reactive nuPlan settings report that changing the surrounding traffic behavior model can alter benchmark scores and reverse pairwise planner rankings [8], [9]. Conclusions about planner performance are therefore conditional on the simulated traffic response. Such sensitivity does not identify which traffic model yields the most credible ranking; that question requires separate validation.

B. Policy Learning with Learned World Models

World models contribute to policy learning through several distinct mechanisms. Learned dynamics can provide imagined transitions for optimization, future prediction can supply auxiliary supervision, and reconstructed environments can expose a policy to states that are rare in recorded demonstrations. These uses should not be conflated with training inside the same type of recurrent simulator.

Think2Drive optimizes driving policies with a learned world model in CARLA, while Imagine-2-Drive uses predictive world modeling within trajectory optimization [66], [67]. Goff *et al.* train an end-to-end driving policy using on-policy simulation that includes a learned world model [68]. In such settings, model error affects the generated future and can propagate into the policy updates produced during optimization.

World modeling can also support learning without serving as the recurrent training environment. DriveVLA-W0 uses future-image prediction as dense self-supervision for vision–language–action learning [69]. Its use of predicted future observations provides a training signal without requiring recurrent policy interaction inside the world model.

Rendered and reactive simulation can further expand the states available during training. RAD performs reinforcement-learning post-training in photorealistic 3D Gaussian Splatting environments [70]. SimScale combines perturbed ego trajectories, a reactive environment, and neural rendering to generate off-log training observations, then uses the synthesized data to improve driving policies [38]. Learned traffic agents can themselves be refined through reinforcement learning in multi-agent simulation [71].

Training amplifies simulator error because optimization can exploit regularities that increase simulated reward without corresponding to sound driving behavior. Rendering artifacts, permissive dynamics, or predictable traffic responses may become shortcuts for the learned policy [7], [34]. A training gain becomes more persuasive when it persists in an independently constructed evaluation environment with different sensing, dynamics, or traffic assumptions.

C. Scenario Generation and Stress Testing

Stress testing begins by constructing conditions that are hazardous, rare, or poorly represented in ordinary driving logs. TrafficGen and Scenario Dreamer generate traffic scenes, actor configurations, and road layouts that can provide controlled initial conditions for evaluation [72], [73]. Such scenes must still satisfy map, geometry, and motion constraints before they can support a meaningful safety test.

Controllable traffic models provide more targeted variation. CTG++, CtRL-Sim, ProSim, and CCDiff allow traffic behavior or interaction objectives to be guided through language, optimization costs, or structured conditioning [60], [61], [74], [75]. This control can concentrate evaluation on selected hazards rather than relying on their frequency in naturalistic data. Controllability specifies the event being generated; it does not by itself establish how traffic responds to the tested policy.

The required feedback depends on the failure mechanism under study. Replay-Constrained environments can support sensor-level stress tests involving altered viewpoints, occlusions, or inserted hazards; NeuroNCAP is an example of this sensor-realistic use [65]. Ego-Responsive feedback is

required when the test concerns whether braking, steering, or another candidate maneuver produces the requested recovery motion. Failure mechanisms involving yielding, merging, cut-in response, or other reciprocal behavior require an attributable traffic response; Waymax provides an executable example, whereas ReactSim-Bench and CausalDrive expose useful off-log protocols or mechanisms whose intervention attribution must be assessed separately [7], [54], [63]. Recurrence becomes necessary when the conclusion concerns how these consequences accumulate across successive policy decisions.

Controlled variation of quantities such as initial gap, relative speed, visibility, or sensing delay can characterize an operational failure boundary within the tested scenario family. Such parameter sweeps identify conditions under which policy behavior changes, but they do not by themselves establish the prevalence of those failures or their corridor- or network-scale consequences.

D. Matching Requirements to the Intended Claim

Application requirements follow the intended claim. Observation-level testing requires credible sensor synthesis; candidate comparison requires command-dependent ego motion; reciprocal-interaction claims require attributable traffic response; and multi-step adaptation requires recurrence. Transfer and transportation-scale validity must then be tested in the domains and at the scales named by the conclusion.

The requirements are not cumulative. A Traffic-Reactive test may prescribe the counterfactual ego trajectory directly and therefore provide no evidence of command-to-motion fidelity. Conversely, a recurrent observation loop may repeatedly return policy observations while non-ego traffic remains Replay-Constrained. Feedback regime identifies the demonstrated intervention dependency, whereas recurrence records whether its consequences persist across policy decisions. Different applications therefore require different feedback depths, but demonstrating the required dependency does not by itself establish that the resulting response is valid.

VII. Evaluating Simulator Validity

Simulator validity is claim-dependent. A neural renderer may reproduce sensor observations near recorded trajectories without supporting claims about reciprocal traffic interaction. A structured traffic simulator may exhibit intervention-dependent non-ego behavior without producing photorealistic sensor data. A reactive environment may still yield unstable planner rankings if its traffic behavior differs from the reference domain. Validation evidence must therefore be matched to the result being claimed rather than reduced to a single realism score.

Fig. 4 distinguishes four claim-specific validity questions: observation fidelity, ego controllability, traffic-response validity, and closed-loop policy validity. These questions are related but not cumulative; the evidence required depends on the simulator components and policy conclusions under study. Validation requirements follow the dependencies of the claimed result, which should be stated explicitly. Transportation scale is assessed separately: encounter-, corridor-, and network-level claims require evidence over the corresponding spatial extent, time horizon, and traffic population.

Table III organizes representative datasets, simulation platforms, and benchmarks by the validity questions they can address. Recorded driving data provide references for sensor appearance, geometry, and observed behavior distributions, but cannot reveal the response that would have followed an unobserved intervention. Executable simulators permit controlled counterfactual variation but inherit assumptions from their sensor, vehicle, and traffic models. Component benchmarks isolate particular dependencies without establishing broader simulator validity.

A. Observation Fidelity: Policy-Relevant Sensor Evidence

Observation fidelity asks whether generated sensor data represent the simulated scene accurately enough for the intended downstream use. Image metrics such as peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and perceptual distance characterize complementary aspects of appearance, while temporal and video-distribution metrics assess consistency across frames. Neural sensor simulators such as UniSim and NeuRAD additionally evaluate geometric or LiDAR properties through depth, intensity, ray-drop behavior, and point-cloud agreement [12], [14].

Sensor similarity alone does not establish that a downstream driving system behaves consistently. Matched real and simulated observations can instead be passed through detection, forecasting, mapping, or planning models to measure whether sensor differences alter task outputs. Neural LiDAR studies show that similar aggregate perception performance can conceal disagreements on matched examples [43]. Camera-rendering studies likewise show that downstream task comparisons provide information beyond image similarity metrics [44].

Driving-specific benchmarks such as WorldLens broaden this evaluation toward geometry, action following, and downstream planning behavior [81]. WorldExam provides a broader, non-driving-specific example of evaluation beyond visual appearance [82]; its reactivity measures should not be interpreted as evidence of traffic-response validity in autonomous-driving simulation.

The supported claim remains bounded by the measured sensor, viewpoint, and downstream-task envelope. High visual fidelity does not establish correct state evolution, ego controllability, or traffic response. Conversely, photorealistic sensing is not a necessary validation target for a structured-state simulator when neither the tested policy nor the claimed result depends on rendered sensor observations.

B. Ego Controllability: Command-to-Motion Fidelity

Ego controllability asks whether a requested command produces the corresponding motion in the generated or simulated future. For a command issued at time t and evaluated over a horizon H, the relevant relation is

