Hybrid Advantage Estimation with Unified Critic for VLM Agentic Reinforcement Learning

Wenxuan Zhang, Yuhui Wang*, Donggang Jia, Xiaoqian Shen, Jian Ding, Ivan Viola, Jürgen Schmidhuber, and Mohamed Elhoseiny*

King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
{wenxuan.zhang,yuhui.wang,donggang.jia,xiaoqian.shen,jian.ding}@kaust.edu.sa
{ivan.viola,juergen.schmidhuber,mohamed.elhoseiny}@kaust.edu.sa

Abstract. Large Vision-Language Models (VLMs) now act as agents in interactive environments, where success requires coherent reasoning and decision-making across turns. Although end-to-end training in agentic environments can improve such multi-turn decision-making abilities, current methods mainly rely on either token-wise optimization over concatenated token trajectories or turn-wise optimization with uniform within-turn credit. In this work, we establish theoretical formulations for the two levels of optimization and derive a hybrid advantage that serves both objectives. Furthermore, with an appropriate choice of discount factor and learning target, we prove that a unified critic model can estimate values for both turn-wise and token-wise. As such, we propose HyGAE, an actor-critic framework that jointly optimizes token- and turn-level objectives with the hybrid advantage and unified critic. We conduct extensive evaluations of HyGAE across five multi-turn decision-making environments, where it achieves an average success rate of 91% and a significant improvement of 10% over other methods. Furthermore, we provide an in-depth analysis showing that the exact analytic form of the hybrid advantage and return is crucial for optimization. Project Page: https://wx-zhang.github.io/hygae-web/.

1 Introduction

With the rapid evolution of Large Vision-Language Models (VLMs) [7,8,20,57], they are no longer limited to single-turn visual question answering. VLMs are now expected to serve as agents that perceive, reason, and act in the real world [1, 14,38]. In this setting, effective agents need to understand complex environments, make sequential decisions, receive feedback, and adapt their future actions over multiple turns.

A primary bottleneck of current VLMs lies in their nearsightedness: they often reason based solely on the most recent observation, neglecting the critical context of past actions and feedback. As illustrated in Fig. 5, when faced with identical observations and feedback indicating that the previous action failed,

* Corresponding authors: Yuhui Wang and Mohamed Elhoseiny.

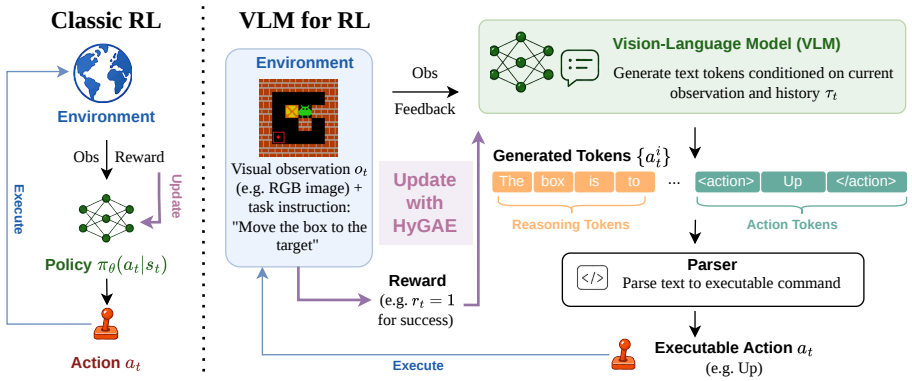

Fig. 1: Multi-turn decision-making tasks for VLMs. We update the VLM policy using hybrid advantages from the token and turn levels, while learning the value with a unified value model.

the model redundantly repeats its previous failed output rather than synthesizing its experience. In this way, multi-turn decision-making is reduced to a series of disjointed single-turn generations, where task success becomes highly opportunistic rather than the result of systematic strategic planning.

This limitation mainly arises from the mismatch between conventional single-turn training paradigms and the requirements of multi-turn interaction. In multi-turn interaction, models can receive intermediate information that guides them to refine or maintain their internal beliefs. Several recent works have noticed this issue and attempted to design algorithms that directly support multi-turn agentic reinforcement learning [10,44,45,52]. A central discussion in this context concerns two levels of reinforcement learning modeling: (1) token-wise RL, which treats each token as a single action, and (2) turn-wise RL, which treats the turn as a whole as one action. Turn-wise optimization better leverages intermediate feedback for intra-turn differentiation, while token-wise training provides fine-grained inner-turn guidance for language generation [9]. More recently, several works have adapted classical hierarchical reinforcement learning ideas [29–31] to combine token-level and turn-level optimization in multi-turn agentic RL [44,56]. To better understand these RL modeling choices, we need to answer an essential question: what is the inherent difference between the two levels of RL modeling?

To answer this question, we first provide a POMDP-based theoretical formulation of token-wise and turn-wise RL, showing that the two formulations differ only in advantage estimation and that their surrogates therefore have the same gradient. Based on this observation, we propose hybrid advantage estimation, which optimizes the two levels of RL simultaneously through a simple linear combination and improves both inner- and intra-turn credit assignment. Furthermore, we show that, with an appropriate choice of discount factor and learning target, we can learn a unified critic model for both turn-wise and token-wise value estimates. We instantiate the resulting hybrid advantage and value

estimation in an actor-critic framework, denoted as HyGAE, which benefits from both language-generation training and turn-wise intermediate feedback during training, as shown in Fig. 1. We further prove that hybrid advantage estimation and unified value training provide a better balance between bias and variance.

We evaluate HyGAE on five multi-turn decision-making benchmarks and compare our algorithm with various baselines. The results show that HyGAE achieves SOTA performance on all tasks, with an average success rate of 0.91. We further provide ablations and qualitative analyses showing that the precise analytic forms of the hybrid advantage and mixed return are crucial for stable optimization. Our contributions are as follows:

- We provide a formal formulation of token-wise and turn-wise RL in multi-turn decision-making tasks and reveal the intrinsic similarities between them.
- We design hybrid advantage estimation with unified value model training to optimize the policy for better inner- and intra-turn credit assignment. This formulation is computationally efficient and provides a better bias-variance trade-off.
- We achieve SOTA performance on five multi-turn decision-making tasks, reaching an average success rate of 0.91.

2 Related Work

Multi-turn Decision-Making Tasks for VLMs. Early approaches for solving vision-related multi-turn decision-making tasks mainly fall into two categories: (1) leveraging pretrained visual representations, *e.g.*, CLIP [27], for decision-making, often trained with external reward models [12, 24], and (2) directly employing pretrained VLMs as agents for sequential reasoning and action generation [54].

Recent works increasingly adopt reinforcement learning to train VLMs for these tasks. Some works propose to obtain high-quality training data from external sources [11, 17, 40, 49], self-refining data through interaction or feedback [28, 53, 55], improving reward modeling [3, 21, 51], and integrating tool usage during rollout [22, 23]. In addition, [2] proposes a comprehensive pipeline spanning pre-training to post-training. Despite these advances, most methods still rely on standard RL algorithms such as PPO [34] or GRPO [15].

