Safety-Aware Shared Autonomy via World-Model Constrained Planning

Zechen Shi[1] , **Jiawei Ye**[1] , **Zeyang Liu**[1] , **Xingyu Chen**[1,*] and **Xuguang Lan**[1,*]

[1]State Key Laboratory of Human-Machine Hybrid Augmented Intelligence, Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University

Abstract

Safety-aware shared autonomy aims to enable an autonomous agent to collaborate with a human operator, completing tasks under safety constraints while maximally preserving human intent. However, existing methods often underperform in long-horizon tasks that require balancing task performance under safety constraints with the degree of intervention—particularly when accurately predicting future unsafe events is critical. This limitation largely stems from their reliance on the current state alone to decide when to intervene and how to modify actions. We propose World Model Assisted Safety Planning (WASP), a model-predictive shared autonomy framework for explicit safety constraint satisfaction. We first formulate a safety-aware world model, and leverage its online predictive reasoning to decide when to intervene by jointly assessing prospective safety violations and degradation in task performance, thereby intervening only when necessary while enforcing safety constraints at decision time. Once intervention is triggered, WASP plans a short-horizon residual correction using world model rollouts, filters out unsafe candidates, and executes the least-deviating correction among the remaining high-return options in a receding-horizon loop. Experiments across diverse vision-based safety-critical domains show that WASP substantially reduces safety violations while preserving task performance and reducing unnecessary interventions over prior shared autonomy baselines.

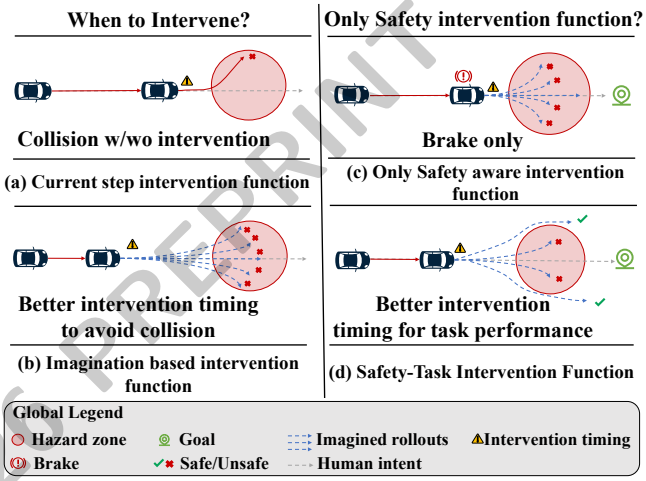

Figure 1: Overview of intervention timing in safety-aware shared autonomy. (a) Current observation-based intervention can be too late under delayed hazards. (b) Imagined rollouts anticipate multi-step risks and enable earlier intervention. (c) A safety-aware intervention function alone can become overly conservative (e.g., brake-only behavior), hindering task performance. (d) Adding a task-aware intervention function when a short correction is expected to improve return helps maintain task performance while preserving safety.

1 Introduction

Shared autonomy has emerged as a practical paradigm for human-machine collaboration, with demonstrated benefits in robotic teleoperation, drone control, and assisted driving [Javdani *et al.*, 2015; Reddy *et al.*, 2018; Li and others, 2023]. Its central objective is to improve task performance and reduce operator workload through timely assistance while preserving the human operator's intent and sense of control [Du *et al.*, 2020; Myers *et al.*, 2024].

In safety-aware and long-horizon tasks, insufficient assistance can lead to unsafe scenarios and poor task performance, while excessive assistance can hinder the human operator's intent [Tan *et al.*, 2022]. Prior work can be broadly categorized into two paradigms. One paradigm relies on an intervention function to decide when to assist the human operator [Oh *et al.*, 2025; McMahan *et al.*, 2024]. For example, IDA estimates the expected values of the agent and human actions to decide whether to execute the agent action instead of the human action [McMahan *et al.*, 2024]. Nevertheless, these methods typically make intervention decisions from the current observation without explicitly anticipating multi-step future risks, which can be insufficient for long-horizon and safety-aware tasks [Huang *et al.*, 2024; Xu *et al.*, 2021]. The other paradigm combines expert and human actions in a stepwise manner for task execution, which improves task performance through continuous

*Corresponding authors.

expert correction [Sun *et al.*, 2025; Yoneda *et al.*, 2023; Schaff and Walter, 2020]. DiffusionSA learns an expert action distribution and uses a diffusion model to correct human actions toward this distribution, while preserving the user's control authority [Yoneda *et al.*, 2023]. However, the assistance can become overly frequent, substantially reducing human autonomy. These limitations (Fig. 1(a,b)) raise a central open question in safety-aware shared autonomy: how to jointly decide intervention timing and design an action correction mechanism in long-horizon, safety-critical tasks.

A desirable assistance mechanism should anticipate multi-step future risks when deciding intervention timing and provide continuous, gradual corrections once intervention is needed. World model constrained planning provides a natural and principled approach to supporting this process [Wang *et al.*, 2025; Vasudevan *et al.*, 2025]. In particular, a world model enables decision making through imagined rollouts, allowing the assistant to reason about future outcomes along candidate trajectories.

Building on these insights, we propose World Model Assisted Safety Planning (WASP), a world model constrained planning framework for safety-aware human-machine collaboration. We first pretrain a safety-aware world model that consists of an RSSM-style latent dynamics model with a reward head and a collision cost head [Huang *et al.*, 2024].

WASP is structured into two stages, Model Predictive Intervention (MPI) and Model Predictive Residual Planning (MPRP). From the perspective of intervention timing, a safety-aware intervention function can avoid collisions, but it often leads to conservative braking behavior that degrades task performance (Fig. 1(c)). Therefore, we design both safety-aware and task-aware intervention functions to address this trade-off in MPI (Fig. 1(d)). The safety-aware intervention function uses imagined rollouts in the pretrained world model to estimate cumulative collision cost and triggers intervention when the cumulative cost is sufficiently high. The task-aware intervention function uses a limited number of imagined rollouts to estimate the long-horizon return gain of a short correction and intervenes only when the expected gain exceeds a threshold. When either function triggers, WASP intervenes and transitions to MPRP.