$$u_t^{\mathrm{ego}} \rightarrow \tau_{t:t+H}^{\mathrm{ego}}.$$

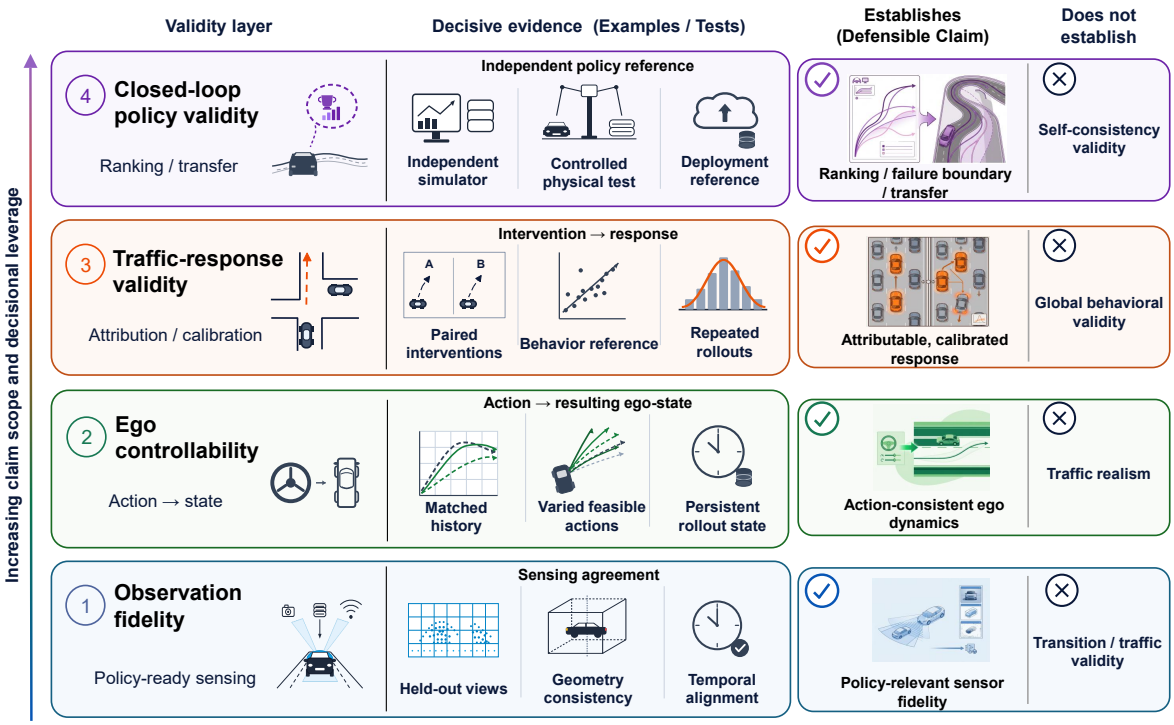

Fig. 4. Claim-specific validity questions for learned driving simulation. Each row links a scientific question to evidence that can support the corresponding claim and to a nearby inference that remains unsupported. The four rows are related but not cumulative; validation requirements depend on the components and dependencies involved in the claimed result. Transportation scale is assessed separately.

Evaluation should measure command-specific trajectory error and physical feasibility over the maneuver range for which controllability is claimed.

Paired commands provide a first test of intervention sensitivity. Starting from the same history, braking, lane keeping, and lane changing should produce distinguishable motions in the requested directions. Such separation is necessary but not sufficient for command adherence: two commands may generate different trajectories while both deviate from their intended motions. ACT-Bench compares requested trajectories with motion recovered from generated futures and reports substantial action-fidelity errors among evaluated driving world models [35].

Off-log testing examines how far this relation extends beyond ordinary demonstrations. ReSim introduces non-expert and hazardous ego trajectories that are sparse in natural driving logs and evaluates action following and motion consistency under these conditions [48]. Responses to infeasible commands should be reported separately and should not be counted as successful command adherence.

Recurrence remains a separate property. A finite branch can establish command-to-motion fidelity without carrying the resulting ego state into another policy decision. Conversely, a recurrent observation loop may persist even when command-to-motion fidelity has not been independently established.

C. Traffic-Response Validity: Intervention Dependence and Plausibility

Traffic-response validity separates two questions. The first is whether relevant non-ego behavior depends systematically on the counterfactual ego trajectory. The second is whether the resulting response is consistent with credible road-user behavior. Intervention dependence is required to establish Traffic-Reactive feedback; behavioral correctness requires additional evidence.

Paired interventions test the first relation. Rollouts begin from matched initial scenes, and the counterfactual ego trajectory is varied while other experimental conditions are controlled as far as practicable. For stochastic models, comparisons should be repeated across matched or multiple random seeds. Actors that are unlikely to be affected by the ego intervention within the evaluated horizon can serve as comparison cases. Equation (2) expresses the intervention dependence tested over the ego-relevant actor set.

ReactSim-Bench supplies a fixed off-log autonomous-vehicle trajectory independently of the surrounding-agent model and measures the resulting agent behavior across interaction conditions and re-inference schedules [63]. This protocol probes response under a prescribed off-log trajectory, but it is not a matched intervention test: without an alternative ego trajectory from the same preceding condition, the reported change cannot be attributed to the ego intervention alone. A paired extension would test that relation without requiring command-to-motion fidelity.

Displacement error against the recorded future answers a different question. Average Displacement Error (ADE) and Final Displacement Error (FDE) measure agreement with the observed trajectory, whereas a legitimate response to an unobserved ego maneuver may depart from that trajectory. Surrogate safety measures such as Time to Collision (TTC),

TABLE III
EVIDENCE SOURCES FOR EVALUATING LEARNED DRIVING ENVIRONMENTS. EACH SOURCE ANSWERS A DIFFERENT QUESTION; NO RECORDED DATASET OR COMPONENT BENCHMARK ALONE VALIDATES A POLICY-INDUCED COUNTERFACTUAL WORLD.

Evidence source	Representative resources	Question addressed collectively	Boundary of the resulting evidence
Recorded multimodal driving corpora	nuScenes [76], Waymo Open Perception [77], and Waymo Open Motion [78]	Does a model reconstruct sensor appearance, geometry, and ordinary trajectory distributions on held-out real scenes?	The log contains only the future that followed the recorded action; it cannot reveal the same encounter under another policy decision.
Executable simulation platforms	CARLA [5], nuPlan [3], Waymax [7], MetaDrive [79], and SUMO [80]	Can policies act repeatedly under controlled dynamics, traffic, maps, signals, and scenario variation?	The policy conclusion inherits the platform's sensor, vehicle, and behavior assumptions; executability alone provides no test of realism or transfer.
Near-log policy evaluation	NAVSIM [45] and Pseudo-Simulation [64]	Can planners be screened efficiently against real scene context and controlled perturbations?	Traffic is fixed or only approximately adjusted, so negotiation and response-dependent policy rankings remain unresolved.
Component and functional diagnostics	ACT-Bench [35], WorldLens [81], and WorldExam [82]	Does generation preserve appearance, geometry, requested ego motion, planner-relevant cues, and reactive scene behavior?	ACT-Bench isolates requested-versus-realized ego motion; WorldLens adds generated-versus-real planner comparisons and reported closed-loop route/adherence tests; WorldExam tests generic subject control and reactivity rather than driving-specific traffic closure.
Interactive traffic and policy benchmarks	WOSAC [62], ReactSim-Bench [63], nuPlan-R [9], and Bench2Drive-R [20]	Do road users evolve coherently after off-log intervention, and does their response change planner behavior?	Attribution, behavior plausibility, and transfer still depend on matched interventions, repeated samples, and an independent reference.

Post-Encroachment Time (PET), and Deceleration Rate to Avoid a Crash (DRAC) characterize conflict proximity and evasive demand [6], [83]. These outcome measures become evidence of traffic response only when paired with an intervention design that attributes the change, and evidence of behavioral plausibility only when compared with an appropriate reference.

Plausibility requires a behavioral reference. Repeated stochastic rollouts can characterize response type, latency, severity, and outcome, but variability alone is not evidence of realism. WOSAC provides rollout-based evidence about behavioral distributions under recorded scenarios [62], but it does not supply the counterfactual response that would have followed a different ego intervention. Rollout-distribution evidence therefore complements rather than replaces paired-intervention tests.

Generative systems such as CausalDrive admit the same paired test: vary the supplied ego trajectory from a matched scene and measure whether relevant non-ego motion changes systematically [54]. Omitting future oracle layouts makes this test possible, but does not itself provide its result.

Interaction families should also be reported separately. Cut-ins probe following and emergency response, highway merges probe gap acceptance and yielding, unprotected turns probe priority negotiation, and pedestrian conflicts probe braking around vulnerable road users. Aggregate scores can conceal failures confined to one interaction mechanism.

Response timing is part of behavioral validity. A qualitatively correct maneuver may still occur too early, too late, or with unstable oscillation. Agent update frequency, modeled reaction delay, and simulation time step can alter the resulting interaction. ReactSim-Bench, for example, varies agent re-inference frequency when evaluating off-log response [63]. These quantities should accompany reported response results when they affect the tested interaction.

D. Closed-Loop Policy Validity: Ranking and Training Transfer

Policy-level evidence asks whether simulator-derived conclusions remain stable under an independently constructed reference. The conclusion may concern a planner ranking, a failure boundary, or an improvement attributed to simulator-based training. Agreement across simulators supports robustness to environment assumptions; physical or observational evidence is still required for claims about real-world validity.

Planner comparison can evaluate the same policies and scenarios in the candidate simulator and in a separately constructed environment. Reactive nuPlan studies illustrate why this matters: changing the surrounding traffic behavior model can alter benchmark scores and reverse pairwise planner rankings [8], [9]. These results establish sensitivity to traffic assumptions, not the correctness of any particular simulator. Ranking robustness is strengthened when conclusions persist across environments built from different behavior, sensing, or dynamics assumptions.

Training transfer poses the corresponding question for learned policies. RAD and SimScale report policy improvements from training or refinement with rendered simulation data [38], [70]. Evaluation on held-out scenarios and independently constructed environments can test whether those gains depend on the simulator that produced the training signal. Evidence from physical testing or real driving data is needed before interpreting cross-simulator agreement as real-world transfer.

Timing can also alter policy-level conclusions. Hardware-in-the-loop and driver-in-the-loop experiments require wall-clock deadlines, whereas offline evaluation may run slower than real time if simulated timestamps, sensing delay, actuation delay, and decision cadence remain consistent. Real-time generators such as OmniDreams target applications with tighter computational deadlines [21]. In either setting, temporal misalignment

can change the effective policy input or action and thereby alter the comparison. Supplementary Table S1 summarizes recurring failure modes and their effect on policy assessment.