Multi-turn Reinforcement Learning for VLMs. An important line of work focuses on the design of training algorithms for foundation models in multi-turn decision-making tasks. Some studies analyze the underlying mechanisms of foundation models in such scenarios [10, 18], while others propose improved optimization strategies, such as advantage-weighted RL with stochasticity-aware estimators [4]. A central topic in this line of research is how to model trajectories at the turn level or token level [9]. On one hand, some approaches introduce complex and computationally expensive hierarchical models to explicitly capture both levels, such as [56]. On the other hand, many methods operate at a single level and rely on heuristic designs for credit assignment. [52] introduces a coefficient to balance

the contribution of reasoning (thinking) tokens and action tokens, and follow-up works further refine control over reasoning processes [46, 47]. Similarly, [44] applies different discount factors to token-level and turn-level rewards, and [13] analyzes credit assignment in GRPO at both the turn and sequence levels. In contrast to these approaches, through theoretical analysis, we find that the two levels of RL are intrinsically related. We therefore propose hybrid advantage estimation, which combines token-wise and turn-wise advantages with a unified value function and achieves a better balance between bias and variance.

3 Preliminaries

POMDP Following [44], we define the multi-turn interaction of VLMs as a Partially Observable Markov Decision Process (POMDP) $(\mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{R}, \Omega, \boldsymbol{\gamma})$. Here, $\mathcal{S}$, $\mathcal{A}$, $\mathcal{O}$, and $\mathcal{R}$ denote the state, action, observation, and reward spaces, respectively; $\mathcal{T}$ represents the state transition dynamics; Ω represents the observation function; and $\boldsymbol{\gamma} \in [0, 1)$ is the discount factor. At each *decision step* t, the agent receives an observation $\boldsymbol{o}_t \in \mathcal{O} \sim \Omega(\cdot \mid \boldsymbol{s}_t)$ conditioned on the system state $\boldsymbol{s}_t \in \mathcal{S}$ and generates an action $\boldsymbol{a}_t \in \mathcal{A}$ conditioned on the history $\tau_t^i := \big(\boldsymbol{o}_0, \boldsymbol{a}_0, \boldsymbol{r}_0, \boldsymbol{o}_1, \boldsymbol{a}_1, \boldsymbol{r}_1, \ldots, \boldsymbol{o}_t, \big(a_t^0, a_t^1, \ldots, a_t^i\big)\big)$. The environment then transitions from state $\boldsymbol{s}_t$ to a new state $\boldsymbol{s}_{t+1} \sim \mathcal{T}(\cdot \mid \boldsymbol{s}_t, \boldsymbol{a}_t)$, produces a reward $\boldsymbol{r}_t \sim \mathcal{R}(\cdot \mid \boldsymbol{s}_t, \boldsymbol{a}_t)$, and sends the agent a new observation $\boldsymbol{o}_{t+1}$.

RL with Actor-Critic Framework The objective of reinforcement learning is to find an optimal policy π_θ parameterized by θ that maximizes the expected return, defined as the sum of the discounted future rewards:

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1} \gamma^t \boldsymbol{r}_t\right] \tag{1}$$

To optimize this objective, it is common to employ the surrogate function:

$$L_{\text{actor}}(\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1} \log \pi_\theta(\boldsymbol{a}_t|\boldsymbol{\tau}_t) \boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t)\right] \tag{2}$$

where $\boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t) = \boldsymbol{Q}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t) - \boldsymbol{V}^{\pi_\theta}(\boldsymbol{\tau}_t)$ is the advantage value function defined as

$$\boldsymbol{Q}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t) := \mathbb{E}\left[\sum_{t'=t}^{T-t} \boldsymbol{\gamma}^{t'-t} \boldsymbol{r}_{t'} \mid \boldsymbol{\tau}_t, \boldsymbol{a}_t\right], V^{\pi_\theta}(\boldsymbol{\tau}_t) := \mathbb{E}_{a \sim \pi_\theta(\cdot|\boldsymbol{\tau}_t)}\left[\boldsymbol{Q}^{\pi_\theta}(\boldsymbol{\tau}_t, a)\right]$$

In practice, the advantage value $\boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t)$ is usually estimated by the Generalized Advantage Estimation (GAE) [33]:

$$\widehat{\boldsymbol{A}}_t = \sum_{t'=0}^{T-t} (\gamma\lambda)^{t'} \delta_{t+t'}, \text{ where } \delta_t := \boldsymbol{r}_t + \gamma \widehat{\boldsymbol{V}}_\psi(\boldsymbol{\tau_{t+1}}) - \widehat{\boldsymbol{V}}_\psi(\boldsymbol{\tau_t}), \tag{3}$$

where δ_t is the temporal difference error, and $\lambda \in [0, 1]$ is a secondary discount factor that controls the bias–variance trade-off. $\widehat{\boldsymbol{V}}_\psi$ is a parameterized value function that estimates the state value.

We employ the actor-critic framework for the value function estimation and policy update.

$\widehat{\boldsymbol{V}}_\psi$ is also called the critic, which is updated together with the actor using the TD error as the target.

$$L_{\text{critic}}(\psi) = \mathbb{E}_t\big[(\widehat{\boldsymbol{V}}_\psi(\boldsymbol{s}_t) - \widehat{\boldsymbol{G}}_t)^2\big], \text{ where } \widehat{\boldsymbol{G}}_t = \widehat{\boldsymbol{A}}_t + \widehat{\boldsymbol{V}}_\psi(\boldsymbol{\tau_t}). \tag{4}$$

4 Method

As discussed above, few prior works have formally formulated the RL process in VLM agentic training. We first derive turn-wise and token-wise objectives under a unified view, revealing their shared surrogate structure and motivating our hybrid actor-critic framework with HyGAE advantage estimation and unified value model training. Finally, we provide a theoretical analysis explaining why the hybrid approach is both effective and efficient.

Notation: Throughout this paper, we use boldface notation to represent turn-wise variables, such as $\boldsymbol{a}_t$, and standard unbolded font with a superscript local token index i to represent token-wise variables, such as a_t^i. Full definitions, theorems, and proofs are provided in Appendix A.

4.1 Turn- and Token-wise Formulations

In multi-turn decision-making tasks, at each turn $t \in [0, T-1]$, the VLM policy generates a token sequence $\boldsymbol{a}_t := (a_t^1, a_t^2, \ldots, a_t^{I_t})$, where a_t^i is the i-th token generated at turn t, and I_t is the length of the token sequence for turn t. The generated tokens encode intermediate reasoning and executable actions. The environment then executes the action, providing a reward $\boldsymbol{r_t}$ and a feedback token sequence $\boldsymbol{o_{t+1}}$ for the next turn.

This generation–interaction loop can be viewed at either the turn or token scale, leading to distinct POMDP formulations and corresponding RL training strategies. Below, we detail these two perspectives.

Token-wise POMDP. A popular perspective treats the generation of each token as an individual action [23, 26, 44]. At each decision step (t-th turn and i-th token generation), the VLM policy π_θ takes the context τ_t^i as input and samples a token $a_t^i \sim \pi_\theta(\cdot\,|\tau_t^i)$. The token-wise reward r_t^i assigned to each token is defined as

$$r_t^i = \begin{cases} \boldsymbol{r}_t, & i = I_t \\ 0, & \text{otherwise} \end{cases}.$$

This indicates that if the token is at the end of the turn, its reward is the environmental reward $\boldsymbol{r}_t$; otherwise, it is 0 for intermediate tokens.