In MPRP, WASP samples residual action sequences and evaluates their imagined trajectories in the world model, retaining candidates that satisfy a predicted cumulative collision-cost constraint. If enough safe candidates remain, WASP optimizes task return while penalizing deviation from the operator command. Otherwise, it prioritizes reducing predicted collision cost.

We evaluate WASP on Safety Gymnasium [Ji *et al.*, 2023], comparing it against value-function-based intervention methods and stepwise action-correction methods. Across three tasks, WASP achieves strong task performance under safety constraints while requiring fewer and smaller interventions.

2 Related Work

Shared autonomy. Shared autonomy combines human inputs with agent assistance to improve task performance while preserving human intent [Abbink *et al.*, 2018]. Prior work includes intent inference, empowerment-based assistance, action correction, and intervention burden [Weiss and Gerdes, 2023]. Among these methods, action correction and intervention quantity have received particular attention. One line of work studies action correction by overriding or adjusting user actions to improve operator task performance [Schaff and Walter, 2020; Li and others, 2023]. Another line explicitly models intervention quantity, e.g., optimizing assistance under hard or soft constraints on the number of interventions to improve user experience and reduce manual tuning [Tan *et al.*, 2022; Broad *et al.*, 2019]. However, safety considerations are often treated as secondary in much of this literature.

Safety-aware shared autonomy. Safety-aware shared autonomy has mainly been explored through two paradigms: action correction and intervention-function-based assistance [Oh *et al.*, 2025; McMahan *et al.*, 2024]. Action-correction methods adjust human actions toward an expert action distribution. For example, diffusion-based approaches map human actions to samples from this distribution [Yoneda *et al.*, 2023; Sun *et al.*, 2025], while residual policy learning trains a model-free assistive policy to minimally adjust human actions under goal-agnostic constraints [Schaff and Walter, 2020; Li and others, 2023]. These methods can improve task performance, but their continuous corrections may lead to overly frequent assistance and reduced operator autonomy [Tan *et al.*, 2022; Broad *et al.*, 2019]. In contrast, intervention-function-based methods decide when the agent should override human actions using value or safety-margin signals, as in interventional assistance [McMahan *et al.*, 2024; Xue and others, 2023; Peng and others, 2023] and human-centered safety filters [Oh *et al.*, 2025]. However, these decisions are often based on current-state value or margin estimates, which may be insufficient when hazards are delayed and require multi-step anticipation. Moreover, some safety-filter approaches require expert priors to construct safety margins [Oh *et al.*, 2025].

Safety-constrained planning with latent world models. Safety-constrained planning with latent world models offers a natural way to anticipate delayed hazards under partial observability [As *et al.*, 2025; Huang *et al.*, 2024]. Latent dynamics models support imagined rollouts for evaluating future outcomes over candidate action sequences. Such rollouts have been widely used for imagination-based control and model-predictive planning [Wang *et al.*, 2025; Vasudevan *et al.*, 2025]. Recent safety-oriented model-based work augments this pipeline with cost or risk prediction and uses predicted cumulative cost to constrain planning or filter unsafe rollouts [Zwane *et al.*, 2023; Hogewind *et al.*, 2022; Jayant and Bhatnagar, 2022].

Relation to prior work. Prior shared-autonomy methods typically address either action correction or value-/margin-based intervention, while safety-constrained latent world models enable multi-step risk-aware planning under partial observability. WASP bridges these lines by using a pretrained safety-aware world model to jointly decide when to intervene and how to correct operator commands.

3 Problem Formulation

Constrained Markov decision process (CMDP). Safety-constrained long-horizon decision making is commonly modeled as a constrained Markov decision process (CMDP), defined as $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p, r, c, \gamma, \kappa)$ [Chow *et al.*, 2018; Dai *et al.*, 2023]. The state and action spaces are $\mathcal{S}$ and $\mathcal{A}$, respectively. The transition kernel $p(s' \mid s, a)$ gives the probability of reaching $s' \in \mathcal{S}$ after taking $a \in \mathcal{A}$ in $s \in \mathcal{S}$. The reward function $r(s, a)$ specifies the immediate reward, and the cost function $c(s, a)$ specifies the immediate safety cost. The discount factor is $\gamma \in (0, 1)$ and κ denotes the allowable discounted cost budget. The discounted return and cumulative cost of a policy $\pi(a \mid s)$ are

$$J_r(\pi) = E_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, r(s_t, a_t)\right], \qquad J_c(\pi) = E_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, c(s_t, a_t)\right]. \tag{1}$$

The CMDP objective is

$$\max_\pi\ J_r(\pi) \quad \text{s.t.} \quad J_c(\pi) \le \kappa, \tag{2}$$

Safety-aware shared autonomy. Based on the CMDP formulation, we formulate safety-aware shared autonomy by coupling a nominal human command with an assistive residual correction [Schaff and Walter, 2020]. At time t, the operator provides a nominal command $a_t^H \in \mathcal{A}$ and the assistant outputs a correction δa_t. The executed action is $a_t = a_t^H + \delta a_t$, followed by projection or clipping to the valid action bounds when necessary.

To make the shared-autonomy information structure explicit, we augment the state with the operator command,

$$\tilde{s}_t = (s_t, a_t^H), \tag{3}$$

and define the assistive policy as a residual policy conditioned on $\tilde{s}_t$:

$$\delta a_t \sim \pi(\delta a \mid \tilde{s}_t). \tag{4}$$

The environment evolves according to the same transition kernel $p(s_{t+1} \mid s_t, a_t)$ under the executed action a_t.

We implement minimal intervention by penalizing correction magnitude while enforcing the cumulative safety-cost constraint. This yields the objective

$$\max_\pi \quad E_\pi\left[\sum_{t=0}^{\infty} \gamma^t \Big(r(s_t, a_t) - \alpha \|\delta a_t\|^2\Big)\right] \quad \text{s.t.} \quad E_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, c(s_t, a_t)\right] \le \kappa, \tag{5}$$

where $\alpha > 0$ controls the penalty on correction magnitude and κ specifies the allowable discounted safety-cost budget.

