HOPE: HANDS-OFF COUNTERFACTUAL POLICY EXPLANATIONS FOR REINFORCEMENT LEARNING FAILURES

Anonymous authors
Paper under double-blind review

ABSTRACT

Explaining a reinforcement learning (RL) failure requires identifying when and how the agent could have acted differently to avoid failure while still completing its task. Existing counterfactual approaches modify observations, generate open-loop action sequences, or learn alternative policies, leaving a gap between explaining a specific failure and preserving the deployed policy's feedback. We present HOPE (**H**ands-**O**ff counterfactual **P**olicy **E**xplanation), a unified framework for generating trajectory-specific counterfactual policies across continuous, discrete, and hybrid action spaces. Each counterfactual policy applies sparse, time-indexed action interventions to change the failure outcome while preserving the deployed policy's feedback at all other times. HOPE separates validity search from intervention reduction, using a graded reach-avoid residual to guide search across continuous, discrete, and hybrid action spaces. Validity-preserving deletion, refitting, and magnitude contraction then reduce intervention burden, while Pareto filtering retains nondominated trade-offs between intervention number and average magnitude. Experiments on 600 failure incidents across three navigation and driving benchmarks yield validity rates of 81%, 74%, and 83% on the common solvable subsets. Across all benchmarks, HOPE ranks first or second in validity and first in single-intervention rate among the evaluated RL counterfactual methods.

1 INTRODUCTION

Although deployed reinforcement learning (RL) policies can achieve high expected return, average performance can conceal rare but consequential failures (Dulac-Arnold et al., 2021). Reward maximization alone does not ensure safety (Alshiekh et al., 2018); in safety-critical tasks, one collision or constraint violation can outweigh otherwise strong performance. Feature-attribution methods such as LIME (Ribeiro et al., 2016) and SHAP (Lundberg & Lee, 2017) identify which inputs influenced an individual decision, but they do not specify action interventions that would change the subsequent trajectory. We therefore ask: *Given an observed failure trajectory, when and how could the deployed policy have acted differently to avoid the failure while still completing the task?*

Counterfactual explanations provide a natural framework for this question. State counterfactuals adapt the small-change principle of supervised counterfactual explanations to synthesize observations that induce alternative actions (Wachter et al., 2017; Olson et al., 2021). Yet a plausible observation may be dynamically unreachable from the observed trajectory (Gajcin & Dusparic, 2024b). Dynamics-aware methods search for executable counterfactual action sequences (Tsirtsis & Rodriguez, 2023; Gajcin & Dusparic, 2023; 2024a), but a realized sequence does not define feedback if the environment reaches another state. Policy-level methods provide state-dependent alternatives through trajectory-conditioned or learned policies (Tsirtsis et al., 2021; Movin et al., 2023; Dong et al., 2025). Existing incident-specific formulations are limited to low-dimensional discrete Markov Decision Processes (MDPs), whereas learned formulations produce task-level alternative policies rather than sparse changes to the deployed policy for one failure (Milani et al., 2024).

We address these limitations with HOPE, a unified framework for generating trajectory-specific counterfactual policies across continuous, discrete, and hybrid action spaces. HOPE explains one observed failure through a trajectory-specific counterfactual policy built from sparse, time-indexed

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

action interventions. At non-intervened times, the original policy selects an action from the counterfactual state actually reached. We evaluate each policy through its closed-loop trajectory under the reference randomization associated with the failure. A policy is valid only if its trajectory avoids the failure set and reaches the goal set. Among valid policies, HOPE returns Pareto-optimal alternatives that trade off intervention number and magnitude.

Generating such a counterfactual policy presents three challenges. First, a binary reach-avoid indicator assigns the same value to all invalid counterfactual policies and thus cannot guide the search toward validity (So et al., 2024). Second, closed-loop dynamics couple action interventions across time, but the environment provides no derivatives of their joint effects, and different action spaces require different update operations. Third, a valid policy may contain redundant or excessive interventions, while removing or shrinking one can alter the subsequent trajectory and invalidate it.

To address these challenges, HOPE separates the search for valid counterfactual policies from intervention reduction. A reach-avoid residual provides a graded loss for ranking invalid counterfactual policies. Using this loss, HOPE employs a joint smoothed local model for continuous action interventions, loss-ranked beam search for discrete action interventions, and alternating type selection and parameter refinement for hybrid action interventions. Once a valid policy is found, validity-preserving deletion and support-restricted refitting reduce the intervention support, while contraction reduces the magnitudes of continuous interventions. Finally, Pareto filtering retains the nondominated alternatives that trade off sparsity and proximity.

Experiments show that HOPE achieves the best validity-sparsity balance among the evaluated methods. Across three benchmarks from Safety-Gymnasium (Ji et al., 2023) and highway-env (Leurent, 2018), it achieves validity rates of 81%, 74%, and 83% in continuous, discrete, and hybrid action spaces, respectively, ranking first, second, and first on the corresponding benchmarks compared to evaluated methods. Furthermore, 24.5%, 40.3%, and 55.7% of the valid counterfactual policies generated by HOPE require only a single action intervention on the three benchmarks, respectively—the highest proportion among the evaluated methods on each benchmark.

2 Related Work

Feature attribution and state counterfactuals. Instance-level RL explanations include feature-attribution and counterfactual methods. LIME fits a locally interpretable model (Ribeiro et al., 2016), while SHAP assigns additive feature attributions using Shapley values (Lundberg & Lee, 2017). These methods identify features associated with a decision but do not construct a dynamically reachable alternative trajectory (Atrey et al., 2019). Counterfactual explanations instead seek small input changes that produce a desired prediction (Wachter et al., 2017). In RL, state-counterfactual methods synthesize minimally modified observations that induce a target action (Olson et al., 2021). However, proximity or plausibility in observation space does not ensure reachability through executable actions and environment dynamics (Karimi et al., 2021; Gajcin & Dusparic, 2024b). Rather than synthesizing states directly, HOPE constructs a counterfactual policy from action interventions and evaluates the resulting counterfactual trajectory under closed-loop environment dynamics.

Dynamics-aware counterfactual trajectories. Dynamics-aware counterfactual methods construct executable trajectories or state-contingent policies. RACCER uses heuristic tree search to reach states that induce the target action sequence (Gajcin & Dusparic, 2023); Tsirtsis & Rodriguez (2023) use A^* search to optimize action sequences; and ACTER applies multi-objective evolutionary search to generate action sequences that avoid an observed failure (Gajcin & Dusparic, 2024a). These sequences describe a realized path but do not prescribe feedback at states outside that path.

Counterfactual policies and strategies. Policy-level methods construct state-dependent alternatives to the deployed policy. In discrete causal MDPs, Tsirtsis et al. (2021) use a trajectory-conditioned Gumbel-Max structural causal model and dynamic programming under a deviation budget, and Movin et al. (2023) learn an auxiliary policy balancing the desired outcome and policy disagreement. For continuous actions, Dong et al. (2025) learn a deterministic counterfactual policy that improves return while penalizing action deviations. Kobialka et al. (2025) synthesize strategies that stay close to an initial policy while limiting undesirable-outcome probability, and Kazemi et al. (2024) derive path-conditioned policies under counterfactual-influence constraints. HOPE instead

explains an observed failure through sparse action interventions around the deployed policy using a unified formulation and validity condition for continuous, discrete, and hybrid action spaces.

Sparse control and runtime safety. Sparse control and runtime safety modify a controller only when intervention is needed (Garcıa & Fernández, 2015; Liu et al., 2019). Their objectives differ from post-hoc counterfactual explanation. Maximum hands-off control (MHOC) minimizes the temporal support of a bounded control input and admits an ℓ_1 formulation for normal finite-horizon linear systems (Nagahara et al., 2015). Shields and control barrier functions instead modify actions online to enforce safety (Alshiekh et al., 2018; Ames et al., 2016). These methods assume a known control model or act prospectively, whereas HOPE explains a recorded failure while preserving goal attainment. HOPE adopts the hands-off temporal-support principle but uses closed-loop simulator evaluation, residual-guided validity search, and validity-preserving intervention reduction under nonlinear simulator dynamics and across heterogeneous action spaces.

3 Problem Formulation

3.1 RL failure and counterfactual policy

Observed RL failure trajectory. Consider an RL problem with state space $\mathcal{S}$, action space $\mathcal{A}$, and time step t. Let $\pi_0 : \mathcal{S} \to \mathcal{A}$ be the deployed policy, producing $\mathbf{a}_t = \pi_0(\mathbf{s}_t)$, where $\mathbf{s}_t$ and $\mathbf{a}_t$ are the factual state and action at time t. For an observed failure, let $\mathcal{F}$ and $\mathcal{G}$ denote the failure and goal sets, respectively. Let $t_f = \min\{t : \mathbf{s}_t \in \mathcal{F}\}$ be the first failure time and H the episode time limit. The observed failure trajectory is $\tau = (\mathbf{s}_0, \mathbf{a}_0, \ldots, \mathbf{s}_{t_f})$.

Counterfactual policy. A counterfactual policy applies time-indexed action interventions while preserving the feedback of the deployed policy. The intervention window begins at time $k < t_f$ and has length $W = t_f - k$. Let $\mathcal{E}$ be the intervention space, $\mathbf{\Delta}_t \in \mathcal{E}$ the intervention at time t, and $\mathbf{\Delta} = (\mathbf{\Delta}_k, \ldots, \mathbf{\Delta}_{t_f-1})$ the intervention sequence, with $\mathbf{\Delta}_t = \mathbf{0}$ outside this window. Its space-specific magnitude is $\|\mathbf{\Delta}_t\|_{\mathcal{E}}$. An intervention map $\varphi : \mathcal{A} \times \mathcal{E} \to \mathcal{A}$ specifies how an intervention modifies an action and satisfies $\varphi(\mathbf{a}, \mathbf{0}) = \mathbf{a}$. Together with the deployed policy π_0, it defines the counterfactual policy

$$\hat{\mathbf{a}}_t = \pi_{\mathbf{\Delta}}(\hat{\mathbf{s}}_t, t) := \varphi(\pi_0(\hat{\mathbf{s}}_t), \mathbf{\Delta}_t)\,, \tag{1}$$

where $\hat{\mathbf{s}}_t$ and $\hat{\mathbf{a}}_t$ are the counterfactual state and action, respectively. When $\mathbf{\Delta}_t = \mathbf{0}$, the deployed policy selects the action $\pi_0(\hat{\mathbf{s}}_t)$ from the counterfactual state actually reached. Appendix A.2 details the intervention map and magnitude measurement.