Based on this formulation, Eq. (1) can be rewritten by re-indexing across both turn and token indices: $J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1}\sum_{i=0}^{I_t-1}\gamma^{i+I_{<t}} r_t^i\right]$, where $I_{<t} = \sum_{k=0}^{t-1} I_k$ is the total number of tokens generated before the t-th turn, and γ is the discount factor. From this objective, we can derive the token-wise surrogate function:

$$L^{\text{token}}(\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i=0}^{I_t-1}\log \pi_\theta\left(a_t^i|\tau_t^i\right) A^{\pi_\theta}\left(\tau_t^i, a_t^i\right)\right] \tag{5}$$

Here $A^{\pi_\theta}(\tau_t^i, a_t^i)$ represents the token-wise advantage function, which estimates the advantage value for each generated token.

Turn-wise POMDP. This perspective treats the generation of the entire token sequence $\boldsymbol{a}_t$ within a single turn as one unified macro-action [52]. We define the turn-wise POMDP as follows: at decision step t (the t-th turn), the VLM policy takes the context $\boldsymbol{\tau}_t$ as input and generates $\boldsymbol{a}_t \sim \pi_\theta(\cdot|\boldsymbol{\tau}_t)$. The context $\boldsymbol{\tau}_t$ is constructed from the observations, actions, and rewards returned by the environment:

$$\boldsymbol{\tau}_t := (\boldsymbol{o}_0, \boldsymbol{a}_0, \boldsymbol{r}_0, \boldsymbol{o}_1, \boldsymbol{a}_1, \boldsymbol{r}_1, \ldots, \boldsymbol{o}_t) \tag{6}$$

The reward in the turn-wise formulation is simply the reward $\boldsymbol{r_t}$ returned by the environment. The objective follows Eq. (1), and we can use the surrogate function from Eq. (2) to optimize it.

4.2 Hybrid Advantage Function and Unified Value Function

Note that the turn action, $\boldsymbol{a}_t := (a_t^i)_{i=0}^{I_t}$, is generated autoregressively by the model. The log-probability of generating the turn-wise action, $\log \pi_\theta(\boldsymbol{a}_t|\boldsymbol{\tau}_t)$, can be decomposed into the sum of the log-probabilities of generating each token:

$$\log \pi_\theta(\boldsymbol{a}_t|\boldsymbol{\tau}_t) = \sum_{i=0}^{I_t-1}\log \pi_\theta\left(a_t^i|\tau_t^i\right), \tag{7}$$

Substituting Equation (7) into the surrogate function in Eq. (2), we obtain the following form of the turn-wise surrogate:

$$L^{\text{turn}}(\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i=0}^{I_t-1}\log \pi_\theta\left(a_t^i|\tau_t^i\right) \boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t)\right] \tag{8}$$

where $\boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t)$ represents the turn-wise advantage function, which estimates the advantage value for the turn-wise action.

The turn-wise and token-wise surrogates in Eq. (8) and Eq. (5) have the same form except for the advantage value: the turn-wise surrogate employs the turn-wise advantage $\boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t)$, whereas the token-wise surrogate uses the token-wise advantage $A^{\pi_\theta}(\tau_t^i, a_t^i)$.

The two estimators focus on different properties during training. As shown in Fig. 2, the turn-wise advantage estimation remains constant across the turn.

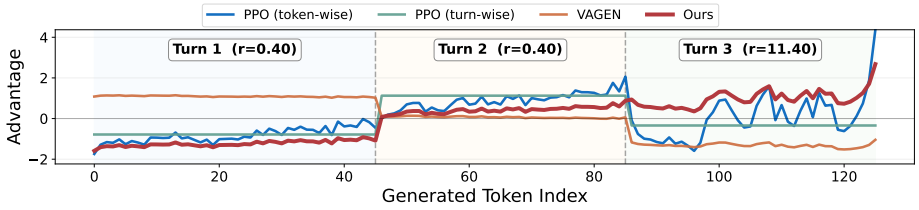

Fig. 2: Estimated advantage scales for different advantage estimators over all tokens in the trajectory.

This preserves the turn-level reward signal and avoids excessive discounting through intermediate reasoning tokens. Conversely, the token-wise advantage estimation provides fine-grained advantage values for each individual token, preserving within-turn differentiation.

Existing methods typically employ either estimator independently or use one signal to guide the other when both advantages are considered. The equivalence above provides a theoretical basis for optimizing with both advantage signals without introducing hierarchical modeling; instead, the two signals can be directly hybridized. This leads to a *hybrid surrogate function* using a hybrid advantage function:

$$L^{\text{hybrid}}(\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1}\sum_{i=0}^{I_t-1} \log \pi_\theta\left(a_t^i|\tau_t^i\right)\left(\alpha \boldsymbol{A}^{\pi_\theta}(\boldsymbol{\tau}_t, \boldsymbol{a}_t) + (1-\alpha)A^{\pi_\theta}\left(\tau_t^i, a_t^i\right)\right)\right] \quad (9)$$

Formally, the following theorem demonstrates that the hybrid surrogate yields the same policy gradient as the original objective $J(\theta)$.

Theorem 1. *Let the discount factors of the turn-wise and token-wise formulations satisfy $\boldsymbol{\gamma}_t = \gamma^{I_t}$, where I_t is the length of turn t. Then, for any $\alpha \in [0,1]$ in Eq. (9), we have*

$$\nabla J(\theta) = \nabla L^{\text{hybrid}}(\theta) = \nabla L^{\text{turn}}(\theta) = \nabla L^{\text{token}}(\theta).$$

As we will show in Sec. 4.4, the hybrid advantage can also balance the bias–variance tradeoff. Therefore, utilizing the hybrid advantage is theoretically sound and effective compared to relying on a single surrogate for advantage estimation.

Naively, the turn-wise and token-wise advantage computations involve different value models, $\widehat{\boldsymbol{V}}_{\boldsymbol{\psi}}$ and $\widehat{V}_{\psi}$. One would need to train two separate value models within the actor-critic framework, which introduces significant computational overhead and memory usage. However, we demonstrate that by appropriately setting the discount factors $\boldsymbol{\gamma}$ and γ, a single, unified value function can be employed for the estimation of both turn-wise and token-wise advantages.

Theorem 2. *Let the discount factors of the turn-wise and token-wise formulations satisfy $\boldsymbol{\gamma}_t = \gamma^{I_t}$, where I_t is the length of turn t. Then the corresponding value functions are consistent at the last token of the trajectory, that is,*

$$\boldsymbol{V}^{\pi}(\boldsymbol{\tau}_t) = V^{\pi}(\tau_t^{I_t}) \quad \textit{for any turn } t. \quad (10)$$

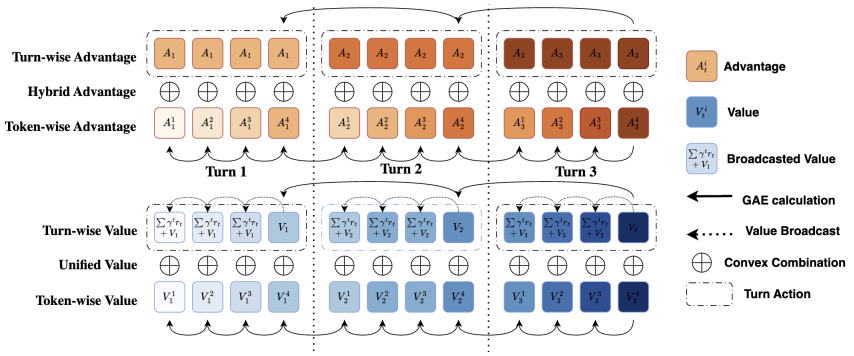

Fig. 3: Actor-critic framework with HyGAE: we combine the token-wise advantage and turn-wise advantage to update the policy model, while broadcasting the turn value with discounted rewards to learn a unified value model.