4 Methods

Our method combines a pretrained safety-aware world model with online intervention gating and residual planning in a closed loop. Figure 2 provides an overview of the entire framework, from environment observation to the execution of the final corrected action, and the subsequent update of the state for the next time step.

4.1 Pretrained Safety-Aware World Model

To pre-evaluate future outcomes for intervention timing and action correction, we pretrain a *safety-aware world model* offline, together with reward and cost critics $\hat{V}_r$ and $\hat{V}_c$, and keep them fixed during online shared-autonomy control.

The world model is parameterized as a recurrent state-space model (RSSM), which summarizes history with a latent state consisting of a deterministic recurrent component h_t and a stochastic latent variable z_t. Given (h_t, z_t) and the executed action a_t, the model updates the recurrent state via $h_{t+1} = f_\theta(h_t, z_t, a_t)$ and predicts the next stochastic latent state through the prior distribution $p_\theta(z_{t+1} \mid h_{t+1})$. When an observation o_t is available, the stochastic latent variable is inferred through the posterior distribution $q_\theta(z_t \mid h_t, o_t)$. During imagination, the model rolls forward using only the prior dynamics.

From the latent state (h_t, z_t), the world model reconstructs the observation via a decoder $D_\theta(o_t \mid h_t, z_t)$ and predicts immediate reward and safety cost through decoders $R_\theta(r_t \mid h_t, z_t)$ and $C_\theta(c_t \mid h_t, z_t)$. We denote the predicted means by $\hat{r}_t$ and $\hat{c}_t$, where $\hat{c}_t \in [0, 1]$ represents the predicted probability of a safety-cost event. Two value critics, $\hat{V}_r^{\psi_r}(h_t, z_t)$ and $\hat{V}_c^{\psi_c}(h_t, z_t)$, provide bootstrapped estimates of long-horizon reward and cost beyond the finite imagination horizon.

The model is pretrained on replayed sequences (o_t, a_t, r_t, c_t) by minimizing

$$\begin{aligned}\mathcal{L}(\theta) = \sum_{t=1}^{T} \alpha_q\, \mathrm{KL}\big[q_\theta(z_t \mid h_t, o_t) \,\|\, \mathrm{sg}(p_\theta(z_t \mid h_t))\big] \\ + \alpha_p\, \mathrm{KL}\big[\mathrm{sg}(q_\theta(z_t \mid h_t, o_t)) \,\|\, p_\theta(z_t \mid h_t)\big] \\ - \beta_o \ln D_\theta(o_t \mid h_t, z_t) - \beta_r \ln R_\theta(r_t \mid h_t, z_t) \\ - \beta_c \ln C_\theta(c_t \mid h_t, z_t),\end{aligned} \tag{6}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator. The first two terms correspond to the representation regularization and future-prediction losses of the RSSM, encouraging consistency between the posterior and the learned prior. The remaining terms are negative log-likelihoods for observation reconstruction, reward prediction, and cost prediction, weighted by β_o, β_r, and β_c, respectively.

The reward and cost critics are trained separately using bootstrapped TD-λ return targets computed from latent rollouts.

4.2 Model-Predictive Intervention Function

Intervention is determined by two model-predictive intervention functions. The *safety-aware intervention function* has priority and gates intervention based on whether nominal operator behavior is predicted to exceed the safety-cost budget. When safety is not at risk, the *task-aware intervention function* gates intervention based on whether a small correction is expected to yield a meaningful long-horizon return gain. This mechanism avoids unnecessary assistance while preventing overly conservative behavior that preserves safety at the expense of task progress.

At each time step, the assistant decides whether to intervene on the operator command, prioritizing safety while

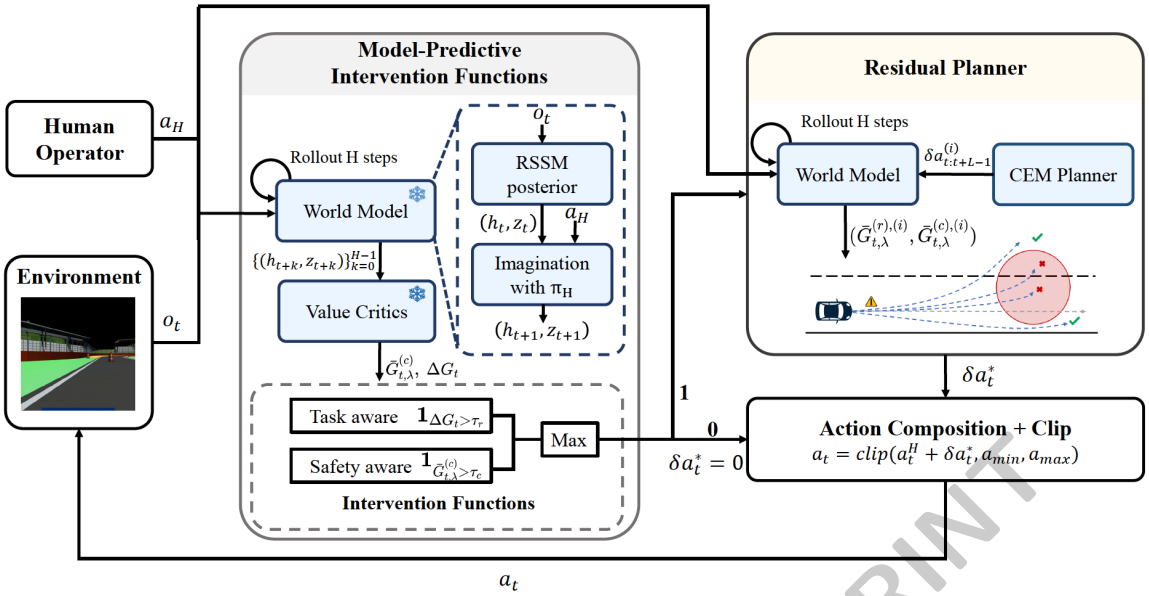

Figure 2: Closed-loop overview of the safety-aware shared autonomy framework. At each time step t, the environment provides observation o_t to both the human operator and the world model posterior, which infers the latent state (h_t, z_t). Using (h_t, z_t), a nominal operator model π_H, and candidate residuals, the world model performs imagined rollouts to compute model-predictive TD-λ estimates for reward and cost. Two intervention functions then decide whether to use the nominal operator command or apply a CEM-planned residual correction. The final action a_t is then executed in the environment, and the updated latent state estimate is used for the next step, closing the control loop. After pretraining, the world model and critics are fixed and not updated during online control.

remaining passive when intervention is unnecessary. This decision is made by simulating nominal operator behavior through imagined rollouts in the pretrained world model.

