weasel

# W0: A dVLM-Based GUI Agent with Block-Diffusion Mode Steering

Minkyu Kang JinHo Kim Eunseop Shin Dayeon Hwang
UNIST (Ulsan National Institute of Science and Technology)
{perelman, justpain02, shineunseop, hdy0319}@unist.ac.kr

## Abstract

Long-horizon graphical user interface (GUI) agents are usually trained and evaluated as stepwise policies over the current screen, yet many software tasks are partially observable: the correct click or keystroke can depend on hidden progress, a previous tab, or earlier form state. We present W0, a vision–language model (VLM)-based GUI agent that pairs a frozen screen-language backbone with bounded persistent memory across interaction blocks and decodes each UI action chunk with a block-discrete diffusion action head. Training is deliberately restricted: after supervised fine-tuning (SFT) aligns a small steering actuator, reinforcement learning with verifiable rewards (RLVR) updates *only* a categorical post-prefix **mode selector** over the frozen model's denoising continuations — a procedure we call **Mode-Selection Policy Optimization** (MSPO). In the ideal frozen-mode abstraction, this turns policy optimization from full reverse-kernel fine-tuning into **support-constrained mode selection** inside the pretrained model's reachable behavior library, which an appendix formalizes as an abstract mode Markov decision process (MDP) with a Bellman contraction, approximate selector improvement, and a finite-cover comparison against full action/residual policy classes. As a technical report, we pair the formulation with a measurement snapshot for the current desktop configuration and specify the diagnostics needed to reproduce or extend the benchmark on mobile neural processing unit (NPU) deployments.

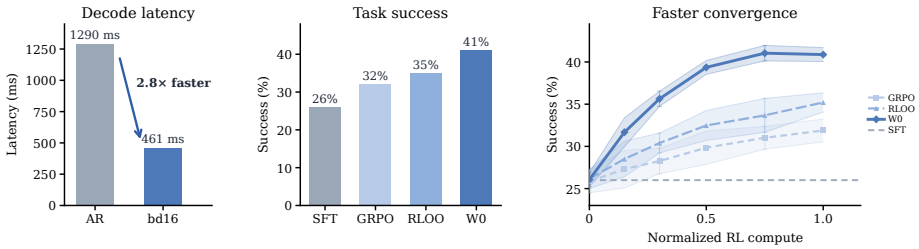



Figure 1: **W0 at a glance: faster block-diffusion decoding, higher GUI task success, and more sample-efficient selector-only RLVR.** Technical-report snapshot (current RTX 4060 desktop; numbers to be finalized). **Left:** at the deployed block size $b^\star$=16 (bd16), block-discrete decoding cuts per-chunk decode latency over an autoregressive decoder ($\approx 2.8\times$). **Middle:** selector-only W0 reaches 41% task success, above SFT (26%) and the two critic-free RLVR baselines applied to the full block-dVLM policy, GRPO (32%) and RLOO (35%). **Right:** W0 reaches the stronger success regime with fewer verifier-driven update rounds. The Snapdragon 8 Gen 3 NPU path is the intended mobile target and is not included in these desktop bars.

**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.

## 1 Introduction

Large language model (LLM)-powered agent services now let end users interact with AI directly, far beyond simple question answering. The most visible instantiation of this shift is the *GUI agent*: an LLM-based system that observes a software interface – a web page, a mobile app, a desktop OS – and manipulates it by emitting clicks, keystrokes, and gestures on the user's behalf (19, 23, 33, 35, 65, 82, 94, 97, 100–102, 105, 106). Powered by multimodal grounding from modern Vision–Language Models (VLMs) (5, 6, 18, 59, 83), the most recent generation collapses perception, grounding, and action prediction into a single end-to-end vision–language–action (VLA) model that operates directly on pixels (49, 65, 94, 97), giving the agent the closed-loop autonomy that a pure QA system lacks.

**Yet agency is only half of what real tasks demand.** Common GUI-agent training and evaluation objectives operationalize a stepwise, Markov decision process (MDP)-style next-action problem: given the provided screen and prompt context, predict the next action $\pi_\theta(a_t \mid o_t, l, c_t)$. The assumption that the provided context is a sufficient state breaks the moment the workflow becomes *partially observable*. A multi-step quiz cannot be solved by looking at the current page if the relevant question state is hidden in earlier turns; a multi-tab workflow requires remembering which tab holds which artifact; a button that has already been pressed looks identical to one that has not. Formally these are partially observable Markov decision processes (POMDPs), in which a policy that conditions only on $o_t$ is not generally optimal; a relevant decision statistic may depend on the history $h_t = (o_{0:t}, a_{0:t-1})$ or on a belief $b_t(s) = \Pr(s_t = s \mid h_t)$ (38).

The deep-RL community has spent the last decade building the machinery this requires — recurrent history compression (27, 31, 40), latent world models (28–30), attention over full trajectories (16, 20, 60), and external differentiable memory (25, 26, 87), all approximating $b_t$ with a fixed-size latent $z_t$. Yet this machinery is still not a first-class object in most native GUI-agent objectives, which instead train screenshot-conditioned action heads or VLM policies whose next-action loss treats the provided context as sufficient (35, 65, 97). Long-horizon GUI behavior falls into the gap between these two literatures.

**Is there a way to give a VLM-based GUI agent a bounded belief-like memory without breaking the frozen backbone or the interaction loop?** The end-to-end collapse buys closed-loop autonomy, but it fuses three concerns with different requirements — a perceiver that should stay frozen, a memory that must persist across the horizon, and a low-latency actuator. Our main hypothesis is that a GUI policy can instead *separate* these three roles: a frozen screen-language model, a bounded history memory, and a compact action expert — so the pretrained perceiver's grounding stays intact while the only learned component is small and cheap to train.[1] Concretely, W0 comprises three components:

- a **frozen screen-language backbone**, instantiated with GUI-Owl-1.5-2B from the Mobile-Agent-v3.5 family (96);
- a **hybrid linear-plus-sliding-window attention memory** that carries bounded history at $O(1)$ recurrent cost per block (2, 41); and
- a **block-discrete diffusion action head** — a discrete-diffusion language model (dLLM) or, with vision conditioning, vision-language model (dVLM) — whose parallel denoising is a natural fit for discrete UI event chunks (4, 48, 57, 70, 92).

The third component, the action head, carries a constraint the memory thread does

[1] The foundation-model/action-head split in robotic VLAs (11, 63) is the architectural analogy, not the application domain — the target policy is a GUI agent over browser, desktop, and mobile screens.

not: real time. A GUI agent must commit its next action before the observed screen goes stale, so per-chunk decode latency is a design constraint. Robotic VLAs meet the analogous control-frequency budget by emitting an action *chunk* per inference (11, 62, 107); W0 borrows the remedy for discrete UI actions, where a block-discrete diffusion head decodes a whole chunk in a few parallel denoising steps rather than token by token (48, 74). The block *size* sets this parallelism: larger blocks decode faster but strain structured-output integrity, so we schedule it with a curriculum that anneals only the distribution center — from the autoregressive anchor $b$=1 to the deploy/eval target ($b^\star \approx 16$) — at fixed spread, with the largest candidate $b$=32 retained as a symmetric training stress tail rather than as the curriculum target.

Freezing the backbone and action head turns RL into a *support-constrained* problem: the selector can only reweight behaviors already in the frozen model's reachable library, never invent new ones. After SFT aligns the steering actuator (the fixed module that turns a selected mode into a concrete action chunk), RLVR (reinforcement learning with verifiable rewards) changes only which frozen *denoising mode* is selected — which of the action head's candidate denoising continuations the agent commits to — an update we name **Mode-Selection Policy Optimization** (MSPO). Optimizing the selector rather than the reverse kernels keeps the expensive pretrained behavior fixed and reduces RL to reweighting a finite library of continuations, the source of the sample efficiency we report. Figure 1 previews three results from the current measurement snapshot (numbers to be finalized): (i) at the deployed block size $b^\star$=16, block-diffusion decoding cuts per-chunk decode latency by roughly 2.8× over an autoregressive decoder; (ii) selector-only W0 improves GUI task success over SFT and the two critic-free RLVR baselines, GRPO and RLOO; and (iii) it reaches that stronger success regime with fewer verifier-driven RL updates.

**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.

**Contributions.**

- **W0**, a long-horizon GUI agent that decouples a frozen screen-language backbone and bounded persistent memory from a block-discrete diffusion (dLLM/dVLM) action head, bringing the foundation-model/action-head split to GUI control.
- **MSPO** (Mode-Selection Policy Optimization): a critic-free, verifier-driven RL update that trains *only* a categorical mode selector over the frozen model's continuations (support-constrained) — the same critic-free RLVR objective our baselines GRPO and RLOO apply to the *full* block-dVLM policy, and distinct from critic-based or latent-noise diffusion RL (OGPO, DSRL).
- A **self-contained performance-transfer analysis** for MSPO in which every gap is derived from named primitives: a Bellman contraction and a finite-sample concentration plus approximate-policy-iteration bound on the learning gap, a return-level lifting bound on the implementation gap, and a simulation-lemma bound on the model-abstraction gap.
- A **degree-2 (Gaussian-in-$\log b$) block-size curriculum** that, in $x = \log b$, is an unnormalized Gaussian; over octave-spaced candidates it is the discrete Gaussian $\mathcal{N}(\mu_x, \sigma_x^2)$ whose multipliers have the *closed form* $\ell_2 = 1/(2\sigma_x^2)$, $\ell_1 = -\mu_x/\sigma_x^2$ — i.e. $\ell_1, \ell_2$ are *derived* from the two moments $\mu_x = \mathbb{E}[\log b]$, $\sigma_x^2 = \mathrm{Var}[\log b]$ as the maximum-entropy (Gibbs) Lagrange multipliers, not tuned (the Boltzmann power law is the $\ell_2 = 0$ special case). The

curriculum anneals only the center (from the autoregressive anchor $b$=1 to the eval target $b^\star \approx 16$) at constant curvature $\ell_2$, together with the finding that large-block parallel unmasking is where structured-output format collapses.

- A **technical-report measurement snapshot and validation protocol**, including an Android-curated reasoning-SFT corpus and decoder/curriculum diagnostics for reproducing or extending the benchmark.

We are deliberate about scope. The only *unconditional* theoretical claim is representational — via the multiplier-to-moment map $\mu_x = -\ell_1/(2\ell_2)$, $\sigma_x^2 = 1/(2\ell_2)$ the two free moments map bijectively to $(\ell_1, \ell_2)$, so the degree-2 law sets the mean and variance of log-block-size independently while a monotone ($\ell_2$=0) law ties spread to mean, and the curriculum varies only the mean (center) at fixed variance, which a monotone one-parameter law cannot; the transfer bounds are explicit but rest on stated primitives, and the curriculum is a design choice rather than a method proven to dominate direct training. The remainder of the paper formalizes the selector objective (Section 2), situates W0 among GUI agents, memory, and diffusion-policy RL (Section 3), details the architecture and training (Section 4), specifies the validation protocol, and proves the transfer analysis (Appendix A) and the curriculum characterization (Appendix B).

## 2 Problem Statement

We state the learning problem before the architecture that solves it. W0 trains exactly one object: a mode selector $q_\psi$. Everything else in the action pipeline is frozen. The goal of this section is to make the selector's objective *explicit* as an optimization problem, and to state precisely what optimum W0 competes with. All symbols are collected in Table 1; we define each at first use.

**The objective, in one line.** Fix the discount $\gamma \in [0, 1)$ and a per-block verifier reward $R(\chi_b, y_b) \in [-R_{\max}, R_{\max}]$, where $\chi_b$ is the outer controller context at block $b$ and $y_b \in \mathcal{Y}$ is the action chunk the environment receives (both defined below). For a selector $q_\psi$ that induces a distribution over interaction rollouts, its **discounted return** is

$$J(q_\psi) = \mathbb{E}_{q_\psi}\left[\sum_{b \geq 0} \gamma^b \, R(\chi_b, y_b)\right], \tag{1}$$

where the expectation is over the rollout generated when the selector $q_\psi$ drives a fixed pretrained pipeline. W0 solves

$$\psi^\star \in \arg\max_{\psi} \; J(q_\psi), \qquad \text{equivalently} \qquad J^\star = \sup_{q \in \mathcal{Q}} J(q), \quad \mathcal{Q} = \{q_\psi\}_\psi, \tag{2}$$

i.e. it maximizes return over the **frozen-mode selector class** $\mathcal{Q}$ of all admissible selectors over the pretrained continuation library. The benchmark $J^\star$ is the *support-constrained frozen-mode optimum*: the best return any selector in $\mathcal{Q}$ can achieve, *not* the global optimum over all GUI action strings. We make no global-optimality claim. For the selector that RLVR returns, Appendix A writes $J^\star$ as the optimal frozen-mode value $J^\star_{\text{mode}}$ and bounds the regret against this benchmark,

$$\underbrace{J^\star - J(q_\psi)}_{\text{regret}} \;\leq\; \Delta_{\text{learn}} + \Delta_{\text{impl}} + \Delta_{\text{model}}, \tag{3}$$

where the three terms are bounded by explicit primitives $\epsilon_R, \epsilon_P, \epsilon_{\text{suff}}, \epsilon_{\text{verifier}}, \epsilon_{\text{opt}}$ (reward/verifier reliability, model abstraction, context sufficiency, verifier bias, and optimization slack; Corollary A.6). These gaps are assumed small but not claimed to vanish; in particular the curriculum is a design choice, not a procedure proven to dominate direct training.

The remaining three blocks define the objects in Eqs. (1)–(2): the outer decision process, the inner denoising realization, and the frozen/trained split that pins down $\mathcal{Q}$.

**Outer decision process.** We model a GUI task as a partially observable discounted control problem. The full interaction history is $h_b = (o_{\le b}, y_{<b})$, but W0 does not feed this raw history to the action decoder; it compresses it into a bounded controller context from the current screen $o_b$, instruction $\ell$, and persistent memory $\xi_{b-1}$:

$$\chi_b = (o_b, \ell, \xi_{b-1}). \tag{4}$$

The environment never sees the token-by-token or denoise-step computation inside the model. At each block it receives one completed UI action chunk and returns a reward and a next context,

$$y_b \in \mathcal{Y}, \qquad \chi_{b+1} \sim P(\cdot \mid \chi_b, y_b), \qquad R(\chi_b, y_b), \tag{5}$$

where $P$ is the abstract Markov kernel induced by the chosen memory state, not an assumption that the raw GUI process is fully observable. Any failure of $\chi_b$ to be Markov-sufficient is charged to $\epsilon_{\text{suff}}$ in Appendix A. Concretely, a click, scroll, typed span, or short tool-call sequence is one outer action $y_b$.

**Inner frozen denoising realization.** W0 realizes each outer action with an inner block-discrete denoising chain of $K$ steps,

$$\tau_b = (z_{b,K}, z_{b,K-1}, \ldots, z_{b,0}), \qquad y_b = \text{decode}(z_{b,0}). \tag{6}$$

For $k > 0$, the intermediate $z_{b,k-1}$ is an internal state of the action head, not an environment action. W0 first runs a *frozen* denoising prefix, reads prefix-measurable denoise-history evidence $H_{b,\mathcal{I}_{\text{obs}}}$ from it over the observed-prefix window $\mathcal{I}_{\text{obs}}$, selects a continuation mode $m_b$, and then completes the chunk through a fixed SFT-aligned actuator. A full diffusion-policy RL method would instead optimize the reverse-kernel likelihood of the whole chain; W0 freezes those kernels and optimizes only the choice among their continuation modes (Appendix A).

**Frozen/trained split.** During RLVR the VLM, memory backbone, dLLM/dVLM action backbone, and actuator are all fixed. The only trainable policy is the **post-prefix mode selector**

$$m_b \sim q_\psi(\cdot \mid C_b), \qquad m_b \in \mathcal{M}_C, \qquad C_b = (\chi_b, H_{b,\mathcal{I}_{\text{obs}}}), \qquad y_b \sim \pi_0(\cdot \mid C_b, m_b), \tag{7}$$

a distribution over a finite continuation-mode set $\mathcal{M}_C$ given the prefix-measurable selector context $C_b$, where the frozen mode-conditioned action law $\pi_0(\cdot \mid C_b, m_b)$ turns a selected mode into an action chunk. Varying $\psi$ in Eq. (2) therefore reweights these frozen laws but cannot create support outside them. This is why the attainable optimum is $J^\star$ over $\mathcal{Q}$ rather than global optimality, and why optimization reduces to a low-dimensional mode choice inside the frozen model's effective support rather than a fine-tuning problem over all reverse denoising kernels.

## 3 Related Work

### 3.1 VLM-Based GUI Agents and MDP-Framed Training

This subsection covers VLM-based GUI agents, i.e., policies over digital interfaces that observe screens, receive a natural-language goal, and emit UI actions. ReAct-style agents first made the reasoning/action loop explicit for language models (100). Mind2Web, WebArena, and VisualWebArena established realistic web-agent benchmarks and evaluation protocols (23, 44, 106), while SeeAct and WebVoyager studied grounded web navigation with modern VLMs (33, 105). Native screenshot systems such as AppAgent, CogAgent, SeeClick/ScreenSpot, OS-Atlas, ShowUI, Aguvis, and UI-TARS scale visual grounding and action modeling across mobile, desktop, and browser settings (19, 35, 49, 65, 94, 97, 102). The Mobile-Agent line progresses from visual mobile operation to multi-agent navigation and self-evolving experience stores (81, 82, 86); Mobile-Agent-v3 then introduces GUI-Owl as a foundational GUI agent model (101). Mobile-Agent-v3.5 introduces GUI-Owl-1.5, a multi-platform native GUI agent family with 2B/4B/8B/32B/235B variants and explicit mobile, desktop, and browser support (96); W0 uses the GUI-Owl-1.5-2B scale as the practical edge-side GUI backbone. UI-TARS-2 is a complementary native GUI-agent direction that emphasizes multi-turn RL, sandboxed rollout infrastructure, and data-flywheel training (80).

The bottleneck is not perception alone. Demonstrations and online rollouts are commonly reduced to tuples of the form $(o_b, \ell, a_b)$, or to a prompt/history string plus the current screen, and the learner is optimized to predict the next action from that available context. This is an MDP-style operational assumption: the training objective treats the provided observation context as sufficient for the next action. Reflection notes, prompt histories, and planner scratchpads help empirically, but they are not equivalent to learning a bounded belief-like state with an explicit decision role. W0 keeps the VLM-based GUI-agent interface but moves the action policy to a belief-conditioned form $\pi(a_b \mid o_b, \ell, \xi_b)$.