Theorem 2 shows that, given the appropriate discount factors, the turn-wise value $\boldsymbol{V}^{\pi}(\boldsymbol{\tau}_t)$ is exactly equal to the token-wise value of the final token $V^{\pi}(\tau_t^{I_t})$. This equivalence implies that we can use exactly one shared value model, V_ψ, which provides turn-wise values at the end of the turn and token-wise values at intermediate token positions.

4.3 HyGAE Estimation

In practice, following Eq. (3), we can estimate the hybrid advantage value by combining the turn-wise and token-wise advantage estimates. As shown in Fig. 3, we first calculate the turn-wise advantage estimate by

$$\widehat{\boldsymbol{A}}_t = \sum_{t'=0}^{T-t} (\boldsymbol{\gamma}\lambda)^{t'} \boldsymbol{\delta}_{t'}, \quad \text{where } \boldsymbol{\delta}_{t'} = \boldsymbol{r}_{t'} + \boldsymbol{\gamma}\widehat{\boldsymbol{V}}_{\boldsymbol{\psi}}(\boldsymbol{\tau}_{t'+1}) - \widehat{\boldsymbol{V}}_{\boldsymbol{\psi}}(\boldsymbol{\tau}_{t'}) \tag{11}$$

We then calculate the token-wise advantage value $A^{\pi_\theta}(\tau_t^i, a_t^i)$ by

$$\widehat{A}_t^i = \sum_{i'=0}^{I-i-I_{<t}} (\gamma\lambda)^{i'} \delta^{(i'+i+I_{<t})}, \text{ where } \delta^{(i')} = r^{(i')} + \gamma\widehat{V}_\psi\left(\tau^{(i'+1)}\right) - \widehat{V}_\psi\left(\tau^{(i')}\right) \tag{12}$$

For simplicity, we use the superscript (i') to denote the global token index in the whole trajectory. Here, $\widehat{V}_\psi$ is the value model in the actor-critic framework.

Finally, we combine the turn-wise and token-wise advantage values to obtain the hybrid advantage value $\widehat{\mathsf{A}}_t^i$ by Eq. (13).

$$\widehat{\mathsf{A}}_t^i = \alpha\widehat{\boldsymbol{A}}_t + (1-\alpha)\widehat{A}_t^i, \quad \alpha \in [0,1] \tag{13}$$

where α is a hyperparameter that weights the turn-wise signal against the token-wise signal. As shown in Fig. 2, the hybrid advantage estimation provides fine-grained advantage values for each individual token, while being less discounted by the intermediate tokens.

Algorithm 1 HyGAE

Input: Policy π_θ; Value Model $\widehat{V}_\psi$.
for each iteration $k = 0$ to K **do**
  Interact with the environment and collect trajectory $\boldsymbol{\tau}$ (see Eq. (6)), which contains T turns.
  for each turn $t = T-1$ to 0 **do**
    Obtain turn-wise advantage $\widehat{\boldsymbol{A}}_t$ and return $\widehat{\boldsymbol{G}}_t$ via turn-wise GAE in Eq. (11) and Eq. (16).
    Obtain token-wise advantage $\widehat{A}_t^i$ and return $\widehat{G}_t^i$ via token-wise GAE in Eq. (12) and Eq. (15).
    for each token $i = I_t - 1$ to 0 **do** ▷ HyGAE advantage estimation
      Calculate the hybrid advantage $\widehat{\mathsf{A}}_t^i$ via Eq. (13).
      Construct the training target for the turn-wise value model by broadcasting the turn-wise return with discounted rewards to each token position via Eq. (16).
      Calculate the hybrid return $\widehat{\mathsf{G}}_t^i$ via Eq. (17).
    end for
  end for
  Update the policy π_θ and value model $\widehat{V}_\psi$ via the policy loss in Eq. (14) and the TD loss with the target in Eq. (17).
end for
Return policy π_θ and value model $\widehat{V}_\psi$.

We employ the clipping technique in Proximal Policy Optimization (PPO) [34] to improve training stability. The empirical surrogate loss function derived from Eq. (9) can be written as

$$\widehat{L}^{\text{hybrid}}(\theta) = \widehat{\mathbb{E}}\left[\sum_{t=0}^{T-1}\sum_{i=0}^{I_t-1} \min\left(\rho_t^i(\theta)\widehat{\mathsf{A}}_t^i, \text{clip}(\rho_t^i(\theta), 1-\epsilon, 1+\epsilon)\widehat{\mathsf{A}}_t^i\right)\right] \tag{14}$$

where $\rho_t^i(\theta) := \pi_\theta(a_t^i|\tau_t^i)/\pi_{\theta'}(a_t^i|\tau_t^i)$ is the likelihood ratio.

A critical practical consideration is the calculation of the target return used to update this shared value model $\widehat{V}_\psi$. While returns from either the turn or token level could be used, they exhibit significantly different bias-variance profiles. Computing the token-wise return following Eq. (3) is straightforward,

$$\widehat{G}_t^i = \widehat{A}_t^i + \widehat{V}_\psi(\tau_t^i) \tag{15}$$

Conversely, the turn-wise GAE only provides a single return at the end of the turn. To compute the hybrid return in the middle of the turn, we must broadcast the turn-wise return back to each intermediate token position by summing the discounted rewards from the current token index to the end of the turn:

$$\widehat{\boldsymbol{G}}_t^i = \widehat{\boldsymbol{A}}_t + \sum_{k=i+1}^{I_t} \gamma^{k-i} r_k + \widehat{V}_\psi(\boldsymbol{\tau}_t) \tag{16}$$

Therefore, as illustrated at the bottom of Fig. 3, we calculate a *hybrid return* to combine the turn-wise and token-wise returns:

$$\widehat{\mathsf{G}}_t^i = \alpha \widehat{\boldsymbol{G}}_t^i + (1-\alpha)\widehat{G}_t^i, \tag{17}$$

We demonstrate in Sec. 5.3 that this specific formulation of the hybrid return is crucial for optimizing the performance of the value model.

Regarding environmental consistency, the token-level rewards should formally aggregate perfectly into the turn-level rewards. In practice, token-level rewards often consist of very small KL-divergence penalties. We found it empirically beneficial *not* to aggregate these micro-rewards into the final turn-level reward $R^{(i)}$. We show both of these design choices in Sec. 5.3.

The full procedure for calculating these mixed signals is summarized in Algorithm 1. Once the estimated advantage A_t^i and return G_t^i are computed, the policy and value networks are updated via the standard PPO loss.

4.4 Bias and Variance Analysis

We compare the bias and variance of the turn-wise and token-wise advantage estimators. We define the bias of the token-wise advantage estimator $\widehat{A}_t^i$ as $\text{bias}\left(\widehat{A}_t^i\right) = \left\| \mathbb{E}\left[\widehat{A}_t^i \mid \tau_t^i, a_t^i\right] - A^\pi(\tau_t^i, a_t^i) \right\|_\infty$, where $\|\cdot\|_\infty$ denotes the infinity norm. Similarly, we can define $\text{bias}(\widehat{\boldsymbol{A}}_t)$ for the turn-wise estimator.