A closed-loop rollout follows the simulator dynamics under the counterfactual policy. Let $f_{\text{sim}} : \mathcal{S}\times\mathcal{A}\times\mathcal{U} \to \mathcal{S}$ be the simulator's one-step transition function, where $\mathcal{U}$ is the per-step randomization space and $\hat{\mathbf{u}}_t \in \mathcal{U}$ is the randomization applied at time t. For the rollout randomization sequence $\hat{\mathbf{u}} = (\hat{\mathbf{u}}_k, \ldots, \hat{\mathbf{u}}_{H-1})$, the closed-loop dynamics are

$$\hat{\mathbf{s}}_{t+1} = f_{\text{sim}}(\hat{\mathbf{s}}_t, \pi_{\mathbf{\Delta}}(\hat{\mathbf{s}}_t, t), \hat{\mathbf{u}}_t)\,, \qquad \hat{\mathbf{s}}_k = \mathbf{s}_k \text{ and } t = k, \ldots, \hat{T} - 1, \tag{2}$$

where the rollout stops at $\hat{T} := \min\big(\{t : \hat{\mathbf{s}}_t \in \mathcal{G} \cup \mathcal{F}\} \cup \{H\}\big)$, the first time $\hat{\mathbf{s}}_t$ enters $\mathcal{G} \cup \mathcal{F}$, or at H if it enters neither. Together, the counterfactual policy and simulator dynamics generate the trajectory $\hat{\tau}(\mathbf{\Delta}; \hat{\mathbf{u}}) = (\hat{\mathbf{s}}_k, \hat{\mathbf{a}}_k, \ldots, \hat{\mathbf{a}}_{\hat{T}-1}, \hat{\mathbf{s}}_{\hat{T}})$. Because Eq. 1 is evaluated at every reached state, an early action intervention changes both subsequent states and their selected actions.

3.2 Counterfactual criteria

We evaluate outcome success and intervention burden using four criteria: validity, sparsity, proximity, and recency. Validity measures reach-avoid success under the reference realization. Sparsity, proximity, and recency quantify the number, magnitude, and timing of interventions, respectively.

Validity. Validity is the binary reach-avoid outcome of a counterfactual policy on a fixed reference rollout. Let $\hat{\mathbf{u}}^{(0)}$ be the reference simulator randomization associated with the observed failure. For any rollout randomization $\hat{\mathbf{u}}$, define the reach-avoid indicator (So et al., 2024; Fisac et al., 2015) as

$$C(\mathbf{\Delta}; \hat{\mathbf{u}}) := \begin{cases} 1, & \text{if } \hat{\mathbf{s}}_{\hat{T}} \in \mathcal{G} \setminus \mathcal{F}, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

A candidate policy is valid if $C(\boldsymbol{\Delta}; \hat{\mathbf{u}}^{(0)}) = 1$, i.e., the trajectory reaches $\mathcal{G}$ without entering $\mathcal{F}$.

Sparsity, proximity, and recency. These criteria measure the number of interventions, their average magnitude, and the time of the first intervention, respectively. Define the temporal support $\mathcal{T}(\boldsymbol{\Delta}) = \{t \in \{k, \dots, t_f - 1\} : \boldsymbol{\Delta}_t \neq \mathbf{0}\}$. The three criteria are

$$N(\boldsymbol{\Delta}) := |\mathcal{T}(\boldsymbol{\Delta})|, \qquad \text{(sparsity)}, \tag{4}$$

$$D(\boldsymbol{\Delta}) := \frac{1}{N(\boldsymbol{\Delta})} \sum_{t \in \mathcal{T}(\boldsymbol{\Delta})} \|\boldsymbol{\Delta}_t\|_{\mathcal{E}}, \qquad \text{(proximity)}, \tag{5}$$

$$R(\boldsymbol{\Delta}) := t_f - \min \mathcal{T}(\boldsymbol{\Delta}), \qquad \text{(recency)}. \tag{6}$$

For discrete action interventions, the magnitude is the action-space-specific distance between the substituted action and the action selected by π_0. Smaller N, D, and R indicate fewer, smaller, and later action interventions, respectively.

A useful counterfactual policy is valid and uses concise, recent action interventions. It satisfies $C(\boldsymbol{\Delta}; \hat{\mathbf{u}}^{(0)}) = 1$ and has small $N(\boldsymbol{\Delta})$, $D(\boldsymbol{\Delta})$, and $R(\boldsymbol{\Delta})$. Sparse and proximal counterfactual policies reduce the number and magnitude of action interventions that a user must examine (Wachter et al., 2017; Miller, 2019; Tsirtsis et al., 2021). For temporal explanations, later action interventions also align with the documented recency of human counterfactual reasoning (Henne et al., 2021).

3.3 Counterfactual Policy Generation

Counterfactual policy generation seeks valid explanations while leaving the choice between sparsity and proximity to the user. Let $\epsilon > 0$ be the maximum admissible magnitude of an action intervention at one time step. The counterfactual policy generation problem is

$$\operatorname*{Pareto\,min}_{\boldsymbol{\Delta}} \big(N(\boldsymbol{\Delta}), D(\boldsymbol{\Delta})\big) \quad \text{s.t.} \quad C(\boldsymbol{\Delta}; \hat{\mathbf{u}}^{(0)}) = 1, \qquad \|\boldsymbol{\Delta}_t\|_{\mathcal{E}} \leq \epsilon \quad \forall t. \tag{7}$$

The resulting Pareto set lets a user choose between counterfactual policies with fewer, larger action interventions and policies with more frequent, smaller action interventions (Mothilal et al., 2020). We represent each policy by its intervention sequence and denote the Pareto policy set by $\mathcal{P}^\star$.

4 Method

HOPE solves the problem in Eq. 7 by a two-phase approach as shown in Fig. 1. First, a reach-avoid residual ranks intervention sequences by their counterfactual trajectories, guiding the search of the policy toward the exact validity condition. Then, validity-preserving reduction deletes, refits, and contracts interventions to reduce their number and magnitude. Each candidate policy is evaluated under Eq. 2 with the fixed reference randomization $\hat{\mathbf{u}}^{(0)}$.

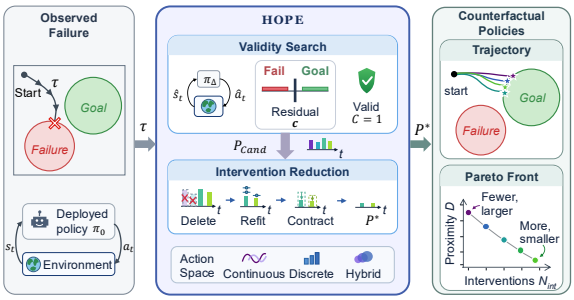


Figure 1: Overview of HOPE.

4.1 Reach-avoid residual

The binary indicator $C(\boldsymbol{\Delta}; \hat{\mathbf{u}}^{(0)})$ is zero for every invalid policy, so it cannot rank candidates during search. To measure how an invalid trajectory falls short of reach-avoid validity, we define the residual vector

$$\mathbf{c}(\boldsymbol{\Delta}; \hat{\mathbf{u}}) := \begin{bmatrix} \mathbf{c}^{\mathcal{F}}(\boldsymbol{\Delta}; \hat{\mathbf{u}}) \\ \mathbf{c}^{\mathcal{G}}(\boldsymbol{\Delta}; \hat{\mathbf{u}}) \end{bmatrix}, \tag{8}$$

where $\mathbf{c}^{\mathcal{F}}$ measures proximity to or penetration of the failure set, and $\mathbf{c}^{\mathcal{G}}$ measures the remaining goal-reaching error.

Each task constructs these components from its state and event information. Geometric tasks use normalized signed distances to $\mathcal{F}$ and $\mathcal{G}$ to measure failure proximity or penetration and remaining

goal-reaching error. Event-based tasks combine closest-approach distances with indicators of failure and missed arrival. Appendix A.3 details the definition of $\mathbf{c}^{\mathcal{F}}$ and $\mathbf{c}^{\mathcal{G}}$.

The corresponding search loss is defined as

$$\mathcal{L}(\mathbf{\Delta}) := \frac{1}{2}\left\|\mathbf{c}(\mathbf{\Delta}; \hat{\mathbf{u}}^{(0)})\right\|_2^2. \tag{9}$$

The loss ranks candidate intervention sequences and guides their updates, while $C(\mathbf{\Delta}; \hat{\mathbf{u}}^{(0)})$ indicates whether the corresponding counterfactual policy is valid.

4.2 Residual-guided validity search

Minimizing the reach-avoid loss requires complete closed-loop rollouts because interventions are coupled across time. These rollouts provide residual values but no derivatives with respect to the intervention sequence. We therefore use the same loss with search operations adapted to each intervention space: a joint smoothed local model for continuous action interventions, loss-ranked beam search for discrete action interventions, and alternating type selection and parameter refinement for hybrid interventions. Each variant starts from $\mathbf{\Delta} = \mathbf{0}$, evaluates the counterfactual trajectory for each intervention sequence under $\hat{\mathbf{u}}^{(0)}$, and returns a sequence whose corresponding policy satisfies $C(\mathbf{\Delta}; \hat{\mathbf{u}}^{(0)}) = 1$. It returns $\emptyset$ if no valid policy is found. In what follows, we explain the intervention search for different action spaces.