E. Transportation-Scale Validity: Matching Evidence to Claims

Transportation scale is separate from the validity question being tested. An attributable response in a single encounter can support a local interaction claim without establishing corridor propagation or network performance. Increasing scene size does not by itself extend the supported scale.

Corridor claims require evidence that repeated local responses compose into credible traffic propagation under stated demand and boundary conditions. Relevant quantities include shockwave speed, queue growth and dissipation, bottleneck discharge, and throughput. Kinematic-wave theory, calibrated microscopic references, and field observations provide independent comparisons for these phenomena [84], [85]. Ring-road experiments and bounded highway studies further demonstrate how vehicle-level control and mixed-traffic penetration can alter wave propagation and throughput [31], [32].

Network claims require additional evidence on route structure, signal control, demand, travel time, throughput, and automation penetration at the corresponding scale [33], [85]. Agreement in one aggregate quantity is insufficient when the claimed effect depends on local response mechanisms that have not been validated.

Table IV relates common simulation claims to the evidence required to support them and to nearby measurements that answer a different question. Sensor-facing claims require held-out geometric, temporal, and downstream-task evidence; candidate-maneuver claims require command-specific motion adherence and feasibility; traffic-interaction claims require intervention-dependent response together with behavioral reference evidence; and policy-ranking or training claims require independently constructed references appropriate to the intended conclusion. Corridor and network claims additionally require scale-matched traffic evidence.

Recurrence and timing matter when the conclusion depends on repeated policy interaction or temporal execution. The empirical scope of a simulation result is therefore defined by the intervention dependencies, recurrence conditions, and transportation scales included in its validation.

VIII. Challenges and Research Directions

Current evidence leaves five questions particularly consequential for learned driving simulation: how long intervention consequences remain coherent, when vehicle-level responses compose into credible traffic dynamics, how timing changes closed-loop outcomes, how rare-event uncertainty should affect optimization, and under which environment assumptions policy conclusions reproduce. These questions concern the reliability of conclusions drawn from simulation, not simply the capacity to generate longer or larger scenes.

A. Persistent State over Long Horizons

Long rollouts expose errors that short generated sequences can hide. Autoregressive prediction can accumulate pose error, geometric inconsistency, identity drift, and loss of interaction history across successive steps [17], [86]. An actor may remain visually plausible while drifting from its lane, losing continuity across views, or producing subsequent motion that no longer reflects an earlier yielding interaction.

Long-horizon simulation therefore requires the information needed by later decisions to remain consistent under repeated off-log updates. This information may be represented through explicit 3D geometry, structured scene state, or latent memory; the relevant question is whether actor identity, pose, motion, signal state, and interaction context remain coherent over the required horizon. HorizonDrive explores rollout-oriented training and recovery mechanisms for reducing long-horizon autoregressive error [86]. Open problems include measuring state drift independently of visual quality, recovering from accumulated error without erasing the consequences of earlier interventions, and determining the horizon over which interaction information remains reliable.

B. From Vehicle-Level Response to Traffic Dynamics

Transportation-scale claims depend on how vehicle-level responses accumulate across vehicles and time. Local braking, merging, and gap acceptance can affect shockwave propagation, queue formation, bottleneck capacity, and network delay [31]–[33]. Larger scenes and more generated agents are useful only if the resulting traffic dynamics are evaluated at the corresponding scale.

Recent multi-agent models broaden scenario and interaction coverage. AutoWorld uses unlabeled driving observations to improve scalable multi-agent traffic simulation, while CounterScene introduces structured counterfactual interventions for safety-critical encounter generation [87], [88]. Mixed-autonomy traffic requires distinct response models for human-driven and automated vehicles. Urban simulation further introduces heterogeneity across cyclists, pedestrians, and other road users. The unresolved question is whether vehicle-level responses, when repeated across vehicles and time, reproduce observed corridor- and network-level traffic dynamics. Wave speed, queue growth and dissipation, throughput, and travel time provide measurable links between local behavior and traffic-scale outcomes [84], [85].

C. Temporal Consistency in Closed-Loop Simulation

Closed-loop simulation depends on the temporal relation among policy decisions, vehicle dynamics, traffic updates, and sensor observations. Offline evaluation may run slower than wall clock if simulated timestamps and modeled delays remain consistent, whereas hardware-in-the-loop and driver-in-the-loop applications additionally impose wall-clock deadlines. Evidence from connected-vehicle and V2X systems illustrates related timing failures: communication or processing delay can change the state available to a controller and alter the resulting interaction [89], [90].

TABLE IV
EVIDENCE REQUIREMENTS AND SUPPORTED SCOPE ACROSS VALIDATION LAYERS. THE THIRD COLUMN HIGHLIGHTS MEASUREMENTS THAT REMAIN INFORMATIVE BUT DO NOT JUSTIFY THE STATED CLAIM ON THEIR OWN.

Simulation claim	Decisive evidence required	Insufficient evidence alone	Supported claim scope
Observation fidelity (Sensor credibility)	Geometric and temporal consistency across held-out viewpoints; multi-view sensor agreement tied to authoritative state transitions [12], [14], [43], [44].	In-distribution visual similarity (PSNR/SSIM/FID); task agreement evaluated without explicit underlying state transitions.	Perception benchmarking and near-log policy rollouts within measured sensor/viewpoint envelopes.
Ego controllability (Action consistency)	Target-to-realized trajectory tracking across commanded ranges (steering, speed, curvature); physical and kinematic feasibility bounds [16], [17], [35], [81].	Presence of action tokens; visual clip diversity without physical trajectory verification; oracle future actor layouts.	Candidate maneuver evaluation and single-step action execution; multi-step rollouts if state persists.
Traffic-response validity (Multi-agent interaction)	Paired counterfactual ego interventions from matched initial scenes; repeated sampling of response type, latency Δt, and severity; synchronized joint states [7], [19], [54], [63].	Open-loop rollout plausibility; prompt-conditioned diversity under fixed ego paths; unreactive log replay.	Interactive negotiation, yielding, and conflict evaluation within tested interaction families and behavioral models.
Policy-verdict validity (Ranking & transfer)	Ranking stability and transfer across held-out scenario suites and independent simulator backends; validation of sim-trained policies [8], [9], [38], [70].	Benchmark performance inside a single simulator susceptible to permissive physics, accommodating traffic, or policy exploitation.	Policy selection, ranking, and training conclusions within validated scenario and environment distributions.
Transportation scale (Corridor & network)	Composition of local responses across vehicles and horizons under stated demand; validation of shockwaves, queue dissipation, and delay against field data [31]–[33], [39], [84], [85].	Isolated encounter realism; aggregate flow matching that fails to test microscopic causal mechanisms.	Network-level throughput, queue dynamics, and delay claims bounded by the highest scale empirically evaluated.

Hybrid environments can separate high-frequency state evolution from more expensive neural rendering, as in Bench2Drive-R, while systems such as OmniDreams target real-time sensor generation [20], [21]. A remaining challenge is to determine how asynchronous component updates alter policy behavior and to distinguish synchronization-induced failures from errors in the underlying dynamics or traffic model. For closed-loop policy evaluation, real-time generation is useful only when the state, observation, and action presented to the policy remain temporally consistent.

D. Rare-Event Learning under Model Uncertainty

Many safety-critical interactions are sparsely represented in recorded driving data, especially near-collisions, aggressive negotiations, and emergency recovery maneuvers. Learned simulators must therefore generate outcomes in regions with weaker empirical support, while policy training and stress testing often place substantial weight on precisely these cases. ReSim expands action coverage with non-expert and emergency trajectories [48], but broader coverage does not remove uncertainty about the corresponding counterfactual dynamics.

A useful distinction is between irreducible variability in road-user behavior and epistemic uncertainty arising from limited model knowledge [91]. Model ensembles provide one practical signal of predictive uncertainty [92], although disagreement is a proxy whose quality depends on ensemble diversity; varying model architectures can improve calibration and robustness under distribution shift relative to fixed-architecture ensembles [93]. The open methodological question is how this uncertainty should change a policy update or stress-test conclusion. Poorly supported transitions can be down-weighted, excluded, or routed to a separate simulator or controlled experiment, but each choice changes the optimization target. Reporting whether a failure, ranking, or training gain survives those choices would be more informative than attaching uncertainty scores to otherwise unchanged rollouts.

E. Cross-Environment Reproducibility

Simulator-derived policy conclusions can change with the renderer, vehicle dynamics, traffic behavior model, or scenario distribution used for evaluation. This dependence is already visible in reactive planning benchmarks, where changing the surrounding traffic model can reverse pairwise planner rankings [8], [9]. Agreement across separately implemented environments with materially different modeling assumptions strengthens evidence that a conclusion is robust to those assumptions, but it does not establish real-world validity without physical or observational support.

The more informative question is where that agreement breaks down. Interaction type, traffic density, response delay, behavioral heterogeneity, sensing error, and scenario difficulty may each alter a planner ranking or erase an apparent training gain. Cross-environment studies should therefore report the conditions under which conclusions reverse, including the interaction family, density, delay, sensing assumptions, and traffic-model assumptions associated with the reversal. Those boundaries are part of the result, rather than exceptions hidden by an aggregate score.