Theorem 3 (Bias Comparison). *Let* $e_{\max} := \|\widehat{V} - V^\pi\|_\infty$, *and* $I = \min_t I_t$. *Then*

$$\text{bias}\Big(\widehat{\boldsymbol{A}}_t\Big) \le \underbrace{\left(1 + \frac{\gamma^I(1-\lambda)}{1-\gamma^I\lambda}\right)}_{b_{\text{turn}}} e_{\max},\ \ \text{bias}\Big(\widehat{A}_t^i\Big) = \underbrace{\left(1 + \frac{\gamma(1-\lambda)}{1-\gamma\lambda}\right)}_{b_{\text{token}}} e_{\max},\ \ b_{\text{turn}} \le b_{\text{token}}.$$

Theorem 4 (Variance Comparison). *Let* $\sigma_{\max} := \sup \text{var}(\delta^{(i)})$, *and* $I = \max_t I_t$. *Then*

$$\text{var}\Big(\widehat{\boldsymbol{A}}_t\Big) \le \underbrace{\left(\frac{1-\gamma^I}{(1-\gamma)(1-\gamma^I\lambda)}\right)^2}_{v_{\text{turn}}} \sigma_{\max},\ \ \text{var}\Big(\widehat{A}_t^i\Big) \le \underbrace{\left(\frac{1}{1-\gamma\lambda}\right)^2}_{v_{\text{token}}} \sigma_{\max},\ \ v_{\text{turn}} > v_{\text{token}}.$$

The above theorems show that the turn-wise estimator tends to have smaller bias but larger variance compared to the token-wise estimator. These results reveal a bias–variance tradeoff between the turn-wise and token-wise estimators, and our hybrid approach balances the two by combining them. All proofs are provided in Appendix A.

5 Experiments

5.1 Setup

Environment: We evaluate HyGAE across five decision-making benchmarks, following the protocols established in [10, 44]. These environments require Vision-Language Models (VLMs) to process visual observations and generate natural language actions to interact with the environment.

- **Sokoban** [32]: Evaluates spatial reasoning and long-term planning. The agent must push boxes onto target positions on a grid using actions: `up, down, left, right`.
- **FrozenLake(F.Lake)** [43]: Tests path planning under constraints. The agent navigates from a start to a goal while avoiding holes, using the same discrete action space as Sokoban.
- **Navigation** [19, 50]: Assesses visual grounding and instruction following. The agent navigates to a target within 1.5 meters using an 8-action space related to move, turn, and look. We test both direct instructions and complex scenarios involving common-sense reasoning.
- **Primitive Skill** [16, 25, 39, 41]: Focuses on robotic manipulation. The agent performs tabletop tasks (place, stack, draw, align) using continuous parameterized actions: `pick(x,y,z), place(x,y,z)`, and `push(x1,y1,z1,x2,y2,z2)`.
- **VIRL** [48]: A real-world navigation task using street-view imagery. The agent follows step-by-step instructions to reach a destination using 10 distinct actions including forward, turn, and stop.

Detailed environment statistics, prompt templates, and evaluation metrics are provided in Section 2.1 of the Supplementary Material.

Comparison: We compare HyGAE against three categories of models:

1. **Proprietary Models:** Leading closed-source models including GPT-5 [38], Gemini 2.5 Pro [14], and Claude Sonnet 4.5 [1].
2. **Frozen Open-Source VLMs:** Top-performing small-scale models.
3. **Reinforcement Learning Methods:** We compare the finetuning performance of the Qwen2.5-VL-3B backbone using standard reinforcement learning methods, including Token-PPO [26, 34], Turn-PPO [9, 34], GRPO [15], RL4VLM [52], and VAGEN-Base [44].

The Qwen2.5-VL-3B backbone was chosen for RL training due to its robust architecture and extensive community support, facilitating reproducible results. Training hyperparameters for all baselines are detailed in Section 2.2 of the Supplementary Material.

Table 1: Comparison of HyGAE with state-of-the-art methods and models on five multi-turn decision-making benchmarks.

Model/Method	**Sokoban**	**F.Lake**	**Navigation**		**Primitive Skill**				**VIRL**		**Avg.**
			Base	CS	Place	Stack	Draw	Align	ID	OOD	
Proprietary Models											
GPT-5 [38]	0.7	0.77	0.75	0.81	**1**	0.63	0	**1**	0.06	0.56	0.63
Gemini 2.5 Pro [14]	0.58	0.78	0.63	0.63	0.63	0.63	0	0.75	0.16	0.39	0.52
Claude Sonnet 4.5 [1]	0.31	0.8	0.67	0.67	0.63	0.5	0	**1**	0.09	0.44	0.51
Frozen Open-Source Models											
Qwen2.5-VL-3B [6]	0.18	0.13	0.34	0.28	0	0	0	0	0	0	0.09
Gemma-3-4B [42]	0.07	0.05	0.03	0.02	0	0	0	0	0	0	0.02
VLM-R1-3B [35]	0.16	0.10	0.34	0.15	0	0	0	0	0.03	0	0.08
Cosmos-R1-7B [2]	0.14	0.23	0.01	0	0	0	0	0	0	0	0.04
Qwen3-VL-4B [5]	0.27	0.08	0.01	0	0	0	0	0	0	0	0.04
Reinforcement Learning Methods											
Token-PPO [26]	0.59	0.72	**0.90**	0.84	**1**	**1**	0.99	0.95	0.16	0.94	0.81
Turn-PPO [9]	0.38	0.70	0.78	0.84	0	0	0	**1**	0.09	0.94	0.47
GRPO [15]	0.20	0.57	0.88	0.81	0	0	0	**1**	0.03	0.28	0.38
RL4VLM [52]	0.20	0.35	0.86	0.78	0	0.95	0	0.98	0.38	**1**	0.55
VAGEN-Base [44]	0.61	0.71	0.78	0.80	**1**	0.88	0.88	0.88	0.06	0.83	0.74
HyGAE(Ours)	**0.83**	**0.80**	**0.90**	**0.86**	**1**	**1**	**1**	0.99	**0.72**	**1**	**0.91**

5.2 Results

As illustrated in Tab. 1, proprietary models consistently outperform open-source alternatives by a wide margin. Specifically, while all proprietary models achieve an average score exceeding 0.5 across the five benchmarks, their open-source counterparts struggle to surpass the 0.1 threshold. Notably, even with the evolution of general-purpose small models, progress remains inconsistent. For instance, while transitioning from Qwen2.5-VL-3B to Qwen3-VL-4B yields a localized improvement on Sokoban (rising from 0.18 to 0.27), this gain is overshadowed by a systemic decline across other benchmarks. Consequently, the overall average score drops from 0.09 to 0.04. This disparity implies the critical necessity of multi-turn RL training to instill robust sequential decision-making capabilities in VLMs, bridging the gap toward real-world applications.

Among reinforcement learning methods, token-wise methods such as Token-PPO achieve an average score of 0.81, and VAGEN-Base reaches 0.74, consistently outperforming their turn-wise counterparts. In contrast, turn-level methodologies such as Turn-PPO and RL4VLM exhibit diminished efficacy, yielding average scores of only 0.47 and 0.55, respectively. This decline is primarily attributed to the omission of fine-grained token-level guidance, which is indispensable for stable autoregressive generation. Furthermore, while GRPO has emerged as a cornerstone for single-turn reasoning, its inability to leverage intermediate rewards at the turn level leads to an average performance of 0.38 across the five benchmarks.

