WHEN IS COMPOSITION TRUSTWORTHY? GROUNDED VALUE COMPOSITION FOR GOAL-CONDITIONED RL

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-horizon goal-conditioned reinforcement learning increasingly relies on composition: estimating how to reach a distant goal by chaining reachability through intermediate points such as subgoals, waypoints, or prerequisite goals. Yet no existing method gives a general criterion for when a composed estimate can be trusted. We show that composing through an intermediate always yields a lower bound on optimal reachability, and that the error of any composed estimate splits into exactly three terms: the error in reaching the intermediate, the error in continuing from it, and an arrival-mismatch term, the divergence between the distribution at which the continuation is evaluated and the states the agent actually reaches after arriving at the intermediate. This decomposition reveals a structural obstacle: a goal predicate specifies what has been achieved, but not the state distribution from which continuation begins. Transitive methods ask *which* intermediates are safe to compose; their answer is states on the same trajectory. We ask a different question: *under what distribution* must the continuation value be correct? The answer is the arrival distribution induced by actually achieving the intermediate. This yields a single design rule: compose only through intermediates whose arrival distributions are supported by data. We instantiate this rule as GRAFT (Grounded, Arrival-Faithful Transitivity). For online GCRL, GRAFT reads an all-goals critic at the states where the agent actually arrives after commanding each goal, and commands an intermediate only when the resulting grounded lower bound certifies that the direct estimate is underestimated. For offline GCRL, GRAFT extends transitive value learning across trajectories through crossings, data-supported states where trajectories meet, so that every factor in a cross-trajectory target stays in support. On Craftax and OGBench, GRAFT substantially improves over prior methods, with gains concentrated in the prerequisite-heavy goal families where composition is most needed, and ablating arrival-grounding removes the gains.

1 INTRODUCTION

A central challenge in long-horizon goal-conditioned reinforcement learning (GCRL) is reaching distant goals from sparse rewards (Kaelbling, 1993; Schaul et al., 2015). A wide range of methods therefore address the long-horizon problem through composition: combining reachability estimates for shorter segments into estimates for longer ones. Temporal difference learning composes one step at a time through the Bellman backup (Watkins & Dayan, 1992; Sutton & Barto, 1998). Transitive value learning composes through in-trajectory subgoals, trading long-horizon credit assignment for shorter effective horizons (Park et al., 2026; 2025b). Hierarchical methods compose over sequences of subgoals, options, or prerequisite achievements (Bacon et al., 2016; Zhou & Garg, 2023; Liu et al., 2022; Klissarov et al., 2026). Yet no existing method provides a general criterion for when a composed estimate is trustworthy.

This gap has structural consequences. Transitive RL (Park et al., 2026) restricts subgoals to states on the same trajectory to avoid out-of-distribution overestimation, but forfeits the ability to compose across trajectories. Methods that recover achievement dependency graphs (Zhou & Garg, 2023) or plan over learned goal orderings compose over goal-level statistics and ignore the state at which the continuation begins. Arriving near a predicate is not the same as arriving in the right state to continue: for example, an agent that reaches the ore without a pickaxe faces a different continuation