IX. CONCLUSION

Closed-loop policy evaluation fundamentally asks counterfactual questions that static logs cannot answer: what consequences would follow an unrecorded decision, and how would surrounding road users respond? Rather than categorizing driving world models by neural architectures, this survey establishes a functional formulation based on the feedback dependencies an environment actually closes under an intervention. We separate simulation into three distinct regimes—Replay-Constrained (prescribed behavioral futures), Ego-Responsive (controllable ego motion), and Traffic-Reactive (reciprocal multi-agent response)—and decouple these causal relations

from policy recurrence, model families, and operational software roles. This structure prevents sensory continuation, command execution, and reciprocal traffic interaction from being conflated under the ambiguous umbrella of "closed-loop simulation."

Beyond feedback mechanics, we articulate the empirical evidence required to validate policy conclusions, decoupling sensor-level observation fidelity and ego controllability from attributable traffic reactivity and policy transfer. Furthermore, we ground learned simulators in transportation science, emphasizing that validity demonstrated within isolated encounters cannot be extrapolated to corridor propagation or network-level performance without scale-matched evidence. Ultimately, a driving world model is not a passive scene generator but an active experimental instrument. Its scientific utility is determined not by visual plausibility alone, but by which counterfactual claims its demonstrated feedback dependencies can defensibly support.

REFERENCES

[1] M. O'Kelly, A. Sinha, H. Namkoong, R. Tedrake, and J. C. Duchi, "Scalable End-to-End Autonomous Vehicle Testing via Rare-Event Simulation," in *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/653c579e3f9ba5c03f2f2f8cf4512b39-Abstract.html

[2] H. Alghodhaifi and S. Lakshmanan, "Autonomous Vehicle Evaluation: A Comprehensive Survey on Modeling and Simulation Approaches," *IEEE Access*, vol. 9, pp. 151 531–151 566, 2021.

[3] H. Caesar *et al.*, "NuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), ADP3*, 2021. [Online]. Available: https://arxiv.org/abs/2106.11810

[4] Q. Zhang *et al.*, "TrajGen: Generating Realistic and Diverse Trajectories With Reactive and Feasible Agent Behaviors for Autonomous Driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 24 474–24 487, 2022.

[5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," in *Proc. Conf. Robot Learn. (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 78. PMLR, 2017, pp. 1–16.

[6] W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A Survey on Safety-Critical Driving Scenario Generation—A Methodological Perspective," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 7, pp. 6971–6988, 2023.

[7] C. Gulino *et al.*, "Waymax: An Accelerated, Data-Driven Simulator for Large-Scale Autonomous Driving Research," in *Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 7730–7742.

[8] S. Hagedorn, L. Donkov, A. Distelzweig, and A. P. Condurache, "When Planners Meet Reality: How Learned, Reactive Traffic Agents Shift nuPlan Benchmarks," arXiv, 2025. [Online]. Available: https://arxiv.org/abs/2510.14677

[9] M. Peng, R. Yao, X. Guo, and J. Ma, "nuPlan-R: A Closed-Loop Planning Benchmark for Autonomous Driving via Reactive Multi-Agent Simulation," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2026, pp. 702–709.

[10] Q. Li *et al.*, "ScenarioNet: Open-Source Platform for Large-Scale Traffic Scenario Simulation and Modeling," in *Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 3894–3920.

[11] X. Hu, S. Li, T. Huang, B. Tang, R. Huai, and L. Chen, "How Simulation Helps Autonomous Driving: A Survey of Sim2real, Digital Twins, and Parallel Intelligence," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 593–612, 2024.

[12] Z. Yang *et al.*, "UniSim: A Neural Closed-Loop Sensor Simulator," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 1389–1399.

[13] Z. Wu *et al.*, "MARS: An Instance-Aware, Modular and Realistic Simulator for Autonomous Driving," in *Proc. CAAI Int. Conf. Artif. Intell. (CICAI)*, ser. Lecture Notes in Artificial Intelligence, vol. 14473, 2024, pp. 3–15.

[14] A. Tonderski, C. Lindström, G. Hess, W. Ljungbergh, L. Svensson, and C. Petersson, "NeuRAD: Neural Rendering for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 14 895–14 904.

[15] A. Hu *et al.*, "GAIA-1: A Generative World Model for Autonomous Driving," arXiv, 2023. [Online]. Available: https://arxiv.org/abs/2309.17080

[16] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the Future: Multiview Visual Forecasting and Planning with World Model for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 14 749–14 759.

[17] S. Gao *et al.*, "Vista: A Generalizable Driving World Model with High Fidelity and Versatile Controllability," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 91 560–91 596.

[18] S. Suo, S. Regalado, S. Casas, and R. Urtasun, "TrafficSim: Learning to Simulate Realistic Multi-Agent Behaviors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 10 395–10 404.

[19] Z. Zhang, A. Liniger, D. Dai, F. Yu, and L. Van Gool, "TrafficBots: Towards World Models for Autonomous Driving Simulation and Motion Prediction," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 1522–1529.

[20] J. You, X. Jia, Z. Zhang, Y. Zhu, and J. Yan, "Bench2Drive-R: Turning Real World Data into Reactive Closed-Loop Autonomous Driving Benchmark by Generative Model," arXiv, 2024. [Online]. Available: https://arxiv.org/abs/2412.09647

[21] NVIDIA *et al.*, "NVIDIA OmniDreams: Real-Time Generative World Model for Closed-Loop Autonomous Vehicle Simulation," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.03159

[22] Y. Guan *et al.*, "World Models for Autonomous Driving: An Initial Survey," *IEEE Trans. Intell. Veh.*, pp. 1–17, 2024.

[23] T. Feng, W. Wang, and Y. Yang, "A Survey of World Models for Autonomous Driving," arXiv, 2025. [Online]. Available: https://arxiv.org/abs/2501.11260

[24] S. Tu *et al.*, "The Role of World Models in Shaping Autonomous Driving: A Comprehensive Survey," arXiv, 2025, rev. 2026. [Online]. Available: https://arxiv.org/abs/2502.10498

[25] R. Zeng and Y. Dong, "Latent World Models for Automated Driving: A Unified Taxonomy, Evaluation Framework, and Open Challenges," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2603.09086

[26] H. Huang *et al.*, "World Models for Autonomous Driving: From Future Generation to Decision Making," SSRN preprint, 2026. [Online]. Available: https://doi.org/10.2139/ssrn.6827179

[27] G. Sarkandi and C. Alecsandru, "Balancing Safety, Efficiency, and Sustainability: A Survey of Modeling Tools for Autonomous Vehicles Interactions," *IEEE Trans. Intell. Transp. Syst.*, vol. 27, no. 7, pp. 7397–7413, 2026.

[28] C. Oefinger, F. R. Schäfer, K. Moller, M. Piccinini, and J. Betz, "Validate the Dream Before You Trust Its Verdict: Admissibility for World-Model Simulators," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2607.07196

[29] Y. Li *et al.*, "Choose Your Simulator Wisely: A Review on Open-Source Simulators for Autonomous Driving," *IEEE Trans. Intell. Veh.*, vol. 9, no. 5, pp. 4861–4876, 2024.

[30] Q. Song, H. Ye, M. Harman, and F. Sarro, "Generative AI for Testing of Autonomous Driving Systems: A Survey," *ACM Trans. Softw. Eng. Methodol.*, 2026, in press.

[31] R. E. Stern *et al.*, "Dissipation of Stop-and-Go Waves via Control of Autonomous Vehicles: Field Experiments," *Transp. Res. Part C: Emerg. Technol.*, vol. 89, pp. 205–221, 2018.

[32] A. Talebpour and H. S. Mahmassani, "Influence of Connected and Autonomous Vehicles on Traffic Flow Stability and Throughput," *Transp. Res. Part C: Emerg. Technol.*, vol. 71, pp. 143–163, 2016.

[33] Z. Gu, Z. Wang, Z. Liu, and M. Saberi, "Network Traffic Instability with Automated Driving and Cooperative Merging," *Transp. Res. Part C: Emerg. Technol.*, vol. 138, p. 103626, 2022.

[34] D. Ha and J. Schmidhuber, "Recurrent World Models Facilitate Policy Evolution," in *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 2451–2463.

[35] H. Arai, K. Ishihara, T. Takahashi, and Y. Yamaguchi, "ACT-Bench: Towards Action Controllable World Models for Autonomous Driving," ICLR Workshop on World Models: Understanding, Modelling and Scaling, 2025. [Online]. Available: https://openreview.net/forum?id=26KlsDgwLi

[36] G. Hess, C. Lindström, M. Fatemi, C. Petersson, and L. Svensson, "SplatAD: Real-Time Lidar and Camera Rendering with 3D Gaussian Splatting for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025, pp. 11 982–11 992.

[37] H. Zhou *et al.*, "HUGSIM: A Real-Time, Photo-Realistic and Closed-Loop Simulator for Autonomous Driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 48, no. 4, pp. 4673–4691, 2026.