HyGAE achieves SOTA performance across all benchmarks. Specifically, our method reaches 0.83 on Sokoban and 0.80 on Frozen Lake, while maintaining a high average score of 0.88 on Navigation. Remarkably, HyGAE achieves a near-perfect average score of 1.0 on Primitive Skill and a robust 0.86 on the VIRL task. By synergistically leveraging the **mixed advantage** for precise credit as-

signment and the **mixed return** for stabilized value training, our method effectively bridges the gap between high-level reasoning and low-level token generation. Notably, HyGAE achieves a remarkable average score of 0.91, significantly outperforming both token-wise and turn-wise baselines and demonstrating its robustness in complex, multi-turn interaction scenarios.

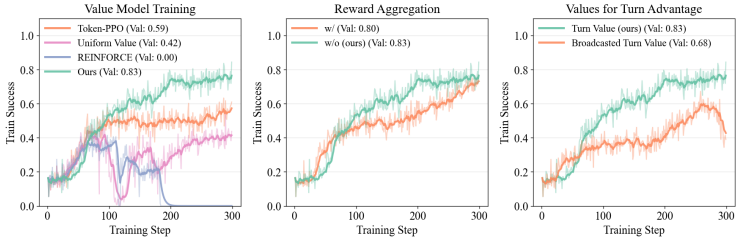

Fig. 4: Key design factors for the performance of our algorithm: value model training, advantage aggregation, and value selection for constructing turn-level advantages.

5.3 Key to the Success of HyGAE

During development, we identify several critical design choices that contribute to the stability and performance of HyGAE. The diagnostic studies in this subsection are conducted on the Sokoban task.

The return must be precise. The formulation of the mixed return in Eq. (17) is critical. We observe that alternative choices, such as broadcasting the turn-level return uniformly across all tokens within a turn or omitting the value model entirely (i.e., using REINFORCE), lead to training collapse, as shown in Fig. 4. This observation aligns with known challenges in PPO for language models, where training the value model is difficult [36,37]. The formulation of the hybrid return in Eq. (17) provides the precise target and necessary variance reduction to stabilize the value function V_ψ throughout training.

Aggregation of small token-wise rewards is optional. As discussed in Sec. 4.3, strictly speaking, additional token-level rewards, *e.g.*, KL-divergence penalties, should be aggregated into the turn reward when computing the turn-level advantages. This ensures that the system provides exactly the same reward from the token-wise and turn-wise RL perspectives. However, since these rewards are typically small, we show in Fig. 4 that aggregating them at the turn level or not leads to similar results.

Which value should be used for the broadcasted turn-wise advantage? As discussed in Sec. 4.3, computing the broadcasted turn-wise advantage for intermediate tokens requires selecting an appropriate value estimate. We evaluate two

strategies: 1) using the token-specific value, which provides fine-grained, state-dependent guidance but relies on potentially noisy broadcasted intermediate value; or 2) using the turn-level value as a constant baseline for all tokens within a turn. Our empirical results in Fig. 4 show that the theoretically grounded choice of using the turn-level value is sufficiently robust.

Numerical analysis on bias-variance. We further examine the bias-variance trade-off on Sokoban by comparing the stability of PPO and HyGAE across runs. PPO obtains a success rate of 0.5797 ± 0.2105, whereas HyGAE reaches 0.8222 ± 0.0711. This result indicates that HyGAE not only improves the average performance but also substantially reduces variance, suggesting a more stable optimization process.

5.4 Ablation Study on α

Table 2: Ablation study on the mixing coefficient α.

α	0	0.25	0.5	0.75	1
Sokoban	0.59	0.64	**0.83**	0.78	0.38
FrozenLake	0.72	0.76	**0.80**	0.73	0.70
Primitive Skill (avg.)	0.98	0.73	**1.00**	0.73	0.25

We ablate the mixing coefficient α used in both the hybrid advantage in Eq. (13) and the hybrid return in Eq. (17), where α controls the balance between turn-level and token-level signals. In this setting, $\alpha = 0$ corresponds to Token PPO, $\alpha = 1$ reduces to Turn PPO with broadcasted values, and $\alpha = 0.5$ is our default configuration. As shown in Tab. 2, the default choice $\alpha = 0.5$ achieves the best performance on Sokoban, FrozenLake, and Primitive Skill. The intermediate mixtures also generally outperform the pure turn-wise setting, suggesting that combining token-level and turn-level credit assignment provides a more reliable training signal.

5.5 Qualitative Analysis

We compare the frozen Qwen2.5-VL and HyGAE model to investigate their decision-making behaviors in Fig. 5. To improve the readability, we modify the output template and highlight the observation, reasoning, prediction, and answer here. The frozen model falls into repetitive and non-adaptive action loops. In both environments, the frozen model consistently outputs the same action across multiple turns, even when the action has no effect on the environment. In contrast, our model demonstrates an active response to environmental feedback: after an initial "Right" action yields no progress, it immediately incorporates this failure and pivots its strategy, correctly executing a "Left" move in the subsequent turn to reach the box. In the VIRL environment, our model explicitly follows the instruction and reaches the destination.

Fig. 5: Qualitative examples on Sokoban (up) and VIRL (bottom) with readable template. The frozen model repeatedly predicts ineffective actions or repeats a static plan, whereas HyGAE updates its actions after receiving feedback and uses previous actions and observations to maintain multi-turn consistency.

6 Conclusion

In this paper, we study the problem of training Large Vision-Language Models for multi-turn decision-making tasks. In these tasks, at each turn, the model must reason over intermediate visual states and make decisions based on its internal reasoning. We propose HyGAE, an agentic reinforcement learning algorithm employing the actor-critic framework. Specifically, we introduce a hybrid advantage that combines a lightly discounted turn-level advantage for decision making with a token-level advantage that better supports general language generation, while using a unified critic for efficient training. We theoretically prove the validity of this hybrid formulation and show that the resulting framework achieves both low variance and low bias. In experiments with 3B-scale VLMs across five decision-making benchmarks, our method consistently outperforms prior approaches. We further provide quantitative and qualitative analyses to enable a deeper understanding of our algorithm.

Acknowledgements

The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST) - Center of Excellence for Generative AI, under award number 5940.