Case I: Intervention search in continuous action space. The continuous search updates all intervention times jointly to account for their temporal dependence. Let $\mathbf{d} = (\mathbf{d}_k, \ldots, \mathbf{d}_{t_f-1})$ be a tentative update to the intervention sequence, where $\mathbf{d}_t$ changes the action intervention at time t. Let $\mathbf{J}_t$ map $\mathbf{d}_t$ to the resulting residual change, and let $\mathbf{J} = [\mathbf{J}_k \ \cdots \ \mathbf{J}_{t_f-1}]$. The joint local model is

$$\mathbf{c}(\mathbf{\Delta} + \mathbf{d}; \hat{\mathbf{u}}^{(0)}) \approx \mathbf{c}(\mathbf{\Delta}; \hat{\mathbf{u}}^{(0)}) + \mathbf{J}\mathbf{d}. \tag{10}$$

Symmetric perturbation rollouts estimate residual sensitivities for the local model (Spall, 1992; Duchi et al., 2015). Because action projection can suppress infinitesimal perturbations, we estimate sensitivity over a finite radius. Let $\mu > 0$ be the smoothing radius, N_{dir} the number of probe directions, $\mathbf{v}_\ell$ a direction in the continuous intervention space, and $\mathbf{V}_{t,\ell}$ the intervention sequence whose time-t block is $\mathbf{v}_\ell$ and whose other blocks are zero. The sensitivity estimate at time t is

$$\widehat{\mathbf{J}}_t := \frac{1}{2\mu N_{\text{dir}}} \sum_{\ell=1}^{N_{\text{dir}}} \left[\mathbf{c}(\mathbf{\Delta} + \mu\mathbf{V}_{t,\ell}; \hat{\mathbf{u}}^{(0)}) - \mathbf{c}(\mathbf{\Delta} - \mu\mathbf{V}_{t,\ell}; \hat{\mathbf{u}}^{(0)})\right] \mathbf{v}_\ell^{\mathsf{T}}. \tag{11}$$

Concatenating the estimates as $\widehat{\mathbf{J}} = [\widehat{\mathbf{J}}_k \ \cdots \ \widehat{\mathbf{J}}_{t_f-1}]$ captures how action interventions throughout the window affect the trajectory-level residual.

The smoothed joint model produces a bounded projected Gauss-Newton update:

$$\min_{\mathbf{d}} \ \frac{1}{2}\left\|\mathbf{c}(\mathbf{\Delta}; \hat{\mathbf{u}}^{(0)}) + \widehat{\mathbf{J}}\mathbf{d}\right\|_2^2 \quad \text{s.t.} \quad \|\mathbf{\Delta}_t + \mathbf{d}_t\|_2 \le \epsilon \quad \forall t. \tag{12}$$

The joint solution coordinates updates across intervention times, while the constraints enforce the intervention bound. A line search evaluates $\mathbf{\Delta} + \eta\mathbf{d}$ and accepts the first step size η that decreases the evaluated search loss. The search then evaluates the validity indicator and repeats the update if the corresponding policy remains invalid.

Case II: Intervention search in discrete action space. The discrete search uses rollout-based loss reduction to rank action substitutions. Let $\mathbf{\Delta}^{(t \leftarrow a)}$ denote the intervention sequence that substitutes action $a \in \mathcal{A}$ at time t, where $a \neq \pi_0(\hat{\mathbf{s}}_t)$. The largest available loss reduction at time t is

$$\ell_t := \mathcal{L}(\mathbf{\Delta}) - \min_{a \in \mathcal{A}:\, a \neq \pi_0(\hat{\mathbf{s}}_t)} \mathcal{L}(\mathbf{\Delta}^{(t \leftarrow a)}). \tag{13}$$

The minimizing action gives the substitution, and ℓ_t ranks its intervention time.

Beam search combines substitutions without committing to a single locally best time. It keeps multiple partial intervention sequences, expands them with high-ranked substitutions, and evaluates the resulting sequences using $\mathcal{L}$ and C. After each expansion, it recomputes the rankings because the reached states and subsequent policy actions have changed. The search terminates when the policy represented by a sequence satisfies the validity condition.

Case III: Intervention search in hybrid action space. The hybrid search alternates discrete type selection and continuous parameter refinement. A coarse parameter grid first ranks candidate intervention types and parameters at each time by their reduction in $\mathcal{L}$. The beam then expands a partial intervention sequence with a high-ranked time and intervention type, after which the joint continuous update refines the parameters on its selected support. The search recomputes the rankings after each trajectory change and alternates these two operations until the corresponding policy satisfies the validity condition.

Algorithm 1 HOPE counterfactual policy generation.

Input: π_0, τ, $\hat{\mathbf{u}}^{(0)}$, $\mathcal{E}$, and ϵ
Output: $\mathcal{P}^\star$ or $\emptyset$

$\boldsymbol{\Delta} \leftarrow \textsc{ValiditySearch}(\mathbf{0})$
$\mathcal{P}_{\text{cand}} \leftarrow \{\boldsymbol{\Delta}\}$
repeat
  $(\boldsymbol{\Delta}, \mathcal{P}_{\text{cand}}, \texttt{deleted}) \leftarrow \textsc{DelSweep}(\boldsymbol{\Delta}, \mathcal{P}_{\text{cand}})$
  if `deleted` **then**
    $\widetilde{\boldsymbol{\Delta}} \leftarrow \textsc{SupportRestrictedRefit}(\boldsymbol{\Delta}, \mathcal{T}(\boldsymbol{\Delta}))$
    if $C(\widetilde{\boldsymbol{\Delta}}; \hat{\mathbf{u}}^{(0)}) = 1$ **then**
      $\boldsymbol{\Delta} \leftarrow \widetilde{\boldsymbol{\Delta}}$
      $\mathcal{P}_{\text{cand}} \leftarrow \mathcal{P}_{\text{cand}} \cup \{\boldsymbol{\Delta}\}$
    end if
  end if
until `deleted` $=$ `false`
if $\mathcal{E}$ has continuous parameters **then**
  $\mathcal{P}_{\text{cand}} \leftarrow \mathcal{P}_{\text{cand}} \cup \textsc{MagContract}(\mathcal{P}_{\text{cand}})$
end if
return $\mathcal{P}^\star = \textsc{ParetoFilter}(\mathcal{P}_{\text{cand}})$

4.3 Validity-preserving intervention reduction

A valid counterfactual policy may still have large N or D. Restricting support before reaching validity may exclude coordinated interventions; deleting an intervention afterward can change the closed-loop trajectory and invalidate the remaining sequence. We therefore reduce intervention number and magnitude only after finding a valid policy and check validity after each accepted change.

Starting from the valid intervention sequence $\boldsymbol{\Delta}$ found in Section 4.2, the reduction procedure initializes $\mathcal{P}_{\text{cand}} = \{\boldsymbol{\Delta}\}$. It alternates deletion sweeps with support-restricted refitting, applies magnitude contraction where continuous parameters are available, and finally performs Pareto filtering. Every $\boldsymbol{\Delta}' \in \mathcal{P}_{\text{cand}}$ defines a counterfactual policy that satisfies $C(\boldsymbol{\Delta}'; \hat{\mathbf{u}}^{(0)}) = 1$.

Validity-preserving deletion. Greedy deletion reduces temporal support while preserving the working policy's validity. The method examines active intervention times in increasing order of $\|\boldsymbol{\Delta}_t\|_{\mathcal{E}}$. For each time t, it forms $\boldsymbol{\Delta}^{(-t)}$ by setting only the time-t action intervention to zero. The deletion is accepted only if $C(\boldsymbol{\Delta}^{(-t)}; \hat{\mathbf{u}}^{(0)}) = 1$. Each accepted sequence becomes the working intervention sequence, is added to $\mathcal{P}_{\text{cand}}$, and decreases the support size by one.

Support-restricted refitting. After each successful deletion sweep, refitting updates only the remaining support while keeping deleted interventions fixed at zero. Continuous refitting applies Eq. 12, discrete refitting reselects substitutions, and hybrid refitting combines type reselection with continuous refinement. A refitted sequence replaces the current one and enters $\mathcal{P}_{\text{cand}}$ only if it remains valid. Deletion and refitting repeat until no remaining intervention can be individually removed without violating validity.

Magnitude contraction. Magnitude contraction reduces D without changing intervention support. For continuous interventions, it evaluates scaled sequences $\boldsymbol{\Delta}(\alpha) = \alpha\boldsymbol{\Delta}$, $\alpha \in (0, 1)$. For hybrid interventions, only continuous parameters are scaled with discrete types fixed. A contracted sequence is added to $\mathcal{P}_{\text{cand}}$ when validity is preserved. Purely discrete interventions are not contracted.

Pareto policy set. Pareto filtering converts the collected valid intervention sequences into counterfactual policies with different sparsity-proximity trade-offs. The policy represented by $\boldsymbol{\Delta}$ is removed if another policy, represented by $\boldsymbol{\Delta}'$, satisfies $N(\boldsymbol{\Delta}') \leq N(\boldsymbol{\Delta})$ and $D(\boldsymbol{\Delta}') \leq D(\boldsymbol{\Delta})$, with at least one strict inequality. The remaining policies form $\mathcal{P}^\star$. Algorithm 1 summarizes the counterfactual policy generation procedure by HOPE.

Table 1: Three benchmarks for evaluation. $\mathcal{A}_d = \{\text{SLOWER}, \text{IDLE}, \text{FASTER}\}$ and $\mathcal{A}_h = \mathcal{K} \times \mathcal{A}_c$, where $\mathcal{K} = \{\text{BRAKE}, \text{THROTTLE}, \text{SET}\}$ and $\mathcal{A}_c = [0, 1]$ is the continuous parameter space.

Benchmark	Environment	Goal $\mathcal{G}$ / Failure $\mathcal{F}$	π_0	$\mathcal{S}$	$\mathcal{A}$	$\mathcal{E}$
B1 (continuous action)	`SafetyPointGoal1-v0`	Goal reached / hazard contact	SAC	$\mathbb{R}^{39}$	$[-1, 1]^2$	$\mathbb{R}^2$
B2 (discrete action)	`intersection-v2`	Planned exit / collision or off-road	Double DQN	$\mathbb{R}^{41}$	$\mathcal{A}_d$	$\mathcal{A}_d$
B3 (hybrid action)	`intersection-v2`	Planned exit / collision or off-road	PPO	$\mathbb{R}^{41}$	$\mathcal{A}_h$	$\mathcal{A}_h$

Table 2: Performance comparison of HOPE and baselines. P_1 is the percentage of valid counterfactuals requiring one intervention; Sparsity, proximity, and recency are medians (25th/75th percentiles) over valid counterfactuals. **Bold** and underlining mark the best and second-best results, respectively.