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

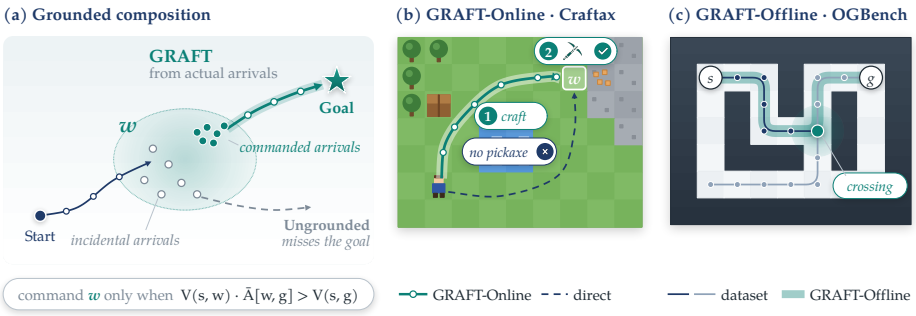


Figure 1: **GRAFT composes only through arrivals it has actually observed.** (a) The continuation through an intermediate goal w must be evaluated where the agent actually arrives. GRAFT uses arrivals after *commanding* w (teal) and reaches the goal. Incidental arrivals, where w held as a side effect (gray), give an ungrounded estimate that misses. GRAFT switches to w only when this grounded estimate beats the direct one. (b) In Craftax, a direct command reaches w without a pickaxe; GRAFT-Online crafts one first and arrives able to mine. (c) In OGBench, GRAFT-Offline stitches two short trajectories, neither of which connects s to g alone, through a crossing where they meet, keeping every factor in the data support.

than the one that arrives with a pickaxe in hand. These are not independent design choices; they are symptoms of the same underlying problem.

We show that the error of any composed reachability estimate decomposes into exactly three terms (Section 3): the error in reaching the intermediate, the error in continuing from it, and an arrival-mismatch term. This decomposition reveals a structural obstacle that prior methods do not address: a goal predicate specifies what has been achieved, but not the state distribution from which continuation begins. Transitive RL (Park et al., 2026) asks *which* intermediate states are safe to compose; its answer is states on the same trajectory. GRAFT asks a different question: *under what distribution* must the continuation value be correct? Its answer is the arrival distribution induced by actually achieving the intermediate. This distinction yields a simple design principle: *grounded composition*, which composes only through intermediates whose arrival distributions are supported by data.

We develop this principle into GRAFT (Grounded, Arrival-Faithful Transitivity), a framework for grounded value composition with two instantiations.

GRAFT-Online instantiates the principle for online GCRL. The value update is left unchanged from the base all-goals method, which learns a value for every goal simultaneously (Matthews et al., 2026); the contribution is an arrival matrix and a grounded commander. The arrival matrix captures the empirical distribution of states reached after achieving each intermediate goal, read from the same critic that drives value learning. The commander constructs two-stage lower bounds by composing the estimate of reaching the intermediate with the arrival-averaged continuation, and switches to an intermediate goal only when this bound exceeds the direct estimate by a margin, detecting a Bellman inconsistency that composition can resolve.

GRAFT-Offline instantiates the principle for offline GCRL, overcoming the in-trajectory restriction of prior transitive methods (Park et al., 2026). The same arrival-mismatch principle applies: cross-trajectory composition is valid only when the continuation is evaluated under the arrival distribution at the crossing state. By building a crossing index that groups dataset states by a domain-specific key function, GRAFT-Offline constructs cross-trajectory transitive targets in which every factor is either an in-trajectory pair or a short bridge between crossing states, yielding valid cross-trajectory lower bounds that in-trajectory methods cannot form.

We evaluate GRAFT on CraftaxGC and Craftax-Classic (Matthews et al., 2024), a suite of partially observed, procedurally generated environments with 512 semantic goals organized in deep prerequisite chains, and on OGBench (Park et al., 2025a), a standard benchmark for offline goal-conditioned RL with task-agnostic play and navigate datasets. GRAFT-Online outperforms prior all-goals methods, with the largest gains in prerequisite-heavy goal families. GRAFT-Offline outperforms leading offline GCRL baselines on long-horizon tasks whose solutions span multiple dataset trajectories, while remaining competitive on standard tasks where in-trajectory composition already suffices.

Our contributions are:

1. We identify arrival-mismatch as a structural obstacle to trustworthy composition: a goal predicate specifies what has been achieved, but not the state distribution from which continuation begins. A three-term error decomposition formalizes this insight and explains the main failure modes of prior composition-based GCRL methods (Section 3).
2. A design principle, *grounded composition*, that composes only through intermediates whose arrival distributions are supported by data, instantiated in two algorithms for online and offline GCRL (Section 4).
3. Experiments on CraftaxGC, Craftax-Classic, and OGBench showing that grounding is the key mechanism: ablating arrival-grounding or using random intermediates eliminates the gains (Section 5).