References

1. Anthropic: Claude sonnet 4.5 system card. https://www.anthropic.com/claude-sonnet-4-5-system-card (2025), accessed: 1 July 2026
2. Azzolini, A., Bai, J., Brandon, H., Cao, J., Chattopadhyay, P., Chen, H., Chu, J., Cui, Y., Diamond, J., Ding, Y., et al.: Cosmos-reason1: From physical common sense to embodied reasoning. arXiv preprint arXiv:2503.15558 (2025)
3. Bai, H., Zhou, Y., Li, L.E., Levine, S., Kumar, A.: Digi-q: Learning vlm q-value functions for training device-control agents. In: The Thirteenth International Conference on Learning Representations (2025)
4. Bai, H., Zhou, Y., Pan, J., Cemri, M., Suhr, A., Levine, S., Kumar, A.: Digirl: Training in-the-wild device-control agents with autonomous reinforcement learning. Advances in Neural Information Processing Systems **37**, 12461–12495 (2024)
5. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report (2025)
6. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report (2025)
7. Chen, J., Guo, H., Yi, K., Li, B., Elhoseiny, M.: Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18030–18040 (2022)
8. Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., Elhoseiny, M.: Minigpt-v2: Large language model as a unified interface for vision-language multi-task learning. arXiv preprint arXiv:2310.09478 (2023)
9. Chen, K., Cusumano-Towner, M., Huval, B., Petrenko, A., Hamburger, J., Koltun, V., Krähenbühl, P.: Reinforcement learning for long-horizon interactive llm agents. arXiv preprint arXiv:2502.01600 (2025)
10. Chu, T., Zhai, Y., Yang, J., Tong, S., Xie, S., Schuurmans, D., Le, Q.V., Levine, S., Ma, Y.: Sft memorizes, rl generalizes: A comparative study of foundation model post-training. In: Proceedings of the 42nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 267, pp. 10818–10838. PMLR (2025)
11. Faldor, M., Zhang, J., Cully, A., Clune, J.: Omni-epic: Open-endedness via models of human notions of interestingness with environments programmed in code. In: The Thirteenth International Conference on Learning Representations (2025)

12. Fan, L., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.A., Zhu, Y., Anandkumar, A.: Minedojo: Building open-ended embodied agents with internet-scale knowledge. Advances in Neural Information Processing Systems **35**, 18343–18362 (2022)
13. Feng, L., Xue, Z., Liu, T., An, B.: Group-in-group policy optimization for llm agent training. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025)
14. Google DeepMind: Gemini 2.5 pro model card. `https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-2-5-Pro-Model-Card.pdf` (2025), updated: 27 June 2025. Accessed: 1 July 2026
15. Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., et al.: Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. Nature **645**, 633–638 (2025). `https://doi.org/10.1038/s41586-025-09422-z`
16. Hiranaka, A., Hwang, M., Lee, S., Wang, C., Fei-Fei, L., Wu, J., Zhang, R.: Primitive skill-based robot learning from human evaluative feedback. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 7817–7824. IEEE (2023)
17. Jin, C., Tan, W., Yang, J., Liu, B., Song, R., Wang, L., Fu, J.: Alphablock: Embodied finetuning for vision-language reasoning in robot manipulation. arXiv preprint arXiv:2305.18898 (2023)
18. Klissarov, M., Hjelm, R.D., Toshev, A.T., Mazoure, B.: On the modeling capabilities of large language models for sequential decision making. In: The Thirteenth International Conference on Learning Representations (2025)
19. Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., Deitke, M., Ehsani, K., Gordon, D., Zhu, Y., et al.: Ai2-thor: An interactive 3d environment for visual ai. arXiv preprint arXiv:1712.05474 (2017)
20. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: Advances in Neural Information Processing Systems. vol. 36, pp. 34892–34916 (2023)
21. Liu, R., Bai, C., Lyu, J., Sun, S., Du, Y., Li, X.: VLP: Vision-language preference learning for embodied manipulation. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 28428–28444. Association for Computational Linguistics (2025)
22. Liu, Z., Zang, Y., Zou, Y., Liang, Z., Dong, X., Cao, Y., Duan, H., Lin, D., Wang, J.: Visual agentic reinforcement fine-tuning. arXiv preprint arXiv:2505.14246 (2025)
23. Lu, M., Xu, R., Fang, Y., Zhang, W., Yu, Y., Srivastava, G., Zhuang, Y., Elhoseiny, M., Fleming, C., Yang, C., et al.: Scaling agentic reinforcement learning for tool-integrated reasoning in vlms. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 26518–26529 (2026)
24. Mazzaglia, P., Verbelen, T., Dhoedt, B., Courville, A., Rajeswar, S.: Genrl: Multimodal-foundation world models for generalization in embodied agents. Advances in neural information processing systems **37**, 27529–27555 (2024)
25. Nasiriany, S., Liu, H., Zhu, Y.: Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 7477–7484. IEEE (2022)
26. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. Advances in neural information processing systems **35**, 27730–27744 (2022)

27. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (18–24 Jul 2021)
28. Sarch, G., Jang, L., Tarr, M., Cohen, W.W., Marino, K., Fragkiadaki, K.: Vlm agents generate their own memories: Distilling experience into embodied programs of thought. Advances in Neural Information Processing Systems **37**, 75942–75985 (2024)
29. Schmidhuber, J.: Towards compositional learning with dynamic neural networks. Inst. für Informatik (1990)
30. Schmidhuber, J.: Learning to generate sub-goals for action sequences. In: Artificial neural networks. pp. 967–972 (1991)
31. Schmidhuber, J., Wahnsiedler, R.: Planning simple trajectories using neural subgoal generators. In: Proceedings of the 2nd International Conference on Simulation of Adaptive Behavior. pp. 196–202 (1992)
32. Schrader, M.P.B.: gym-sokoban. `https://github.com/mpSchrader/gym-sokoban` (2018), accessed: 1 July 2026
33. Schulman, J., Moritz, P., Levine, S., Jordan, M., Abbeel, P.: High-dimensional continuous control using generalized advantage estimation. In: International Conference on Learning Representations (2016)
34. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
35. Shen, H., Liu, P., Li, J., Fang, C., Ma, Y., Liao, J., Shen, Q., Zhang, Z., Zhao, K., Zhang, Q., Xu, R., Zhao, T.: Vlm-r1: A stable and generalizable r1-style large vision-language model. arXiv preprint arXiv:2504.07615 (2025)
36. Shen, W., Hu, J., Zhao, P., He, X., Chen, L.: Advanced tricks for training large language models with proximal policy optimization. `https://swtheking.notion.site/eb7b2d1891f44b3a84e7396d19d39e6f?v=01bcb084210149488d730064cbabc99f&pvs=74` (2024), notion Blog. Accessed: 28 June 2026
37. Sheng, G., Zhang, C., Ye, Z., Wu, X., Zhang, W., Zhang, R., Peng, Y., Lin, H., Wu, C.: Hybridflow: A flexible and efficient rlhf framework. arXiv preprint arXiv: 2409.19256 (2024)
38. Singh, A., et al.: Openai gpt-5 system card. `https://openai.com/index/gpt-5-system-card/` (2025), accessed: 1 July 2026
39. Srivastava, S., Wang, K., Chan, Y.C., Dai, T., Li, M., Zhang, R., Xu, M., Wu, J., Fei-Fei, L.: Rosetta: Constructing code-based reward from unconstrained language preference. In: Workshop on Continual Robot Learning from Humans (2025)
40. Tang, G., Rajkumar, S., Zhou, Y., Walke, H.R., Levine, S., Fang, K.: Kalie: Fine-tuning vision-language models for open-world manipulation without robot data. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 9507–9515. IEEE (2025)
41. Tao, S., Xiang, F., Shukla, A., Qin, Y., Hinrichsen, X., Yuan, X., Bao, C., Lin, X., Liu, Y., Chan, T.k., et al.: Maniskill3: Gpu parallelized robotics simulation and rendering for generalizable embodied ai. In: Robotics: Science and Systems (2025)
42. Team, G., Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., Matejovicova, T., Ramé, A., Rivière, M., Rouillard, L., Mesnard, T., Cideron, G., bastien Grill, J., Ramos, S., Yvinec, E., Casbon, M., Pot, E., Penchev, I., Liu,