Bench.	Method	Validity ↑	$\mathbf{P_1}$ ↑	Sparsity (N) ↓	Proximity (D) ↓	$\mathbf{N \cdot D}$ ↓	Recency (R) ↓
B1	Random Support	0.47	6.4%	6.00 (2.75/12.00)	0.50 (0.50/0.50)	3.00 (1.37/6.00)	82.00 (76.00/86.50)
	HIGHLIGHTS	0.12	0.0%	12.00 (7.00/12.00)	0.50 (0.50/0.50)	6.00 (3.50/6.00)	**20.00** (16.50/25.50)
	RACCER	0.49	6.4%	4.00 (2.25/4.75)	0.50 (0.50/0.50)	2.00 (1.12/2.38)	90.00 (85.25/90.00)
	ACTER	**0.81**	0.0%	20.00 (18.00/23.00)	**0.25** (0.22/0.27)	5.05 (4.26/5.94)	85.00 (80.75/89.00)
	HOPE (ours)	**0.81**	**24.5%**	**2.00** (1.00/3.00)	0.33 (0.21/0.45)	**0.67** (0.28/1.22)	78.00 (73.75/83.00)
B2	Random Support	**0.81**	20.8%	2.00 (1.25/3.00)	0.54 (0.50/0.80)	1.50 (1.00/2.00)	28.00 (22.00/31.00)
	HIGHLIGHTS	0.56	1.4%	6.00 (4.00/7.00)	0.67 (0.58/0.75)	3.75 (2.50/5.12)	**16.00** (12.75/23.25)
	RACCER	0.36	16.7%	2.00 (1.00/3.00)	**0.50** (0.50/0.50)	**1.00** (0.50/1.88)	33.00 (31.00/35.00)
	ACTER	0.72	0.0%	10.00 (7.75/13.25)	0.63 (0.58/0.69)	7.00 (4.50/8.62)	27.50 (21.75/30.00)
	HOPE (ours)	0.74	**40.3%**	**1.00** (1.00/2.00)	**0.50** (0.50/0.75)	**1.00** (0.50/1.50)	29.00 (20.00/32.00)
B3	Random Support	0.81	44.3%	**1.00** (1.00/3.00)	0.37 (0.25/0.46)	0.58 (0.28/1.03)	29.00 (19.00/31.00)
	HIGHLIGHTS	0.47	5.7%	8.00 (3.00/12.00)	0.36 (0.21/0.49)	1.98 (1.00/3.99)	**15.00** (12.00/19.00)
	RACCER	0.56	25.7%	2.00 (1.00/3.00)	0.28 (0.21/0.48)	**0.50** (0.25/1.01)	33.00 (31.00/36.00)
	ACTER	0.79	0.0%	31.00 (27.00/35.00)	**0.25** (0.19/0.29)	7.83 (5.60/9.46)	32.00 (30.00/36.00)
	HOPE (ours)	**0.83**	**55.7%**	**1.00** (1.00/2.00)	0.50 (0.44/0.75)	0.75 (0.46/1.39)	26.50 (18.00/30.00)

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Benchmarks and incidents. Table 1 presents the three benchmarks. B1 uses SafetyPointGoal1-v0 from Safety-Gymnasium (Ji et al., 2023), where a planar robot uses continuous actions to reach a goal while avoiding hazards. B2 and B3 use intersection-v2 from highway-env (Leurent, 2018) with discrete and hybrid actions, respectively, where a vehicle must reach its planned exit without collision or roadway departure. Each benchmark contains 200 failure incidents evenly collected from five frozen deployed policies with distinct configurations (40 per policy).

Baselines. We compare HOPE against two RL counterfactual methods, RACCER (Gajcin & Dusparic, 2023) and ACTER (Gajcin & Dusparic, 2024a), and two diagnostic controls, HIGHLIGHTS (Amir & Amir, 2018) and Random Support (Hooker et al., 2019). RACCER searches a tree of contiguous open-loop actions, and ACTER uses NSGA-II to optimize open-loop actions over the full window. HIGHLIGHTS uses Q-value importance to select intervention states and alternative actions. Random Support samples intervention times uniformly to isolate the effect of state ranking.

Evaluation protocol. For an incident with first failure time t_f, action interventions are allowed over $[k, t_f)$, with $W = t_f - k$. B1 uses $W = \min(90, t_f)$, $H = 150$, and $|\mathbf{\Delta}_t|_2 \leq 0.5$; B2 and B3 use $W = 36$ and $H = 60$. B2 assigns magnitudes 0.5 and 1.0 to adjacent and two-level action substitutions, respectively; B3 normalizes target-speed changes by the maximum target speed to obtain magnitudes in $[0, 1]$. For each incident, all methods start from the same state at k, receive the same disturbance sequence, and use the same reach-avoid test over H. An incident is solvable if the dense oracle described in Appendix B.3 finds a valid counterfactual policy. This yields 188, 144, and 140 solvable incidents for B1, B2, and B3 among the 200 failure incidents, respectively.

Appendix B provides implementation details, including benchmark settings, policies, incidents, rollout and reporting protocols, baselines, and ablation studies.

5.2 MAIN RESULTS

Table 2 reports our main results in terms of validity, sparsity (N), proximity (D), and recency (R). HOPE achieves a strong validity-sparsity trade-off: its residual-guided closed-loop search and deletion/refitting yield smaller supports than open-loop RACCER and ACTER, with the best validity

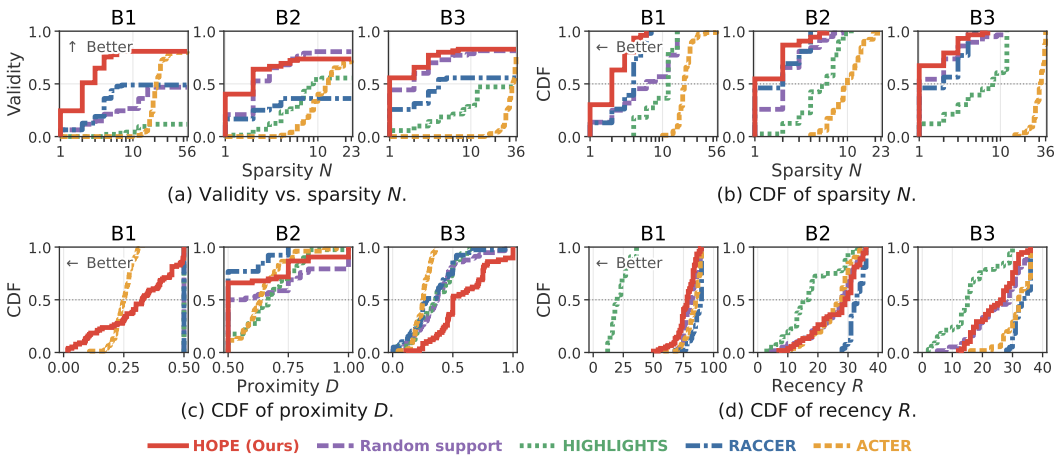


Figure 2: Counterfactual-policy quality on the three benchmarks.

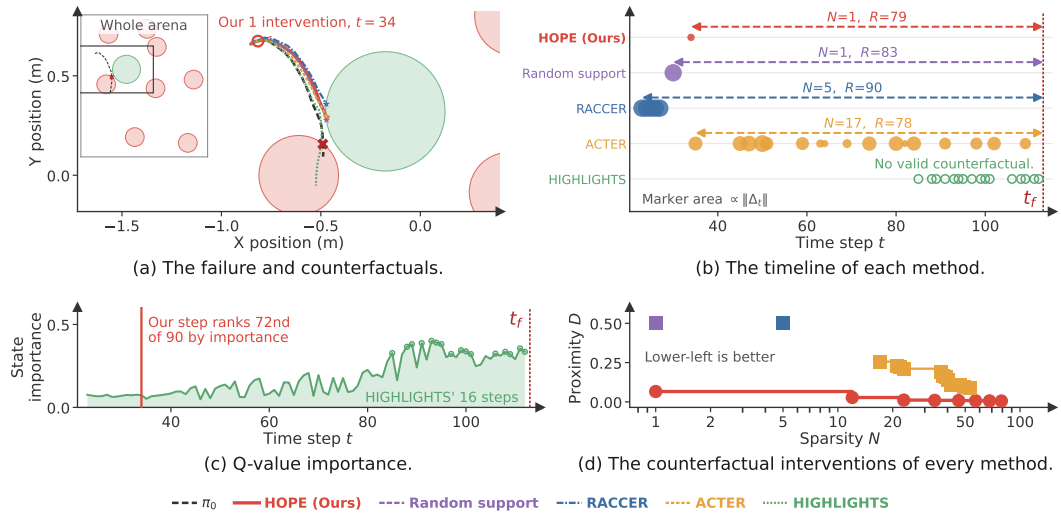


Figure 3: Counterfactual policies for one failure incident in B1.

in most tested cases. Random Support has slightly higher B2 validity through broader coverage of the small discrete space, but HOPE yields substantially more single-intervention solutions (P_1). HIGHLIGHTS favors recency but has low validity. Since D is averaged over the support, $N \cdot D$ captures total intervention magnitude: HOPE and RACCER are consistently lowest or near-lowest, Random Support is intermediate, and ACTER and HIGHLIGHTS have larger intervention burdens.

Figure 2 shows validity versus intervention budget and the distributions of sparsity, proximity, and recency. HOPE gains validity fastest with small intervention budgets and reaches competitive or best final validity compared to other policies.