[38] H. Tian *et al.*, "SimScale: Learning to Drive via Real-World Simulation at Scale," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 36 365–36 374.

[39] S. C. Calvert and B. van Arem, "A Generic Multi-Level Framework for Microscopic Traffic Simulation with Automated Vehicles in Mixed Traffic," *Transp. Res. Part C: Emerg. Technol.*, vol. 110, pp. 291–311, 2020.

[40] W. Li *et al.*, "AADS: Augmented Autonomous Driving Simulation Using Data-Driven Algorithms," *Sci. Robot.*, vol. 4, no. 28, p. eaaw0863, 2019.

[41] X. Zhou, Z. Lin, X. Shan, Y. Wang, D. Sun, and M.-H. Yang, "DrivingGaussian: Composite Gaussian Splatting for Surrounding Dynamic Autonomous Driving Scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 21 634–21 643.

[42] D. Jannussi, S. C. Lambertenghi, C. Carste, and A. Stocco, "Cam2Sim: Neural Scenario Reconstruction for Closed-Loop Autonomous Driving Simulation," in *Proc. 41st IEEE/ACM Int. Conf. Autom. Softw. Eng. (ASE), Tools and Datasets Track*, 2026.

[43] S. Manivasagam, I. A. Bârsan, J. Wang, Z. Yang, and R. Urtasun, "Towards Zero Domain Gap: A Comprehensive Study of Realistic LiDAR Simulation for Autonomy Testing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 8238–8248.

[44] C. Lindström *et al.*, "Are NeRFs Ready for Autonomous Driving? Towards Closing the Real-to-Simulation Gap," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2024, pp. 4461–4471.

[45] D. Dauner *et al.*, "NAVSIM: Data-Driven Non-Reactive Autonomous Vehicle Simulation and Benchmarking," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 28 706–28 719.

[46] X. Wang, Z. Zhu, G. Huang, X. Chen, J. Zhu, and J. Lu, "DriveDreamer: Towards Real-World-Drive World Models for Autonomous Driving," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, ser. Lecture Notes in Computer Science, vol. 15106, 2024, pp. 55–72.

[47] F. Jia *et al.*, "ADriver-I: A General World Model for Autonomous Driving," arXiv, 2023. [Online]. Available: https://arxiv.org/abs/2311.13549

[48] J. Yang *et al.*, "ReSim: Reliable World Simulation for Autonomous Driving," in *Adv. Neural Inf. Process. Syst.*, vol. 38, 2025, pp. 185 877–185 908.

[49] L. Bergamini *et al.*, "SimNet: Learning Reactive Self-driving Simulations from Real-world Observations," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 5119–5125.

[50] D. Xu, Y. Chen, B. Ivanovic, and M. Pavone, "BITS: Bi-level Imitation for Traffic Simulation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 2929–2936.

[51] Z. Zhou *et al.*, "BehaviorGPT: Smart Agent Simulation for Autonomous Driving with Next-Patch Prediction," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 79 597–79 617.

[52] W. Wu, X. Feng, Z. Gao, and Y. Kan, "SMART: Scalable Multi-agent Real-time Motion Generation via Next-token Prediction," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 114 048–114 071.

[53] X. Hu, M. Jia, X. Guo, Q. Zhang, X.-x. Long, and W. Yin, "DrivingWorld: Constructing World Model for Autonomous Driving via Video GPT," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2026. [Online]. Available: https://arxiv.org/abs/2412.19505

[54] T. Yan *et al.*, "CausalDrive: Real-time Causal World Models for Autonomous Driving," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.15341

[55] M. Treiber, A. Hennecke, and D. Helbing, "Congested Traffic States in Empirical Observations and Microscopic Simulations," *Phys. Rev. E*, vol. 62, no. 2, pp. 1805–1824, 2000.

[56] A. Kesting, M. Treiber, and D. Helbing, "General Lane-Changing Model MOBIL for Car-Following Models," *Transp. Res. Rec.*, vol. 1999, no. 1, pp. 86–94, 2007.

[57] P. G. Gipps, "A Behavioural Car-Following Model for Computer Simulation," *Transp. Res. Part B: Methodol.*, vol. 15, no. 2, pp. 105–111, 1981.

[58] R. Wiedemann, *Simulation des Straßenverkehrsflusses*. Karlsruhe, Germany: Institut für Verkehrswesen, Universität Karlsruhe, 1974.

[59] Q. Sun, X. Huang, B. C. Williams, and H. Zhao, "InterSim: Interactive Traffic Simulation via Explicit Relation Modeling," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2022, pp. 11 416–11 423.

[60] Z. Zhong *et al.*, "Language-Guided Traffic Simulation via Scene-Level Diffusion," in *Proc. Conf. Robot Learn. (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 229, 2023, pp. 144–177.

[61] L. Rowe *et al.*, "CtRL-Sim: Reactive and Controllable Driving Agents with Offline Reinforcement Learning," in *Proc. Conf. Robot Learn. (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 270, 2025, pp. 3600–3621.

[62] N. Montali *et al.*, "The Waymo Open Sim Agents Challenge," in *Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 59 151–59 171.

[63] Z. Zhang *et al.*, "ReactSim-Bench: Benchmarking Reactive Behavior World Model Simulation in Autonomous Driving," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.14058

[64] W. Cao *et al.*, "Pseudo-Simulation for Autonomous Driving," in *Proc. 9th Conf. Robot Learn. (CoRL)*, ser. Proc. Mach. Learn. Res., vol. 305, 2025, pp. 4709–4722.

[65] W. Ljungbergh *et al.*, "NeuroNCAP: Photorealistic Closed-Loop Safety Testing for Autonomous Driving," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, ser. Lecture Notes in Computer Science, vol. 15088, 2024, pp. 161–177.

[66] Q. Li, X. Jia, S. Wang, and J. Yan, "Think2Drive: Efficient Reinforcement Learning by Thinking with Latent World Model for Autonomous Driving (in CARLA-V2)," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 142–158.

[67] A. Garg and K. M. Krishna, "Imagine-2-Drive: Leveraging High-Fidelity World Models via Multi-Modal Diffusion Policies," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2025, pp. 4188–4195.

[68] M. Goff *et al.*, "Learning to Drive from a World Model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2025, pp. 1955–1964.

[69] Y. Li *et al.*, "DriveVLA-W0: World Models Amplify Data Scaling Law in Autonomous Driving," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2026. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2026/hash/0d70423f59c5fdd24f0dd3fa52e34623-Abstract-Conference.html

[70] H. Gao *et al.*, "RAD: Training an End-to-End Driving Policy via Large-Scale 3DGS-based Reinforcement Learning," in *Adv. Neural Inf. Process. Syst.*, vol. 38, 2025.

[71] Z. Peng *et al.*, "Improving Agent Behaviors with RL Fine-Tuning for Autonomous Driving," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, ser. Lecture Notes in Computer Science, vol. 15083, 2024, pp. 165–181.

[72] L. Feng, Q. Li, Z. Peng, S. Tan, and B. Zhou, "TrafficGen: Learning to Generate Diverse and Realistic Traffic Scenarios," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 3567–3575.

[73] L. Rowe, R. Girgis, A. Gosselin, L. Paull, C. Pal, and F. Heide, "Scenario Dreamer: Vectorized Latent Diffusion for Generating Driving Simulation Environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025, pp. 17 207–17 218.

[74] S. Tan *et al.*, "Promptable Closed-loop Traffic Simulation," in *Proc. Conf. Robot Learn. (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 270. PMLR, 2025, pp. 5087–5105.

[75] H. Lin *et al.*, "Causal Composition Diffusion Model for Closed-loop Traffic Generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025, pp. 27 542–27 552.

[76] H. Caesar *et al.*, "nuScenes: A Multimodal Dataset for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 11 618–11 628.

[77] P. Sun *et al.*, "Scalability in Perception for Autonomous Driving: Waymo Open Dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 2443–2451.

[78] S. Ettinger *et al.*, "Large Scale Interactive Motion Forecasting for Autonomous Driving: The Waymo Open Motion Dataset," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 9690–9699.

[79] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, "MetaDrive: Composing Diverse Driving Scenarios for Generalizable Reinforcement Learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3461–3475, 2023.

[80] P. A. Lopez *et al.*, "Microscopic Traffic Simulation Using SUMO," in *Proc. 21st IEEE Int. Conf. Intell. Transp. Syst. (ITSC)*, 2018, pp. 2575–2582.

[81] A. Liang *et al.*, "WorldLens: Full-Spectrum Evaluations of Driving World Models in Real World," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 36 385–36 399.

[82] Y. Yang *et al.*, "WorldExam: Benchmarking World Models from Apparent Appearance to Inherent Reactivity," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2608.02603

[83] D. Gettman and L. Head, "Surrogate Safety Measures from Traffic Simulation Models," Federal Highway Administration, Tech. Rep. FHWA-RD-03-050, 2003. [Online]. Available: https://www.fhwa.dot.gov/publications/research/safety/03050/

[84] M. J. Lighthill and G. B. Whitham, "On Kinematic Waves II. A Theory of Traffic Flow on Long Crowded Roads," *Proc. R. Soc. A*, vol. 229, no. 1178, pp. 317–345, 1955.