G., Visin, F., hleen Kenealy, K., Beyer, L., Zhai, X., Tsitsulin, A., Busa-Fekete, R., Feng, A., Sachdeva, N., Coleman, B., Gao, Y., Mustafa, B., Barr, I., Parisotto, E., Tian, D., Eyal, M., Cherry, C., Peter, J.T., Sinopalnikov, D., Bhupatiraju, S., Agarwal, R., Kazemi, M., Malkin, D., Kumar, R., Vilar, D., Brusilovsky, I., Luo, J., Steiner, A., Friesen, A., Sharma, A., Sharma, A., Gilady, A.M., Goedeckemeyer, A., Saade, A., Feng, A., Kolesnikov, A., Bendebury, A., Abdagic, A., Vadi, A., György, A., Pinto, A.S., Das, A., Bapna, A., Miech, A., Yang, A., Paterson, A., Shenoy, A., Chakrabarti, A., Piot, B., Wu, B., Shahriari, B., Petrini, B., Chen, C., Lan, C.L., Choquette-Choo, C.A., Carey, C., Brick, C., Deutsch, D., Eisenbud, D., Cattle, D., Cheng, D., Paparas, D., Sreepathihalli, D.S., Reid, D., Tran, D., Zelle, D., Noland, E., Huizenga, E., Kharitonov, E., Liu, F., Amirkhanyan, G., Cameron, G., Hashemi, H., Klimczak-Plucińska, H., Singh, H., Mehta, H., Lehri, H.T., Hazimeh, H., Ballantyne, I., Szpektor, I., Nardini, I., Pouget-Abadie, J., Chan, J., Stanton, J., Wieting, J., Lai, J., Orbay, J., Fernandez, J., Newlan, J., yeong Ji, J., Singh, J., Black, K., Yu, K., Hui, K., Vodrahalli, K., Greff, K., Qiu, L., Valentine, M., Coelho, M., Ritter, M., Hoffman, M., Watson, M., Chaturvedi, M., Moynihan, M., Ma, M., Babar, N., Noy, N., Byrd, N., Roy, N., Momchev, N., Chauhan, N., Sachdeva, N., Bunyan, O., Botarda, P., Caron, P., Rubenstein, P.K., Culliton, P., Schmid, P., Sessa, P.G., Xu, P., Stanczyk, P., Tafti, P., Shivanna, R., Wu, R., Pan, R., Rokni, R., Willoughby, R., Vallu, R., Mullins, R., Jerome, S., Smoot, S., Girgin, S., Iqbal, S., Reddy, S., Sheth, S., Põder, S., Bhatnagar, S., Panyam, S.R., Eiger, S., Zhang, S., Liu, T., Yacovone, T., Liechty, T., Kalra, U., Evci, U., Misra, V., Roseberry, V., Feinberg, V., Kolesnikov, V., Han, W., Kwon, W., Chen, X., Chow, Y., Zhu, Y., Wei, Z., Egyed, Z., Cotruta, V., Giang, M., Kirk, P., Rao, A., Black, K., Babar, N., Lo, J., Moreira, E., Martins, L.G., Sanseviero, O., Gonzalez, L., Gleicher, Z., Warkentin, T., Mirrokni, V., Senter, E., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Matias, Y., Sculley, D., Petrov, S., Fiedel, N., Shazeer, N., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Alayrac, J.B., Anil, R., Dmitry, Lepikhin, Borgeaud, S., Bachem, O., Joulin, A., Andreev, A., Hardin, C., Dadashi, R., Hussenot, L.: Gemma 3 technical report (2025)

43. Towers, M., Kwiatkowski, A., Terry, J.K., Balis, J.U., de Cola, G., Deleu, T., Goulão, M., Kallinteris, A., Krimmel, M., KG, A., et al.: Gymnasium: A standard interface for reinforcement learning environments. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2025)
44. Wang, K., Zhang, P., Wang, Z., Gao, Y., Li, L., Wang, Q., Chen, H., Lu, Y., Yang, Z., Wang, L., et al.: Vagen: Reinforcing world model reasoning for multi-turn vlm agents. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025)
45. Wang, Z., Wang, K., Wang, Q., Zhang, P., Li, L., Yang, Z., Jin, X., Yu, K., Nguyen, M.N., Liu, L., et al.: Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. arXiv preprint arXiv:2504.20073 (2025)
46. Wei, T., Yang, Y., Xing, J., Shi, Y., Lu, Z., Ye, D.: Gtr: Guided thought reinforcement prevents thought collapse in rl-based vlm agent training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18855–18865 (2025)
47. Wei, T., Yang, Y., Zhang, C., Xing, J., Shi, Y., Lu, Z., Ye, D.: Gtr-turbo: Merged checkpoint is secretly a free teacher for agentic vlm training. In: Proceedings of the

IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 26476–26486 (2026)
48. Yang, J., Ding, R., Brown, E., Qi, X., Xie, S.: V-irl: Grounding virtual intelligence in real life. In: European conference on computer vision. pp. 36–55. Springer (2024)
49. Yang, J., Dong, Y., Liu, S., Li, B., Wang, Z., Tan, H., Jiang, C., Kang, J., Zhang, Y., Zhou, K., et al.: Octopus: Embodied vision-language programmer from environmental feedback. In: European conference on computer vision. pp. 20–38. Springer (2024)
50. Yang, R., Chen, H., Zhang, J., Zhao, M., Qian, C., Wang, K., Wang, Q., Koripella, T.V., Movahedi, M., Li, M., et al.: Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. In: Proceedings of the 42nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 267, pp. 70576–70631. PMLR (2025)
51. Yang, S., Du, Y., Ghasemipour, S.K.S., Tompson, J., Kaelbling, L.P., Schuurmans, D., Abbeel, P.: Learning interactive real-world simulators. In: The Twelfth International Conference on Learning Representations (2024)
52. Zhai, S., Bai, H., Lin, Z., Pan, J., Tong, P., Zhou, Y., Suhr, A., Xie, S., LeCun, Y., Ma, Y., et al.: Fine-tuning large vision-language models as decision-making agents via reinforcement learning. Advances in neural information processing systems **37**, 110935–110971 (2024)
53. Zhang, L., Liu, Y., Zhang, Z., Aghaei, M., Hu, Y., Gu, H., Alomrani, M.A., Bravo, D.G.A., Karimi, R., Hamidizadeh, A., et al.: Mem2ego: Empowering vision-language models with global-to-ego memory for long-horizon embodied navigation. In: Workshop on Foundation Models Meet Embodied Agents at CVPR 2025 (2025)
54. Zhou, E., Qin, Y., Shi, Z., Yin, Z., Huang, Y., Zhang, R., Sheng, L., Shao, J.: Chain-of-imagination for reliable instruction following in decision making. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 10010–10017. IEEE (2025)
55. Zhou, Y., Yang, Q., Lin, K., Bai, M., Zhou, X., Wang, Y.X., Levine, S., Li, L.E.: Proposer-agent-evaluator (PAE): Autonomous skill discovery for foundation model internet agents. In: Proceedings of the 42nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 267, pp. 79490–79528. PMLR (2025)
56. Zhou, Y., Zanette, A.: Archer: training language model agents via hierarchical multi-turn rl. In: Proceedings of the 41st International Conference on Machine Learning. pp. 62178–62209 (2024)
57. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. In: International Conference on Learning Representations (2024)