To roll out nominal behavior, we follow the standard shared-autonomy evaluation setup and use a simulated nominal operator policy π_H to generate future operator commands during imagination. At each imagined step, the future nominal command is sampled from π_H conditioned on the imagined latent state, $a_{t+k}^H \sim \pi_H(\cdot \mid h_{t+k}^{(m)}, z_{t+k}^{(m)})$. In our experiments, π_H is constructed from an RL-trained expert policy with injected imperfections, including action noise, lagged execution, limited skill, and delayed response. The same π_H is used consistently across imagined rollouts and real-environment evaluation for a given operator setting.

Starting from the current filtered latent state (h_t, z_t), we sample M imagined trajectories of horizon H under π_H and the world model dynamics, producing per-step predictions $\{\hat{r}_{t+k}^{(m)}\}_{k=0}^{H-1}$ and $\{\hat{c}_{t+k}^{(m)}\}_{k=0}^{H-1}$. Each rollout m is summarized by TD-λ returns with critic bootstrapping. For a generic signal $x \in \{r, c\}$ with per-step prediction $\hat{x}_{t+k}^{(m)}$ and critic $\hat{V}_x$, initialize

$$\hat{G}_{t+H,\lambda}^{(x),(m)} = \hat{V}_x(h_{t+H}^{(m)}, z_{t+H}^{(m)}), \tag{7}$$

and for $k = H-1, \ldots, 0$ compute

$$\hat{G}_{t+k,\lambda}^{(x),(m)} = \hat{x}_{t+k}^{(m)} + \gamma\Big((1-\lambda)\hat{V}_x(h_{t+k+1}^{(m)}, z_{t+k+1}^{(m)}) + \lambda\, \hat{G}_{t+k+1,\lambda}^{(x),(m)}\Big). \tag{8}$$

Here, $\lambda \in (0, 1]$ controls the rollout–bootstrap trade-off. Setting $x = r$ yields a model-predictive estimate of long-horizon task return, and setting $x = c$ yields a model-predictive estimate of discounted cumulative safety cost.

Safety-aware intervention function. Safety intervention takes priority and is driven by the CMDP constraint on discounted cumulative cost. Under nominal operator behavior, we compute the mean TD-λ cost return across M imagined rollouts:

$$\bar{G}_{t,\lambda}^{(c)} = \frac{1}{M}\sum_{m=1}^{M} \hat{G}_{t,\lambda}^{(c),(m)}. \tag{9}$$

Intervention is activated whenever $\bar{G}_{t,\lambda}^{(c)} > \tau_c$, where τ_c is the model-predictive cost threshold. When this condition holds, the task-aware gate is skipped.

Task-aware intervention function. When the safety-aware intervention function is inactive, task intervention is applied only if a short residual correction is expected to yield a meaningful improvement in long-horizon return.

Let $\bar{G}_{t,\lambda}^{(r)}$ denote the mean imagined TD-λ reward return under nominal operator behavior:

$$\bar{G}_{t,\lambda}^{(r)} = \frac{1}{M}\sum_{m=1}^{M} \hat{G}_{t,\lambda}^{(r),(m)}. \tag{10}$$

To test whether task intervention is warranted, we run a budgeted sampling probe over an L-step residual sequence $\delta a_{t:t+L-1}$. This probe is used only for gating and evaluates a finite set of candidate residual sequences without iterative distribution updates. Each candidate sequence is evaluated by composing it with nominal operator commands: the

first L actions are corrected, and the remaining actions follow the nominal commands generated by π_H in imagination. Let $\bar{G}^{(r)}_{t,\lambda}(\delta a_{t:t+L-1})$ and $\bar{G}^{(c)}_{t,\lambda}(\delta a_{t:t+L-1})$ denote the mean imagined TD-λ reward and cost returns under this composed rollout. We compute a candidate correction by solving

$$\begin{aligned}\delta a^{\text{corr}}_{t:t+L-1} \in \underset{\delta a_{t:t+L-1}}{\arg\max} \quad & \bar{G}^{(r)}_{t,\lambda}(\delta a_{t:t+L-1}) - \alpha \sum_{k=0}^{L-1} \|\delta a_{t+k}\|^2 \\ \text{s.t.} \quad & \bar{G}^{(c)}_{t,\lambda}(\delta a_{t:t+L-1}) \le \tau_c.\end{aligned} \tag{11}$$

Denote $\bar{G}^{\text{corr}}_t = \bar{G}^{(r)}_{t,\lambda}(\delta a^{\text{corr}}_{t:t+L-1})$ and define

$$\Delta G_t = \bar{G}^{\text{corr}}_t - \bar{G}^{(r)}_{t,\lambda}. \tag{12}$$

The task-aware intervention function activates only when $\Delta G_t > \tau_r$, where τ_r is the return-gain threshold.

4.3 Model-Predictive Residual Planning

When intervention is applied, residual corrections are planned online in a receding-horizon manner using imagined rollouts from the pretrained world model. At time t, we optimize an L-step residual sequence while evaluating task return and safety over a longer prediction horizon H. In practice, near the safety boundary, many sampled residual sequences lead to predicted safety-constraint violations, leaving too few safe candidates for reliable CEM distribution updates. We therefore handle planning in two cases: one where the sampled batch contains enough safe candidates, and one where safe candidates are scarce.