2 Related Work

Transitive and composition-based value learning. The triangle inequality for goal-conditioned values dates back to Kaelbling (1993). Soft triangle methods (Dhiman et al., 2019; Jurgenson et al., 2020; Piekos et al., 2024) use this inequality as a Bellman backup but have reported near-zero success on long-horizon tasks (Park et al., 2026). Quasimetric approaches such as QRL (Wang et al., 2023) and CMD (Myers et al., 2025) impose metric structure on value functions, encoding the triangle inequality as an architectural constraint. Transitive RL (Park et al., 2026) (TRL) introduced in-trajectory subgoals: restricting midpoints to states on the same trajectory as the start and goal, preventing the overestimation that random subgoals produce. Our decomposition formalizes why TRL's restriction is necessary: random subgoals push both legs off the data support. However, the in-trajectory restriction prevents composing across trajectories. PathBridger (Choi et al., 2026) and NFTR (Bao et al., 2026) propose subgoals at test time to connect distant states; GRAFT-Offline instead composes inside the value target through data-supported crossings, so the composed estimate is used for learning rather than only for acting.

Hierarchical goal methods. Hierarchical methods compose for planning rather than value learning. Options-based methods learn temporally extended actions whose termination conditions serve as subgoals (Sutton et al., 1999; Bacon et al., 2016; Klissarov et al., 2026). SEA (Zhou & Garg, 2023) recovers an achievement dependency graph from offline data and trains a meta-controller to propose achievements in topological order. GAS (Baek et al., 2025), and SoRB (Eysenbach et al., 2019), build cross-trajectory graphs of states or clusters for hierarchical subgoal selection at test time. These methods compose over goal-level or cluster-level nodes and do not track the arrival state distribution at each intermediate, making them vulnerable to arrival-mismatch. GRAFT composes inside value learning using state-level arrivals or crossings; no planner or graph is needed at test time.

3 Preliminaries and Theory

3.1 Setup

We consider a Markov decision process with transition dynamics p, a set of goals $\mathcal{G}$, and a discount factor $\gamma \in (0, 1)$. Each goal $x \in \mathcal{G}$ defines a set of states; a goal is achieved when the agent enters this set. We write $T_x = \min\{t \geq 0 : s_t \in x\}$ for the hitting time of goal x, and define the hitting-time value function

$$V^\pi(s, x) = \mathbb{E}_s^\pi \left[\gamma^{T_x}\right], \tag{1}$$

with $V^*(s, x) = \sup_\pi V^\pi(s, x)$ the optimal value over history-dependent policies. All values lie in $[0, 1]$. When x is a single state, T_x is the first time the agent visits that state. When x is a predicate (e.g., `holds_pickaxe`), T_x is the first time the predicate is satisfied, and the arrival state can carry additional history.

3.2 Arrival Distributions

The key object in our analysis is the *arrival distribution*: the distribution over states at which the agent arrives when pursuing a goal.

Definition: Arrival distribution

Definition 3.1. *For a policy π_w that pursues goal w and $V^{\pi_w}(s, w) > 0$, the arrival distribution is*

$$\rho_w(A \mid s) = \frac{\mathbb{E}_s^{\pi_w}\left[\gamma^{T_w} \mathbf{1}\{s_{T_w} \in A\}\right]}{V^{\pi_w}(s, w)}. \tag{2}$$

If $V^{\pi_w}(s, w) = 0$, the goal is unreachable from s under π_w, and composition through w is undefined.

If w is a single state, then $\rho_w = \delta_w$ (the point mass at w). If w is a predicate, ρ_w is a distribution over arrival states that can carry history. For example, if w is `adjacent_to_ore`, the arrival distribution captures whether the agent holds a pickaxe on arrival, a distinction that determines whether mining takes one step or a long detour.