### 3.2 Bounded Memory for POMDP GUI Control

The memory module has a different role from the dVLM action head. It must carry task history across blocks while preserving a fixed deployment shape. Linear attention gives an $O(1)$ recurrent state (41); structured state-space duality (SSD) connects this recurrence to structured state-space models (22); hybrid linear-plus-window attention restores local precision at high throughput (2). This motivates W0's hybrid linear-plus-window attention memory. A linear branch carries long-horizon belief, while a small full attention branch anchors local screen details and exact goal cues.

This memory is not a replacement for the dVLM's denoise-history cache. The persistent memory $\xi_b$ survives across GUI actions. The denoise-history cache is block-local evidence used to select the current continuation mode and is discarded after the action chunk is decoded.

### 3.3 VLM-Based Agents and VLA Action Heads

Robotic VLAs supply the architectural analogy. RT-2 and OpenVLA map images and instructions to action tokens (43, 107), while $\pi_0$ and $\pi_{0.5}$ pair a VLM with a diffusion or flow action expert (11, 63). This decoupled template is useful for GUI agents: keep the expensive screen-language model frozen, and train a compact action expert that targets interaction-rate decoding.

The GUI setting changes the action codomain. UI events are naturally discrete strings or chunks: coordinates, key codes, scroll operations, tool calls, and typed text. FAST shows that action tokenization is a first-order design choice for VLA policies (62). Discrete Diffusion VLA shows that discretized action chunks can be modeled by discrete diffusion inside a unified transformer, preserving pretrained vision-language priors while supporting parallel refinement (48). Fast-dVLA is related as an efficiency/adaptation direction for discrete-diffusion VLA policies (74). W0 adopts the dLLM/dVLM action-head view for GUI control.

### 3.4 Block-Parallel dLLM/dVLM Decoding

Autoregressive decoding emits one token per forward pass, often leaving GPU tensor-core parallelism underused at batch size one. Block diffusion changes the unit of generation. A block is denoised over $K$ refinement steps, and all tokens inside the block are updated in parallel. D3PM and MDLM provide the discrete/masked diffusion foundation (4, 70); LLaDA scales this recipe to language modeling (57). BD3-LM combines block-level autoregression with intra-block diffusion (3); D2F adds block-wise autoregressive (AR) structure and following-token prediction to obtain faster-than-AR inference (85); Fast-dLLM and Fast-dLLM v2 add cache-aware and block-diffusion acceleration (91, 92); Fast-dVLM converts AR VLMs into block-diffusion VLMs with key–value (KV) cache-compatible parallel decoding (90). BARD similarly bridges AR and diffusion VLMs, progressively merging small decoding blocks into larger ones via stage-wise distillation from a fixed small-block anchor (15). At the agent level, DLLM Agent isolates the decoding paradigm by holding the agent framework and supervision fixed while swapping an AR backbone for a diffusion one, and reports faster, fewer-round inference at comparable task success alongside structured tool-call reliability challenges specific to diffusion policies (104).

This lineage is the reason W0 uses a block-discrete action expert. The relevant common structure is intra-block parallel denoising: the environment sees one completed GUI action chunk, while the model internally refines a fixed-size block over $K$ steps. Inter-block decoding may remain sequential, cache-reused, or pipelined depending on the backbone. W0's steering claim concerns the block-local denoising trajectory, not a particular inter-block scheduler. W0 confronts the same speed-versus-structured-output tradeoff that DLLM Agent reports, but in the GUI setting: the block-discrete action head targets faster, fewer-round decoding (Figure 1), while the curriculum diagnostics track strict JSON/action recovery against repaired outputs (Figure 2).

### 3.5 RL for Diffusion Policies and Frozen Mode Selection

DDPO treats denoising as a multi-step decision process and optimizes the reverse kernels with policy gradients (10). DPPO applies the same view to diffusion policies for robot control (69). DSRL freezes a diffusion or flow policy and learns over its initial latent-noise input (78); OGPO uses an off-policy critic and updates the full generative control policy (61). W0 is narrower than all of these: the Fast-dVLM/dVLA backbone and SFT-aligned actuator are fixed during RLVR, and only a post-prefix continuation-mode selector is updated.

Phase-transition analyses of diffusion models motivate a testable hypothesis for why such a selector can be low dimensional. High-level mode commitment may concentrate in a small portion of the denoising trajectory, while later steps mainly refine intra-mode details (9, 68). The validation protocol tests this hypothesis through critical-window ablations rather than assuming it as a theorem.

Table 1: Notation used throughout the paper.

| Symbol | Meaning |
|---|---|
| $o_b,\ \ell,\ \xi_b$ | screenshot, language instruction, and bounded persistent memory (after block $b$) |
| $\chi_b = (o_b, \ell, \xi_{b-1})$ | outer controller context / Bellman state |
| $y_b \in \mathcal{Y}$ | decoded action chunk (the environment action) |
| $b$ | interaction-block index; in the curriculum, the block *size* $b \in \{1, 2, 4, 8, 16, 32\}$ (the value range disambiguates the two uses; $b$=32 is a non-deployed training stress tail, deployment/eval clamped to $b^\star$=16) |
| $z_{b,k}$ | internal denoising state at block $b$, step $k$ (not an environment action) |
| $K$ | number of intra-block denoising steps |
| $C_b = (\chi_b, H_{b,\mathcal{I}_{\text{obs}}})$ | prefix-measurable selector context |
| $m_b \in \mathcal{M}_C$ | continuation mode (selector command) from the finite mode set $\mathcal{M}_C$ |
| $q_\psi(m \mid C)$ | trainable mode selector (the only object updated by RLVR) |
| $\pi_0(\cdot \mid C, m)$ | frozen mode-conditioned action law |
| $P(b)$ | degree-2 Gaussian-in-log $b$ block-size curriculum, the discrete Gaussian $\mathcal{N}(\mu_x, \sigma_x^2)$ in $x = \log b$ over octave candidates $b \in \{1, 2, 4, 8, 16, 32\}$, multipliers $\ell_2 = 1/(2\sigma_x^2)$, $\ell_1 = -\mu_x/\sigma_x^2$ (Sec. 4.6) |
| $b^\star$ | deploy/eval center (mode) of the curriculum, $b^\star \approx 16$ |
| $\mu_x,\ \sigma_x^2$ | mean and variance of log-block-size, $\mu_x = \mathbb{E}[\log b]$, $\sigma_x^2 = \mathrm{Var}[\log b]$ |
| $\gamma,\ R_{\max}$ | discount factor and reward bound |

# 4 Method

## 4.1 Overview

W0 is a long-horizon GUI agent built from three components: a frozen vision-language backbone that encodes the screen and task, a bounded memory module that carries history across the interaction, and a parallel action expert that emits the next UI event chunk. At outer block $b$, the agent receives a screenshot $o_b$, a language instruction $\ell$, and persistent memory $\xi_{b-1}$, and it outputs a discrete action chunk $y_b$ such as a click, scroll, typed string, or short sequence of UI events. The environment transitions only after $y_b$ is decoded; the intermediate denoising states used to produce $y_b$ are internal policy states.

The design separates three trainable objects. First, the VLM and dLLM action backbone are pretrained and then frozen. Second, a small steering actuator is SFT-aligned so that requested mode commands produce bounded residual changes in the denoising chain. Third, RLVR trains only a selector $q_\psi(m \mid C_b)$ over post-prefix continuation modes. In the ideal frozen-mode abstraction, policy optimization is therefore a low-dimensional mode-selection problem inside the frozen model's effective support rather than a full fine-tuning problem over all reverse denoising kernels; implementation leakage is handled separately in the appendix.

## 4.2 VLM-Based GUI Backbone

W0 uses a frozen screenshot-native VLM as its screen-language encoder. The default instantiation is **GUI-Owl-1.5-2B-Instruct** (96), which is GUI-specialized from the **Qwen3-VL-2B** foundation model (5). We use this backbone as the baseline screen-language model and detail its architecture — including the *DeepStack* multi-level vision–language fusion (54) that W0 inherits — in Appendix E.

**GUI specialization and the distillation target.** GUI-Owl-1.5-2B-Instruct adds GUI competence on top of this foundation: the Mobile-Agent-v3.5 recipe continually pretrains and post-trains the backbone on large-scale multi-platform (mobile/desktop/browser) trajectories through a hybrid data flywheel, a unified thought-synthesis pipeline, and the MRPO reinforcement-learning objective (96); its predecessor GUI-Owl was specialized analogously from Qwen2.5-VL (101). The *Instruct* variant is CoT-free and tuned for low-latency, high-efficiency execution, which is why it fits the interaction-rate budget of Section 4.4. This single backbone plays two roles in W0: it is the frozen perceiver of Eq. (8), and its native clean/autoregressive branch — a stop-gradient stream of the *same* network, not a separate model — is the teacher that the block-diffusion action head distills toward (Section 4.5). Reusing one GUI-pretrained backbone for both the perceiver and the action-head teacher is intended to let the action head inherit that grounding from the shared backbone rather than relearn it.

For each observation block, the frozen VLM consumes $(o_b, \ell)$ and produces visual-language features

$$H_b^{\mathrm{VLM}} = \mathrm{VLM}(o_b, \ell) \in \mathbb{R}^{S_b \times D_{\mathrm{VLM}}}. \tag{8}$$

A lightweight adaptor maps these features to the action-head hidden dimension. The VLM pass is computed once per outer block and cached; the memory module and the dLLM/dVLM action head consume the cached features without updating VLM parameters.

### 4.3 Memory Module

W0 maintains a persistent state $\xi_b$ that summarizes the interaction history. The main branch is a linear-attention/SSD recurrence with fixed-size state, augmented by a short full-attention window for current-screen precision:

$$\xi_b = \mathrm{Mem}_\omega(\xi_{b-1}, H_b^{\mathrm{VLM}}, y_{b-1}). \tag{9}$$

This state plays the role of an approximate belief statistic for the GUI POMDP. It is deliberately separated from the block-local denoise-history cache: $\xi_b$ persists across blocks, while denoise-history evidence is discarded after the current action chunk has been generated.

### 4.4 Action Decoder and Deployment Assumptions

The canonical W0 action decoder is a separate block-discrete diffusion head conditioned on frozen GUI-backbone features and persistent memory. It follows the discrete-diffusion VLA action representation (48): a GUI action chunk is tokenized as a fixed-length block containing action type, arguments, coordinates, tool-call fields, or text spans. It is Fast-dVLM/Fast-dLLM style in its decoding schedule: a fixed block is refined for $K$ denoising steps with intra-block parallel updates. Unless explicitly measured, we claim only fixed-shape compatibility, not hardware optimality: deployment fixes the block length, denoising-step budget, attention masks, and visual-token cap.

The decoder samples a noisy/masked action block $z_{b,K}$ and denoises

$$\tau_b = (z_{b,K}, z_{b,K-1}, \ldots, z_{b,0}), \qquad y_b = \mathrm{decode}(z_{b,0}). \tag{10}$$

For $k > 0$, $z_{b,k-1}$ is not an environment action; it is an internal denoising update. W0 first runs a frozen prefix $(z_{b,K}, \ldots, z_{b,\kappa})$, computes prefix evidence $H_{b,\mathcal{I}_{\mathrm{obs}}}$, and forms the selector context $C_b = (\chi_b, H_{b,\mathcal{I}_{\mathrm{obs}}}) = (o_b, \ell, \xi_{b-1}, H_{b,\mathcal{I}_{\mathrm{obs}}})$. The selector samples a continuation mode

$m_b \sim q_\psi(\cdot \mid C_b)$, and the SFT-aligned actuator injects bounded residuals only in the future continuation window $\mathcal{I}_{\text{inj}} \subseteq \{0, \dots, \kappa - 1\}$:

$$u_{b,k} = A_{\eta_{\text{SFT}}}(h^0_{b,k}; C_b, m_b), \qquad r_{b,k} = w_k P_0 u_{b,k}, \qquad k \in \mathcal{I}_{\text{inj}}. \tag{11}$$

Here $h^0_{b,k}$ is the frozen pre-residual hidden state, $P_0$ is a fixed residual projection, and $w_k$ gates steering toward the critical denoising window. The dLLM backbone, VLM, memory backbone, and SFT-aligned actuator are fixed during RLVR; only $q_\psi$ changes.

### 4.5 Block-Diffusion Forward Process and Dual-Stream Training

**Forward (masking) process.** A response action chunk is a token sequence partitioned into blocks of size $b$. Following masked/absorbing discrete diffusion (4, 70), each block is corrupted independently: block $j$ draws a mask rate $p_j \sim \mathcal{U}[\epsilon, 1]$, and each of its tokens is replaced by the absorbing [MASK] symbol i.i.d. with probability $p_j$, producing a **noisy** stream $\tilde{y}$ and its unmasked **clean** complement. Drawing the noise *per block* (rather than one global level) exposes each forward pass to a spectrum of corruption rates; drawing the block size $b$ itself from the curriculum $P(b)$ of Section 4.6 (Figure 3) exposes it to a spectrum of block sizes.

**Dual-stream parameterization.** The network is run on the concatenation

$$\big[\ \underbrace{\tilde{y}}_{\text{noisy text stream}}\ \big|\ \underbrace{(o, \ell, \xi, y_{<b})}_{\text{clean multimodal stream}}\ \big],$$

under a single asymmetric attention mask with three rules: noisy tokens attend (i) *bidirectionally within their own block* (intra-block parallel denoising) and (ii) *causally to the clean stream of earlier blocks*; (iii) the clean stream is autoregressive-causal and carries the grounded visual prefix, instruction, memory $\xi$, and committed tokens $y_{<b}$. The noisy stream predicts the masked response tokens while the clean stream supplies grounding and a same-network teacher (below); at deployment only the block-parallel noisy path is unrolled for $K$ steps per chunk.

**Stage-1 objective.** Let $\text{CE}_{\text{noisy}}$ and $\text{CE}_{\text{clean}}$ be the token-level cross-entropies of the two streams on their masked/shifted response targets, and let $\text{KD}_\bullet = \text{KL}\big(\text{sg}[\, p_{\text{clean}}/T \,] \,\big\|\, p_{\text{noisy}}\big)$ be forward-KL distillation from the *stop-gradient* clean/autoregressive teacher into the noisy student at temperature $T$, where $\text{KD}_{\text{fs}}$ isolates the heavily-masked, largest-block pair. The behavior-cloning objective is

$$\mathcal{L}_{\text{BC}} = \text{CE}_{\text{noisy}} + \beta_{\text{c}}\, \text{CE}_{\text{clean}} + \beta_{\text{kd}}\, \text{KD}_{\text{noisy}} + \lambda_{\text{fs}}(b)\, \text{KD}_{\text{fs}}, \tag{12}$$

with $\beta_{\text{c}}$=0.75, $\beta_{\text{kd}}$=0.25, and a block-scaled few-step distillation weight

$$\lambda_{\text{fs}}(b) = \lambda_0\, \text{ramp}(t)\, \min\big(b/b_{\text{ref}},\, c\big), \qquad \lambda_0{=}0.25,\ b_{\text{ref}}{=}4,\ c{=}4,$$

saturating at the deploy center $b^\star$=16 (the clamp ceiling $c$=4=16/4) so that the non-deployed $b$=32 stress tail receives no *more* distillation pressure than the $b$=16 center — it is regularized, not optimized as a deploy target. Larger blocks — where factorized parallel unmasking is hardest (Finding B.3) — are therefore pulled more strongly toward the small-block/clean teacher, realizing the stage-wise distillation of BARD (15) inside a single network rather than across separate checkpoints. The VLM backbone, memory, the dLLM reverse kernels, and (after Stage 2) the steering actuator are frozen; Stage 3 updates only the selector $q_\psi$.