5.3 Case Study

Figure 3(a) illustrates a representative B1 failure trajectory and its counterfactual trajectories generated by the evaluated methods. The deployed policy enters the hazard at $t_f = 113$, with action interventions allowed over $t = [23, 113)$. We have the following observations.

O1: Sparse interventions improve interpretability. Figure 3(b) shows that HOPE and Random Support generate valid counterfactual policies with a single action intervention, while RACCER and ACTER require 5 and 17, respectively; HIGHLIGHTS generates no valid policy. Sparse interventions make outcome-changing decisions easier to identify. This observation is consistent with prior findings that dense sequences distribute changes across multiple steps, obscuring the contribution of any single intervention (Miller, 2019; Ribeiro et al., 2016).

Table 3: Ablation results on the three benchmarks. Contraction is not applicable to B2, and Group-OMP is tested only on B1. **Bold** and underlining mark the best and second-best results, respectively.

Bench.	Variant	Validity ↑	$\mathbf{P_1}$ ↑	Sparsity (N) ↓	Proximity (D) ↓	$\mathbf{N \cdot D}$ ↓	Recency (R) ↓
B1	Sparsity-First	0.61	**37.2%**	**1.00** (1.00/2.00)	**0.17** (0.09/0.30)	**0.20** (0.10/0.46)	79.00 (68.00/86.00)
	Open-loop	0.69	8.5%	3.00 (2.00/5.00)	0.45 (0.29/0.50)	1.38 (0.80/1.93)	79.00 (73.00/86.00)
	w/o Refit	**0.81**	9.6%	4.00 (2.00/7.00)	0.18 (0.09/0.27)	0.73 (0.29/1.09)	81.00 (74.00/85.25)
	w/o Contraction	**0.81**	24.5%	2.00 (1.00/3.00)	0.46 (0.33/0.50)	0.92 (0.50/1.41)	**78.00** (73.75/83.00)
	Group-OMP	0.48	11.7%	2.00 (2.00/3.00)	0.26 (0.22/0.29)	0.56 (0.34/0.94)	79.00 (75.00/86.00)
	HOPE (ours)	**0.81**	24.5%	2.00 (1.00/3.00)	0.33 (0.21/0.45)	0.67 (0.28/1.22)	**78.00** (73.75/83.00)
B2	Sparsity-First	0.62	**40.3%**	**1.00** (1.00/2.00)	0.75 (0.50/1.00)	1.00 (0.50/1.50)	28.00 (18.00/31.00)
	Open-loop	0.49	37.5%	**1.00** (1.00/1.00)	**0.50** (0.50/0.78)	**0.50** (0.50/1.00)	**25.00** (17.00/29.50)
	w/o Refit	**0.74**	**40.3%**	**1.00** (1.00/2.00)	**0.50** (0.50/0.75)	1.00 (0.50/1.50)	29.00 (20.00/32.00)
	HOPE (ours)	**0.74**	**40.3%**	**1.00** (1.00/2.00)	**0.50** (0.50/0.75)	1.00 (0.50/1.50)	29.00 (20.00/32.00)
B3	Sparsity-First	0.73	**55.7%**	**1.00** (1.00/1.00)	**0.50** (0.45/0.75)	0.62 (0.49/1.00)	27.00 (18.00/30.00)
	Open-loop	0.51	42.9%	**1.00** (1.00/1.00)	**0.50** (0.44/0.75)	**0.50** (0.48/0.79)	**20.50** (15.50/28.25)
	w/o Refit	**0.83**	**55.7%**	**1.00** (1.00/2.00)	**0.50** (0.44/0.75)	0.75 (0.46/1.39)	26.50 (18.00/30.00)
	w/o Contraction	**0.83**	**55.7%**	**1.00** (1.00/2.00)	**0.50** (0.46/0.75)	0.75 (0.50/1.39)	26.50 (18.00/30.00)
	HOPE (ours)	**0.83**	**55.7%**	**1.00** (1.00/2.00)	**0.50** (0.44/0.75)	0.75 (0.46/1.39)	26.50 (18.00/30.00)

O2: Q-value importance does not identify effective intervention times. Figure 3(c) shows that 16 selected steps occur at $t = 85$-112, near the failure, yet none yields a valid counterfactual policy. In contrast, the four successful methods first intervene at $t = 23 - 35$; HOPE intervenes at $t = 34$, although this step ranks only 72nd of 90 by Q-value importance. In this incident, Q-value importance highlights large action-value differences near failure rather than effective corrective timing.

O3: Pareto front offers incident-specific alternatives. Figure 3(d) shows that HOPE and ACTER produce multi-point Pareto fronts, while Random Support and RACCER each return only a single counterfactual policy. A practitioner can therefore choose between fewer, larger action interventions for a concise explanation and more frequent, smaller action interventions when per-step deviation is constrained, without specifying a fixed scalar weighting in advance. For this incident, HOPE also provides the lowest-burden counterfactual policy with $N \cdot D = 0.066$.

5.4 Ablation Study

Table 3 evaluates five design choices in HOPE. *Sparsity-First* prioritizes sparsity before validity, while *Open-loop* replays factual actions at non-intervened steps. *Group-OMP* (Tropp, 2004) replaces the joint local model on B1 with greedy selection of intervention times. *w/o Refit* and *w/o Contraction* remove support-restricted refitting and magnitude contraction, respectively.

The results show that validity-first search, closed-loop feedback, and joint temporal optimization are important for valid counterfactual policies. Sparsity-First reduces validity by $12\% - 25\%$ on three benchmarks; although it increases P_1 on B1, this gain comes with lower overall validity. Open-loop execution further reduces validity by $15\% - 39\%$, showing the importance of adapting actions to the reached counterfactual states. Group-OMP decreases validity by 41% on B1, confirming the benefit of coordinating interventions over time. Removing refitting or contraction preserve validity but increase intervention burden on sparsity or proximity. Overall, the search components improve validity, while refitting and contraction improve sparsity and proximity without reducing validity.

6 Conclusion

We presented HOPE, a unified framework for generating trajectory-specific counterfactual policies that explain RL failures through sparse, time-indexed action interventions while preserving the deployed policy's closed-loop feedback. HOPE combines residual-guided validity search with validity-preserving intervention reduction, enabling counterfactual explanations across continuous, discrete, and hybrid action spaces. Experiments on 600 failure incidents show that HOPE achieves competitive or best validity while consistently producing substantially sparser counterfactual policies. These results demonstrate that separating validity search from intervention reduction provides an effective way to generate concise, actionable counterfactual explanations for sequential decision-making failures.

AI USE STATEMENT

In this work, we used generative AI tools only to polish the manuscript for clarity, grammar, and readability. We did not use generative AI tools to generate or refine research ideas, hypotheses, methodology, experimental designs, or literature analysis. All technical content and research contributions were developed by the authors. We reviewed all AI-assisted work and take full responsibility for the final manuscript, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE transactions on automatic control*, 62(8): 3861–3876, 2016.

Dan Amir and Ofra Amir. Highlights: Summarizing agent behavior to people. In *Proceedings of the 17th international conference on autonomous agents and multiagent systems*, pp. 1168–1176, 2018.

Akanksha Atrey, Kaleigh Clary, and David Jensen. Exploratory not explanatory: Counterfactual analysis of saliency maps for deep reinforcement learning. *arXiv preprint arXiv:1912.05743*, 2019.

Zana Buçinca, Phoebe Lin, Krzysztof Z Gajos, and Elena L Glassman. Proxy tasks and subjective measures can be misleading in evaluating explainable ai systems. In *Proceedings of the 25th international conference on intelligent user interfaces*, pp. 454–464, 2020.

Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134(1):19–67, 2005.

Shuyang Dong, Shangtong Zhang, and Lu Feng. Counterfactual explanations for continuous action reinforcement learning. *arXiv preprint arXiv:2505.12701*, 2025.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Gabriel Dulac-Arnold, Nir Levine, Daniel J Mankowitz, Jerry Li, Cosmin Paduraru, Sven Gowal, and Todd Hester. Challenges of real-world reinforcement learning: definitions, benchmarks and analysis. *Machine Learning*, 110(9):2419–2468, 2021.

Jaime F Fisac, Mo Chen, Claire J Tomlin, and S Shankar Sastry. Reach-avoid problems with time-varying dynamics, targets and constraints. In *Proceedings of the 18th international conference on hybrid systems: computation and control*, pp. 11–20, 2015.

Jasmina Gajcin and Ivana Dusparic. Raccer: Towards reachable and certain counterfactual explanations for reinforcement learning. *arXiv preprint arXiv:2303.04475*, 2023.

Jasmina Gajcin and Ivana Dusparic. Acter: Diverse and actionable counterfactual sequences for explaining and diagnosing rl policies. *arXiv preprint arXiv:2402.06503*, 2024a.

Jasmina Gajcin and Ivana Dusparic. Redefining counterfactual explanations for reinforcement learning: Overview, challenges and opportunities. *ACM Computing Surveys*, 56(9):1–33, 2024b.

Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Paul Henne, Aleksandra Kulesza, Karla Perez, and Augustana Houcek. Counterfactual thinking and recency effects in causal judgment. *Cognition*, 212:104708, 2021.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32, 2019.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. *Advances in Neural Information Processing Systems*, 36:18964–18993, 2023.

Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 353–362, 2021.

Milad Kazemi, Jessica Lally, Ekaterina Tishchenko, Hana Chockler, and Nicola Paoletti. Counterfactual influence in markov decision processes. *arXiv preprint arXiv:2402.08514*, 2024.

Paul Kobialka, Lina Gerlach, Francesco Leofante, Erika Abrahám, Silvia Lizeth Tapia Tarifa, and Einar Broch Johnsen. Counterfactual strategies for markov decision processes. *arXiv preprint arXiv:2505.09412*, 2025.

Edouard Leurent. An environment for autonomous driving decision-making. `https://github.com/eleurent/highway-env`, 2018.

Vincent Liu, Raksha Kumaraswamy, Lei Le, and Martha White. The utility of sparse representations for control in reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4384–4391, 2019.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Stephanie Milani, Nicholay Topin, Manuela Veloso, and Fei Fang. Explainable reinforcement learning: A survey and comparative review. *ACM Computing Surveys*, 56(7):1–36, 2024.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 607–617, 2020.