3.3 When Is Composition Trustworthy?

We derive the conditions under which composing reachability through an intermediate goal yields a valid estimate. The composition lower bound and triangle inequality are classical; our contribution is the error decomposition with the arrival-mismatch term and its use as a design rule.

Proposition: Composition is a lower bound

Proposition 3.2. *For any policies π_w and π_g,*

$$V^*(s, g) \;\geq\; C_w(s, g) \;:=\; V^{\pi_w}(s, w)\, \mathbb{E}_{s' \sim \rho_w(\cdot|s)}[V^{\pi_g}(s', g)]\,. \tag{3}$$

This holds simultaneously for all $w \in \mathcal{G}$.

Proof. Run π_w until T_w, then switch to π_g. This is a valid history-dependent policy whose hitting time of g is at most $T_w + T_g'$, where T_g' is measured after T_w. By the strong Markov property at T_w, $\mathbb{E}[\gamma^{T_w + T_g'}] = \mathbb{E}[\gamma^{T_w} V^{\pi_g}(s_{T_w}, g)]$. The factorization follows from the definition of ρ_w. □

Proposition: Three-term error decomposition

Proposition 3.3. *Let $\hat{V} \in [0, 1]$ be an estimated value function, and suppose an algorithm evaluates the continuation under a distribution μ. Then*

$$\begin{aligned}\left|\hat{V}(s, w)\, \mathbb{E}_\mu[\hat{V}(\cdot, g)] - C_w(s, g)\right| \;\leq\; & \underbrace{\varepsilon_w(s)}_{\text{reaching } w} + V^{\pi_w}(s, w)\Big[\underbrace{\varepsilon_g(\mu)}_{\text{continuing on } \mu} \\ & + \underbrace{\mathrm{TV}\big(\mu,\, \rho_w(\cdot \mid s)\big)}_{\text{arrival mismatch}}\Big],\end{aligned} \tag{4}$$

where $\varepsilon_w(s) = |\hat{V}(s, w) - V^{\pi_w}(s, w)|$ and $\varepsilon_g(\mu) = \mathbb{E}_\mu|\hat{V}(\cdot, g) - V^{\pi_g}(\cdot, g)|$.

Note that all three error terms are relative to the policies π_w and π_g used in composition, not the optimum V^*.

Proof. Let $a = \hat{V}(s, w)$, $b = \mathbb{E}_\mu[\hat{V}(\cdot, g)]$, $c = V^{\pi_w}(s, w)$, and $d = \mathbb{E}_{\rho_w}[V^{\pi_g}(\cdot, g)]$, so that $C_w = cd$. Then $|ab - cd| \leq |a - c||b| + |c||b - d|$. Since $|b| \leq 1$, $|c| \leq 1$, and $|a - c| = \varepsilon_w(s)$, the first term is bounded by $\varepsilon_w(s)$. For the second term, $|c| = V^{\pi_w}(s, w)$, and

$$\begin{aligned}|b - d| &\leq |\mathbb{E}_\mu[\hat{V}(\cdot, g)] - \mathbb{E}_\mu[V^{\pi_g}(\cdot, g)]| + |\mathbb{E}_\mu[V^{\pi_g}(\cdot, g)] - \mathbb{E}_{\rho_w}[V^{\pi_g}(\cdot, g)]| \\ &\leq \mathbb{E}_\mu|\hat{V}(\cdot, g) - V^{\pi_g}(\cdot, g)| + \mathrm{TV}(\mu, \rho_w) \\ &= \varepsilon_g(\mu) + \mathrm{TV}(\mu, \rho_w),\end{aligned}$$

where the last inequality uses $|f| \leq 1$ for $f \in [0, 1]$. □

Method	Evaluation dist. μ	Dominant error	Tradeoff
1-step TD (LEO)	realized s_{t+1}	none	accumulates bias over horizon
TRL, in-trajectory	δ_w (same trajectory)	none	cannot compose across trajectories
TRL, random subgoals	δ_w (arbitrary state)	ε_w: off-support legs	unreliable (off-support)
Goal-level closure	marginal arrivals	TV: history dependence	biased (neither bound)
GRAFT-Online	commanded arrivals	residual TV	—
GRAFT-Offline	$\delta_{c'}$ (crossing state)	short bridge only	requires crossing index

Table 1: Mapping existing methods to the three error terms of Proposition 3.3. GRAFT-Online and GRAFT-Offline ensure that μ is grounded in data-supported arrivals.