We plan over an L-step residual sequence $\delta a_{t:t+L-1}$ and evaluate it by composing the residuals with nominal operator commands:

$$a_{t+k} = \begin{cases} a^H_{t+k} + \delta a_{t+k}, & k < L, \\ a^H_{t+k}, & k \ge L, \end{cases} \tag{13}$$

where a^H_t is the current operator command and future nominal commands a^H_{t+k} for $k \ge 1$ are generated by π_H during imagination. In implementation, the composed actions are clipped to the valid action bounds before being passed to the world model.

For each sampled candidate $\delta a^{(i)}_{t:t+L-1}$, we roll out the world model for horizon H and compute its predicted discounted cumulative safety cost via the mean TD-λ cost return $\bar{G}^{(c),(i)}_{t,\lambda}$, averaged over M imagined rollouts. A candidate is treated as safe if its predicted cost satisfies the model-predictive cost threshold:

$$\mathcal{F}_s = \left\{ \delta a^{(i)}_{t:t+L-1} \;\middle|\; \bar{G}^{(c),(i)}_{t,\lambda} \le \tau_c \right\}. \tag{14}$$

We denote the number of safe candidates by $|\mathcal{F}_s|$. Planning is handled in two cases depending on whether $|\mathcal{F}_s|$ is at least a threshold N_f.

Case I ($|\mathcal{F}_s| \ge N_f$). When the batch contains enough safe candidates, we update the CEM sampling distribution using elites selected from $\mathcal{F}_s$. For each candidate $\delta a^{(i)}_{t:t+L-1}$, we estimate its mean imagined TD-λ reward return:

$$\bar{G}^{(r),(i)}_{t,\lambda} = \frac{1}{M} \sum_{m=1}^{M} \hat{G}^{(r),(i,m)}_{t,\lambda}, \tag{15}$$

where each rollout uses the composed action sequence in Eq. (13). We score safe candidates by

$$S^{(i)}_{\text{task}} = \bar{G}^{(r),(i)}_{t,\lambda} - \alpha \sum_{k=0}^{L-1} \|\delta a^{(i)}_{t+k}\|^2, \tag{16}$$

and update the sampling distribution using the highest-scoring candidates within $\mathcal{F}_s$ as elites.

Case II ($|\mathcal{F}_s| < N_f$). When safe candidates are scarce, $\mathcal{F}_s$ provides too few elites for a reliable CEM distribution update. We therefore perform a recovery update by selecting elites that minimize predicted safety-cost violation while still penalizing large corrections.

Using the predicted cost return $\bar{G}^{(c),(i)}_{t,\lambda}$ for each candidate i, we define

$$S^{(i)}_{\text{safe}} = \left[\bar{G}^{(c),(i)}_{t,\lambda} - \tau_c\right]_+ + \alpha \sum_{k=0}^{L-1} \|\delta a^{(i)}_{t+k}\|^2, \tag{17}$$

where $[x]_+ = \max(x, 0)$. We select the N_f candidates with the smallest $S^{(i)}_{\text{safe}}$ as elites to update the sampling distribution. In subsequent CEM iterations, if the sampled batch contains at least N_f safe candidates, planning switches back to Case I.

Given the selected residual sequence, the action executed in the environment is obtained by composing the nominal operator command with the first residual and enforcing action bounds:

$$a_t = \text{clip}\big(a^H_t + \delta a^*_t,\, a_{\min},\, a_{\max}\big), \tag{18}$$

where δa^*_t is the first element of the selected residual sequence: the highest-scoring feasible candidate under S_{task} if feasible candidates are available, and otherwise the candidate with the smallest S_{safe}. We then apply a_t in the real environment, observe the next transition, update the filtered latent state using the new observation, and repeat the intervention-function gating and residual planning at time $t{+}1$. This receding-horizon loop closes the control pipeline by continuously replanning corrections from updated latent state estimates.

5 Experiments

Our experiments evaluate WASP in terms of task performance, safety, autonomy, and intervention necessity. Unless otherwise specified, results are averaged over three random seeds, and main results are aggregated over the imperfect operator variants.

5.1 Experimental Setup

We evaluate WASP on three Safety-Gymnasium tasks [Ji *et al.*, 2023] with visual observations and explicit safety costs: *SafetyCarRace1-v0*, *SafetyCarFormulaOne1-v0*, and *SafetyPointGoal1-v0*. All tasks adopt distance-based reward shaping with an additional goal-completion bonus, provide front- and rear-view camera observations, and use a continuous two-dimensional control space for steering and longitudinal motion. These tasks go beyond obstacle avoidance by combining goal-directed navigation with multiple safety-cost sources, including hazards, collisions, and barrier contacts.