### 4.6 Training Objective

W0 follows a three-stage training recipe. Stage 1 pretrains the dLLM action head by masked/discrete diffusion behavior cloning on GUI action chunks. Following the progressive small-to-large block-merging schedule of BARD (15), the block size used by this branch is not fixed but drawn from a *degree-2 Gaussian-in-*$\log b$ *curriculum* $P_t(b) \propto \exp\big(-\ell_1(t)\log b - \ell_2(t)(\log b)^2\big)$ over octave-spaced candidate block sizes $b \in \{1, 2, 4, 8, 16, 32\}$. In the energy coordinate $x = \log b$ this is an unnormalized Gaussian, and over octave candidates ($x = k \ln 2$, $k = \log_2 b$) it is exactly the discrete Gaussian $\mathcal{N}(\mu_x, \sigma_x^2)$ whose coefficients are the closed-form maximum-entropy (Gibbs) Lagrange multipliers

$$\ell_2(t) = \frac{1}{2\,\sigma_x^2}, \qquad \ell_1(t) = -\frac{\mu_x(t)}{\sigma_x^2} \qquad \Big(\text{inverse:}\;\; \sigma_x^2 = \tfrac{1}{2\ell_2},\;\; \mu_x = -\tfrac{\ell_1}{2\ell_2}\Big),$$

$$x = \log b, \qquad \mu_x = \mathbb{E}[\log b], \qquad \sigma_x^2 = \mathrm{Var}[\log b],$$

so $\ell_1, \ell_2$ are *derived* from two moments, not tuned (Appendix B.1, Remark B.2(b)). Deployment and evaluation fix the center $\mu_k = \log_2(16) = 4$ (i.e. $\mu_x = \ln 16$); the spread is one octave, $\sigma_k = 1$ (the natural spacing of octave candidates), so $\sigma_x = \ln 2$ and $\ell_2 = 1/(2(\ln 2)^2) \approx 1.04$ is held *constant* through training. The curriculum anneals *only* the center, from the autoregressive anchor $b = 1$ ($\mu_k = 0$, so $\ell_1 = 0$) to the eval target $b = 16$ ($\mu_k = 4$, so $\ell_1 = -\ln 16/(\ln 2)^2 \approx -5.77$), at constant curvature $\ell_2$ — not by sharpening the spread. The start center $b = 1$ gives the allocation $P \approx \{1{:}0.57,\ 2{:}0.35,\ 4{:}0.08\}$ with mode $b = 1$, and the end center $b = 16$ gives $P \approx \{8{:}0.26,\ 16{:}0.42,\ 32{:}0.26\}$ with mode $b = 16$, the deploy/eval target (Figure 3); the end allocation is symmetric on $\{8, 16, 32\}$. The training mean $\mathbb{E}[b] \approx 17.3$ at the end center is inflated by the symmetric $b = 32$ stress tail, but deployment is at the mode/center $b = 16$, never the mean. The largest candidate $b = 32$ is *included* precisely so the eval-centered end Gaussian is symmetric ($\{8, 16, 32\}$) rather than boundary-truncated; it is a non-deployed training-time stress tail (bd32 structured-output collapse, Finding B.3) that regularizes the $b = 16$ center, while deployment and evaluation are clamped to $b^\star = 16$. The one-parameter Boltzmann power law $P(b) \propto b^{-\lambda}$ used for the measurement snapshot of this report is the $\ell_2 = 0$ special case; here, by contrast, $\ell_2$ is held fixed nonzero ($\ell_2 = 1/(2(\ln 2)^2)$) and only $\ell_1$ (the center) anneals, so the deployed degree-2 curriculum is *not* the $\ell_2 = 0$ case (that was only the measurement-snapshot Boltzmann run). Via the closed-form Lagrange multipliers above, this family is the maximum-entropy block-size law for a fixed mean — and, for $\ell_2 > 0$, variance — of log-block-size (Appendix B.1, Proposition B.1 and Remark B.2); its only unconditional advantage is representational: a one-parameter monotone law ties its spread to its mean, so it can neither place an interior mode nor hold a controlled-variance low/mid-block tail at a high mean — it can only collapse monotonically toward the largest block — whereas the degree-2 family sets the mean and variance of $\log b$ independently (this follows because $(\mu_x, \sigma_x^2)$ map bijectively to $(\ell_1, \ell_2)$ via $\ell_2 = 1/(2\sigma_x^2)$, $\ell_1 = -\mu_x/\sigma_x^2$; Remark B.2(a)–(b)); the benefit of the small-to-large ordering itself is empirical, motivated by the coarse-to-fine learning order of image diffusion (79, 98) and carrying no proven optimality (Remark B.4). Stage 2 aligns the steering actuator with supervised mode commands while the VLM and dLLM backbone remain frozen. Mode-command labels can be formed by clustering frozen continuation evidence, by optimal-transport/radial-basis-function (OT/RBF) prototype assignment, or by manually specified intervention tags for known failure forks; these labels supervise the actuator, not the RL selector's reward objective. Stage 3 performs the **MSPO** update: critic-free RLVR on the selector. Given a batch of mode continuations for the same context $C_b$, a verifier supplies sparse success rewards

and, when available, dense milestone rewards from the GUI environment. The selector update uses a grouped verifier baseline,

$$\widehat{g}_{\mathrm{W0}} = \widehat{A}_b \nabla_\psi \log q_\psi(m_b \mid C_b), \tag{13}$$

where $\widehat{A}_b$ is a group-normalized verifier advantage. Logged selector samples may also be reused through bounded selector-level importance ratios $q_\psi(m \mid C)/q_{\mathrm{old}}(m \mid C)$ in fixed-context or contextual-bandit reuse; sequential off-policy reuse requires Monte-Carlo returns, trajectory importance sampling, a learned model, or a critic. Unlike OGPO-style full generative-policy optimization, no Bellman critic is required and no reverse denoising kernel is updated. The proof in Appendix A does not claim that this estimator alone proves convergence; it assumes a separate selector-level bandit, policy-gradient, or approximate-policy-iteration result supplies the final learning gap $\Delta_{\mathrm{learn}}$.

**Android-curated reasoning SFT corpus.** The Stage 1 and Stage 2 supervised phases are trained on a reasoning-augmented GUI corpus that we re-curate for the Android phone form factor, assembling CoT supervision, action trajectories, and element-grounding from public GUI-agent datasets plus in-house OpenMobile traces, and synthesizing additional teacher trajectories where ready-made reasoning traces are insufficient. The curation pass restricts every source to Android phone trajectories, normalizes them to the decoder's mobile UI action schema, keeps reasoning-dense steps, and reformats them into the dLLM action head's block-chunk decoding target, yielding an Android-specific mixture aligned with the deployed action space rather than a generic multi-platform corpus. Full dataset sources and curation steps are listed in Appendix C.

## 5 Empirical Validation Requirements

The formal claim in Appendix A is conditional: the transfer statement compares W0 with the best reachable frozen-mode policy only when the selector-learning, model-abstraction, and implementation gaps are small. The empirical section of a completed benchmark study therefore needs to instantiate three interfaces, rather than only report task success: **decoder speed**, **selector-only adaptation**, and **mode-abstraction validity**. Figures 1 and 2 serve as the paper's technical-report measurement summary: they show the current RTX 4060 desktop behavior and the block-size/curriculum signals that motivated the deployed decoder design. The same reporting template should be reused for Snapdragon 8 Gen 3 NPU measurements, where the static-shape denoising graph can be placed on the NPU while the GUI/VLM context path remains outside the block decoder.

**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.

**Finding: block size and structured-output integrity.** A block-size sweep on the 116-task AndroidWorld suite, decoding from a single training epoch (the $\ell_2$=0 Boltzmann special case of the degree-2 curriculum), shows that block-diffusion decoding is near-lossless up to a block size of four — bd1 (autoregressive), bd2, and bd4 reach the same task success — and degrades only gracefully through bd16, while raw structured-output validity stays high (strict-JSON validity $\geq 0.945$ through bd16). The degradation concentrates at the largest block size: at bd32 strict-JSON validity falls to $\approx 0.569$ (about 43% of raw outputs malformed), i.e., large-block parallel unmasking is where the action/JSON output

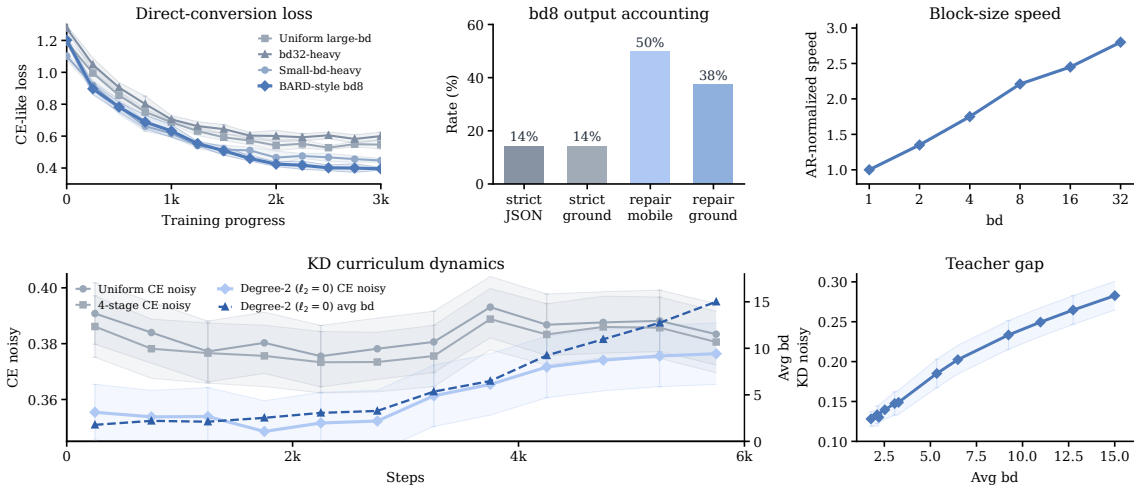



Figure 2: **Block-size and curriculum diagnostics.** The block-diffusion decoder is not only an inference optimization; it also changes the training variables exposed to the action head. The direct conversion panel tracks CE-like loss under different block-size schedules, the accounting panel separates strict JSON/mobile-use recovery from repaired outputs, and the speed panel shows the AR-normalized latency trend as block size increases. The lower panels summarize how the curriculum's block allocation moves probability mass from small anchor blocks toward larger blocks while monitoring noisy-branch CE and teacher-gap behavior. These diagnostics explain why W0 reports decoder speed, block size, and denoising-step budget together with task success.

format breaks. This is the same structured tool-call failure mode reported for diffusion-based agents (104); W0 encounters it as well. The block-size curriculum (Figure 3) responds to this by *including* $b$=32 only as a non-deployed training stress tail (for end-Gaussian symmetry) while concentrating the mode and clamping deployment and evaluation at the usable center $b^\star \approx 16$ — we never serve the broken block size. We therefore report decoder speed together with block size and strict-format validity rather than task success alone.

**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.

**Benchmark sources.** Mind2Web, WebArena, VisualWebArena, OSWorld, and AndroidWorld provide natural GUI-agent task families for these checks (23, 44, 67, 95, 106). AndroidWorld is especially suitable for RLVR because task success checkers and intermediate state predicates can provide sparse and dense verifier rewards without introducing a learned critic. The most relevant subsets are **hidden-state stress tasks**: delayed goal recall, multi-tab or multi-window state, form-progress memory, state toggles whose visual appearance is ambiguous, and mobile tasks in which intermediate predicates reveal progress before final success.

**Required baselines.** A complete comparison uses four matched policies at fixed verifier-call and compute budgets. **SFT** is the non-RL recipe (Stage 1 behavior-clones the action decoder, Stage 2 supervised-aligns the steering actuator; both frozen for evaluation). The two RL baselines are *critic-free, verifiable-reward* policy-gradient methods — the same regime as

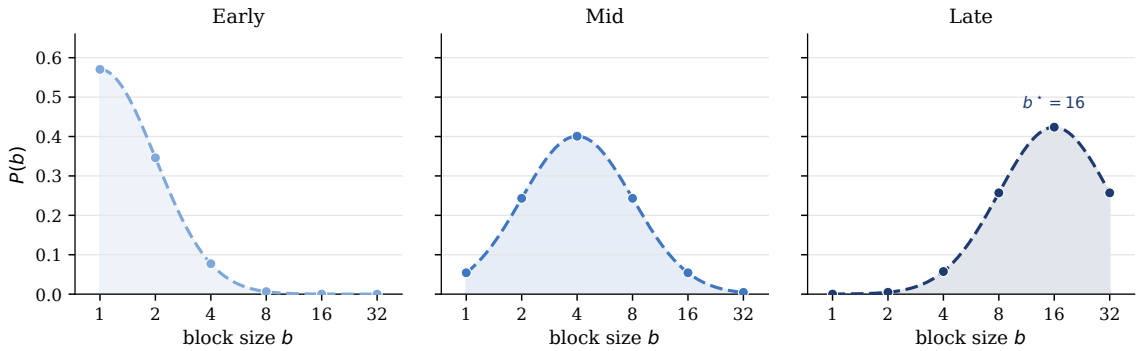



Figure 3: **Degree-2 (Gaussian-in-$\log b$) block-size curriculum.** Block sizes for the diffusion behavior-cloning branch are sampled from the degree-2 maximum-entropy law $P_t(b) \propto \exp(-\ell_1(t)\log b - \ell_2(t)(\log b)^2)$ over the support $b \in \{1, 2, 4, 8, 16, 32\}$; the Boltzmann power law is the $\ell_2$=0 boundary case. Equivalently this is a discretized log-normal in $b$, i.e. a Gaussian in the dyadic index $\log_2 b$. Across the early, mid, and late phases, markers give the on-grid allocation $P(b)$ at the six dyadic block sizes and the dashed curve is the underlying continuous Gaussian-in-$\log b$ density; both share one normalizer over the support, so the markers lie exactly on the curve (they are its on-grid samples). The allocation is a moving bump of *constant* spread ($\sigma_k$=1, one octave) whose center/mode advances from the autoregressive anchor $b$=1 toward the deploy/eval center $b^\star \approx 16$ (e.g. $1 \to 4 \to 16$); the spread does not narrow ($\ell_2$ fixed). The largest candidate $b$=32 is the symmetric stress tail that makes the end Gaussian $\{8, 16, 32\}$ symmetric and is never deployed.

W0 — applied to the *full* block-dVLM action policy: **GRPO**, which uses a group-relative advantage over $K$ sampled action chunks (73), and **RLOO**, which uses a leave-one-out baseline (1); both weight the diffusion action chunk's log-likelihood by the verifier-reward advantage and require no learned critic. Neither imposes a post-prefix mode restriction. W0 instead freezes the VLM, memory backbone, diffusion action decoder, and SFT-aligned actuator, applying the same critic-free verifiable-reward update to only the **post-prefix mode selector**. In all RLVR variants the outer objective is

$$J(\pi) = \mathbb{E}_\pi\left[\sum_{b\geq 0} \gamma^b r_b^{\text{ver}}\right],$$

but the score term differs: full-policy baselines use $\nabla \log \pi(y_b \mid \chi_b)$ or the corresponding reverse-kernel sum, while W0 uses only $\nabla \log q_\psi(m_b \mid C_b)$.

**Decoder speed.** For each completed outer action block, measure wall-clock decode time $t_i$ under a fixed hardware and decoding configuration. Report

$$\widehat{L}_{\text{mean}} = \frac{1}{N}\sum_{i=1}^{N} t_i, \qquad \widehat{L}_{95} = \text{Quantile}_{0.95}(t_{1:N}), \qquad \widehat{T}_{\text{chunk}} = \frac{N}{\sum_i t_i},$$

together with peak memory, block size, denoising-step budget, visual-token cap, and whether the decoder uses AR or block-parallel denoising.

**Required diagnostics.** Five estimators target the gaps in Appendix A: **mode stability** $\widehat{S}(C)$ and **verifier disagreement** $\widehat{D}_{\text{ver}}$ over $R$ frozen continuations, **prefix sufficiency** $\widehat{\Delta}_{\text{pref}}$, **critical-window sensitivity** $\widehat{\Delta}_{\text{win}}$, and **actuator leakage** ($\widehat{\Delta}^{J}_{\text{leak}}$ and a TV distance $\widehat{\epsilon}_{\text{leak}}$). Appendix D gives the estimator definitions.

## 6 Discussion and Limitations

W0 is intentionally more restricted than full generative-policy fine-tuning. This improves stability: the frozen VLM and frozen dLLM cannot suffer parameter-level forgetting, and RLVR can only reweight reachable continuation modes, up to measured actuator leakage. The same restriction defines the main limitation: if the frozen model contains no useful mode, if the denoise-prefix evidence is not predictive of reward, or if the SFT-aligned actuator leaks outside the mode abstraction, the conditional support-constrained transfer bound in Appendix A does not imply good task performance. W0 is therefore best understood as a support-constrained alignment method for pretrained GUI/VLA action experts: a selector over pretrained capabilities rather than a mechanism for creating skills absent from the frozen model. Its sample-efficiency claim is meaningful only when the validation quantities in Section 5 are favorable.

## 7 Conclusion

W0 combines a frozen screen-language backbone, bounded persistent memory, a discrete diffusion action head, and RLVR-trained post-prefix mode selection. It targets the regime where pretrained diffusion action experts already contain useful GUI behaviors but choose the wrong mode under long-horizon or out-of-distribution context. By training only a small selector over denoising continuation modes, the ideal W0 abstraction stays inside the frozen model's effective support, avoids parameter-level full-policy forgetting, and restricts optimization to the finite mode-codebook abstraction formalized in the appendix; the implemented system must still verify actuator leakage empirically.

## References


[1] Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. URL https://arxiv.org/abs/2402.14740. REINFORCE Leave-One-Out (RLOO), a critic-free baseline estimator.

[2] Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, Dylan Zinsley, James Zou, Atri Rudra, and Christopher Ré. Simple linear attention language models balance the recall–throughput tradeoff. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2402.18668.

[3] Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2503.09573. Oral.

[4] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. URL https://arxiv.org/abs/2107.03006.

[5] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

[6] Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[7] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning (ICML)*, pages 41–48, 2009.

[8] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[9] Giulio Biroli, Tony Bonnaire, Valentin de Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 2024.

[10] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/14f75513f0f1ca01de1e826b52e6b840-Abstract-Conference.html.

[11] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, et al. $\pi_0$: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems (RSS)*, 2025. URL https://www.roboticsproceedings.org/rss21/p010.pdf.

[12] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, Oxford, 2013. ISBN 9780199535255. doi: 10.1093/acprof:oso/9780199535255.001.0001.

[13] Andrea Burns, Deniz Arsan, Sanjna Agrawal, Ranjitha Kumar, Kate Saenko, and Bryan A. Plummer. A dataset for interactive vision-language navigation with unknown command feasibility. In *European Conference on Computer Vision (ECCV)*, pages 312–328, 2022. URL https://arxiv.org/abs/2202.02312.

[14] Yuxiang Chai, Siyuan Huang, Yazhe Niu, Han Xiao, Liang Liu, Guozhi Wang, Dingyu Zhang, Shuai Ren, and Hongsheng Li. AMEX: Android multi-annotation expo dataset for mobile GUI agents. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2138–2156, 2025. URL https://aclanthology.org/2025.findings-acl.110/.

[15] Baoyou Chen, Hanchen Xia, Peng Tu, Haojun Shi, Liwei Zhang, Weihao Yuan, and Siyu Zhu. BARD: Bridging autoregressive and diffusion vision-language models via highly efficient progressive block merging and stage-wise distillation. *arXiv preprint arXiv:2604.16514*, 2026. URL https://arxiv.org/abs/2604.16514.

[16] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. URL https://arxiv.org/abs/2106.01345.

[17] Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo, Yuan Yao, Yankai Lin, Zhiyuan Liu, and Maosong Sun. GUICourse: From general vision language model to versatile GUI agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21936–21959, 2025. URL https://aclanthology.org/2025.acl-long.1065/.

[18] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24185–24198, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chen_InternVL_Scaling_up_Vision_Foundation_Models_and_Aligning_for_Generic_CVPR_2024_paper.html.

[19] Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. Seeclick: Harnessing GUI grounding for advanced visual GUI agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 9313–9332, 2024. doi: 10.18653/v1/2024.acl-long.505. URL https://aclanthology.org/2024.acl-long.505/.

[20] Egor Cherepanov, Alexey Staroverov, Dmitry Yudin, Alexey K. Kovalev, and Aleksandr I. Panov. Recurrent action transformer with memory. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2306.09459.

[21] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

[22] Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2405.21060.

[23] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. URL https://arxiv.org/abs/2306.06070.

[24] Longxi Gao, Li Zhang, Shihe Wang, Pengzhi Gao, Wei Liu, Jian Luan, Shangguang Wang, Yuanchun Li, and Mengwei Xu. MobileViews: A million-scale and diverse mobile GUI dataset. *arXiv preprint arXiv:2409.14337*, 2024. URL https://arxiv.org/abs/2409.14337.