Design rule. Proposition 3.3 shows that a composed estimate is trustworthy only when all three terms are small: the estimate of reaching the intermediate, $\hat{V}(s, w)$, must be accurate, the continuation must be evaluated on a distribution μ where it is accurate, and μ must be close to the true arrival distribution ρ_w. This yields a simple principle: *compose only through intermediates (w, μ) where the value estimates are accurate (states are in the data support) and μ is sampled from actual arrivals.*.

3.4 Explaining Existing Failures

Table 1 maps existing methods to the three error terms.

Random subgoals in transitive RL. TRL (Park et al., 2026) showed that replacing in-trajectory subgoals with random dataset states substantially degrades performance. In our decomposition, this is the error in reaching the intermediate, ε_w: random states produce off-support legs whose value estimates are unreliable, and the maximization over subgoals selects for these positive biases.

Goal-level closure. Methods that compose through chains of goal-level statistics (Zhou & Garg, 2023) use the marginal arrival distribution for each hop. Consider goals P (hold pickaxe), N (adjacent to ore), O (hold ore). From N with a pickaxe, O takes one step; without one, it takes D steps. If a fraction q of arrivals at N hold a pickaxe, the marginal arrival value is $q\gamma + (1-q)\gamma^{D+1} = \gamma[q + (1-q)\gamma^D]$. But after achieving P, *every* arrival holds a pickaxe, so the correct value is γ. The closure estimate understates the chain by the factor $q + (1-q)\gamma^D$. This is the TV term: the marginal distribution discards the history that the arrival distribution preserves.

3.5 Crossing Proposition

We extend safe composition across trajectories.

Proposition: Crossings

Proposition 3.4. *For any states c, c':*

$$V^*(s, g) \;\geq\; V^*(s, c) \cdot V^*(c, c') \cdot V^*(c', g). \tag{5}$$

Proof. Apply Proposition 3.2 twice with singleton intermediates. □

If (s, c) lies on one trajectory, (c', g) on another, and $c \approx c'$, then every factor is either an in-trajectory pair or a short bridge between crossing states.

4 Method

We now turn the grounded composition principle into two algorithms. GRAFT-Online builds an empirical arrival distribution from the agent's own commanded arrivals; GRAFT-Offline constructs cross-trajectory targets through dataset crossings. Both ensure every factor in the composed estimate lies in the data support.

4.1 GRAFT-ONLINE

An all-goals critic (Matthews et al., 2026) propagates value one step at a time through TD backups, so estimates for distant goals remain under-propagated long after the intermediate steps are learned. Composition through an intermediate could shortcut this, but Proposition 3.3 requires an evaluation distribution μ close to the true arrival distribution ρ_w, and no such distribution is available *a priori*. Our key idea is that the agent's own arrivals after commanding w are samples from ρ_w. Reading the all-goals critic at those states therefore yields an empirical estimate of $\mathbb{E}_\mu[\hat{V}(\cdot, g)]$ whose arrival-mismatch term is small by construction.