Task	Method	Return↑	Succ.↑	Cost↓	Viol.↓	Int.↓	Mag.↓
Race1	Human-only	4.97 ± 6.17	19.80 ± 2.40	68.35 ± 80.20	0.18 ± 0.17	0	0
	IDA	8.06 ± 7.51	27.80 ± 9.70	28.70 ± 25.95	0.09 ± 0.07	0.18 ± 0.15	$\mathbf{0.13 \pm 0.13}$
	Diffusion	11.56 ± 5.49	51.70 ± 5.30	16.80 ± 28.06	0.04 ± 0.06	1.00 ± 0.00	0.44 ± 0.25
	SafeRL-Res	10.50 ± 5.20	40.00 ± 6.00	25.50 ± 20.00	0.05 ± 0.04	1.00 ± 0.00	0.30 ± 0.20
	WASP (ours)	$\mathbf{13.74 \pm 4.15}$	$\mathbf{78.00 \pm 4.00}$	$\mathbf{0.00 \pm 0.00}$	$\mathbf{0.00 \pm 0.00}$	$\mathbf{0.15 \pm 0.08}$	0.20 ± 0.15
FormulaOne1	Human-only	0.34 ± 0.40	8.00 ± 6.90	126.80 ± 72.51	0.32 ± 0.20	0	0
	IDA	1.20 ± 3.34	8.90 ± 5.70	120.10 ± 60.09	0.32 ± 0.17	0.40 ± 0.19	0.53 ± 0.27
	Diffusion	$\mathbf{11.65 \pm 8.82}$	$\mathbf{66.70 \pm 25.20}$	10.30 ± 2.65	0.10 ± 0.03	1.00 ± 0.00	0.71 ± 0.21
	SafeRL-Res	9.50 ± 4.00	43.00 ± 7.00	20.00 ± 15.00	0.08 ± 0.05	1.00 ± 0.00	0.45 ± 0.20
	WASP (ours)	11.06 ± 10.59	65.00 ± 5.00	$\mathbf{1.20 \pm 1.98}$	$\mathbf{0.03 \pm 0.03}$	$\mathbf{0.20 \pm 0.15}$	$\mathbf{0.30 \pm 0.18}$
Goal1	Human-only	7.41 ± 5.17	10.20 ± 8.30	47.00 ± 37.61	0.05 ± 0.04	0	0
	IDA	13.94 ± 2.72	24.50 ± 9.70	43.60 ± 34.73	0.06 ± 0.04	$\mathbf{0.23 \pm 0.14}$	0.25 ± 0.16
	Diffusion	14.72 ± 6.99	29.90 ± 5.30	35.90 ± 25.54	0.08 ± 0.03	1.00 ± 0.00	0.65 ± 0.28
	SafeRL-Res	13.50 ± 5.20	28.00 ± 6.00	40.00 ± 30.00	0.07 ± 0.04	1.00 ± 0.00	0.35 ± 0.22
	WASP (ours)	$\mathbf{23.24 \pm 3.61}$	$\mathbf{56.70 \pm 6.80}$	$\mathbf{13.50 \pm 19.54}$	$\mathbf{0.01 \pm 0.02}$	0.25 ± 0.12	$\mathbf{0.25 \pm 0.15}$

Table 1: Main results aggregated over four imperfect operator variants, their severity settings, and three random seeds for each environment. We report task performance, safety, and autonomy metrics.

We use $H = 15$, $L = 5$, $N_{\text{probe}} = 300$ for the task-aware probe without iterative updates, and CEM with population $N = 300$, elites $N_e = 100$, and $I = 6$ iterations for MPRP.

We compare our approach against several baselines that represent key design axes in prior work. The Human-only baseline executes the operator's actions without assistance, serving as a reference for evaluating the effectiveness of shared autonomy. IDA (Intervention-Driven Assistance) leverages a learned Q-function to decide when to override the operator's action, substituting an expert action when necessary [McMahan *et al.*, 2024]. Diffusion continuously applies corrections toward expert-like behavior, while SafeRL-Res employs a model-free residual controller trained under safety constraints [Yoneda *et al.*, 2023; Schaff and Walter, 2020].

To model realistic operator imperfections, we simulate four variants: noisy operators introduce stochastic perturbations, laggy operators repeat the previous action with probability p_{lag}, limited-skill operators use a less skilled expert checkpoint trained with fewer interactions, and reaction-time operators apply delayed commands. These variants capture noise, temporal inconsistency, limited skill, and sluggish response, respectively, and assess the ability of predictive intervention to prevent violations caused by sequences of suboptimal actions. Because several variants are stochastic, such as noisy actions and probabilistic lag, the nominal operator model used in planning is an approximate predictor rather than an exact copy of the realized operator behavior.

5.2 Main Results

Table 1 presents a comparison across three tasks: `Race1`, `FormulaOne1`, and `Goal1`, evaluated under the protocol described in Sec. 5.1. The table reports results aggregated over four imperfect operator variants: noisy, laggy, limited-skill, and reaction-time operators. We compare against several baselines: Human-only, IDA, Diffusion action correction, and SafeRL-Res. These baselines represent key design approaches such as value-based gating (IDA), stepwise correction (Diffusion), and model-free residual assistance (SafeRL-Res).

We evaluate the methods on three aspects: task performance, safety, and autonomy. Task performance is measured by episodic return and success rate, while safety is assessed by cumulative cost and violation rate. Autonomy is evaluated by the assistant's deviation from the nominal operator command [Oh *et al.*, 2025]. Specifically, we define the induced residual as $\delta a_t = a_t - a_t^H$, where the intervention rate is the fraction of steps where $\|\delta a_t\| > 0$, and the correction magnitude is the empirical mean of $\|a_t - a_t^H\|$.

Across all three environments, WASP demonstrates a strong balance between safety and task performance. The effect is most pronounced in `Race1` and `FormulaOne1`, where violations often emerge after a sequence of small, suboptimal actions. Compared with IDA, WASP substantially reduces both cost and violation rate while requiring fewer or smaller deviations from the operator command. This result highlights the benefit of predictive rollouts, which allow WASP to intervene more selectively before safety violations occur.

Compared with Diffusion action correction and SafeRL-Res, WASP achieves comparable or superior task performance while substantially reducing intervention rate and correction magnitude. This indicates that short-horizon residual planning avoids the continual micro-corrections often seen in stepwise correction methods while still maintaining safety. Diffusion often requires large corrections because it pushes the current action toward expert-like behavior, which can disregard the operator's current intent.

5.3 Counterfactual Evaluation of Intervention Necessity

We assess whether interventions occur before risky events rather than being applied as persistent corrections. To do this, we perform counterfactual rollouts in the real environment at timesteps where intervention occurs. At each such timestep,

Variant	Return↑	Succ.↑	Cost↓	Viol.↓	Int.↓	Mag.↓
WASP (full)	11.06 ± 10.59	65.00 ± 5.00	1.20 ± 1.98	0.03 ± 0.03	0.20 ± 0.15	0.30 ± 0.18
Safety-only gate (remove task gate)	9.50 ± 8.00	18.00 ± 15.00	1.13 ± 2.50	0.05 ± 0.06	0.15 ± 0.20	0.40 ± 0.30
Task-only gate (remove safety gate)	12.35 ± 7.50	67.00 ± 3.00	37.50 ± 5.60	0.28 ± 0.08	0.12 ± 0.10	0.25 ± 0.20
No Case II recovery (Case I only)	8.50 ± 5.20	45.00 ± 6.00	6.80 ± 3.20	0.12 ± 0.10	0.25 ± 0.21	0.35 ± 0.40
Rollout-disabled (only use single-step V)	7.80 ± 6.00	42.00 ± 8.00	18.40 ± 4.70	0.19 ± 0.12	0.45 ± 0.35	0.27 ± 0.15

Table 2: Ablations on FormulaOne1 averaged over four operator variants.