Maria Movin, Guilherme Dinis Junior, Jaakko Hollmén, and Panagiotis Papapetrou. Explaining black box reinforcement learning agents through counterfactual policies. In *International Symposium on Intelligent Data Analysis*, pp. 314–326. Springer, 2023.

Masaaki Nagahara, Daniel E Quevedo, and Dragan Nešić. Maximum hands-off control: a paradigm of control effort minimization. *IEEE Transactions on Automatic Control*, 61(3):735–747, 2015.

Matthew L Olson, Roli Khanna, Lawrence Neal, Fuxin Li, and Weng-Keen Wong. Counterfactual state explanations for reinforcement learning agents via generative deep learning. *Artificial Intelligence*, 295:103455, 2021.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Oswin So, Cheng Ge, and Chuchu Fan. Solving minimum-cost reach avoid using reinforcement learning. *Advances in Neural Information Processing Systems*, 37:30951–30984, 2024.

James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 1992.

Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information theory*, 50(10):2231–2242, 2004.

Stratis Tsirtsis and Manuel Rodriguez. Finding counterfactually optimal action sequences in continuous state spaces. *Advances in Neural Information Processing Systems*, 36:3220–3247, 2023.

Stratis Tsirtsis, Abir De, and Manuel Rodriguez. Counterfactual explanations in sequential decision making under uncertainty. *Advances in Neural Information Processing Systems*, 34:30127–30139, 2021.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in neural information processing systems*, 33:14129–14142, 2020.

APPENDIX

The appendix contains:

- **Appendix A** summarizes the notation and specifies the technical detail.
- **Appendix B** provides the implementation details.
- **Appendix C** examines sensitivity to the intervention window W and the per-step magnitude bound ϵ.
- **Appendix D** discusses limitations and future work.

A TECHNICAL DETAILS

This appendix gives the notation and technical details used by HOPE. It defines intervention maps and magnitudes, specifies the benchmark residuals, and adds the search details.

A.1 NOTATION

Table 4 lists the notation used in the formulation and method. Factual variables are unmarked, while hats mark variables in a counterfactual trajectory.

A.2 INTERVENTION MAPS AND MAGNITUDES

The intervention map and magnitude give a common interface across action spaces. The symbol $\mathbf{0}$ denotes the null action intervention. For every action space, $\varphi(\mathbf{a}, \mathbf{0}) = \mathbf{a}$ and $\|\mathbf{0}\|_{\mathcal{E}} = 0$. The value $\|\boldsymbol{\Delta}_t\|_{\mathcal{E}}$ measures the size of the action change under the metric chosen for $\mathcal{E}$; the subscript therefore identifies the intervention space, not an additional vector coordinate. This quantity need not be a vector norm for discrete or hybrid interventions.

Continuous actions. For a bounded continuous action space, let $\boldsymbol{\Gamma}$ scale normalized interventions to physical action units and let $\Pi_{\mathcal{A}}$ project onto the admissible action set. A standard map and magnitude are

$$\varphi(\mathbf{a}, \boldsymbol{\Delta}_t) = \Pi_{\mathcal{A}}(\mathbf{a} + \boldsymbol{\Gamma}\boldsymbol{\Delta}_t), \qquad \|\boldsymbol{\Delta}_t\|_{\mathcal{E}} = \|\boldsymbol{\Delta}_t\|_2. \tag{14}$$

For a box action space with lower and upper limits ℓ_j and u_j on coordinate j, respectively, we use

$$\boldsymbol{\Gamma} = \sqrt{d_a}\,\mathrm{diag}(u_1 - \ell_1, \ldots, u_{d_a} - \ell_{d_a}). \tag{15}$$

This definition supports any continuous action dimension d_a.

Discrete actions. For a discrete action space, a nonzero action intervention substitutes an action $\mathbf{a}'$ for the deployed action $\mathbf{a}$. The map sets $\varphi(\mathbf{a}, \boldsymbol{\Delta}_t) = \mathbf{a}'$, and the magnitude is a task-defined action distance

$$\|\boldsymbol{\Delta}_t\|_{\mathcal{E}} = d_{\mathcal{A}}(\mathbf{a}, \mathbf{a}'). \tag{16}$$

An unordered action set may use the indicator distance, while an ordered set may use normalized graph or index distance. Thus, the definition applies to finite action spaces of any size and does not require every substitution to have unit magnitude.

Hybrid actions. For a hybrid action space, a nonzero action intervention contains a discrete type and one or more continuous parameters. The map converts this pair into an admissible action. Its magnitude uses a task-defined distance between the deployed and intervened actions, with continuous coordinates normalized by their physical ranges when needed. The distance may also include a discrete type cost, but it must be zero when the action intervention leaves the executed action unchanged. Appendix B.2 gives the three benchmark definitions.

A.3 REACH-AVOID RESIDUALS

The reach-avoid residual gives validity search a graded search signal while the exact indicator C decides validity. Let $\mathbf{o}_j$ and $r_j^{\mathcal{F}}$ denote the center and radius of failure region j, respectively, and let $\hat{\mathbf{x}}_t$ be the position component of $\hat{\mathbf{s}}_t$. The signed failure margin is

$$g_j(\hat{\mathbf{s}}_t) = r_j^{\mathcal{F}} - \|\hat{\mathbf{x}}_t - \mathbf{o}_j\|_2, \tag{17}$$

Table 4: Notation used in the formulation and method.

Notation	Definition
Environment and deployed policy	
t, H, t_f	time step, rollout horizon, and first failure time
$\mathcal{S}, \mathcal{A}, \mathcal{U}$	state, action, and simulator-randomization spaces
$\mathbf{s}_t, \mathbf{a}_t$	factual state and action at time t
f_{sim}	simulator transition function
π_0	deployed policy, with $\mathbf{a}_t = \pi_0(\mathbf{s}_t)$
$\mathcal{F}, \mathcal{G}$	failure and goal sets
τ	observed failure trajectory
Counterfactual policy and criteria	
k, W	intervention start time and window length, with $W = t_f - k$
$\mathcal{E}, \mathbf{\Delta}_t$	intervention space and action intervention at time t
$\mathbf{\Delta}$	intervention sequence over $[k, t_f)$
φ	intervention map, with $\varphi(\mathbf{a}, \mathbf{0}) = \mathbf{a}$
$\pi_{\mathbf{\Delta}}$	counterfactual policy induced by $\mathbf{\Delta}$
$\hat{\mathbf{s}}_t, \hat{\mathbf{a}}_t, \hat{\mathbf{u}}_t$	counterfactual state, action, and simulator randomization
$\hat{\tau}, \hat{T}$	counterfactual trajectory and stopping time
$\|\mathbf{\Delta}_t\|_{\mathcal{E}}, \epsilon$	action-intervention magnitude and per-step bound
$C(\mathbf{\Delta}; \hat{\mathbf{u}})$	reach-avoid validity indicator
$\mathcal{T}(\mathbf{\Delta})$	temporal support of the intervention sequence
N, D, R	sparsity, proximity, and recency
Search	
$\mathbf{c}^{\mathcal{F}}, \mathbf{c}^{\mathcal{G}}$	failure-avoidance and goal-attainment residuals
$\mathbf{c}, \mathcal{L}$	combined residual and search loss
μ, N_{dir}	smoothing radius and number of probe directions
$\mathbf{v}_\ell, \mathbf{V}_{t,\ell}$	probe direction ℓ and its embedding in time block t
$\widehat{\mathbf{J}}_t, \widehat{\mathbf{J}}$	estimated sensitivity block and full sensitivity matrix
$\mathbf{d}$	candidate continuous update
ℓ_t	best discrete loss reduction at time t
$\mathcal{P}_{\text{cand}}, \mathcal{P}^\star$	candidate set and returned Pareto policy set

where a positive value places the state inside the failure region. For $\beta > 0$, define $\sigma_\beta(x) = \beta^{-1}\log(1+\exp(\beta x))$ and

$$\text{smin}_\beta(x_{1:n}) = -\beta^{-1}\log\sum_{i=1}^{n}\exp(-\beta x_i). \tag{18}$$

B1 residual. The B1 residual uses normalized signed distances to the hazard and goal sets. Let $b_{\mathcal{F}}$ and $b_{\mathcal{G}}$ be the hazard and goal buffers, respectively, let $r_{\mathcal{G}}$ be the goal radius, and let $d_{\mathcal{G}}(\hat{\mathbf{s}}_t)$ be signed distance to the goal, with nonpositive values inside $\mathcal{G}$. The residual components are

$$\begin{aligned} c^{\mathcal{F}}_{j,t} &= \sigma_\beta\left(\frac{g_j(\hat{\mathbf{s}}_t) + b_{\mathcal{F}}}{r^{\mathcal{F}}_j}\right), \\ c^{\mathcal{G}} &= \sigma_\beta\left(\text{smin}_\beta\left(\left\{\frac{d_{\mathcal{G}}(\hat{\mathbf{s}}_t)}{r_{\mathcal{G}}}\right\}_{t=k}^{H}\right) + \frac{b_{\mathcal{G}}}{r_{\mathcal{G}}}\right). \end{aligned} \tag{19}$$

The vector $\mathbf{c}$ contains all $c^{\mathcal{F}}_{j,t}$ terms and the goal term $c^{\mathcal{G}}$. We use $\beta = 3$.

B2 and B3 residuals. The deadline benchmarks combine geometric distances with simulator events. Let $m^{\max}_{\mathcal{F}}$ be the largest normalized hazard-approach margin, $d^{\min}_{\mathcal{G}}$ the smallest remaining goal distance, $I_{\mathcal{F}}$ the failure indicator, and I_{miss} the missed-arrival indicator. With event penalties $\gamma_{\mathcal{F}}$ and $\gamma_{\mathcal{G}}$, the residual components are

$$c^{\mathcal{F}} = \sigma_\beta(m^{\max}_{\mathcal{F}}) + \gamma_{\mathcal{F}} I_{\mathcal{F}}, \qquad c^{\mathcal{G}} = \sigma_\beta\left(\frac{d^{\min}_{\mathcal{G}}}{r_{\mathcal{G}}}\right) + \gamma_{\mathcal{G}} I_{\text{miss}}. \tag{20}$$

The exact rollout outcome determines validity; these residuals only guide the search.