[25] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural Turing machines. *arXiv preprint arXiv:1410.5401*, 2014. URL https://arxiv.org/abs/1410.5401.

[26] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016. doi: 10.1038/nature20101.

[27] Jake Grigsby, Linxi Fan, and Yuke Zhu. AMAGO: Scalable in-context reinforcement learning for adaptive agents. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.09971.

[28] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning (ICML)*, 2019. URL https://arxiv.org/abs/1811.04551.

[29] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://arxiv.org/abs/1912.01603.

[30] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. URL https://arxiv.org/abs/2301.04104.

[31] Matthew Hausknecht and Peter Stone. Deep recurrent Q-learning for partially observable MDPs. In *AAAI Fall Symposium Series*, 2015. URL https://arxiv.org/abs/1507.06527.

[32] Elad Hazan, Kfir Yehuda Levy, and Shai Shalev-Shwartz. On graduated optimization for stochastic non-convex problems. In *International Conference on Machine Learning (ICML)*, 2016. URL https://arxiv.org/abs/1503.03712.

[33] Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an end-to-end web agent with large multimodal models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. URL https://arxiv.org/abs/2401.13919.

[34] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

[35] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxuan Zhang, Juanzi Li, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. CogAgent: A visual language model for GUI agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. URL https://arxiv.org/abs/2312.08914.

[36] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106(4): 620–630, 1957.

[37] Nan Jiang, Alex Kulesza, and Satinder Singh. Abstraction selection in model-based reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 179–188, 2015.

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

[39] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 267–274, 2002.

[40] Steven Kapturowski, Georg Ostrovski, John Quan, Rémi Munos, and Will Dabney. Recurrent experience replay in distributed reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://openreview.net/forum?id=r1lyTjAqYX.

[41] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning (ICML)*, 2020. URL https://arxiv.org/abs/2006.16236.

[42] Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49(2–3):209–232, 2002.

[43] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P. Foster, Pannag R. Sanketi, Quan Vuong, et al. OpenVLA: An open-source vision-language-action model. In *Proceedings of the 8th Conference on Robot Learning (CoRL)*, pages 2679–2713, 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

[44] Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In

*Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 881–905, 2024. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/.

[45] Hongxin Li, Jingran Su, Jingfan Chen, Zheng Ju, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. UIPro: Unleashing superior interaction capability for GUI agents. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1613–1623, 2025. URL https://arxiv.org/abs/2509.17328.

[46] Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI grounding for professional high-resolution computer use. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM)*, pages 8778–8786, 2025. URL https://arxiv.org/abs/2504.07981.

[47] Wei Li, William E. Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on UI control agents. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024. URL https://arxiv.org/abs/2406.03679.

[48] Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Liuao Pei, Tian Nian, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, Yao Mu, and Ping Luo. Discrete diffusion VLA: Bringing discrete diffusion to action decoding in vision-language-action policies. In *International Conference on Machine Learning (ICML)*, 2026. URL https://arxiv.org/abs/2508.20072.

[49] Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Stan Weixian Lei, Lijuan Wang, and Mike Zheng Shou. ShowUI: One vision-language-action model for GUI visual agent. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19498–19508, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Lin_ShowUI_One_Vision-Language-Action_Model_for_GUI_Visual_Agent_CVPR_2025_paper.html.

[50] Yuhang Liu, Pengxiang Li, Congkai Xie, Xavier Hu, Xiaotian Han, Shengyu Zhang, Hongxia Yang, and Fei Wu. InfiGUI-R1: Advancing multimodal GUI agents from reactive actors to deliberative reasoners. *arXiv preprint arXiv:2504.14239*, 2025. URL https://arxiv.org/abs/2504.14239.

[51] Zhaoyang Liu, Jingjing Xie, Zichen Ding, Zehao Li, Bowen Yang, Zhenyu Wu, Xuehui Wang, Qiushi Sun, Shi Liu, Weiyun Wang, et al. ScaleCUA: Scaling open-source computer use agents with cross-platform data. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2509.15221.

[52] Quanfeng Lu, Wenqi Shao, Zitao Liu, Lingxiao Du, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, and Ping Luo. GUIOdyssey: A comprehensive dataset for cross-app GUI navigation on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22404–22414, 2025. URL https://arxiv.org/abs/2406.08451.

[53] Run Luo, Lu Wang, Wanwei He, Longze Chen, Jiaming Li, and Xiaobo Xia. GUI-R1: A generalist R1-style vision-language action model for GUI agents. *arXiv preprint arXiv:2504.10458*, 2025. URL https://arxiv.org/abs/2504.10458.

[54] Lingchen Meng, Jianwei Yang, Rui Tian, Xiyang Dai, Zuxuan Wu, Jianfeng Gao, and Yu-Gang Jiang. DeepStack: Deeply stacking visual tokens is surprisingly simple and effective for LMMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. doi: 10.52202/079017-0739. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/29cd7f8331d13ede6dc6d6ef3dfacb70-Abstract-Conference.html.

[55] Rémi Munos. Error bounds for approximate policy iteration. In *International Conference on Machine Learning (ICML)*, pages 560–567, 2003.

[56] Rémi Munos. Performance bounds in $L_p$-norm for approximate value iteration. *SIAM Journal on Control and Optimization*, 46(2):541–561, 2007.

[57] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2502.09992.

[58] Mang Ning, Mingxiao Li, Jianlin Su, Haozhe Jia, Lanmiao Liu, Martin Beneš, Wenshuo Chen, Albert Ali Salah, and Itir Onal Ertugrul. DCTdiff: Intriguing properties of image generative modeling in the DCT space. In *International Conference on Machine Learning (ICML)*, pages 46498–46524, 2025. URL https://proceedings.mlr.press/v267/ning25c.html.

[59] OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. URL https://arxiv.org/abs/2303.08774.

[60] Emilio Parisotto, H. Francis Song, Jack W. Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant M. Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, Matthew M. Botvinick, Nicolas Heess, and Raia Hadsell. Stabilizing transformers for reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2020. URL https://arxiv.org/abs/1910.06764.

[61] Sarvesh Patil, Mitsuhiko Nakamoto, Manan Agarwal, Shashwat Saxena, Jesse Zhang, Giri Anantharaman, Cleah Winston, Chaoyi Pan, Douglas Chen, Nai-Chieh Huang, Zeynep Temel, Oliver Kroemer, Sergey Levine, Abhishek Gupta, Hongkai Da, Paarth Shah, and Max Simchowitz. OGPO: Sample efficient full-finetuning of generative control policies. *arXiv preprint arXiv:2605.03065*, 2026. URL https://arxiv.org/abs/2605.03065.

[62] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, et al. FAST: Efficient action tokenization for vision-language-action models. In *Robotics: Science and Systems (RSS)*, 2025. URL https://www.roboticsproceedings.org/rss21/p012.pdf.

[63] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *Proceedings of the*

*9th Conference on Robot Learning (CoRL)*, pages 17–40, 2025. URL https://arxiv.org/abs/2504.16054.

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

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

[66] Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy P. Lillicrap. AndroidInTheWild: A large-scale dataset for android device control. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. URL https://arxiv.org/abs/2307.10088.

[67] Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Toyama, Robert Berry, Divya Tyamagundlu, Timothy Lillicrap, and Oriana Riva. AndroidWorld: A dynamic benchmarking environment for autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2405.14573.

[68] Gabriel Raya and Luca Ambrogioni. Spontaneous symmetry breaking in generative diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[69] Allen Z. Ren, Justin Lidard, Lars L. Ankile, Anthony Simeonov, Pulkit Agrawal, Anirudha Majumdar, Benjamin Burchfiel, Hongkai Dai, and Max Simchowitz. Diffusion policy policy optimization. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/c0749c39aaff9e9e4c91f7118bf21b1e-Abstract-Conference.html.

[70] Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander M. Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://arxiv.org/abs/2406.07524.

[71] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2202.00512.

[72] Bruno Scherrer, Mohammad Ghavamzadeh, Victor Gabillon, Boris Lesner, and Matthieu Geist. Approximate modified policy iteration and its application to the game of Tetris. *Journal of Machine Learning Research*, 16(49):1629–1676, 2015.

[73] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint*

*arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300. introduces Group Relative Policy Optimization (GRPO), a critic-free policy-gradient method.

[74] Wenxuan Song, Jiayi Chen, Shuai Chen, Jingbo Wang, Pengxiang Ding, Han Zhao, Yikai Qin, Xinhu Zheng, Donglin Wang, Yan Wang, and Haoang Li. Fast-dVLA: Accelerating discrete diffusion VLA to real-time performance. *arXiv preprint arXiv:2603.25661*, 2026. URL https://arxiv.org/abs/2603.25661.

[75] Liangtai Sun, Xingyu Chen, Lu Chen, Tianle Dai, Zichen Zhu, and Kai Yu. META-GUI: Towards multi-modal conversational agents on mobile GUI. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6699–6712, 2022. URL https://aclanthology.org/2022.emnlp-main.449/.

[76] Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, and Zhiyong Wu. OS-Genesis: Automating GUI agent trajectory construction via reverse task synthesis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5555–5579, 2025. URL https://aclanthology.org/2025.acl-long.277/.

[77] Sagar Gubbi Venkatesh, Partha Talukdar, and Srini Narayanan. UGIF: UI grounded instruction following. *arXiv preprint arXiv:2211.07615*, 2022. URL https://arxiv.org/abs/2211.07615.

[78] Andrew Wagenmaker, Yunchu Zhang, Mitsuhiko Nakamoto, Seohong Park, Waleed Yagoub, Anusha Nagabandi, Abhishek Gupta, and Sergey Levine. Steering your diffusion policy with latent space reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2025. URL https://arxiv.org/abs/2506.15799.

[79] Binxu Wang and Cengiz Pehlevan. An analytical theory of spectral bias in the learning dynamics of diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2503.03206.

[80] Haoming Wang, Haoyang Zou, Huatong Song, Jiazhan Feng, Junjie Fang, Junting Lu, Longxiang Liu, Qinyu Luo, Shihao Liang, Shijue Huang, Wanjun Zhong, Yining Ye, Yujia Qin, et al. UI-TARS-2 technical report: Advancing GUI agent with multi-turn reinforcement learning. *arXiv preprint arXiv:2509.02544*, 2025. URL https://arxiv.org/abs/2509.02544.

[81] Junyang Wang, Haiyang Xu, Haitao Jia, Xi Zhang, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-Agent-v2: Mobile device operation assistant with effective navigation via multi-agent collaboration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://arxiv.org/abs/2406.01014.

[82] Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-Agent: Autonomous multi-modal mobile device agent with visual perception. *arXiv preprint arXiv:2401.16158*, 2024. URL https://arxiv.org/abs/2401.16158.

[83] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, et al. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[84] Xinyuan Wang, Bowen Wang, Dunjie Lu, Junlin Yang, Tianbao Xie, Junli Wang, Jiaqi Deng, Xiaole Guo, Yiheng Xu, Chen Henry Wu, et al. OpenCUA: Open foundations for computer-use agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2508.09123.

[85] Xu Wang, Chenkai Xu, Yijie Jin, Jiachun Jin, Hao Zhang, and Zhijie Deng. Diffusion LLMs can do faster-than-AR inference via discrete diffusion forcing. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/pdf/8c334f769df9e5a9bf4b042fd7eba800553f48d3.pdf.

[86] Zhenhailong Wang, Haiyang Xu, Junyang Wang, Xi Zhang, Ming Yan, Ji Zhang, Fei Huang, and Heng Ji. Mobile-Agent-E: Self-evolving mobile assistant for complex tasks. *arXiv preprint arXiv:2501.11733*, 2025. URL https://arxiv.org/abs/2501.11733.

[87] Greg Wayne, Chia-Chun Hung, David Amos, Mehdi Mirza, Arun Ahuja, Agnieszka Grabska-Barwińska, Jack Rae, Piotr Mirowski, Joel Z. Leibo, Adam Santoro, Mevlana Gemici, Malcolm Reynolds, Tim Harley, Josh Abramson, Shakir Mohamed, Danilo Rezende, David Saxton, Adam Cain, Chloe Hillier, David Silver, Koray Kavukcuoglu, Matt Botvinick, Demis Hassabis, and Timothy Lillicrap. Unsupervised predictive memory in a goal-directed agent. *arXiv preprint arXiv:1803.10760*, 2018. URL https://arxiv.org/abs/1803.10760.

[88] Daphna Weinshall and Dan Amir. Theory of curriculum learning, with convex loss functions. *Journal of Machine Learning Research*, 21(222):1–19, 2020. URL https://arxiv.org/abs/1812.03472.

[89] Daphna Weinshall, Gad Cohen, and Dan Amir. Curriculum learning by transfer learning: Theory and experiments with deep networks. In *International Conference on Machine Learning (ICML)*, pages 5238–5246, 2018. URL https://arxiv.org/abs/1802.03796.

[90] Chengyue Wu, Shiyi Lan, Yonggan Fu, Sensen Gao, Jin Wang, Jincheng Yu, Jose M. Alvarez, Pavlo Molchanov, Ping Luo, Song Han, Ligeng Zhu, and Enze Xie. Fast-dVLM: Efficient block-diffusion VLM via direct conversion from autoregressive VLM. *arXiv preprint arXiv:2604.06832*, 2026. URL https://arxiv.org/abs/2604.06832.

[91] Chengyue Wu, Hao Zhang, Shuchen Xue, Shizhe Diao, Yonggan Fu, Zhijian Liu, Pavlo Molchanov, Ping Luo, Song Han, and Enze Xie. Fast-dLLM v2: Efficient block-diffusion LLM. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=1NZ3DHF9nT.

[92] Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=3Z3Is6hnOT.

[93] Qinzhuo Wu, Weikai Xu, Wei Liu, Tao Tan, Jianfeng Liu, Ang Li, Jian Luan, Bin Wang, and Shuo Shang. MobileVLM: A vision-language model for better intra- and inter-UI understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10231–10251, 2024. URL https://aclanthology.org/2024.findings-emnlp.599/.

[94] Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. OS-Atlas: A foundation action model for generalist GUI agents. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/0faa4bc5f522076947a030273629d4fe-Abstract-Conference.html.

[95] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5d413e48f84dc61244b6be550f1cd8f5-Abstract-Datasets_and_Benchmarks_Track.html.

[96] Haiyang Xu, Xi Zhang, Haowei Liu, Junyang Wang, Zhaozai Zhu, Shengjie Zhou, Xuhao Hu, Feiyu Gao, Junjie Cao, Zihua Wang, Zhiyuan Chen, Jitong Liao, Qi Zheng, Jiahui Zeng, Ze Xu, Shuai Bai, Junyang Lin, Jingren Zhou, and Ming Yan. Mobile-agent-v3.5: Multi-platform fundamental GUI agents. *arXiv preprint arXiv:2602.16855*, 2026. URL https://arxiv.org/abs/2602.16855.

[97] Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lu, Tianbao Xie, Amrita Saha, Doyen Sahoo, Tao Yu, and Caiming Xiong. Aguvis: Unified pure vision agents for autonomous GUI interaction. In *International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2412.04454.

[98] Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *Communications in Computational Physics*, 28(5):1746–1767, 2020. URL https://arxiv.org/abs/1901.06523.

[99] Rui Yang, Qianhui Wu, Zhaoyang Wang, Hanyang Chen, Ke Yang, Hao Cheng, Huaxiu Yao, Baolin Peng, Huan Zhang, Jianfeng Gao, and Tong Zhang. GUI-Libra: Training native GUI agents to reason and act with action-aware supervision and partially verifiable RL. *arXiv preprint arXiv:2602.22190*, 2026. URL https://arxiv.org/abs/2602.22190.

[100] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2210.03629.

[101] Jiabo Ye, Xi Zhang, Haiyang Xu, Haowei Liu, Junyang Wang, Zhaoqing Zhu, Ziwei Zheng, Feiyu Gao, Junjie Cao, Zhengxi Lu, Jitong Liao, Qi Zheng, Fei Huang, Jingren Zhou, and Ming Yan. Mobile-agent-v3: Fundamental agents for GUI automation. *arXiv preprint arXiv:2508.15144*, 2025. URL https://arxiv.org/abs/2508.15144.

[102] Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. AppAgent: Multimodal agents as smartphone users. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 70:1–70:20, 2025. URL https://arxiv.org/abs/2312.13771.

[103] Jiwen Zhang, Jihao Wu, Yihua Teng, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. Android in the zoo: Chain-of-action-thought for GUI agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12016–12031, 2024. URL https://aclanthology.org/2024.findings-emnlp.702/.

[104] Huiling Zhen, Weizhe Lin, Renxi Liu, Kai Han, Yiming Li, Yuchuan Tian, Hanting Chen, Xiaoguang Li, Xiaosong Li, Chen Chen, Xianzhi Yu, Mingxuan Yuan, Youliang Yan, Peifeng Qin, Jun Wang, Yu Wang, Dacheng Tao, and Yunhe Wang. DLLM agent: See farther, run faster. *arXiv preprint arXiv:2602.07451*, 2026. URL https://arxiv.org/abs/2602.07451.

[105] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. GPT-4V(ision) is a generalist web agent, if grounded. In *International Conference on Machine Learning (ICML)*, 2024. URL https://arxiv.org/abs/2401.01614.

[106] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2307.13854.

[107] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of the 7th Conference on Robot Learning (CoRL)*, pages 2165–2183, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

# A Problem Statement and RLVR Mode-Selector Formalization

**Scope.** This appendix states the formal object used by W0. The claim is not global optimality over all possible GUI action strings. The claim is a **support-constrained performance-transfer statement**: if selector learning reaches a near-optimal abstract mode policy, then the implemented W0 policy competes with that frozen continuation-mode target up to explicit learning, modeling, and implementation gaps.

**Three levels of policy.** We keep three objects separate. The **outer GUI policy** emits one completed action chunk $y_b$. The **inner denoising chain** produces intermediate states $z_{b,K}, \dots, z_{b,0}$ that are not environment actions. The trainable **mode selector** $q_\psi(m_b \mid C_b)$ observes only prefix-measurable denoising evidence and selects a continuation mode for a fixed actuator.