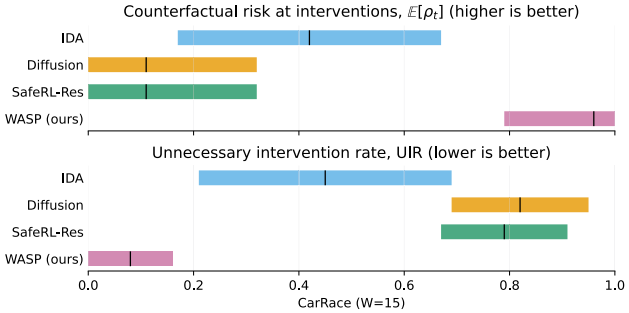

Figure 3: Counterfactual evaluation of intervention necessity on Race1 ($W = 15$).

we record the environment state, including relevant randomization factors and the operator's internal state. From this recorded state, we run two branches for W steps: at the first step, the assisted branch executes the method's action, while the counterfactual branch executes the nominal operator command. Both branches then continue with operator-only control for the remaining $W - 1$ steps.

We measure counterfactual risk using the mean cumulative cost over W steps:

$$\rho_t = \frac{1}{W} \sum_{k=0}^{W-1} c_{t+k}^{\mathrm{cf}}. \tag{19}$$

We also compute the local benefit of intervention by comparing cost and reward differences between the assisted and counterfactual branches:

$$\begin{aligned} \Delta C_t &= \sum_{k=0}^{W-1} c_{t+k}^{\mathrm{cf}} - \sum_{k=0}^{W-1} c_{t+k}^{\mathrm{asst}}, \\ \Delta R_t &= \sum_{k=0}^{W-1} r_{t+k}^{\mathrm{asst}} - \sum_{k=0}^{W-1} r_{t+k}^{\mathrm{cf}}. \end{aligned} \tag{20}$$

An intervention is counted as unnecessary if it does not reduce cost and its return improvement is below a threshold τ_R. We define the unnecessary intervention rate as

$$\mathrm{UIR} = E_{t \in \mathcal{T}_{\mathrm{int}}} \left[\mathbf{1} \left\{ \Delta C_t \leq 0 \ \wedge \ \Delta R_t \leq \tau_R \right\} \right], \tag{21}$$

where $\mathcal{T}_{\mathrm{int}}$ denotes the set of intervention timesteps.

Figure 3 shows results on `Race1` with $W = 15$. WASP selectively intervenes in high-risk states, resulting in a low unnecessary intervention rate. In contrast, Diffusion and SafeRL-Res apply corrections even in low-risk states, leading to higher unnecessary intervention rates. IDA focuses more on high-risk states than stepwise correction methods but still intervenes in some low-risk cases. These results indicate that WASP intervenes more selectively, reducing unnecessary corrections while maintaining safety.

5.4 Ablation Study

Table 2 summarizes the ablation results, showing the impact of different components on task performance, safety, and autonomy. Removing the safety-aware gate and relying only on the task-aware gate causes a significant increase in violations, as the system may fail to intervene when the operator takes unsafe actions that still appear beneficial for task progress. This emphasizes the importance of multi-step safety screening for reducing safety violations. Conversely, using only the safety-aware gate largely preserves safety but leads to overly conservative interventions, which can hinder task progress and reduce performance.

Ablating Case II recovery degrades performance near the constraint boundary, where too few feasible candidates are available for stable CEM updates. This leads to higher safety costs and less stable corrections, especially under temporally inconsistent operator behavior, highlighting the importance of recovery updates for maintaining stability under constraints.

Disabling imagined rollouts and relying on a single-step value estimate increases intervention rates while worsening both cumulative cost and task return. This suggests that single-step estimates can trigger unnecessary interventions while missing delayed risks, leading to poorer intervention timing. These results highlight the importance of multi-step rollouts for accurate intervention timing and effective safety-aware residual planning.

6 Conclusion

We introduced World Model Assisted Safety Planning (WASP), a safety-aware shared autonomy framework that uses a pretrained world model for predictive intervention and residual planning. WASP anticipates future safety risks and applies low-deviation corrections only when needed. Experiments on vision-based benchmarks show that WASP substantially reduces safety violations and unnecessary interventions while maintaining competitive task performance. Although evaluated on driving and navigation tasks, WASP only assumes a nominal operator command, residual correction, and cumulative safety-cost constraint, making it applicable to broader shared-autonomy settings.

Acknowledgements

This work was supported in part by the NSFC under grant No. U23A20339, No. 62125305, No. 62573339, No. 62503380, Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China under grant No. JYB2025XDXM210, and Natural Science Foundation of Shaanxi Province under Grant 2025SYSSYSZD-083, the National Key R&D Program of China under Grant No. 2024YFB4707600, State Grid Corporation of China Science and Technology Projects 52060025005B-439-ZN, and Shaanxi Provincial Science Fund for Distinguished Young Scholars under Grant 2025JC-JCQN-077.

Contribution Statement

Zechen Shi, Jiawei Ye, and Zeyang Liu contributed equally to this work.

References

[Abbink *et al.*, 2018] David A. Abbink, Tom Carlson, Mark Mulder, Joost C. F. de Winter, Farzad Aminravan, Tricia L. Gibo, and Erwin R. Boer. A topology of shared control systems—finding common ground in diversity. *IEEE Transactions on Human-Machine Systems*, 48(5), 2018.