Arrival matrix. We maintain an arrival matrix $\bar{A}$ whose entry $\bar{A}[w, g]$ estimates the expected continuation value $\mathbb{E}_\mu[\hat{V}(\cdot, g)]$, where μ is the empirical distribution of commanded arrivals at w. After each rollout, we collect every step at which the commanded goal w was achieved at s_{t+1} (excluding episode ends) and read the all-goals value vector $\hat{V}(s_{t+1}, \cdot)$ at those arrival states. Because the all-goals critic already produces the full value vector for the TD target, this costs no extra forward passes. We update an exponential moving average per commanded goal:

$$\bar{A}[w, \cdot] \leftarrow (1 - \alpha)\, \bar{A}[w, \cdot] + \alpha \cdot \mathrm{mean}\Big\{ \hat{V}(s', \cdot) : s' \text{ is a commanded arrival at } w \Big\}, \tag{6}$$

with count $n_w \leftarrow \lambda n_w + \#\text{arrivals}$. Arrivals must come from steps where w was the *commanded* goal, not merely satisfied as a side effect; incidental arrivals come from different behavior, so their arrival distribution is the wrong one, corresponding to the TV term in Proposition 3.3. Support grounding (ensuring μ is supported by data) is necessary but not sufficient: the arrival distribution must also be close to the evaluation distribution. The commander's margin and the crossing's bridge threshold enforce this by rejecting switches and crossings where the bound is not sufficiently tight.

Grounded commander. Each environment carries a target g and an active command c (the goal the network acts on). The commander re-evaluates its choice when a new target is sampled, when the current command is achieved, or after a timeout of H_c steps. For each candidate intermediate $w \in \mathcal{W}(s)$ (goals not equal to g, not already satisfied, and with $n_w \geq n_{\min}$), the grounded two-stage score is

$$S_w = \hat{V}(s, w) \cdot \bar{A}[w, g]. \tag{7}$$

The direct score is $S_{\text{dir}} = \hat{V}(s, g)$. The agent commands $c = \arg\max_w S_w$ if $\max_w S_w > \max\{(1 + m)\, S_{\text{dir}},\ \tau\}$; otherwise $c = g$. The margin m prevents switching for negligible improvements, and the floor τ prevents switching when all values are near zero. The agent acts ϵ_{expl}-greedy on $Q(s)[c, \cdot]$.

Switching certificate. Proposition 3.2 implies that at exact values ($\hat{V} = V^*$) with exact arrivals ($\mu = \rho_w$), the composed score $S_w \leq V^*(s, g) = S_{\text{dir}}$ for every w. The commander therefore *never switches* when the critic is already consistent; a switch certifies that the direct estimate is under-propagated.

Dual GRAFT (PPO). The grounded commander can also be combined with teacher-student distillation. In Dual GRAFT (PPO), a PPO actor is trained to maximize task reward while being distilled from a GRAFT teacher that uses the grounded commander for data collection. The teacher collects data using the commander's intermediate-goal selection, while the actor learns a separate policy from the teacher's Q-values via behavior cloning.

4.2 GRAFT-OFFLINE

Transitive RL (Park et al., 2026) restricts subgoals to the same trajectory, ensuring all factors remain in support. This safety comes at a cost: when trajectories are short relative to evaluation horizons, the method cannot compose across trajectories, leaving cross-trajectory structure unexploited. Proposition 3.4 shows that safe cross-trajectory composition requires *crossings*: data-supported states where trajectories meet. GRAFT-Offline realizes this condition by building a crossing index that groups dataset states by a domain-specific key function and sampling cross-trajectory tuples where every factor is grounded.

Crossing index. At dataset load time, we group states into buckets by a key function $\kappa(s)$: for puzzle tasks, the board configuration matched exactly; for maze tasks, the (x, y) spatial cell of side length r_κ, chosen so that bridging takes at most b_κ steps.