[85] K. E. Wunderlich, M. Vasudevan, and P. Wang, "Traffic Analysis Toolbox Volume III: Guidelines for Applying Traffic Microsimulation Modeling Software, 2019 Update to the 2004 Version," U.S. Department of Transportation, Federal Highway Administration, Tech. Rep. FHWA-HOP-18-036, 2019. [Online]. Available: https://ops.fhwa.dot.gov/publications/fhwahop18036/

[86] C. Zhang *et al.*, "HorizonDrive: Self-Corrective Autoregressive World Model for Long-horizon Driving Simulation," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2605.11596

[87] M. Pourkeshavarz, T. Liu, and N. Rhinehart, "AutoWorld: Learning Multi-Agent Traffic Simulation with Self-Supervised World Models," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2603.28963

[88] B. Jing, R. Hao, W. Zhou, and H. Yu, "CounterScene: Counterfactual Causal Reasoning in Generative World Models for Safety-Critical Closed-Loop Evaluation," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2603.21104

[89] Z. H. Khattak, J. Rios-Torres, and M. D. Fontaine, "Impact of Communications Delay on Safety and Stability of Connected and Automated Vehicle Platoons: Empirical Evidence from Experimental Data," *IEEE Access*, vol. 11, pp. 128 549–128 568, 2023.

[90] S. Ren *et al.*, "Interruption-Aware Cooperative Perception for V2X Communication-Aided Autonomous Driving," *IEEE Trans. Intell. Veh.*, vol. 9, no. 4, pp. 4698–4714, 2024.

[91] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5574–5584.

[92] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6402–6413.

[93] S. Zaidi, A. Zela, T. Elsken, C. C. Holmes, F. Hutter, and Y. W. Teh, "Neural ensemble search for uncertainty estimation and dataset shift," in *Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 7898–7911.

Supplementary Material for
"From Foresight to Reactive Traffic Worlds: A Survey of Driving World Models for Closed-Loop Simulation"

Literature Identification and Evidence Basis

The survey covers representative work on learned driving simulation published from 2017 through 15 August 2026, together with earlier studies that establish necessary concepts or validation practices. Candidate literature was identified through arXiv, Crossref, IEEE Xplore, CVF Open Access, NeurIPS, PMLR, ECCV, and OpenReview, supplemented by backward and forward reference tracing. The search vocabulary combined three term groups. Driving-context terms included "autonomous driving," "driving world model," and "traffic simulation." Learned-environment terms included "world model," "neural simulation," "neural rendering," and "generative simulation." Interaction and use terms included "closed loop," "action conditioned," "reactive traffic," "policy evaluation," and "policy training." Technical statements were checked against primary papers, proceedings versions, or publicly available author manuscripts; when a proceedings version was available by the cutoff date, it was preferred to an earlier preprint.

The synthesis includes studies that contribute a distinct simulation mechanism, counterfactual capability, application, benchmark, or validity test. The unit of comparison is a reported system setting rather than a permanent label attached to a model name. The configuration comparison focuses on studies that inform these dimensions; it is not intended to estimate publication frequencies.

Selection and Evidence Interpretation

We included a study in the configuration comparison when its public description exposed at least one property relevant to policy evaluation or directly tested such a property. The comparison records the observation returned after an off-log query, the relation between an ego command and realized motion, the response of non-ego actors to a changed ego trajectory, and whether a downstream policy acts again from the resulting state. Prediction and generation studies also enter the synthesis when they provide direct evidence about action adherence, observation fidelity, or finite-horizon counterfactual generation. Capability assignments rely on demonstrated system behavior rather than model names, scene size, visual realism, or the ability to invoke a component repeatedly.

Each configuration was characterized from the system description and the evaluation reported in the cited source. A regime is assigned only when the corresponding intervention relation is demonstrated. *Replay-Constrained* denotes an off-log observation query with a prescribed behavioral future. *Ego-Responsive* requires an ego command to produce the intended realized motion, without a demonstrated change in relevant non-ego behavior. *Traffic-Reactive* requires relevant non-ego behavior to change when the counterfactual ego trajectory is varied from a matched preceding condition; that ego trajectory may be generated from a command or prescribed for the test. A configuration is reported as *not classified* when its mechanism is compatible with a relation that the reported evaluation does not expose. Table S2 abbreviates the three regimes as *Replay*, *Ego*, and *Traffic*. Policy recurrence is recorded separately and requires a downstream policy to receive an updated state or observation, act again, and alter an uncommitted continuation. Transportation scale is reported as *Encounter* only when the evaluation directly assesses a local policy or interaction outcome; component-level fidelity and controllability experiments are reported as *Not assessed*.

Table S2 distinguishes four evidence states for the named relation. *Established* denotes a direct evaluation of that relation. *Partial* denotes a direct test of a necessary element without isolating the full relation, such as sensitivity without adherence or an off-log response without a matched intervention. *Not established* denotes a compatible mechanism or qualitative demonstration without a direct test of the relation. *Not assessed* denotes an evaluation directed at another property. Recurrence affects these judgments only when recurrent policy interaction is itself the claim. The states describe the scope of the reported evidence rather than overall model quality, and the table does not infer a capability merely from an architecture that could support it.

Table S1 relates recurring failure modes to observable evaluation tests, and Table S2 compares system settings with source-specific evidence. This is a purposive, concept-driven synthesis rather than an exhaustive systematic review. Its coverage can therefore be affected by public-source availability, rapid revision of recent preprints, and judgment in separating a component capability from the assembled system setting; the cutoff date and explicit comparison criteria bound the resulting claims.

TABLE S1
FAILURE MODES IN LEARNED DRIVING SIMULATION, OBSERVABLE EVALUATION TESTS, AND IMPLICATIONS FOR POLICY ASSESSMENT. REPRESENTATIVE STUDIES MOTIVATE EACH MECHANISM OR PROVIDE AN INTERFACE THROUGH WHICH IT CAN BE TESTED.

Failure mode	Observable signature or evaluation test	Effect on policy assessment	Representative studies
visual drift	Long-horizon autoregressive comparison using Fréchet inception distance (FID), Fréchet video distance (FVD), or perceptual stability, supplemented by temporal-consistency and human/qualitative inspection.	Later frames may no longer preserve stable geometry or semantics, so policy behavior can be driven by generator drift rather than the intended scenario.	[S1]–[S3]
oracle-layout leakage	Remove future non-player character (NPC) layouts or human-trajectory endpoints from conditioning and compare action/interaction fidelity under initial-state-only inputs.	Future-layout or goal leakage can make rendered futures appear accurate while bypassing the causal prediction required after an off-log ego action.	[S4], [S5]
non-reactive traffic	Deviate the ego from the logged action and inspect whether surrounding trajectories adapt; compare replay with reactive-agent simulation and collision outcomes.	Replay agents can collide unrealistically or conceal negotiation failures, invalidating closed-loop safety and interaction judgments.	[S4], [S6], [S7]
optimistic simulator bias	Compare planner scores/rankings across open-loop, non-reactive/rule-based, and learned reactive evaluations on matched planner sets.	Simplified traffic can overestimate planner performance and alter rankings, hiding failures that emerge with more realistic interactions.	[S5], [S8]
runtime mismatch	Report end-to-end frames per second (FPS) and latency on stated hardware and compare them with the planner/control frequency and intended online training or evaluation loop.	A simulator that cannot meet interaction-rate requirements changes the control loop or becomes unusable for online policy learning and hardware-in-the-loop evaluation.	[S3], [S4], [S9]
long-horizon compounding error	Evaluate multiple rollout horizons, compare teacher/ground-truth history with self-rollout history, and track realism, collision, overlap, or prediction error over time.	Errors feed back into subsequent states, so long-horizon policy outcomes can reflect accumulated model error rather than true environment dynamics.	[S1], [S4], [S10], [S11]
cross-module desynchronization	Compare the behavior state, vehicle dynamics, rendered actor pose, signal state, and sensor timestamps after the same intervention; vary module update rates and action-to-observation delay.	A policy can react to stale geometry or inconsistent intent even when the behavior model and renderer each pass isolated tests.	[S12]–[S14]

TABLE S2

REPRESENTATIVE SETTINGS UNDERLYING THE CROSS-PAPER COMPARISON. A REGIME TAG IS ASSIGNED ONLY WHEN THE CORRESPONDING INTERVENTION RELATION IS DIRECTLY DEMONSTRATED; "NOT CLASSIFIED" DENOTES A COMPATIBLE SETTING WHOSE REPORTED EVALUATION DOES NOT EXPOSE THAT RELATION, AND "DIAGNOSTIC" DENOTES A BENCHMARK RATHER THAN A SIMULATOR REGIME. ESTABLISHED EVIDENCE DIRECTLY EVALUATES THE NAMED RELATION. PARTIAL EVIDENCE TESTS A NECESSARY ELEMENT WITHOUT ISOLATING THE FULL RELATION. NOT ESTABLISHED INDICATES THAT THE REPORTED EVIDENCE DOES NOT DIRECTLY DEMONSTRATE THE RELATION, EVEN WHEN THE ARCHITECTURE IS COMPATIBLE, WHEREAS NOT ASSESSED IDENTIFIES AN EXPERIMENT DIRECTED AT ANOTHER PROPERTY. THESE LABELS CHARACTERIZE SUPPORT FOR THE RELATION DISCUSSED IN EACH ROW; THEY ARE NOT RATINGS OF OVERALL MODEL CAPABILITY OR MATURITY. POLICY RECURRENCE IS RECORDED SEPARATELY AND DOES NOT CHANGE THE STATUS OF EGO- OR TRAFFIC-RESPONSE EVIDENCE. THE FINAL COLUMN DISTINGUISHES ENCOUNTER-LEVEL TRANSPORTATION EVIDENCE FROM COMPONENT TESTS; NONE OF THE LISTED EVALUATIONS ESTABLISHES CORRIDOR PROPAGATION OR NETWORK PERFORMANCE.