[As *et al.*, 2025] Yarden As, Bhavya Sukhija, Lenart Treven, Carmelo Sferrazza, Stelian Coros, and Andreas Krause. Actsafe: Active exploration with safety constraints for reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2025. Poster.

[Broad *et al.*, 2019] Alexander Broad, Todd Murphey, and Brenna Argall. Highly parallelized data-driven mpc for minimal intervention shared control. In *Robotics: Science and Systems (RSS)*, 2019.

[Chow *et al.*, 2018] Yinlam Chow, Ofir Nachum, Emilio Duéñez-Guzmán, and Mohammad Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 8092–8101, 2018.

[Dai *et al.*, 2023] Juntao Dai, Jiaming Ji, Long Yang, Qian Zheng, and Gang Pan. Augmented proximal policy optimization for safe reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2023.

[Du *et al.*, 2020] Yuqing Du, Stas Tiomkin, Emre Kiciman, Daniel Polani, Pieter Abbeel, and Anca D. Dragan. AvE: Assistance via empowerment. In *Advances in Neural Information Processing Systems*, 2020.

[Hogewind *et al.*, 2022] Yannick Hogewind, Thiago D. Simao, Tal Kachman, and Nils Jansen. Safe reinforcement learning from pixels using a stochastic latent representation. arXiv preprint arXiv:2210.01801, 2022.

[Huang *et al.*, 2024] Weidong Huang, Jiaming Ji, Borong Zhang, Chunhe Xia, and Yaodong Yang. Safedreamer: Safe reinforcement learning with world models. In *International Conference on Learning Representations (ICLR)*, 2024.

[Javdani *et al.*, 2015] Shervin Javdani, Siddhartha S. Srinivasa, and J. Andrew Bagnell. Shared autonomy via hindsight optimization. In *Robotics: Science and Systems (RSS)*, 2015.

[Jayant and Bhatnagar, 2022] Ashish K. Jayant and Shalabh Bhatnagar. Model-based safe deep reinforcement learning via a constrained proximal policy optimization algorithm. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[Ji *et al.*, 2023] 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. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[Li and others, 2023] Zhaoyuan Li et al. Human-ai shared control via policy dissection. In *Advances in Neural Information Processing Systems*, 2023.

[McMahan *et al.*, 2024] Brandon J. McMahan, Zhenghao Peng, Bolei Zhou, and Jonathan C. Kao. Shared autonomy with IDA: Interventional diffusion assistance. In *Advances in Neural Information Processing Systems*, 2024.

[Myers *et al.*, 2024] Vivek Myers, Evan Ellis, Sergey Levine, Benjamin Eysenbach, and Anca D. Dragan. Learning to assist humans without inferring rewards. In *Advances in Neural Information Processing Systems*, 2024.

[Oh *et al.*, 2025] Donggeon David Oh, Justin Lidard, Haimin Hu, Himani Sinhmar, Elle Lazarski, Deepak Gopinath, Emily S. Sumner, Jonathan A. DeCastro, Guy Rosman, Naomi Ehrich Leonard, and Jaime Fernández Fisac. Safety with agency: Human-centered safety filter with application to AI-assisted motorsports. In *Proceedings of Robotics: Science and Systems*, 2025.

[Peng and others, 2023] Xue Bin Peng et al. Learning from active human involvement through proxy value propagation. In *Advances in Neural Information Processing Systems*, 2023.

[Reddy *et al.*, 2018] Siddharth Reddy, Anca D. Dragan, and Sergey Levine. Shared autonomy via deep reinforcement learning. In *Proceedings of Robotics: Science and Systems*, 2018.

[Schaff and Walter, 2020] Charles Schaff and Matthew R. Walter. Residual policy learning for shared autonomy. In *Proceedings of Robotics: Science and Systems*, 2020.

[Sun *et al.*, 2025] Luzhe Sun, Jingtian Ji, Xiangshan Tan, and Matthew R. Walter. Flashback: Consistency model-accelerated shared autonomy. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2025. CoRL 2025 Poster.

[Tan *et al.*, 2022] Weihao Tan, David Koleczek, Siddhant Pradhan, Nicholas Perello, Vivek Chettiar, Vishal Rohra, Aaslesha Rajaram, Soundararajan Srinivasan, H. M. Sajjad Hossain, and Yash Chandak. On optimizing interventions in shared autonomy. In *Proceedings of the Thirty-Sixth*

AAAI Conference on Artificial Intelligence, pages 5341–5349, 2022.

[Vasudevan *et al.*, 2025] Arun Balajee Vasudevan, Neehar Peri, Jeff Schneider, and Deva Ramanan. Planning with adaptive world models for autonomous driving. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14938–14945, Atlanta, GA, USA, May 2025. IEEE.

[Wang *et al.*, 2025] Hang Wang, Xin Ye, Feng Tao, Chenbin Pan, Abhirup Mallik, Burhaneddin Yaman, Liu Ren, and Junshan Zhang. Adawm: Adaptive world model based planning for autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2025.

[Weiss and Gerdes, 2023] E. Weiss and J. Christian Gerdes. Follow my lead: Designing an ADAS that shares decision making and control with the driver. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023.

[Xu *et al.*, 2021] Yunkun Xu, Zhenyu Liu, Guifang Duan, Jiangcheng Zhu, Xiaolong Bai, and Jianrong Tan. Look before you leap: Safe model-based reinforcement learning with human intervention. arXiv:2111.05819, 2021. CoRL 2021 accepted.

[Xue and others, 2023] Yifan Xue et al. Guarded policy optimization with imperfect online demonstrations. In *International Conference on Learning Representations*, 2023.

[Yoneda *et al.*, 2023] Takuma Yoneda, Luzhe Sun, Ge Yang, and Matthew R. Walter. To the noise and back: Diffusion for shared autonomy. In *Proceedings of Robotics: Science and Systems*, 2023.

[Zwane *et al.*, 2023] Sicelukwanda Zwane, Denis Hadjivelichkov, Yicheng Luo, Yasemin Bekiroglu, Dimitrios Kanoulas, and Marc Peter Deisenroth. Safe trajectory sampling in model-based reinforcement learning. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, 2023.