**Proof organization.** Definitions and assumptions appear first. The proof section then gives the standard RL ingredients and wires them together: a Bellman contraction (Lemma A.7) and one-step approximate selector improvement (Proposition A.9), whose uniform estimation event is discharged from finite verifier samples by a Hoeffding/union-bound concentration lemma (Lemma A.8), are combined into an approximate-policy-iteration bound on the selector-learning gap (Corollary A.10); a one-step leakage bound (Lemma A.11) is lifted to a return-level implementation gap (Lemma A.12); a simulation-lemma argument (Lemma A.14) bounds the mode-abstraction slack $\Delta_{\text{model}}$ from named per-step primitives (Definition A.13); these feed the support-constrained transfer corollary (Corollary A.6), so every gap is derived from explicit primitives, leaving only the irreducible reward-model, transition-kernel, prefix-sufficiency, verifier-proxy, and optimization errors as assumed-small inputs, while the statistical error $\epsilon_{\text{est}}$ is itself driven to zero by sampling (Lemma A.8). A finite-cover comparison (Proposition A.15) bounds the selector policy-class size.

## A.1 Outer GUI/VLA MDP and Inner Denoising Policy

Let the outer interaction be a discounted MDP or a belief-MDP abstraction with bounded controller context $\chi_b = (o_b, \ell, \xi_{b-1})$ and environment action chunk $y_b \in \mathcal{Y}$. The **outer action** is this completed chunk, not an intermediate denoising state. The transition kernel on $\chi$ is an abstract Markov kernel induced by the memory update; Markov-sufficiency error is included in $\Delta_{\text{model}}$. A block-discrete diffusion action head emits one chunk only after an **inner denoising rollout**

$$\tau_b = (z_{b,K}, z_{b,K-1}, \dots, z_{b,0}), \qquad y_b = \text{decode}(z_{b,0}). \tag{14}$$

For $k > 0$, $z_{b,k-1}$ is an internal denoising update, not an environment action. A **full diffusion-policy RL** update would optimize the reverse-kernel likelihood

$$p_\phi(\tau_b \mid \chi_b) = p(z_{b,K}) \prod_{k=K}^{1} p_\phi(z_{b,k-1} \mid z_{b,k}, \chi_b, H_{b,k}), \tag{15}$$

with a policy-gradient estimator proportional to

$$\hat{A}_b \sum_{k=K}^{1} \nabla_\phi \log p_\phi(z_{b,k-1} \mid z_{b,k}, \chi_b, H_{b,k}). \tag{16}$$

W0 uses Eqs. (15)–(16) only as the reference policy object. It freezes the reverse kernels and trains a selector over their continuation modes.

### A.2 Post-Prefix Mode Selection

Choose a selection index $\kappa$. The **observed prefix window** and the **future injection window** are separated:

$$\mathcal{I}_{\text{obs}} \subseteq \{\kappa, \kappa+1, \ldots, K\}, \qquad \mathcal{I}_{\text{inj}} \subseteq \{0, 1, \ldots, \kappa-1\}. \tag{17}$$

The frozen backbone first samples a prefix

$$\tau_b^{\text{pre}} = (z_{b,K}, z_{b,K-1}, \ldots, z_{b,\kappa}) \sim P_{\theta_0}^{\text{pre}}(\cdot \mid \chi_b), \qquad C_b = (\chi_b, H_{b,\mathcal{I}_{\text{obs}}}). \tag{18}$$

The selector observes $C_b$ and samples a **runtime command** $m_b \sim q_\psi(\cdot \mid C_b)$. For the ideal abstract library, let $\bar{m}_b = \iota_{C_b}(m_b)$ be the corresponding **frozen basin label**. The calibration map $\iota_C$ is defined in the frozen-library definition below. The **ideal frozen continuation** is

$$\tau_b^{\text{post}} = (z_{b,\kappa-1}, \ldots, z_{b,0}) \sim \bar{P}_{\theta_0}^{\text{post}}(\cdot \mid \chi_b, \tau_b^{\text{pre}}, \bar{m}_b). \tag{19}$$

Thus the selector is prefix-measurable. It does not observe $z_{b,0}$ or the verifier reward before choosing $m_b$. The implemented actuator-conditioned denoising kernel is compared with this ideal basin law through $\Delta_{\text{impl}}$.

**Definition A.1** (Step-wise continuation modes)**.** For each $k \in \mathcal{I}_{\text{inj}}$, let

$$\mathcal{C}_{C,k} = \{1, \ldots, M_k(C)\}$$

be a finite candidate set. A trajectory-level mode is an admissible path

$$m = (m_k)_{k \in \mathcal{I}_{\text{inj}}} \in \mathcal{M}_C \subseteq \prod_{k \in \mathcal{I}_{\text{inj}}} \mathcal{C}_{C,k}. \tag{20}$$

The subset $\mathcal{M}_C$ excludes impossible step combinations. A single-step selector is the special case $\mathcal{M}_C = \mathcal{C}_{C,k^\dagger}$.

**Definition A.2** (Frozen mode library)**.** Let $\bar{\mathcal{M}}_C$ be the set of post-hoc frozen basins, and let $\iota_C : \mathcal{M}_C \to \bar{\mathcal{M}}_C$ map a runtime selector command to its intended frozen basin. Let $\bar{\pi}_0(y \mid C, \bar{m})$ denote the frozen final-chunk law conditioned on basin $\bar{m}$. For compact notation define

$$\pi_0(y \mid C, m) := \bar{\pi}_0(y \mid C, \iota_C(m)).$$

The ideal W0 mixture is

$$\pi_{\text{id}}^q(y \mid C) = \sum_{m \in \mathcal{M}_C} q(m \mid C)\pi_0(y \mid C, m). \tag{21}$$

If $C$ records only the observed denoising prefix rather than the full latent denoising trajectory, then $\pi_0(y \mid C, m)$ is understood as the posterior average of the frozen basin law over latent prefixes consistent with $C$. For discrete neural policies, raw support can be nearly all strings, so the paper uses a thresholded effective support

$$\text{supp}_\epsilon(p) = \{y : \ p(y) \geq \epsilon\}, \qquad \epsilon > 0.$$

The ideal mixture stays inside the frozen model's effective mode support:

$$\text{supp}_\epsilon(\pi_{\text{id}}^q(\cdot \mid C)) \subseteq \bigcup_{m \in \mathcal{M}_C} \text{supp}_{\epsilon/|\mathcal{M}_C|}(\pi_0(\cdot \mid C, m)). \tag{22}$$

Indeed, if $\sum_{m \in \mathcal{M}_C} q(m \mid C)\pi_0(y \mid C, m) \geq \epsilon$, then at least one component satisfies $q(m \mid C)\pi_0(y \mid C, m) \geq \epsilon/|\mathcal{M}_C|$, hence $\pi_0(y \mid C, m) \geq \epsilon/|\mathcal{M}_C|$.

There are two mode objects. A frozen basin label $\bar{m}$ is a post-hoc label used to define the ideal library $\{\bar{\pi}_0(\cdot \mid C, \bar{m})\}$. A runtime command $m$ is sampled from $q_\psi(\cdot \mid C)$ before the post-prefix continuation is generated. The calibration map $\iota_C$ records which basin command $m$ is meant to invoke. The SFT-aligned actuator is assumed to make command $m$ approximate the corresponding frozen basin law; any mismatch between the implemented steered continuation and the ideal library is charged to $\Delta_{\text{impl}}$ below.

**Optimal-transport/RBF mode prototypes.** Offline, mode prototypes may be built from frozen denoising evidence. Let $E_0^C(\tau^{\text{post}})$ be hidden/key–value (KV) evidence from the post-prefix chain and choose prototypes $\mu_m(C)$. With cost $c(e, \mu_m)$ and prior weights $a_m(C)$, the entropic one-point OT selector is

$$q_\psi(\cdot \mid C) = \arg\min_{q \in \Delta(\mathcal{M}_C)} \left\{ \sum_m q_m c(e_\psi(C), \mu_m(C)) + \lambda \,\mathrm{KL}(q \,\|\, a(C)) \right\}, \tag{23}$$

equivalently

$$q_\psi(m \mid C) = \frac{a_m(C) \exp(-c(e_\psi(C), \mu_m(C))/\lambda)}{\sum_n a_n(C) \exp(-c(e_\psi(C), \mu_n(C))/\lambda)}. \tag{24}$$

With uniform $a_m \equiv a$ and squared cost this is an RBF selector, for which, if $m^\star$ has margin

$$\mathfrak{m}(C) = \min_{n \neq m^\star} \{c(e_\psi(C), \mu_n(C)) - c(e_\psi(C), \mu_{m^\star}(C))\} > 0, \tag{25}$$

then

$$1 - q_\psi(m^\star \mid C) \leq (|\mathcal{M}_C| - 1) \exp(-\mathfrak{m}(C)/\lambda). \tag{26}$$

This is a selector-confusion bound, not yet a bound on implemented residual-policy leakage.

### A.3 Steering Hook

One denoising step is written as an ordered computation to avoid an algebraic loop. First form the frozen pre-residual state

$$h_{b,k}^0 = F_{\theta_0}^{\text{pre}}(z_{b,k}, \chi_b). \tag{27}$$

Then the fixed SFT-aligned actuator produces

$$u_{b,k}^{(m_b)} = A_{\eta_{\text{SFT}}}(h_{b,k}^0, C_b, m_b), \qquad r_{b,k} = w_k P_0 u_{b,k}^{(m_b)}. \tag{28}$$

Finally,

$$(h_{b,k}, z_{b,k-1}) = F_{\theta_0}^{\text{post}}(z_{b,k}, \chi_b, r_{b,k}). \tag{29}$$

During RLVR, $\theta_0, \eta_{\text{SFT}}, P_0$, and the gate schedule $w_k$ are fixed; only $q_\psi$ is updated.

### A.4 Abstract Mode MDP

Let $\bar{R}(C, m)$ and $\bar{P}(\chi' \mid C, m)$ be the reward and next context kernel induced by selecting mode $m$ and then executing the frozen mode-conditioned final chunk. For selector $q$,

$$(T_q V)(\chi) = \mathbb{E}_{\tau^{\text{pre}}} \left[ \sum_{m \in \mathcal{M}_C} q(m \mid C) \left[ \bar{R}(C, m) + \gamma \mathbb{E}_{\chi' \sim \bar{P}(\cdot \mid C, m)} V(\chi') \right] \right], \tag{30}$$

and the optimal frozen-mode Bellman operator is

$$(T_{\mathcal{M}}V)(\chi) = \mathbb{E}_{\tau^{\mathrm{pre}}}\left[\max_{m\in\mathcal{M}_C}\left[\bar{R}(C,m) + \gamma\mathbb{E}_{\chi'\sim\bar{P}(\cdot|C,m)}V(\chi')\right]\right]. \tag{31}$$

There are two common specializations. In the discounted abstract MDP, $\bar{R}(C,m)$ is a one-step verifier reward and the future value is propagated by $\gamma\mathbb{E}_{\chi'}V(\chi')$. In verifier-only contextual-bandit reuse, set $\gamma = 0$ and replace $\bar{R}(C,m)$ by a return-level verifier estimate $G_{\mathrm{ver}}(C,m)$, so future reward is not counted a second time by the Bellman term.

**Assumption A.3** (Abstract mode MDP regularity)**.** Rewards are bounded, $|\bar{R}(C,m)| \le R_{\max}$; $\bar{P}$ is a valid Markov kernel on a Markov-sufficient abstraction $\chi$; the selector class maps each prefix context $C$ to a distribution over the finite set $\mathcal{M}_C$; and $0 \le \gamma < 1$ for the discounted case. Finite horizon tasks use the same definitions with time-indexed Bellman operators and backward induction.

Under this assumption, $T_q$ and $T_{\mathcal{M}}$ map bounded functions to bounded functions and are $\gamma$-contractions in the supremum norm (Lemma A.7); by the Banach fixed-point theorem each admits a unique fixed point. Let $V_q$ be the unique fixed point of $T_q$, and let $V^\star_{\mathcal{M}}$ be the unique fixed point of $T_{\mathcal{M}}$. For an initial context distribution $\mu_0$,

$$J_{\mathrm{mode}}(q) = \mathbb{E}_{\chi_0\sim\mu_0}[V_q(\chi_0)], \qquad J^\star_{\mathrm{mode}} = \mathbb{E}_{\chi_0\sim\mu_0}[V^\star_{\mathcal{M}}(\chi_0)]. \tag{32}$$

The ideal frozen-mode mixture return integrates over the frozen prefix distribution before evaluating the outer policy:

$$\bar{\pi}^q_{\mathrm{id}}(y \mid \chi) = \mathbb{E}_{\tau^{\mathrm{pre}}\sim P^{\mathrm{pre}}_{\theta_0}(\cdot|\chi)}\left[\pi^q_{\mathrm{id}}(y \mid C(\chi,\tau^{\mathrm{pre}}))\right], \qquad J_{\mathrm{ideal}}(q) = J(\bar{\pi}^q_{\mathrm{id}}). \tag{33}$$

At the realized prefix-context level,

$$\pi^q_{\mathrm{id}}(y \mid C) = \sum_{m\in\mathcal{M}_C} q(m \mid C)\pi_0(y \mid C, m).$$

In the verifier-only contextual-bandit specialization, replace $V_q$ by

$$J_{\mathrm{bandit}}(q) = \mathbb{E}_C\left[\sum_{m\in\mathcal{M}_C} q(m \mid C)G_{\mathrm{ver}}(C,m)\right], \qquad G_{\mathrm{ver}}(C,m) = \mathbb{E}_{y\sim\pi_0(\cdot|C,m)}[R(C,y)]. \tag{34}$$

**Assumption A.4** (Abstract mode optimization gap)**.** After $T$ selector-improvement rounds with the actuator fixed, the final selector satisfies

$$J_{\mathrm{mode}}(q_{\psi_T}) \ge J^\star_{\mathrm{mode}} - \Delta_{\mathrm{learn}}, \qquad \Delta_{\mathrm{learn}} \ge 0. \tag{35}$$

This gap is discharged by Corollary A.10, which iterates the one-step improvement of Proposition A.9 under the $\gamma$-contraction of Lemma A.7 into a standard approximate-policy-iteration bound; alternatively it may be supplied by any policy-gradient or bandit guarantee. It is not proved by the transfer corollary below.

**Assumption A.5** (Return-level transfer gaps)**.** For the final selector,

$$|J_{\text{ideal}}(q_{\psi_T}) - J_{\text{mode}}(q_{\psi_T})| \le \Delta_{\text{model}}, \qquad |J(\widetilde{\pi}_{\psi_T,\eta_{\text{SFT}}}) - J_{\text{ideal}}(q_{\psi_T})| \le \Delta_{\text{impl}}, \tag{36}$$

with $\Delta_{\text{model}}, \Delta_{\text{impl}} \ge 0$. Neither gap is left bare: the first is discharged by the model-abstraction bound of Lemma A.14, and the second by the return-level lifting of Lemma A.12. Concretely, with the named per-step primitives of Definition A.13,

$$\Delta_{\text{model}} \le \frac{\epsilon_R}{1-\gamma} + \frac{2\gamma R_{\max}\,\epsilon_P}{(1-\gamma)^2} + \epsilon_{\text{suff}}, \qquad \Delta_{\text{impl}} \le \frac{2R_{\max}\,\varepsilon_{\text{leak}}}{(1-\gamma)^2},$$

so the modeling slack is expressed through the explicit reward-model error $\epsilon_R$, transition-kernel error $\epsilon_P$, and prefix-sufficiency bias $\epsilon_{\text{suff}}$ rather than the informal triple $\epsilon_{\text{mode}} + \epsilon_{\text{suff}} + \epsilon_{\text{abs}}$. Of these, $\epsilon_R, \epsilon_P, \epsilon_{\text{suff}}$ (modeling), together with the verifier proxy-bias $\epsilon_{\text{verifier}}$ and the optimization slack $\epsilon_{\text{opt}}$, are the irreducible primitives: they are now named and quantified, but remain *assumed small* and cannot be estimated from selector data alone. By contrast the statistical estimation error $\epsilon_{\text{est}}(n,\delta)$ entering $\Delta_{\text{learn}}$ is *not* assumed but derived from finite samples by Lemma A.8, and vanishes as $n \to \infty$.