System setting and realized regime	Learned function	Persistent state	Ego-conditioned evolution	Traffic evolution	Supports policy recurrence?	Reported evidence and source location	Highest transportation scale directly assessed
UniSim (*Replay*)	Neural camera/LiDAR renderer	Reconstructed scene; the ego state evolves in the autonomy loop while actor trajectories are supplied to the renderer	New sensor views follow the executed ego pose	The reported closed-loop demonstration uses human-specified counterfactual actor trajectories	Yes, in the integrated autonomy loop	**Established.** Scene controllability and closed-loop autonomy (Sec. 4.5 and App. A1) exercise recurrent policy observation under specified actor trajectories [S15]	Encounter
MARS (*Replay*)	Instance-aware neural renderer	Reconstructed background and object nodes; queried ego and actor poses	Novel camera views follow the supplied sensor pose	Logged or user-specified actor trajectories; no behavior model in the renderer	No recurrent policy loop in the reported system setting	**Not assessed.** Rendering and instance editing (Secs. 3.1–3.2) do not exercise policy feedback [S16]	Not assessed
NeuRAD (*Replay*)	Neural camera/LiDAR renderer	Reconstructed scene, actor tracks, and queried sensor pose	Off-path observations follow the supplied ego pose	Logged or edited actor states; behavior remains external	No recurrent policy query in the renderer evaluation	**Not assessed.** Novel-view and novel-scenario generation (Secs. 4.2–4.3) omit policy recurrence [S17]	Not assessed
SplatAD (*Replay*)	Real-time 3D Gaussian splatting (3DGS) camera/LiDAR renderer	Reconstructed dynamic scene with externally supplied pose and actor tracks	Rendered sensor data changes with the queried ego pose	Recorded or externally supplied motion; no actor-decision model	No downstream policy recurrence is reported	**Not assessed.** Camera/LiDAR rendering experiments (Secs. 4.1–4.2) do not evaluate policy feedback [S18]	Not assessed
NeRF downstream comparison (*Replay*)	Neural renderer evaluated against real sensor data	Matched reconstructed scenes and camera trajectories	Matched-pose rendering tests observation replacement rather than new dynamics	Traffic follows the paired real/log sequence	No recurrent environment; autonomy modules consume rendered inputs	**Partial.** Real–rendered downstream comparisons (Secs. 4.1–4.3) test observation replacement, not environment recurrence [S19]	Not assessed
GAIA-1 (*not classified*)	Autoregressive video-token future generator	Video, text, and action-token context within the generated sequence	Vehicle-action tokens steer the continuation; command-to-motion adherence is not isolated	Surrounding motion is generated jointly; conditional response is not isolated	No externally executed terminal state enters another policy query	**Not established.** Long-video and qualitative action-control demonstrations (Secs. 7.1–7.3) show conditional generation without directly establishing command-to-motion execution [S20]	Not assessed
Drive-WM planning branch (*Ego*)	Multiview future generator and candidate evaluator	Common input history within a finite planning branch; no exposed executed terminal state	Candidate trajectory conditions a distinct visual future	Non-ego motion is jointly generated; intervention-specific response is not isolated	No external execution; branches are reused only inside the planning tree	**Partial.** Candidate rollouts and image-reward planning (Secs. 4.1–4.2 and 5.4) stop before external execution [S21]	Not assessed
Vista iterative rollout (*Ego*)	Action- and trajectory-conditioned video generator	Recent real or generated frames carried across successive clips	Trajectory, command, and speed conditions alter the generated ego future	Surrounding motion is jointly generated; paired response attribution is absent	The generator recurs, but no downstream policy query is reported	**Partial.** Action control and extended rollout (Secs. 3.2 and 4.2) exercise the generator without a downstream policy loop [S22]	Not assessed

TABLE S2 (Continued)

System setting and realized regime	Learned function	Persistent state	Ego-conditioned evolution	Traffic evolution	Supports policy recurrence?	Reported evidence and source location	Highest transportation scale directly assessed
DrivingWorld (*not classified*)	Autoregressive video-token world model	Tokenized frame and vehicle-state history within the rollout	Vehicle location and orientation control the generated continuation; adherence is not isolated	Traffic is generated jointly without a separately identified mechanism	No external policy query in the reported system setting	**Not established.** Qualitative controllable long-horizon generation (Secs. 4.2–4.3) does not directly establish command-to-motion execution [S23]	Not assessed
ACT-Bench adherence test (*diagnostic*)	Trajectory-conditioned generator under an adherence test	Fixed context and requested trajectory for a finite generated clip	Realized ego motion is estimated and compared with the instruction	Non-ego motion remains part of the generator; response is not tested	No; the benchmark evaluates the generated clip	**Established for the diagnostic.** Action fidelity is directly evaluated on finite clips (Secs. 3–5); recurrence is not assessed [S24]	Not assessed
TrafficSim (*not classified*)	Learned joint multi-agent policy	Current agent/map state is fed back through the rollout	Sampled agent actions advance the joint scene; a separate autonomous vehicle (AV) intervention is not the focus	The learned joint policy updates simulated actors	The agent rollout recurs; no separate downstream AV policy query is reported	**Partial.** Joint-agent rollout and realism/safety tests (Secs. 3.3 and 4.2) omit a separate AV intervention required to assign the Traffic-Reactive regime [S25]	Encounter
TrafficBots (*not classified*)	Learned recurrent multi-agent behavior model	Joint agent state, recurrent agent memory, destination, and personality	Player action advances through vehicle dynamics	Learned reactive agents; logged or scripted actors may coexist	Behavior state recurs; no separate downstream AV policy query is reported	**Partial.** Recurrent agent rollouts (Fig. 1; Sec. V) execute reactive traffic, but the reported evaluation does not isolate matched ego interventions; downstream AV-policy recurrence is not assessed [S10]	Encounter
BehaviorGPT (*not classified*)	Fully autoregressive agent behavior generator	Scene and agent histories updated after each generated motion patch	The main simulation protocol does not isolate an exogenous ego intervention	Learned next-patch updates generate the simulated agents	Generated state recurs; no separate AV policy loop is reported	**Partial.** Closed-loop agent simulation (Secs. 4.1 and 4.3) does not isolate the external ego intervention required to assign a feedback regime [S26]	Encounter
SMART (*not classified*)	Next-token multi-agent behavior model	Map tokens and rolling motion-token history	The main Waymo Open Sim Agents Challenge (WOSAC) evaluation does not separately control ego intervention	Learned next-token rollout updates the simulated agents	Generated state recurs; no separate AV policy query is reported	**Partial.** Long-rollout WOSAC evaluation (Secs. 4.1–4.2) omits the separate ego intervention required to assign a feedback regime [S27]	Encounter
Waymax with reactive agents (*Traffic*)	Learned or rule-based agent model inside a structured simulator	Dynamic joint state plus static road graph and routes	User-policy action advances AV dynamics	Reactive IDM agents; logged or learned agents are alternatives	Yes; the successor state is observed by the policy	**Established.** Reactive-agent simulation and planner evaluation (Secs. 3.5 and 5.3) exercise recurrence [S28]	Encounter
HUGSIM (*Traffic*)	Neural sensor renderer in a hybrid simulator	Reconstructed scene plus structured ego and actor state	A linear–quadratic regulator (LQR) executes ego commands before the next rendering query	IDM controls ordinary actors; an attack policy supplies aggressive actors	Yes; the driving system acts on successive rendered observations	**Established.** Closed-loop integration, actor response, and benchmark (Secs. IV-B–IV-C and VI) [S29]	Encounter

TABLE S2 (Continued)