A.4 Supplementary Search Details

The search uses action-space-specific operations with the same loss and exact validity test. This section states the details needed to reproduce the three validity searches.

Table 5: Training configurations of the five deployed policies for the three benchmarks. P1 uses the base configuration, while P2–P5 each modify one hyperparameter. Each benchmark contains 200 evaluation incidents, with 40 incidents from each policy. γ is the discount factor; Here, k and M denote 10^3 and 10^6, respectively.

Setting	B1: SAC	B2: Double DQN	B3: PPO
Base configuration (P1)			
Network	$(64, 64)$	$(256, 256)$	$(256, 256)$
Learning rate	3×10^{-4}	5×10^{-4}	3×10^{-4}
Batch size	128	128	64
Discount γ	0.99	0.98	0.98
Replay buffer	50k	50k	–
Policy variants			
P1	Base	Base	Base
P2	$\gamma = 0.98$	$\gamma = 0.95$	$\gamma = 0.95$
P3	$\gamma = 0.995$	$\gamma = 0.99$	$\gamma = 0.99$
P4	$(128, 128)$	$(128, 128)$	$(128, 128)$
P5	$(256, 256)$	$(512, 512)$	$(512, 512)$

Continuous search. The estimator in Eq. 11 perturbs one time block at a time. The sequence $\mathbf{V}_{t,\ell}$ contains direction $\mathbf{v}_\ell \in \mathbb{R}^{d_a}$ in block t and zero in every other block. When N_{dir} is a multiple of d_a, scaled orthogonal directions satisfy

$$\frac{1}{N_{\mathrm{dir}}} \sum_{\ell=1}^{N_{\mathrm{dir}}} \mathbf{v}_\ell \mathbf{v}_\ell^{\mathsf{T}} = \mathbf{I}_{d_a}. \tag{21}$$

Here, $\mathbf{I}_{d_a}$ is the d_a-dimensional identity matrix. Estimating all W blocks uses $2WN_{\mathrm{dir}}$ closed-loop rollouts per Jacobian. A line search accepts the first tested update that lowers the unperturbed loss $\mathcal{L}$.

Discrete search. One leverage sweep evaluates every alternative action at every intervention time and uses $W(|\mathcal{A}|-1)$ rollouts. The beam first evaluates supports of size one and two, then adds high-leverage times. It recomputes leverage after each accepted expansion because the counterfactual trajectory and later deployed-policy actions may change.

Hybrid search. The hybrid search first evaluates each discrete type on a finite parameter grid. It uses these evaluations to rank intervention times and initialize the types and parameters. With the selected types fixed, the continuous search refines their parameters and updates the ranking after the trajectory changes.

B Implementation Details

This appendix gives the implementation details needed to interpret and reproduce the experiments. It describes the benchmarks and incidents, action interfaces and replay, evaluation protocol, and fixed search settings.

B.1 Benchmarks, Deployed Policies, and Incidents

The three benchmarks test continuous, discrete, and hybrid action interfaces, respectively. Their deployed policies and action spaces are shown in Table 1. Table 5 summarizes the training configurations of the five deployed policies for the three benchmarks.

B1. B1 uses `SafetyPointGoal1-v0` from Safety-Gymnasium (Ji et al., 2023) with horizon $H = 150$. Its 39-dimensional state describes the robot, goal, hazards, movable object, and remaining time. The action $\mathbf{a} \in [-1, 1]^2$ has two continuous components: the first drives the robot forward or backward, and the second turns it. Failure is hazard contact, and success is entry into the goal region. We use five SAC policies trained with different configurations.

B2. B2 uses `intersection-v2` from highway-env 1.12.1 (Leurent, 2018) with horizon $H = 60$, a policy frequency of 5 Hz, and a simulator frequency of 15 Hz. The 41-dimensional state represents

the ego vehicle, route progress, and nearby traffic. The three discrete actions set the target speed of the ego vehicle on the grid $\{0, 4.5, 9\}$ m/s: SLOWER and FASTER move it one level down and up, respectively, and IDLE keeps it. Failure is a collision or roadway departure, and success is arrival at the planned exit. We use five Double DQN policies trained with different configurations.

B3. B3 uses the same highway-env task, state, and reach-avoid condition as B2 but the hybrid action space $\mathcal{A}_h = \mathcal{K} \times \mathcal{A}_c$. Each action consists of a type in $\mathcal{K}$ and a parameter $a_c \in \mathcal{A}_c = [0, 1]$. BRAKE lowers the target speed to a fraction $1 - a_c$ of the current speed, THROTTLE raises it by a fraction a_c of the remaining speed range, and SET sets it to a fraction a_c of the maximum target speed. We use five PPO policies trained with different configurations.

The implementation of HIGHLIGHT baseline needs to use Q-value. On B1 and B2, the action values come from the critic of SAC and the Q-network of Double DQN, respectively. Since the PPO policies of B3 learn only a state value $V(\mathbf{s})$, we compute the action value by one simulator step, $Q(\mathbf{s}_t, \mathbf{a}) = r_t + \gamma V(\mathbf{s}_{t+1})$, where $\mathbf{s}_{t+1}$ is reached from the factual state $\mathbf{s}_t$ under the recorded randomness.

Each benchmark contains 200 evaluation incidents collected evenly from its five deployed policies, with 40 incidents per policy. For each incident, we store the simulator snapshot at k, indexed simulator randomness, the first failure time t_f, factual actions, policy identity, and per-step state checksums. We include an incident only when restoring the snapshot and replaying the stored randomness reproduces its checksums and failure.

B.2 Action Interfaces

Each benchmark defines the intervention map and magnitude from the physical effect of its action interface. All methods use the same definition within a benchmark.

B1. For B1, Eqs. 14 and 15 give

$$\varphi(\mathbf{a}, \mathbf{\Delta}_t) = \Pi_{\mathcal{A}}(\mathbf{a} + \mathbf{\Gamma}\mathbf{\Delta}_t), \qquad \|\mathbf{\Delta}_t\|_{\mathcal{E}} = \|\mathbf{\Delta}_t\|_2, \tag{22}$$

with $\|\mathbf{\Delta}_t\|_2 \leq 0.5$.

B2. A nonzero B2 action intervention substitutes one of the three ordered actions. Its normalized distance is

$$\|\mathbf{\Delta}_t\|_{\mathcal{E}} = \frac{|\hat{a}_t - \pi_0(\hat{\mathbf{s}}_t)|}{|\mathcal{A}| - 1}, \tag{23}$$

where the action indices follow SLOWER, IDLE, and FASTER. Adjacent and two-level substitutions therefore have magnitudes 0.5 and 1.0, respectively.

B3. A B3 action intervention produces a target-speed command v_t^{cmd}, while the deployed policy would produce target speed $v_t^{\pi_0}$ at the reached counterfactual state. Let $v_{\max}$ be the maximum target speed. The magnitude is

$$\|\mathbf{\Delta}_t\|_{\mathcal{E}} = \frac{|v_t^{\mathrm{cmd}} - v_t^{\pi_0}|}{v_{\max}}. \tag{24}$$

This value lies in $[0, 1]$ and is zero when the action intervention leaves the target speed unchanged.

B.3 Dense Oracle

The dense oracle extends the observed solvable set beyond the evaluated methods. For each incident left unsolved, it runs full-support cross-entropy search (De Boer et al., 2005) under the stored replay conditions. The search minimizes the reach-avoid loss without a sparsity term and does not apply intervention reduction. Each of four independent restarts evaluates 48 candidates per iteration, retains an elite fraction of $1/8$, and runs for at most 40 iterations. The oracle receives eight times the corresponding per-incident HOPE simulator-transition budget. It adds an incident to the observed solvable set only when exact replay verifies a valid counterfactual policy.

B.4 Settings and Reproducibility

We select all settings on the development incidents and freeze them before evaluation. Table 6 lists the fixed search hyperparameters. Simulator transitions define the common compute budget because

Table 6: Frozen method settings used for the evaluation set. A hyphen denotes an action-space-specific parameter that is not used.

Setting	B1 continuous	B2 discrete	B3 hybrid
Validity-search iterations	8	-	-
Inner updates per iteration	3	-	-
Smoothing radius μ	0.35	-	0.15
Probe directions N_{dir}	2 orthogonal	-	2
Maximum support	-	12	12
Beam width	-	8	6
Leverage shortlist	-	16	12
Initial parameter grid	-	-	$\{0, 0.25, 0.5, 0.75, 1\}$
Stagnation patience / relative tolerance	4 / 0.01	4 / 0.01	4 / 0.01

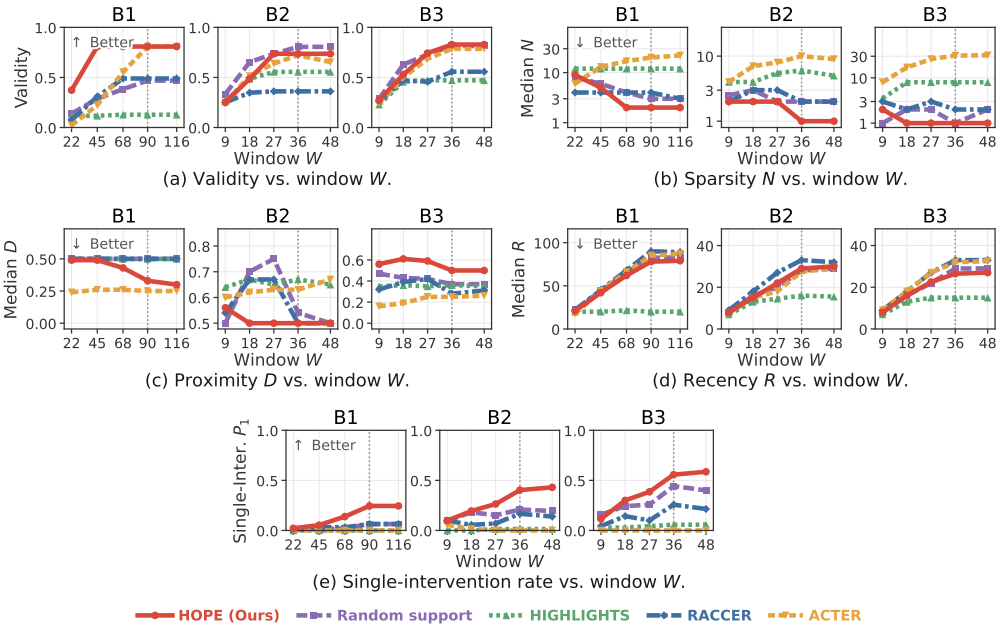


Figure 4: Sensitivity to the intervention window W. The panels show (a) validity, (b) median sparsity N, (c) median proximity D, (d) median recency R and (e) single-intervention rate P_1 of each method at five windows. Validity is the fraction of the solvable incidents that a method solves, and P_1 is the fraction that it solves with one action intervention; the medians are over the valid counterfactual policies returned by each method.

optimizer iterations have different rollout costs. On B2 and B3, each baseline and ablation receives the corresponding per-incident HOPE transition budget. On B1, each baseline receives a 44.86M aggregate budget, while the reported HOPE run uses 24.10M. Every budget includes candidate generation and verification rollouts.