**Corollary A.6** (Support-constrained performance transfer)**.** *Under the abstract mode MDP regularity assumption and Eqs.* (35)–(36)*,*

$$J(\widetilde{\pi}_{\psi_T,\eta_{\text{SFT}}}) \ge J^{\star}_{\text{mode}} - \Delta_{\text{learn}} - \Delta_{\text{model}} - \Delta_{\text{impl}}. \tag{37}$$

### A.5 Proofs and Auxiliary Claims

**Lemma A.7** (**Bellman contraction**)**.** *Under Assumption A.3, for any selector $q$ and any bounded value functions $V, W$,*

$$\|T_q V - T_q W\|_\infty \le \gamma \|V - W\|_\infty, \qquad \|T_{\mathcal{M}} V - T_{\mathcal{M}} W\|_\infty \le \gamma \|V - W\|_\infty.$$

*Proof.* Let $\delta = \|V - W\|_\infty$ and define the one-step mode value

$$Q_V(C,m) = \bar{R}(C,m) + \gamma \mathbb{E}_{\bar{P}(\cdot|C,m)} V(\chi').$$

Then, for every prefix context $C$ and mode $m$,

$$|Q_V(C,m) - Q_W(C,m)| \le \gamma\delta \qquad \because\ \bar{R}(C,m) \text{ cancels, } \bar{P}(\cdot \mid C,m) \text{ unit-mass.}$$

Therefore,

$$|(T_q V)(\chi) - (T_q W)(\chi)| \le \mathbb{E}_{\tau^{\text{pre}}} \mathbb{E}_{m\sim q(\cdot|C)} |Q_V(C,m) - Q_W(C,m)| \le \gamma\delta,$$
$$|(T_{\mathcal{M}} V)(\chi) - (T_{\mathcal{M}} W)(\chi)| \le \mathbb{E}_{\tau^{\text{pre}}} \max_{m\in\mathcal{M}_C} |Q_V(C,m) - Q_W(C,m)| \le \gamma\delta,$$

where the second line uses that max is 1-Lipschitz under the supremum norm. Taking $\sup_\chi$ proves the claim. These are the standard sup-norm $\gamma$-contraction properties of (mode) Bellman operators (64, Sec. 6.2); see also Bertsekas and Tsitsiklis (8). □

**Lemma A.8** (**Uniform verifier-mean estimation (Monte-Carlo discharge of the estimation event)**)**.** *Consider the verifier-only contextual-bandit specialization ($\gamma = 0$), in which the action value of a mode is the return-level verifier mean*

$$Q(C,m) \;=\; G_{\text{ver}}(C,m) \;=\; \mathbb{E}_{y\sim\pi_0(\cdot|C,m)}\big[R(C,y)\big], \qquad |R(C,y)| \le R_{\max}\ \text{a.s.}$$

*Fix a finite context cover $\mathcal{C}$ with $|\mathcal{C}| < \infty$ and a per-context mode budget $\sup_{C\in\mathcal{C}} |\mathcal{M}_C| \le M_{\max}$. For each pair $(C, m)$ with $C \in \mathcal{C}$, $m \in \mathcal{M}_C$, draw $n$ i.i.d. verifier rollouts $y^{(1)}_{C,m}, \dots, y^{(n)}_{C,m} \sim \pi_0(\cdot \mid C, m)$ from the frozen actuator and form the empirical mean*

$$\widehat{Q}(C, m) \;=\; \frac{1}{n} \sum_{i=1}^{n} R\big(C, y^{(i)}_{C,m}\big).$$

*Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ the uniform estimation event holds:*

$$\sup_{C\in\mathcal{C}} \sup_{m\in\mathcal{M}_C} \big|\widehat{Q}(C, m) - Q(C, m)\big| \;\le\; \epsilon_{\text{est}}(n, \delta) \;:=\; R_{\max} \sqrt{\frac{2\,\ln\big(2|\mathcal{C}|\,M_{\max}/\delta\big)}{n}}.$$

*Equivalently, to guarantee $\epsilon_{\text{est}}(n, \delta) \le \epsilon$ at confidence $1 - \delta$ it suffices to take*

$$n \ge 2R^2_{\max}\, \epsilon^{-2}\, \ln\big(2|\mathcal{C}|M_{\max}/\delta\big),$$

*so the per-pair sample budget grows only logarithmically in the cover size and in $1/\delta$ and as $\epsilon^{-2}$ in the target accuracy.*

*Proof.* Fix a single pair $(C, m)$ with $C \in \mathcal{C}$ and $m \in \mathcal{M}_C$. The rollouts $y^{(1)}_{C,m}, \dots, y^{(n)}_{C,m}$ are i.i.d. from $\pi_0(\cdot \mid C, m)$, so the rewards $X_i := R(C, y^{(i)}_{C,m})$ are i.i.d., bounded in the interval $[-R_{\max}, R_{\max}]$ of width $b - a = 2R_{\max}$, with common mean $\mathbb{E}[X_i] = G_{\text{ver}}(C, m) = Q(C, m)$; here we use that the rollout law $\pi_0(\cdot \mid C, m)$ depends only on the frozen actuator and not on the selector $q$, so $\widehat{Q}(C, m)$ is an unbiased sample mean of $Q(C, m)$ and the argument is non-circular.

*Step 1 (per-pair two-sided Hoeffding).* Hoeffding's inequality (34) for the mean of $n$ independent variables with $X_i \in [a, b]$ gives, for any $t > 0$,

$$\Pr\big[\,\widehat{Q}(C, m) - Q(C, m) \ge t\,\big] \le \exp\Big(-\tfrac{2n^2t^2}{\sum_{i=1}^{n}(b-a)^2}\Big) = \exp\Big(-\tfrac{2n^2t^2}{n\,(2R_{\max})^2}\Big) = \exp\Big(-\tfrac{n\,t^2}{2R^2_{\max}}\Big),$$

and symmetrically for the lower tail; a union bound over the two tails yields

$$\Pr\big[\,|\widehat{Q}(C, m) - Q(C, m)| \ge t\,\big] \le 2\exp\Big(-\tfrac{n\,t^2}{2R^2_{\max}}\Big).$$

*Step 2 (union bound over the finite cover).* There are at most $N := |\mathcal{C}|\,M_{\max} \ge \sum_{C\in\mathcal{C}} |\mathcal{M}_C|$ distinct pairs $(C, m)$. By subadditivity of probability — a union bound over the finite family with the per-pair Hoeffding tail (12, §2.6) —

$$\begin{aligned}\Pr\Big[\sup_{C\in\mathcal{C}} \sup_{m\in\mathcal{M}_C} |\widehat{Q}(C, m) - Q(C, m)| \ge t\Big] \;&\le\; \sum_{C\in\mathcal{C}} \sum_{m\in\mathcal{M}_C} \Pr\big[|\widehat{Q}(C, m) - Q(C, m)| \ge t\big] \\ &\le\; 2N \exp\Big(-\tfrac{n\,t^2}{2R^2_{\max}}\Big).\end{aligned}$$

Independence *across* pairs is not needed: only the per-pair tail and subadditivity are used.

*Step 3 (invert at level $\delta$).* Setting the right-hand side equal to $\delta$ and solving for $t$,

$$2N \exp\Big(-\tfrac{n\,t^2}{2R^2_{\max}}\Big) = \delta \iff \tfrac{n\,t^2}{2R^2_{\max}} = \ln\tfrac{2N}{\delta} \iff t = R_{\max}\sqrt{\tfrac{2\,\ln(2N/\delta)}{n}}.$$

With $N = |\mathcal{C}|M_{\max}$ this is $\epsilon_{\text{est}}(n, \delta)$. Since the tail bound is monotone in $N$, replacing $\sum_C |\mathcal{M}_C|$ by its upper bound $|\mathcal{C}|M_{\max}$ only enlarges $\epsilon_{\text{est}}$, so the statement holds as written. Finally, requiring $\epsilon_{\text{est}}(n, \delta) \le \epsilon$ and squaring gives the sample-complexity bound $n \ge 2R^2_{\max}\epsilon^{-2} \ln(2|\mathcal{C}|M_{\max}/\delta)$, and $\epsilon_{\text{est}}(n, \delta) = O\big(R_{\max}\sqrt{\ln(|\mathcal{C}|M_{\max}/\delta)/n}\big) \to 0$ as $n \to \infty$. □

**Proposition A.9** (**One-step approximate selector improvement**). *Fix a selector $q$ and let*

$$Q^q(C,m) = \bar{R}(C,m) + \gamma \mathbb{E}_{\bar{P}(\cdot|C,m)} V_q(\chi').$$

*Suppose a new selector $q^+$ satisfies, for every prefix context $C$,*

$$|\widehat{Q}(C,m) - Q^q(C,m)| \le \epsilon_{\text{stat}}, \qquad \mathbb{E}_{m\sim q^+}\widehat{Q}(C,m) \ge \max_m \widehat{Q}(C,m) - \epsilon_{\text{opt}},$$

*where $\epsilon_{\text{stat}} = \epsilon_{\text{est}} + \epsilon_{\text{verifier}}$. In the verifier-only contextual-bandit specialization the statistical slot $\epsilon_{\text{est}}$ is not assumed but* derived*: Lemma A.8 discharges this uniform estimation event with probability at least $1-\delta$ at $\epsilon_{\text{est}}(n,\delta) = R_{\max}\sqrt{2\ln(2|\mathcal{C}|M_{\max}/\delta)/n}$ from $n$ i.i.d. verifier rollouts per $(C,m)$, so only the verifier proxy-bias $\epsilon_{\text{verifier}}$ remains assumed-small. Then, with*

$$\epsilon_{\text{imp}} = 2\epsilon_{\text{stat}} + \epsilon_{\text{opt}},$$

*we have*

$$T_{q^+}V_q \ge T_{\mathcal{M}}V_q - \epsilon_{\text{imp}} \quad \textit{and} \quad V_{q^+} \ge V_q - \frac{\epsilon_{\text{imp}}}{1-\gamma}.$$

*Proof.* For each $C$,

$$\begin{aligned}\mathbb{E}_{m\sim q^+}Q^q(C,m) &\ge \mathbb{E}_{m\sim q^+}\widehat{Q}(C,m) - \epsilon_{\text{stat}}\\ &\ge \max_m \widehat{Q}(C,m) - \epsilon_{\text{opt}} - \epsilon_{\text{stat}}\\ &\ge \max_m Q^q(C,m) - \epsilon_{\text{imp}}.\end{aligned}$$

Taking $\mathbb{E}_{\tau^{\text{pre}}}[\cdot \mid \chi]$ gives

$$(T_{q^+}V_q)(\chi) \ge (T_{\mathcal{M}}V_q)(\chi) - \epsilon_{\text{imp}} \ge (T_qV_q)(\chi) - \epsilon_{\text{imp}} = V_q(\chi) - \epsilon_{\text{imp}}.$$

Let $d = \sup_\chi(V_q(\chi) - V_{q^+}(\chi))$. By monotonicity and discounted constant-shift equivariance of $T_{q^+}$,

$$\begin{aligned}V_q(\chi) &\le (T_{q^+}V_q)(\chi) + \epsilon_{\text{imp}}\\ &\le (T_{q^+}V_{q^+})(\chi) + \gamma d + \epsilon_{\text{imp}} = V_{q^+}(\chi) + \gamma d + \epsilon_{\text{imp}}.\end{aligned}$$

Thus $d \le \gamma d + \epsilon_{\text{imp}}$, so $V_{q^+} \ge V_q - \epsilon_{\text{imp}}/(1-\gamma)$. □

**Corollary A.10** (**Selector-iteration learning gap**). *Apply Proposition A.9 for $T$ rounds, each with per-round error $\epsilon_{\text{imp}} = 2\epsilon_{\text{stat}} + \epsilon_{\text{opt}}$, starting from any bounded initial selector and, in the verifier-only specialization, on the probability-$(1-\delta)$ uniform estimation event of Lemma A.8 (so that $\epsilon_{\text{stat}} = \epsilon_{\text{est}}(n,\delta) + \epsilon_{\text{verifier}}$). For $\gamma > 0$ the single $(1-\delta)$ estimation event of Lemma A.8 no longer suffices—the action value depends on the round through $V_q$—so there $\epsilon_{\text{stat}}$ is treated as an assumed per-round estimation error rather than discharged. Then the optimization gap in Eq. (35) obeys*

$$\Delta_{\text{learn}} = J^\star_{\text{mode}} - J_{\text{mode}}(q_{\psi_T}) \;\le\; \underbrace{\frac{2\gamma\,\epsilon_{\text{imp}}}{(1-\gamma)^2}}_{\textit{amplified per-round error}} + \underbrace{\gamma^T \frac{2R_{\max}}{1-\gamma}}_{\textit{initialization decay}}.$$

*Proof.* The first term is the standard supremum-norm approximate-policy-iteration error-accumulation bound (8, Prop. 6.2), instantiated with the per-round Bellman error $\epsilon_{\text{imp}}$ supplied by Proposition A.9; the second is the geometric decay of the initial value error,

which holds because $T_q$ is a $\gamma$-contraction (Lemma A.7) and $|V| \le R_{\max}/(1-\gamma)$. If the per-round error is controlled only in a weighted $L_2$ (regression) sense, the constant takes the concentrability form $\frac{2\gamma}{(1-\gamma)^2}\sqrt{C(\nu)}\,\epsilon_{\text{imp}}$ (55, 56, 72). Either way $\Delta_{\text{learn}}$ is bounded by the per-round selector error rather than assumed. □

*Proof of Corollary A.6.* The three nonnegative gaps are return-level quantities:

$$J_{\text{mode}}(q_{\psi_T}) \ge J^\star_{\text{mode}} - \Delta_{\text{learn}},$$
$$|J_{\text{ideal}}(q_{\psi_T}) - J_{\text{mode}}(q_{\psi_T})| \le \Delta_{\text{model}},$$
$$|J(\widetilde{\pi}_{\psi_T,\eta_{\text{SFT}}}) - J_{\text{ideal}}(q_{\psi_T})| \le \Delta_{\text{impl}}.$$

Chaining these inequalities yields

$$\begin{aligned}
\underbrace{J(\widetilde{\pi}_{\psi_T,\eta_{\text{SFT}}})}_{\text{implemented W0}} &\ge \underbrace{J_{\text{ideal}}(q_{\psi_T})}_{\text{ideal frozen-mode mixture}} - \Delta_{\text{impl}} \\
&\ge \underbrace{J_{\text{mode}}(q_{\psi_T})}_{\text{abstract mode MDP}} - \Delta_{\text{model}} - \Delta_{\text{impl}} \\
&\ge \underbrace{J^\star_{\text{mode}}}_{\text{best reachable frozen-mode policy}} - \Delta_{\text{learn}} - \Delta_{\text{model}} - \Delta_{\text{impl}}.
\end{aligned}$$

□

**Lemma A.11 (One-step leakage bound).** *For any two one-step action distributions $p, q$ and any bounded reward with $|r(a)| \le R_{\max}$,*

$$|\mathbb{E}_p r - \mathbb{E}_q r| \le 2R_{\max}\,\mathrm{TV}(p,q).$$

*Proof.*

$$\begin{aligned}
|\mathbb{E}_p r - \mathbb{E}_q r| &= \left|\sum_a (p(a)-q(a))r(a)\right| \\
&\le R_{\max}\sum_a |p(a)-q(a)| = 2R_{\max}\,\mathrm{TV}(p,q).
\end{aligned}$$

□

**Lemma A.12 (Return-level lifting).** *Let $\pi, \pi'$ be policies on the discounted abstract mode MDP with $|\bar{R}| \le R_{\max}$ and per-context total-variation gap $\sup_\chi \mathrm{TV}\big(\pi(\cdot \mid \chi), \pi'(\cdot \mid \chi)\big) \le \varepsilon$. Then*

$$|J(\pi) - J(\pi')| \;\le\; \frac{2R_{\max}}{(1-\gamma)^2}\,\varepsilon.$$

*In particular, if the implemented actuator differs from the ideal frozen-mode mixture by at most $\varepsilon_{\text{leak}}$ in per-context TV, then $\Delta_{\text{impl}} \le 2R_{\max}\varepsilon_{\text{leak}}/(1-\gamma)^2$; in the verifier-only contextual-bandit specialization ($\gamma = 0$), $\Delta_{\text{impl}}$ reduces to the one-step bound of Lemma A.11.*

*Proof.* The performance-difference lemma (39) writes $J(\pi) - J(\pi')$ as the $\pi$-discounted-occupancy average of the advantage $\mathbb{E}_{a\sim\pi}A^{\pi'}(\chi,a)$. Bounding the per-context term by $2\|Q^{\pi'}\|_\infty\,\mathrm{TV}(\pi,\pi') \le \frac{2R_{\max}}{1-\gamma}\varepsilon$ (via Lemma A.11 with $|Q^{\pi'}| \le R_{\max}/(1-\gamma)$) and the occupancy normalization by $1/(1-\gamma)$ yields the stated constant $2R_{\max}/(1-\gamma)^2$. □

**Definition A.13** (**Model-abstraction primitives**)**.** The abstract mode MDP $(\bar{R}, \bar{P})$ optimized by the selector is an **approximate model** of the ideal frozen-mode dynamics $(R_{\text{true}}, P_{\text{true}})$ on the same context space, mode sets $\mathcal{M}_C$, and discount $\gamma$. Here $R_{\text{true}}(C, m)$ and $P_{\text{true}}(\cdot \mid C, m)$ denote the reward and next-context law *actually induced* by issuing command $m$ and running the ideal frozen continuation $\bar{\pi}_0(\cdot \mid C, \iota_C(m))$, so that $V^q_{\text{true}}$ is the fixed point of the true mode Bellman operator and $J_{\text{suff}}(q) = \mathbb{E}_{\chi_0 \sim \mu_0}[V^q_{\text{true}}(\chi_0)]$ is the return of that Markov-sufficient true-dynamics MDP (distinct in general from $J_{\text{ideal}}(q) = J(\bar{\pi}^q_{\text{id}})$ of Eq. (33); their gap is exactly $\epsilon_{\text{suff}}$ below). Define three per-step primitives:

$$\epsilon_R = \sup_{C,\, m \in \mathcal{M}_C} \left|\bar{R}(C, m) - R_{\text{true}}(C, m)\right| \qquad \text{(reward-model error)}, \tag{38}$$

$$\epsilon_P = \sup_{C,\, m \in \mathcal{M}_C} \mathrm{TV}\big(\bar{P}(\cdot \mid C, m),\, P_{\text{true}}(\cdot \mid C, m)\big) \qquad \text{(transition-kernel error)}, \tag{39}$$

$$\epsilon_{\text{suff}} = \left|J_{\text{ideal}}(q) - J_{\text{suff}}(q)\right| \qquad \text{(prefix-sufficiency bias)}. \tag{40}$$

In (40), $J_{\text{suff}}(q)$ is the return of the $(R_{\text{true}}, P_{\text{true}})$ MDP whose state is a Markov-sufficient latent denoising prefix, while $J_{\text{ideal}}(q)$ uses the posterior-averaged law $\pi_0(\cdot \mid C, m)$ obtained when $C$ records only the observed prefix window $\mathcal{I}_{\text{obs}}$ (Eq. (17)); thus $\epsilon_{\text{suff}}$ quantifies the bias introduced by replacing the latent state with the prefix-measurable statistic $C$ and by the $\bar{P}$-Markov assumption of Assumption A.3.