System setting and realized regime	Learned function	Persistent state	Ego-conditioned evolution	Traffic evolution	Supports policy recurrence?	Reported evidence and source location	Highest transportation scale directly assessed
Bench2Drive-R (*Traffic*)	Generative observation model coupled to a reactive controller	Structured ego/actor state and autoregressive rendering context	Planned ego trajectory is executed	Reactive behavior controller supplies surrounding motion	Yes; the policy acts over a closed-loop episode	**Established.** Iterative framework and closed-loop results (Secs. 3.2 and 4.2) exercise the policy–traffic loop [S12]	Encounter
SimScale (*not classified*)	3DGS renderer in a data-generation and training environment	Perturbed joint state seeds a second finite-horizon rollout	LQR follows the prescribed perturbed trajectory; a pseudo-expert supplies the second-stage ego trajectory	IDM updates surrounding agents in each finite stage	No downstream-policy recurrence in the reported generator; the second stage uses a pseudo-expert trajectory	**Partial.** Reactive two-stage generation and planner scaling (Secs. 2.3 and 3.3) execute traffic response, but the reported evaluation does not isolate matched ego interventions; recurrence is not assessed [S13]	Encounter
CausalDrive (*not classified*)	End-to-end generative sensor world model	Generated history within a coupled rollout	Off-log ego trajectory conditions the future	Learned non-player-character (NPC) motion without future layouts	Yes, in the reported policy loop	**Not established.** The design permits an intervention test, but prompt-conditioned reactions at a fixed ego trajectory (Fig. 1) do not isolate how NPC motion changes when that trajectory is varied [S4]	Encounter
OmniDreams (*Replay*)	Stateful causal generative sensor model	Streaming visual context with a key–value (KV) cache plus simulation state	Latest policy trajectory changes the synthesized view	Dynamic-agent trajectories are supplied in advance through the abstract world-scenario input	Yes; policy inference and sensor synthesis alternate	**Established.** Closed-loop integration and multi-policy evaluation (Secs. 6 and 9.4) exercise recurrent sensor synthesis under supplied actor trajectories [S14]	Encounter
ReactSim-Bench fixed off-log-AV protocol (*not classified*)	Learned world-agent model conditioned on a fixed external off-log AV rollout	Joint scene history carried across agent rollout and re-inference	One preselected off-log AV trajectory is injected per scenario; agents condition on its realized states	The evaluated world agents recompute surrounding actions	No external-policy recurrence; world agents are re-inferred autoregressively at tested intervals	**Partial.** The fixed off-log protocol tests response under a prescribed AV trajectory (Secs. 3.1–3.2 and 4.2–4.3), but does not provide a matched alternative trajectory for intervention attribution; policy recurrence is not assessed [S30]	Encounter
nuPlan-R (*Traffic*)	Learned reactive traffic model in a planning benchmark	Structured joint agent/map state across planner steps	The planner trajectory advances the ego vehicle	Learned reactive agents are compared with fixed or rule-based traffic	Yes; the planner is queried repeatedly	**Established.** Reactive task definition and joint agent/planner experiments (Secs. III-A and IV-A–IV-B) [S31]	Encounter
Traffic-model sensitivity study (*Traffic*)	Learned SMART traffic model compared with IDM	The same nuPlan scene is carried through repeated replanning	The same planner/scenario is executed under each traffic model	SMART or IDM supplies surrounding behavior	Yes; each rollout repeatedly queries the planner	**Established.** Matched-scene traffic-model comparison (Fig. 1; Sec. IV) measures planner and ranking changes [S8]	Encounter

REFERENCES

[S1] K. Zhang *et al.*, "Epona: Autoregressive Diffusion World Model for Autonomous Driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2025, pp. 27 220–27 230.

[S2] A. Liang *et al.*, "WorldLens: Full-Spectrum Evaluations of Driving World Models in Real World," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 36 385–36 399.

[S3] L. Russell *et al.*, "GAIA-2: A Controllable Multi-View Generative World Model for Autonomous Driving," arXiv, 2025. [Online]. Available: https://arxiv.org/abs/2503.20523

[S4] T. Yan *et al.*, "CausalDrive: Real-time Causal World Models for Autonomous Driving," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.15341

[S5] D. Dauner *et al.*, "NAVSIM: Data-Driven Non-Reactive Autonomous Vehicle Simulation and Benchmarking," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 28 706–28 719.

[S6] H. Caesar *et al.*, "NuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), ADP3*, 2021. [Online]. Available: https://arxiv.org/abs/2106.11810

[S7] Q. Zhang *et al.*, "TrajGen: Generating Realistic and Diverse Trajectories With Reactive and Feasible Agent Behaviors for Autonomous Driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 24 474–24 487, 2022.

[S8] S. Hagedorn, L. Donkov, A. Distelzweig, and A. P. Condurache, "When Planners Meet Reality: How Learned, Reactive Traffic Agents Shift nuPlan Benchmarks," arXiv, 2025. [Online]. Available: https://arxiv.org/abs/2510.14677

[S9] Z. Xiong *et al.*, "UniDrive-WM: Unified understanding, planning and generation world model for autonomous driving," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2026, to be published. [Online]. Available: https://arxiv.org/abs/2601.04453

[S10] Z. Zhang, A. Liniger, D. Dai, F. Yu, and L. Van Gool, "TrafficBots: Towards World Models for Autonomous Driving Simulation and Motion Prediction," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 1522–1529.

[S11] S. Tan *et al.*, "SceneDiffuser++: City-Scale Traffic Simulation via a Generative World Model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025, pp. 1570–1580.

[S12] J. You, X. Jia, Z. Zhang, Y. Zhu, and J. Yan, "Bench2Drive-R: Turning Real World Data into Reactive Closed-Loop Autonomous Driving Benchmark by Generative Model," arXiv, 2024. [Online]. Available: https://arxiv.org/abs/2412.09647

[S13] H. Tian *et al.*, "SimScale: Learning to Drive via Real-World Simulation at Scale," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 36 365–36 374.

[S14] NVIDIA *et al.*, "NVIDIA OmniDreams: Real-Time Generative World Model for Closed-Loop Autonomous Vehicle Simulation," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.03159

[S15] Z. Yang *et al.*, "UniSim: A Neural Closed-Loop Sensor Simulator," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 1389–1399.

[S16] Z. Wu *et al.*, "MARS: An Instance-Aware, Modular and Realistic Simulator for Autonomous Driving," in *Proc. CAAI Int. Conf. Artif. Intell. (CICAI)*, ser. Lecture Notes in Artificial Intelligence, vol. 14473, 2024, pp. 3–15.

[S17] A. Tonderski, C. Lindström, G. Hess, W. Ljungbergh, L. Svensson, and C. Petersson, "NeuRAD: Neural Rendering for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 14 895–14 904.

[S18] G. Hess, C. Lindström, M. Fatemi, C. Petersson, and L. Svensson, "SplatAD: Real-Time Lidar and Camera Rendering with 3D Gaussian Splatting for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2025, pp. 11 982–11 992.

[S19] C. Lindström *et al.*, "Are NeRFs Ready for Autonomous Driving? Towards Closing the Real-to-Simulation Gap," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2024, pp. 4461–4471.

[S20] A. Hu *et al.*, "GAIA-1: A Generative World Model for Autonomous Driving," arXiv, 2023. [Online]. Available: https://arxiv.org/abs/2309.17080

[S21] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the Future: Multiview Visual Forecasting and Planning with World Model for Autonomous Driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 14 749–14 759.

[S22] S. Gao *et al.*, "Vista: A Generalizable Driving World Model with High Fidelity and Versatile Controllability," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 91 560–91 596.

[S23] X. Hu, M. Jia, X. Guo, Q. Zhang, X.-x. Long, and W. Yin, "DrivingWorld: Constructing World Model for Autonomous Driving via Video GPT," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2026. [Online]. Available: https://arxiv.org/abs/2412.19505

[S24] H. Arai, K. Ishihara, T. Takahashi, and Y. Yamaguchi, "ACT-Bench: Towards Action Controllable World Models for Autonomous Driving," ICLR Workshop on World Models: Understanding, Modelling and Scaling, 2025. [Online]. Available: https://openreview.net/forum?id=26KlsDgwLi

[S25] S. Suo, S. Regalado, S. Casas, and R. Urtasun, "TrafficSim: Learning to Simulate Realistic Multi-Agent Behaviors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 10 395–10 404.

[S26] Z. Zhou *et al.*, "BehaviorGPT: Smart Agent Simulation for Autonomous Driving with Next-Patch Prediction," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 79 597–79 617.

[S27] W. Wu, X. Feng, Z. Gao, and Y. Kan, "SMART: Scalable Multi-agent Real-time Motion Generation via Next-token Prediction," in *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 114 048–114 071.

[S28] C. Gulino *et al.*, "Waymax: An Accelerated, Data-Driven Simulator for Large-Scale Autonomous Driving Research," in *Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 7730–7742.

[S29] H. Zhou *et al.*, "HUGSIM: A Real-Time, Photo-Realistic and Closed-Loop Simulator for Autonomous Driving," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 48, no. 4, pp. 4673–4691, 2026.

[S30] Z. Zhang *et al.*, "ReactSim-Bench: Benchmarking Reactive Behavior World Model Simulation in Autonomous Driving," arXiv, 2026. [Online]. Available: https://arxiv.org/abs/2606.14058

[S31] M. Peng, R. Yao, X. Guo, and J. Ma, "nuPlan-R: A Closed-Loop Planning Benchmark for Autonomous Driving via Reactive Multi-Agent Simulation," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2026, pp. 702–709.