C Experiments

This appendix reports additional sensitivity results for the evaluation in Section 5.

C.1 Sensitivity to the intervention window

The intervention window W determines how far before the first failure time t_f an action intervention may occur. We evaluate every method in Section 5.1 at five windows ($W = 22, 45, 68, 90$) per benchmark while fixing the other settings. Figure 4 reports the validity, median sparsity N, median proximity D, median recency R and single-intervention rate P_1 of each method at five windows.

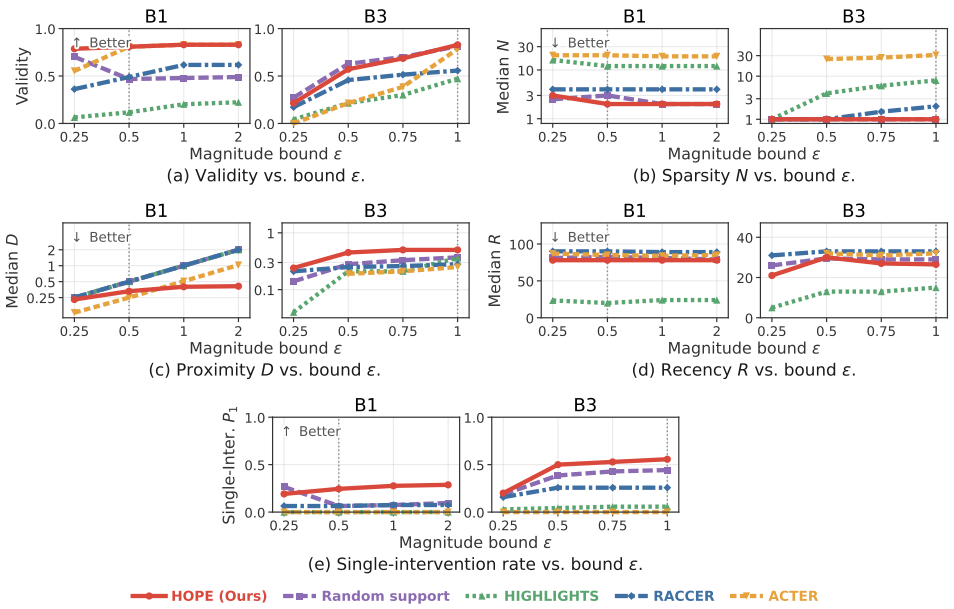


Figure 5: Sensitivity to the per-step magnitude bound ϵ on B1 and B3. The panels show (a) validity, (b) median sparsity N, (c) median proximity D, (d) median recency R, and (e) single-intervention rate P_1 of each method at four bounds. Validity is the fraction of the solvable incidents that a method solves, and P_1 is the fraction that it solves with one action intervention. Each median is taken over the valid counterfactual policies of a method. Dotted vertical lines mark the bounds of the main comparison. ACTER has no valid policy on B3 at $\epsilon = 0.25$, so its lines in (b), (c) and (d) start at $\epsilon = 0.5$.

Larger windows improve validity until method-specific saturation. On B1, HOPE increases from 0.37 at $W = 22$ to 0.81 at $W = 45$, and remains unchanged afterward; ACTER reaches the same level only for $W \geq 90$. On B2, Random Support is stronger at small and large windows, while comparable to HOPE at $W = 27$. On B3, Random Support leads at small windows, while HOPE becomes superior from $W = 27$ onward and reaches 0.83. These results show that a larger feasible window helps only until the method can already access the time steps it uses effectively.

HOPE generally achieves validity with sparser interventions. It has the lowest or comparable median N in 12 of 15 settings and the highest or comparable P_1 in 14 of 15. At larger windows, its median N decreases to at most two on B1 and B3, whereas ACTER grows to 22 and 31.5, respectively. At the main settings, HOPE achieves P_1 values of 24.5%, 40.3%, and 55.7% on B1–B3, confirming its strong tendency toward single-step interventions.

Magnitude and recency further explain this sparsity. On B1, the median D of HOPE falls from 0.49 to 0.30 as W grows, while its median N falls from 9 to 2. On B2, its median D is 0.50 for $W \geq 18$. On B3, however, its median D stays between 0.50 and 0.61 and is higher than those of the baselines, so its small support is achieved with larger individual changes. Median recency generally grows with the available window for Random Support, RACCER, ACTER, and HOPE, whereas HIGHLIGHTS remains near the failure: its median R stays around 20 on B1 and 15 on B2 and B3. For HOPE on B1, validity has already saturated at $W = 45$, but median R continues from 42 to 79. Hence, the main-setting recency is governed partly by the permitted window, not only by the lead time needed for validity.

C.2 Sensitivity to the magnitude bound

The bound ϵ limits the magnitude of each action intervention in Eq. 7. We evaluate every method at four bounds on B1 and B3 while holding the other settings fixed. The bounds are 0.25, 0.5, 1, and 2 on B1 and 0.25, 0.5, 0.75, and 1 on B3. B2 is excluded because an intervention substitutes one of three discrete meta-actions and has no continuous per-step bound to sweep. Figure 5 reports the

validity, median sparsity N, median proximity D, median recency R and single-intervention rate P_1 of each method at four bounds.

On B1, HOPE maintains stable validity while adapting intervention magnitude. Its validity remains within 0.79–0.83 across all bounds, outperforming ACTER at $\epsilon = 0.25$ and matching it at larger bounds. ACTER improves markedly from 0.55 to 0.83, whereas Random Support decreases from 0.70 to below 0.50 as the bound is relaxed, indicating solver-related nonmonotonic behavior rather than reduced feasibility. For HOPE, median N decreases from 3 to 2, median D increases moderately from 0.23 to 0.41, and median R remains near 78. In comparison, several baselines tend to exploit the full perturbation bound, while ACTER retains a much larger support of about 19–20 interventions. HOPE also achieves the highest P_1 for $\epsilon \geq 0.5$, increasing from 24.5% to 28.7%.

On B3, the intervention bound mainly determines feasibility. As ϵ increases from 0.25 to 1, validity rises substantially from 0.21 to 0.83 for HOPE, with similar trends for Random Support and ACTER. Random Support is slightly stronger at smaller bounds, with differences below 0.06, whereas HOPE reaches the highest validity at the main setting. Despite this sensitivity in feasibility, HOPE consistently maintains median $N = 1$ and the highest P_1, increasing from 20.0% to 55.7%. Its median D increases from 0.24 to 0.50 before saturating, while median R remains within 21–30. Thus, relaxing ϵ primarily affects whether a valid counterfactual can be found and the required intervention magnitude, while the sparse one-step structure of HOPE remains stable.

D Limitations and Future Work

This section summarizes the main limitations of the current study and outlines the future work needed to further validate the method. These limitations relate to the method's access assumptions, the scope of the evaluation, and the practical usefulness of the generated explanations.

Simulator access. The method assumes a simulator that can restore the state before an observed incident and replay its reference randomization. This enables controlled, closed-loop evaluation of each candidate, but it may be unavailable for physical platforms, or incidents represented only by logs. Learned world models and offline transition models could relax this requirement; however, counterfactual interventions may leave the support of the training data, allowing model errors to distort validity judgments (Janner et al., 2019; Yu et al., 2020). Future work should therefore combine model-based replay with uncertainty estimates, coverage checks, and conservative acceptance criteria, followed by independent high-fidelity or real-system verification when feasible.

Evaluation scope. The evaluation covers continuous, discrete, and hybrid action spaces across three benchmarks from two simulators, demonstrating applicability across action representations. The current study focuses on finite-horizon reach–avoid tasks with fixed policies and access to simulator states. More challenging settings, including longer horizons, partial observability, and multi-agent interactions, would increase temporal reasoning, search, and explanation complexity. Real-world deployment would also introduce sensing noise, actuator delays, model mismatch, and operational constraints. Future work should evaluate additional domains, policy types, failure modes, and deployment conditions, with scalability and robustness assessed alongside explanation quality.

User evaluation. The study evaluates explanation quality using validity, sparsity, proximity, recency, and the single-intervention rate. These metrics capture whether an explanation succeeds and the extent and timing of the required changes, but do not directly assess practical usefulness (Buçinca et al., 2020). Sparse or proximal explanations may still be difficult to interpret or implement, and a Pareto set requires users to resolve trade-offs among multiple solutions. Future studies should include safety engineers and operators and assess comprehension, actionability, decision time, calibrated trust, and consistency in policy selection. They should also compare full Pareto-set presentation with single-policy recommendations and evaluate the effects of domain-specific costs, operational constraints, and visualization methods on user decisions.