**Lemma A.14** (**Model-abstraction bound (simulation lemma)**)**.** *Under Assumption A.3 and Definition A.13, for any selector $q$ the per-step reward/kernel errors propagate to a uniform value gap*

$$\left\|V^q - V^q_{\text{true}}\right\|_\infty \;\le\; \frac{\epsilon_R}{1-\gamma} \;+\; \frac{2\gamma R_{\max}\, \epsilon_P}{(1-\gamma)^2}, \tag{41}$$

*and therefore the model-abstraction slack of Eq. (36) obeys*

$$\Delta_{\text{model}} = \left|J_{\text{ideal}}(q) - J_{\text{mode}}(q)\right| \;\le\; \frac{\epsilon_R}{1-\gamma} \;+\; \frac{2\gamma R_{\max}\, \epsilon_P}{(1-\gamma)^2} \;+\; \epsilon_{\text{suff}}. \tag{42}$$

*In the verifier-only contextual-bandit specialization ($\gamma = 0$, $\bar{R} = G_{\text{ver}}$) the kernel term vanishes and*

$$\Delta_{\text{model}} \;\le\; \epsilon_R + \epsilon_{\text{suff}}. \tag{43}$$

*Proof.* Fix $q$ and write the model and true mode-$Q$ functions at the model value $V^q$ and true value $V^q_{\text{true}}$,

$$Q^q(C, m) = \bar{R}(C, m) + \gamma \mathbb{E}_{\bar{P}(\cdot \mid C, m)} V^q, \qquad Q^q_{\text{true}}(C, m) = R_{\text{true}}(C, m) + \gamma \mathbb{E}_{P_{\text{true}}(\cdot \mid C, m)} V^q_{\text{true}}.$$

Let $\Delta_V = \|V^q - V^q_{\text{true}}\|_\infty$. Decompose the kernel difference at the common test function $V^q$,

$$\mathbb{E}_{\bar{P}} V^q - \mathbb{E}_{P_{\text{true}}} V^q_{\text{true}} = \underbrace{\big(\mathbb{E}_{\bar{P}} - \mathbb{E}_{P_{\text{true}}}\big) V^q}_{\text{kernel}} \;+\; \underbrace{\mathbb{E}_{P_{\text{true}}}\big(V^q - V^q_{\text{true}}\big)}_{\le \Delta_V}.$$

For the kernel term, since $\bar{P}$ and $P_{\text{true}}$ are probability measures their difference integrates to zero, so subtracting the midrange constant $c = \frac{1}{2}(\sup V^q + \inf V^q)$ and using $\sum_{\chi'} |\bar{P} - P_{\text{true}}| = 2\ \mathrm{TV}(\bar{P}, P_{\text{true}})$ gives

$$\left|\big(\mathbb{E}_{\bar{P}} - \mathbb{E}_{P_{\text{true}}}\big) V^q\right| = \left|\big(\mathbb{E}_{\bar{P}} - \mathbb{E}_{P_{\text{true}}}\big)(V^q - c)\right| \le \tfrac{1}{2}\,\mathrm{span}(V^q) \cdot 2\ \mathrm{TV}(\bar{P}, P_{\text{true}}) \le \frac{2R_{\max}}{1-\gamma}\,\epsilon_P,$$

where $\mathrm{span}(V^q) = \sup V^q - \inf V^q \le 2R_{\max}/(1-\gamma)$ by Assumption A.3. Combining,

$$\left|Q^q(C,m) - Q^q_{\text{true}}(C,m)\right| \le \epsilon_R + \gamma\Big(\tfrac{2R_{\max}}{1-\gamma}\epsilon_P + \Delta_V\Big).$$

Because $V^q(\chi) = \mathbb{E}_{\tau^{\text{pre}}}\mathbb{E}_{m\sim q}Q^q(C,m)$ and likewise for the truth, taking $\mathbb{E}_{\tau^{\text{pre}}}\mathbb{E}_{m\sim q}$ and then $\sup_\chi$ preserves the bound (averaging is 1-Lipschitz in sup-norm), so

$$\Delta_V \le \epsilon_R + \frac{2\gamma R_{\max}}{1-\gamma}\epsilon_P + \gamma\Delta_V.$$

Solving the geometric recursion yields (41). Since $J_{\text{mode}}(q) = \mathbb{E}_{\mu_0}V^q$ and $J_{\text{suff}}(q) = \mathbb{E}_{\mu_0}V^q_{\text{true}}$, the expectation over $\mu_0$ is a 1-Lipschitz contraction of the sup-norm, giving $|J_{\text{mode}}(q) - J_{\text{suff}}(q)| \le \|V^q - V^q_{\text{true}}\|_\infty$. The triangle inequality with $|J_{\text{ideal}}(q) - J_{\text{suff}}(q)| \le \epsilon_{\text{suff}}$ then gives (42). For $\gamma = 0$ the recursion terminates at $\Delta_V \le \epsilon_R$ and the kernel term is absent, which is (43). This is the discounted simulation lemma (42) specialized to a fixed policy and a shared state–mode space; the additive $\epsilon_{\text{suff}}$ is the abstraction-selection bias that the simulation lemma does not remove because $C$ is a non-Markov statistic of the latent prefix (37). □

The two powers of $1/(1-\gamma)$ on the kernel term match the return-level lifting constant of Lemma A.12: a kernel/TV perturbation costs one factor $1/(1-\gamma)$ from the value magnitude $\|V^q\|_\infty \le R_{\max}/(1-\gamma)$ and a second from the geometric horizon sum, exactly as the per-context TV perturbation $\varepsilon_{\text{leak}}$ does in $\Delta_{\text{impl}}$. The reward term carries a single $1/(1-\gamma)$ because it is a direct, non-propagated perturbation. The factor 2 on the kernel term is the worst-case-tight span constant ($\sum|\bar{P} - P_{\text{true}}| = 2$ TV); it is unavoidable under the TV convention (39) and is the same factor 2 that appears in Lemmas A.11–A.12.

**Proposition A.15** (**Finite-cover mode-class comparison**). *Let $\mathcal{B}$ be a finite cover of a full action/residual policy class for one context, and let $\mathcal{C}$ be a finite context cover. If*

$$|\mathcal{B}| = g^{D_{\text{act}}}, \qquad \sup_{C\in\mathcal{C}}|\mathcal{M}_C| \le g^{D_{\text{mode}}}, \qquad 1 < g,\ D_{\text{mode}} < D_{\text{act}},$$

*then each context has fewer mode choices than the full cover. For deterministic selectors over $\mathcal{C}$ (or for finite covers of stochastic selectors with the same cardinality bound), the induced selector-policy cover is smaller:*

$$|\Pi_{\text{W0}}| \le \prod_{C\in\mathcal{C}}|\mathcal{M}_C| \le g^{D_{\text{mode}}|\mathcal{C}|} < g^{D_{\text{act}}|\mathcal{C}|} = |\mathcal{B}|^{|\mathcal{C}|} = |\Pi_{\text{base}}|.$$

*Proof.* Since $g > 1$ and $D_{\text{mode}} < D_{\text{act}}$, for every $C \in \mathcal{C}$,

$$|\mathcal{M}_C| \le g^{D_{\text{mode}}} < g^{D_{\text{act}}} = |\mathcal{B}|.$$

Multiplying over the finite context cover gives the displayed bound. □

This cardinality claim does not prove optimization by itself; it is a **finite-cover comparison** showing that W0 ranges over a strictly smaller mode-codebook policy class under the stated dimension assumptions.

## B Block-Diffusion Training and Diagnostic Figures

This appendix collects the diagnostic views behind Figure 2. They are included to make the technical-report narrative auditable: block size affects both decode latency and the noisy-branch training terms, so W0 reports scheduling, output-format recovery, and variance behavior rather than only a final success number.

### B.1 Block-Size Sampling Law (Degree-2 Gaussian-in-$\log b$ Curriculum)

The Stage-1 curriculum samples the block size $b$ from a finite ordered grid $S = \{b_1 < \cdots < b_m\} \subset \mathbb{R}_{>0}$ (here $S = \{1, 2, 4, 8, 16, 32\}$), writing $g(b) = \log b$. Three things matter, in order: *how* the sampling law is derived (maximum entropy / Lagrange multipliers), *why* the degree-2 Gaussian-in-$\log b$ family is the justified choice, and *why* the deployed schedule has its shape. Its $\ell_2$=0 base case is the Boltzmann law $P_\lambda(b) \propto b^{-\lambda(t)}$ of Section 4.6 used by this report's measurement snapshot. Every claim here is about the *sampling distribution* only, never the behavior-cloning objective.

**How the law is derived: max-entropy moments, with the exponents as the dual multipliers.** *The sampling law is the maximum-entropy distribution on $S$ that fixes prescribed moments of $\log b$, and its exponents are exactly the dual Lagrange multipliers; the base case fixes one moment.*

**Proposition B.1** (Form, direction, and naming of the block-size curriculum)**.** (i) Maximum-entropy form. *For any feasible mean $\mu \in (\log b_1, \log b_m)$, the unique maximizer of the Shannon entropy $H(q) = -\sum_{b\in S} q(b) \log q(b)$ over distributions $q$ on $S$ subject to $\sum_{b\in S} q(b)\, g(b) = \mu$ is the Gibbs law*

$$P_\lambda(b) = \frac{b^{-\lambda}}{Z(\lambda)}, \qquad Z(\lambda) = \sum_{b\in S} b^{-\lambda},$$

*with $\lambda = \lambda(\mu)$ the Lagrange multiplier dual to the mean constraint; equivalently $\{P_\lambda\}$ is the one-parameter exponential family with sufficient statistic $g(b) = \log b$ and natural parameter $-\lambda$.*

(ii) Monotone control of expected block size. *For any statistic $f$, $\frac{d}{d\lambda}\mathbb{E}_{P_\lambda}[f] = -\operatorname{Cov}_{P_\lambda}(f, g)$; with $f = \mathrm{id}$, comonotonicity of $b$ and $\log b$ gives $\frac{d}{d\lambda}\mathbb{E}_{P_\lambda}[b] = -\operatorname{Cov}_{P_\lambda}(b, \log b) < 0$, so $\mathbb{E}_{P_\lambda}[b]$ is a strictly decreasing bijection of $\lambda$ and the family is ordered by first-order stochastic dominance ($\lambda' < \lambda \Rightarrow P_{\lambda'} \succeq_{\mathrm{FOSD}} P_\lambda$, since the likelihood ratio $P_{\lambda'}(b)/P_\lambda(b) \propto b^{\lambda-\lambda'}$ is increasing in $b$). The anneal $\lambda : 1.5 \to -0.5$ thus drives $\mathbb{E}_{P_\lambda}[b]$ monotonically from $\approx 1.93$ to $\approx 16.54$ on $S$. Because the log-density is affine in $\log b$, the profile can only sharpen monotonically toward the largest block: it cannot raise the mean while retaining a controlled low/mid-block tail. This inability to set mean and spread independently is the representational limit lifted by the degree-2 family of Remark B.2.*

(iii) Tempering. *$P_\lambda(b) = \exp(-\lambda\, E(b))/Z(\lambda)$ with energy $E(b) = \log b$ is the Boltzmann measure at inverse temperature $\lambda$, so annealing $\lambda$ is temperature scheduling on the block-size latent.*

*Proof. (i)* Maximizing the strictly concave $H(q)$ over the compact convex feasible polytope (nonempty for $\mu \in (\log b_1, \log b_m)$) with multipliers $(\nu, -\lambda)$ for the normalization and mean constraints gives the stationarity condition $-\log q(b) - 1 + \nu - \lambda g(b) = 0$, i.e. $q(b) \propto e^{-\lambda g(b)} = b^{-\lambda}$; strict concavity makes the maximizer unique and of exponential-family form (21, 36).

*(ii)* $\log Z(\lambda)$ is the cumulant generating function of $-\lambda g$, so $\frac{d}{d\lambda}\mathbb{E}_{P_\lambda}[f] = -\operatorname{Cov}_{P_\lambda}(f, g)$; since $b$ and $\log b$ are both strictly increasing on $S$, their covariance under any non-degenerate $P_\lambda$ is strictly positive (Chebyshev's sum inequality), and the monotone likelihood ratio $b^{\lambda-\lambda'}$ is equivalent to first-order stochastic dominance.

*(iii)* is the definition of a Gibbs measure. □

The proposition justifies only the form $b^{-\lambda}$ (under the modeling choice of energy $E(b) = \log b$) and the small-block-first ordering *in expectation*; it makes *no* objective claim, neither that the schedule lowers BC loss, improves sample complexity or structured-output validity, nor that it beats a fixed/reversed schedule, and it does not cover the intra-model distillation term. Those benefits are reported empirically (Figure 3 and the diagnostics below); the small-to-large ordering is motivated by — not derived from — the coarse-to-fine learning order of continuous image diffusion (58, 79, 98) and the progressive-distillation recipes of Salimans and Ho (71) and BARD (15).

**Why this family: it is the least-assumptive law that can place a controlled-width interior bump.** *Adding the second moment of* $\log b$ *is the least-assumptive way to gain a controlled-width bump at the interior eval center* $b^\star$=16 *that a monotone power law provably cannot represent; representability is the only unconditional claim.*

*Remark* B.2 (Parameterization: power law vs. its two-moment generalization). This compares *which sampling family to use*; it is structural, not a claim that any family minimizes BC loss. *(a) The power law is the one-moment law and is necessarily monotone.* By Proposition B.1(i), $P_\lambda(b) \propto b^{-\lambda} = e^{-\lambda \log b}$ is the unique maximum-entropy law on $S$ under the *single* constraint $\mathbb{E}[\log b] = \mu$. Its log-density is affine in $g(b) = \log b$, so its likelihood ratio $b^{\lambda-\lambda'}$ is monotone (part (ii)) and it is always FOSD-monotone with mode at an endpoint. A one-parameter law therefore cannot represent a *localized* interior bump (mass concentrated at an interior $b$ with controlled spread): it shifts mass only by sliding the whole monotone profile.

*(b) The two-moment generalization is again max-entropy, and its coefficients are* derived *as Lagrange multipliers.* Maximize $H(q)$ on $S$ subject to normalization and to fixing *both* the mean $\mu_x$=$\mathbb{E}[\log b]$ and variance $\sigma_x^2$=$\operatorname{Var}[\log b]$ of $g(b) = \log b$ (the sufficient statistics $(\log b, (\log b)^2)$). The Lagrangian

$$\begin{aligned}\mathcal{L}(q) = & -\sum_{b\in S} q(b)\log q(b) + \nu\Big(\sum_{b\in S} q(b) - 1\Big) \\ & - \ell_1\Big(\sum_{b\in S} q(b)\log b - \mu_x\Big) - \ell_2\Big(\sum_{b\in S} q(b)(\log b)^2 - (\mu_x^2+\sigma_x^2)\Big)\end{aligned}$$

has stationarity condition $\partial\mathcal{L}/\partial q(b) = 0$, i.e. $\log q(b) = (\nu - 1) - \ell_1 \log b - \ell_2(\log b)^2$, giving uniquely — by strict concavity of $H$ (21, Thm. 12.1.1)(36) — the two-parameter exponential family

$$P_{\ell_1,\ell_2}(b) = \frac{\exp\big(-\ell_1 \log b - \ell_2(\log b)^2\big)}{Z(\ell_1,\ell_2)}, \qquad Z(\ell_1,\ell_2) = \sum_{b\in S} \exp\big(-\ell_1 \log b - \ell_2(\log b)^2\big),$$

a discretized log-normal in $b$ / Gaussian-in-index in $\log b$. It is still a Gibbs law in the same energy coordinate $g(b) = \log b$, so it inherits the entire Boltzmann/tempering reading of Proposition B.1; the power law is the boundary case $\ell_2$=0. With $\ell_2 > 0$ the log-density is

strictly concave in $\log b$, giving an interior mode at $\log b^\star = -\ell_1/(2\ell_2)$ with curvature set by $\ell_2$. Matching the Gibbs law to a target $\mathcal{N}(\mu_x, \sigma_x^2)$ in $x{=}\log b$ gives the closed form

$$\ell_2 = \frac{1}{2\sigma_x^2}, \qquad \ell_1 = -\frac{\mu_x}{\sigma_x^2}, \qquad \Longleftrightarrow \qquad \sigma_x^2 = \frac{1}{2\ell_2}, \qquad \mu_x = -\frac{\ell_1}{2\ell_2}, \qquad x = \log b, \tag{44}$$

so $(\ell_1, \ell_2)$ *are* the maximum-entropy multipliers dual to fixing the mean and variance of $\log b$, strengthening Proposition B.1(i) (whose one-moment map is $\lambda{=}\lambda(\mu)$) to a two-multiplier closed form. The two free moments map bijectively to $(\ell_1, \ell_2)$: the degree-2 law sets the mean and variance of $\log b$ *independently*, while the $\ell_2{=}0$ law ties spread to mean. This permits a *moving constant-width bump* that the always-monotone power law cannot produce, and $\ell_2 \downarrow 0$ recovers Proposition B.1 exactly.[2]

*Status.* Representability is the only *unconditional* claim: the degree-2 law can advance the mode to the usable frontier $b^\star \approx 16$ while keeping a controlled-variance low/mid-block tail, which (a) shows the power law provably cannot. Whether to advance the center by an open-loop clock or a closed-loop competence gate on held-out format validity is the least-settled choice and should be set by a multi-scale validity sweep (Finding B.3); the schedule's theoretical status is in Remark B.4.

**Why this deployed configuration: slide the center at fixed spread, clamp deploy/eval to the usable frontier.** *The center slides* bd1 $\to$ bd16 *at fixed spread* $\sigma_x{=}1$ *octave; the grid keeps* $b{=}32$ *only as a symmetric stress tail, and deploy/eval are clamped to* $b^\star{=}16$ *because bd32 collapses structured-output validity.*

Concretely, the spread is held at one octave ($\sigma_x{=}\ln 2$), so the curvature $\ell_2$ is *constant* while only the center anneals through $\ell_1$:

$$\ell_2 = \tfrac{1}{2(\ln 2)^2} \approx 1.04, \qquad \ell_1 :\ 0 \xrightarrow{\ b:1\to 16\ } -\tfrac{\ln 16}{(\ln 2)^2} \approx -5.77.$$

Four choices fix the rest:

- **Center, not spread.** Only the center $\mu_x$ anneals, from the autoregressive anchor $b{=}1$ to the eval center $b{=}16$; the curvature $\ell_2$ is fixed.
- **Symmetric stress tail.** The support *includes* $b{=}32$ as a non-deployed candidate, so the eval-centered end Gaussian $\{8, 16, 32\}$ is symmetric, not boundary-truncated.
- **Built-in replay.** The constant-$\ell_2$ low/mid-block tail keeps mass on small blocks throughout, a built-in replay against forgetting.
- **Anchored distillation.** Each stage distills from a fixed small-block anchor ($b \le 4$), following BARD (15).

The clamp to the deploy/eval center $b^\star{=}16$ is forced by the empirical cliff below.

*Finding* B.3 (Capacity-determined critical block size). On 116-task AndroidWorld (single epoch, sampled from the $\ell_2{=}0$ Boltzmann measurement-snapshot special case of the degree-2 curriculum, not the deployed $\ell_2{\neq}0$ schedule), task success is near-lossless for $b \le 4$ ($\approx 6.0\%$, matching autoregressive decoding) and degrades gracefully at $b \in \{8, 16\}$ ($\approx 5.2\%$). The *raw* (un-repaired) strict-JSON/action-format validity, however, shows a sharp cliff,

$$\text{validity}(b):\quad \underbrace{1.000}_{b=1},\quad \underbrace{0.952}_{b=4},\quad \underbrace{0.945}_{b=16},\quad \underbrace{0.569}_{b=32},$$

[2] A Beta / Beta–Binomial prior on the grid index is a flexible interior-bump alternative, but it is the max-entropy law for $\mathbb{E}[\log x]$ and $\mathbb{E}[\log(1-x)]$ (21, §12) — not for moments of the curriculum energy $g(b) = \log b$ — so it forfeits the single "Boltzmann in $\log b$" characterization and is not theory-preserving in the sense of (a)–(b).

so $\approx 43\%$ of $b$=32 outputs are malformed. Thus $b$=16 holds while $b$=32 breaks, indicating a capacity-determined critical block size $b_{\text{crit}}$ with $16 < b_{\text{crit}} \leq 32$; deployment uses the usable frontier just below it, the center $b^{\star}$=16.

**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.

A consistent mechanism: larger $b$ unmasks more tokens in parallel under a factorized denoiser, ignoring more intra-block dependency (total correlation); since well-formed JSON/action structure *is* a long-range intra-block dependency, it collapses first once $b$ exceeds capacity. Deployment and evaluation are therefore clamped to the center $b^{\star}$=16, while $b$=32 is retained in training only as a non-deployed symmetric stress tail.

We make no claim that $b^{\star}$ is invariant across model scale or training progress; a 2B-vs-8B validity sweep is needed to decide whether an open-loop clock suffices or a closed-loop competence gate is warranted.

*Remark* B.4 (Honest theoretical status of the schedule). The annealed, easy-to-hard schedule is an instance of graduated optimization (32). *If* the Stage-1 loss at small $b$ is a smoothed, near-convex level of the loss at large $b$ along a valid homotopy (the $\sigma$-niceness condition of 32), the staged schedule reaches an $\varepsilon$-approximate global solution in $O(1/\varepsilon^2)$ stochastic-gradient steps; but this condition is *unverified* for the block-discrete-diffusion loss, and the rate only *matches* the stochastic non-convex SGD lower bound — it gives global-convergence reassurance, *not* a provable speedup over direct training. The convex curriculum theorems of Weinshall and Amir (88), Weinshall et al. (89) predict faster early convergence when $b$ is read as a difficulty proxy, but hold only for convex, *static* orderings and transfer to our non-convex, non-stationary, structured-output setting by analogy only (7).

We therefore claim theoretical superiority of *one* kind only — the unconditional representability characterization of Remark B.2(a)–(b) (a degree-1 / monotone law ties its spread to its mean and so can neither place an interior mode nor retain a controlled-variance low/mid-block tail at a high mean, whereas the degree-2 Gaussian-in-log $b$ law is the minimal exponential family that sets the mean and variance of $\log b$ independently, because the multiplier-to-moment map $\ell_2 = 1/(2\sigma_x^2)$, $\ell_1 = -\mu_x/\sigma_x^2$ is a bijection, Eq. (44)) — and treat the center-equals-frontier choice (deploying at the mode $b^{\star}$=16 rather than the largest candidate $b$=32), the frozen distillation anchor, the retained-tail replay, and competence-gated pacing as well-motivated design principles, not as theorems that the curriculum beats its alternatives. The block-size/quality trade-off itself follows the block-diffusion model class of Arriola et al. (3).

**Real training-loss trace.** The deployed degree-2 curriculum above is the schedule we actually train under. Figure 4 shows the training-loss trace of our in-house run; Figure 5 then compares schedules to illustrate the phenomena the curriculum is designed around.

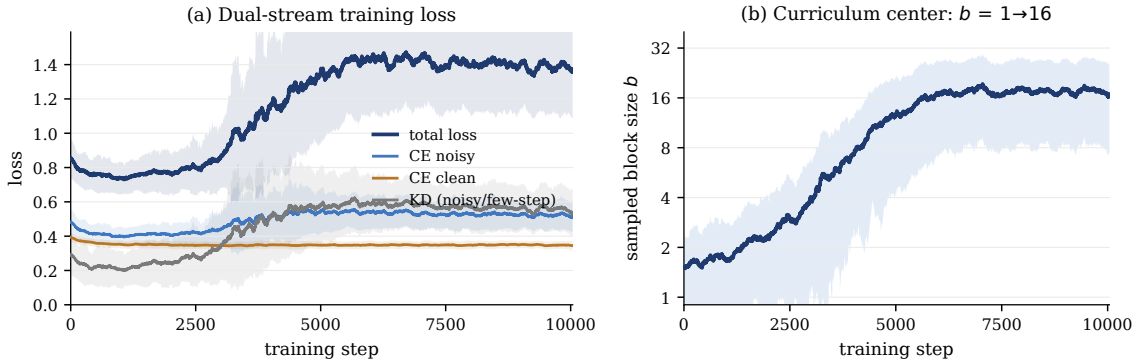
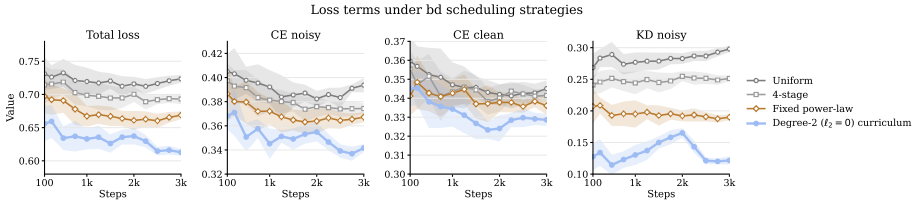
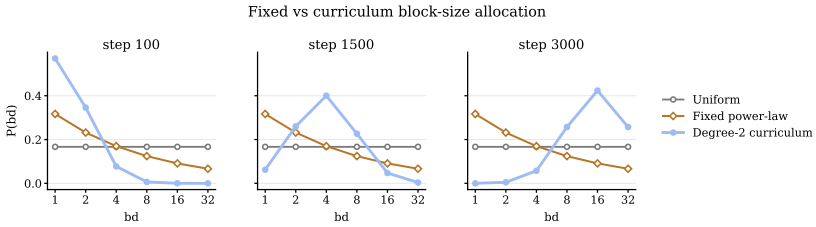



Figure 4: **Real training-loss trace of the deployed degree-2 curriculum** (in-house single run; v6e-16, 10,040 steps, 3 epochs; teacher = same-network clean branch with stop-gradient). **(a)** the dual-stream loss components — CE-clean stays flat while CE-noisy and KD rise as the curriculum moves to larger, harder blocks; **(b)** the sampled block size $b$ sliding $1 \to 16$ (mass settling on $\{8, 16, 32\}$). Unlike the schedule comparison in Figure 5, this is the actual run we trained.

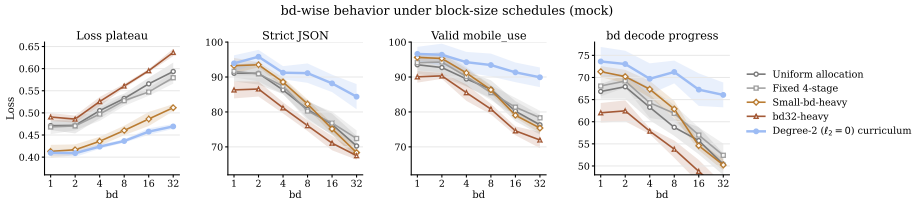
**Not yet formally validated.** These are preliminary results we observed in our own small-scale in-house testing.



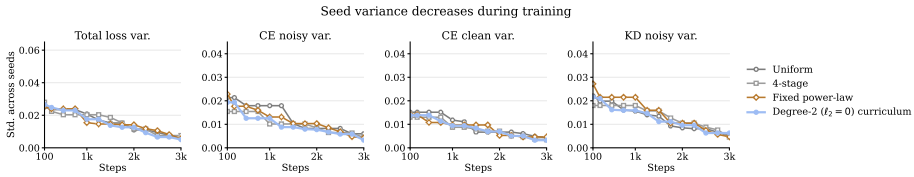

(a) **Loss terms.** The noisy diffusion branch vs. the cleaner behavior-cloning branch, and which terms respond to block-size scheduling.



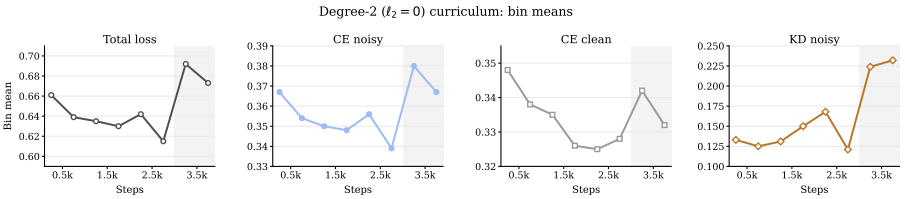

(b) **Block-size allocation.** How the curriculum allocation starts from small-block anchors and shifts mass toward larger block sizes.




(c) **Per-block-size behavior.** Whether a schedule preserves strict JSON formatting, valid mobile-use actions, and decode progress across block sizes, not just aggregate loss.




(d) **Loss-term variance.** Whether the schedule stabilizes the noisy branch rather than merely lowering the mean loss.




(e) **Training-window bins.** Bin-level view of the degree-2 Gaussian-in-log $b$ curriculum (measured on the $\ell_2$=0 snapshot run), flagging later-window instability before final benchmark claims.

Figure 5: **Block-diffusion curriculum diagnostics** (in-house snapshot). (a) loss-term mechanics, (b) block-size allocation, (c) per-block-size structured-output behavior, (d) loss-term variance, and (e) training-window bin means.

## C Android-Curated Reasoning SFT Corpus

The Stage 1 and Stage 2 supervised phases are trained on a reasoning-augmented GUI corpus that we re-curate for the Android phone form factor. We assemble chain-of-thought (CoT) supervision from publicly available GUI-agent reasoning SFT datasets — GUI-Libra, Agu-vis Stage-2, ScaleCUA/GUI-Net, Android-in-the-Zoo (Chain-of-Action-Thought), GUI-R1, and AndroidControl (47, 51, 53, 97, 99, 103). The underlying Android action trajectories are curated from core mobile-control datasets (13, 14, 52, 66, 75, 77), and element-grounding supervision is drawn from large-scale cross-platform GUI corpora (17, 19, 24, 45, 46, 49, 93, 94), complemented by our in-house OpenMobile traces covering answer and long-press actions. Where ready-made reasoning traces are insufficient, we synthesize additional teacher trajectories with reverse-task-synthesis and reflective-CoT pipelines (76, 84, 96), and we follow spatial-reasoning distillation to transfer teacher reasoning into the compact action head (50). Our curation pass (i) restricts every source to Android phone trajectories, (ii) normalizes heterogeneous logs to the mobile UI action schema used by the decoder (tap, swipe, type, system keys, and tool calls), (iii) keeps reasoning-dense steps while discarding pure-action samples that carry no thought, and (iv) reformats the retained CoT-plus-action steps into the block-chunk decoding target of the dLLM action head. This yields an Android-specific reasoning SFT mixture aligned with the deployed action space, rather than a generic multi-platform action corpus.

## D Diagnostic Estimators

These estimate the gaps appearing in Appendix A. For a fixed context $C$, draw $R$ frozen continuations and their basin labels $\bar{m}^{(1:R)}$. **Mode stability** is

$$\widehat{S}(C) = \max_{\bar{m} \in \bar{\mathcal{M}}_C} \frac{1}{R} \sum_{r=1}^{R} \mathbf{1}\{\bar{m}^{(r)} = \bar{m}\}.$$

For a verifier $v(y)$ applied repeatedly to identical decoded chunks, **Verifier disagreement** is the pairwise rate

$$\widehat{D}_{\text{ver}} = \frac{2}{R(R-1)} \sum_{i<j} \mathbf{1}\{v_i(y) \neq v_j(y)\}.$$

**Prefix sufficiency** is measured as the return gap between a state-only selector and a denoise-history selector, $\widehat{\Delta}_{\text{pref}} = J(q_{\text{hist}}) - J(q_{\text{state}})$. **Critical-window sensitivity** compares the selected window against random or all-window variants, $\widehat{\Delta}_{\text{win}} = J(q_{\mathcal{I}_{\text{obs}}}) - \max\{J(q_{\text{rand}}), J(q_{\text{all}})\}$. **Actuator leakage** is reported both as a return gap and as a distributional distance,

$$\widehat{\Delta}^{J}_{\text{leak}} = |J(\widetilde{\pi}_{\psi,\eta}) - J_{\text{ideal}}(q_\psi)|, \qquad \widehat{\epsilon}_{\text{leak}} = \mathbb{E}_C \, \text{TV}\big(\widetilde{\pi}_{\psi,\eta}(\cdot \mid C), \pi^{q_\psi}_{\text{id}}(\cdot \mid C)\big) \, .$$

## E Baseline Backbone: Qwen3-VL Architecture

This appendix describes the baseline screen-language model W0 freezes: the default instantiation GUI-Owl-1.5-2B-Instruct (96) is the Qwen3-VL-2B foundation model (5) after GUI post-training (Section 4.2).

Qwen3-VL (5) builds upon the Qwen2-VL (83) architecture with several improvements. The vision encoder uses a ViT-600M backbone (27 transformer blocks, hidden dimension $d$=1152) with native dynamic resolution support through 2D Rotary Position Embedding

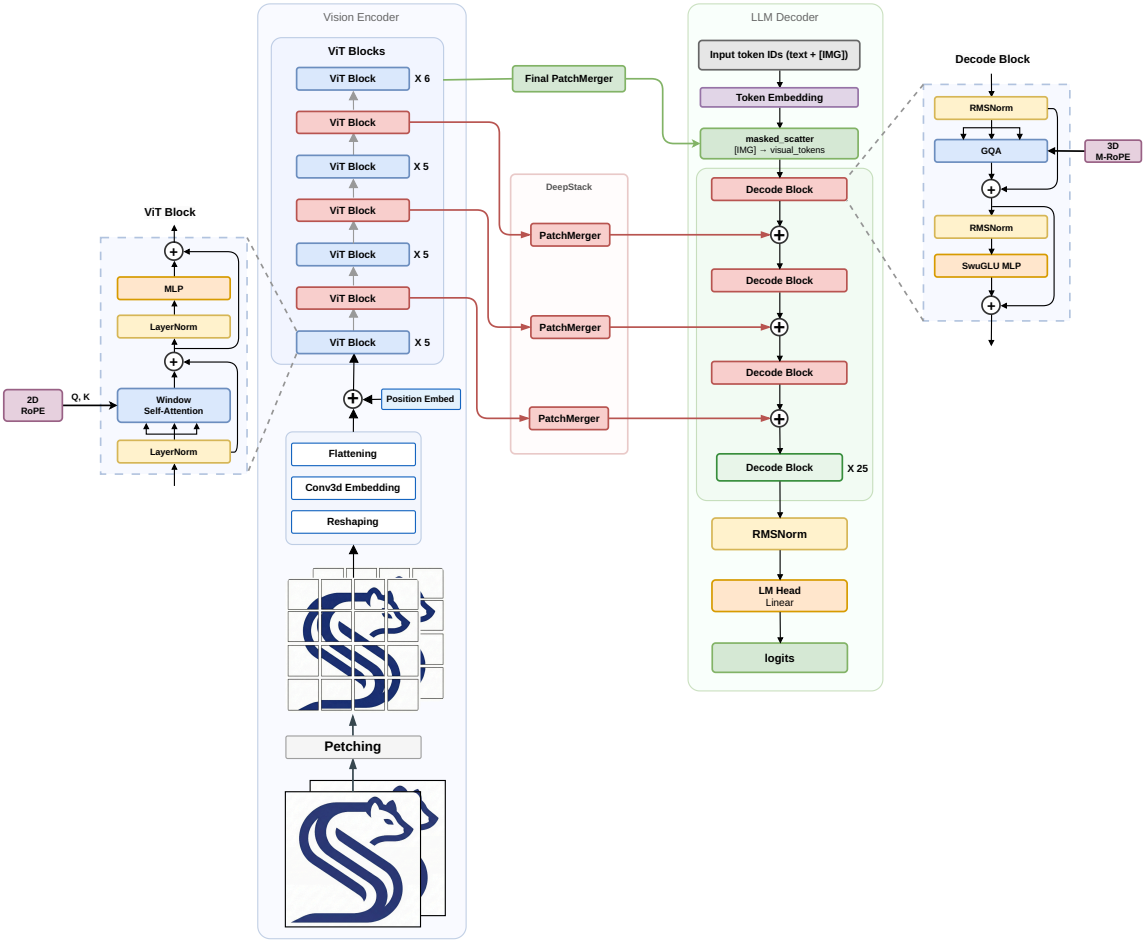



Figure 6: **Qwen3-VL architecture overview.** The vision encoder (ViT-600M) processes visual inputs with 2D-RoPE, extracts multi-scale DeepStack features from blocks $\{8, 16, 24\}$, and compresses them via a patch merger ($2 \times 2 \to 1$). The 1.5B-parameter transformer decoder receives DeepStack injections at its early layers and fuses visual tokens with language tokens.

(2D-RoPE) combined with learnable absolute position embeddings, eliminating the need for fixed-resolution resizing. A key innovation is **DeepStack**: the encoder extracts intermediate features from blocks $\{8, 16, 24\}$, each processed by a dedicated PatchMerger,

$$\boldsymbol{F}^{(k)} = \text{PatchMerger}_k\big(\boldsymbol{h}^{(l_k)}\big) \in \mathbb{R}^{N/4 \times d_{\text{out}}}, \quad l_k \in \{8, 16, 24\}, \tag{45}$$

and injects these multi-scale features into the LLM decoder's early layers ($j \in \{0, 1, 2\}$) via additive residuals at visual-token positions:

$$\boldsymbol{h}^{(j)}_{\text{LLM}}[\text{vis}] \;\mathrel{+}=\; \boldsymbol{F}^{(k)}. \tag{46}$$

This supplies the decoder with low-level textures (block 8), mid-level object parts (block 16), and high-level scene semantics (block 24) from the very first layers. The patch merger compresses $2 \times 2$ spatial patches into single tokens using a learned two-layer MLP, reducing the visual token count by $4\times$.

The language backbone is a 1.5B-parameter transformer decoder with 28 layers, hidden dimension $D_{\text{VLM}} = 1536$, and 12 attention heads. Visual and language tokens are concatenated along the sequence dimension with special delimiter tokens, and the model uses grouped-query attention (GQA) with 2 KV heads for efficient inference. The "2B" in the model name refers to the total parameter count ($\approx$600M vision encoder + $\approx$1.5B language decoder), so the 2B name and the 1.5B-parameter decoder are the same model, not different sizes.

W0 freezes this backbone and reads its final-layer features $H_b^{\text{VLM}}$ (Eq. (8)); the same network's clean/autoregressive branch is also the distillation teacher for the block-diffusion action head (Section 4.5).