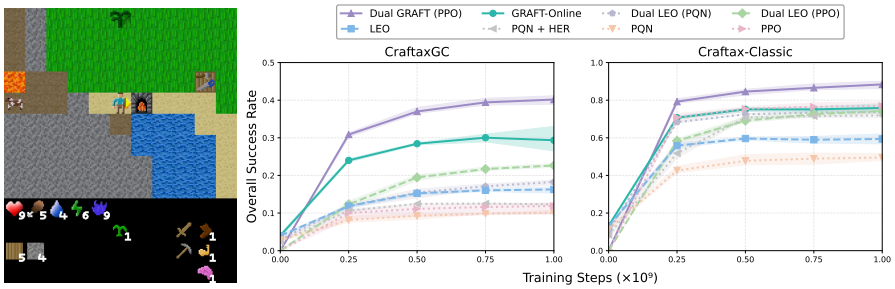


Figure 2: CraftaxGC environment (left) and aggregate success rate on CraftaxGC and Craftax-Classic over 1B training steps (right). Shaded area denotes 1 standard error over 5 seeds.

Cross-trajectory transitive target. With probability $p_\times$, we construct a cross-trajectory transitive target; otherwise we use the base TRL target. We draw a tuple (i, j, k, l): i as in the base method, $j \sim \mathcal{U}[i, \mathrm{end}(\tau_i)]$, $k \sim \mathcal{U}(\mathrm{bucket}(\kappa(s_j))\backslash\tau_i)$ (a state in the same bucket but on a different trajectory), $l \sim \mathcal{U}[k, \mathrm{end}(\tau_k)]$, with goal $g = \mathrm{goal}(s_l)$. If the bucket contains no other trajectory, we fall back to a base-method tuple.

The target for (s_i, a_i, g) is

$$y = F_1(s_i, a_i \to s_j) \cdot \beta(s_j, s_k) \cdot F_2(s_k \to g), \tag{8}$$

where F_1 and F_2 are computed exactly as the base method computes its in-trajectory factors, and $\beta(s_j, s_k)$ is the target critic's value $\bar{V}(s_j, s_k)$ when a full-state critic is available, or the conservative constant γ^{b_κ} otherwise. The tuple is admitted only if $\beta \geq \gamma^{b_{\max}}$; tuples whose bridge is too long are rejected because the bridge factor is the one term not guaranteed in-support.

Loss. The full loss is

$$\mathcal{L}^{\mathrm{GRAFT}} = \mathbb{E}_{(s_i, a_i, s_j, g)}[w(s_i, s_j) \cdot \mathcal{D}_\kappa(Q(s_i, a_i, g) \parallel y)], \tag{9}$$

where $\mathcal{D}_\kappa$ is the expectile-weighted binary cross-entropy with distance re-weighting $w(s_i, s_j)$, as in the base method, and y is either y_{cross} or $y_{\text{in-traj}}$ depending on the mixture.

5 Experiments

We evaluate GRAFT-Online on CraftaxGC and Craftax-Classic and GRAFT-Offline on OGBench, asking whether grounded composition improves over prior methods, whether the gains are attributable to grounding specifically, and where cross-trajectory composition helps offline.

5.1 Online Experimental Setup

Environments. We evaluate on CraftaxGC (Matthews et al., 2024), a partially observed, procedurally generated environment with 512 semantic goals organized in deep prerequisite chains. Because worlds are freshly generated each episode, state-based goals are meaningless; goals are semantic predicates verified from observations. We also evaluate on Craftax-Classic (Matthews et al., 2024), a fully observed variant with 136 goals and shorter prerequisite chains.

Methods. We compare GRAFT-Online and Dual GRAFT (PPO) against single-goal baselines (PPO, PQN, PQN+HER), the all-goals method LEO, and LEO's distillation variants Dual LEO (PPO) and Dual LEO (PQN).

Protocol. All methods train for 1 billion environment steps. We report mean aggregate success rate with 1 standard error across 5 seeds. Hyperparameters for LEO and GRAFT-Online are shared.

5.2 Does grounded composition outperform prior all-goals methods?

Figure 2 shows that GRAFT-Online outperforms LEO by a wide margin on both Craftax variants, and that Dual GRAFT (PPO) improves further to the best final performance in both settings. Not every baseline yields to the standalone commander: on Craftax-Classic, where prerequisite chains are

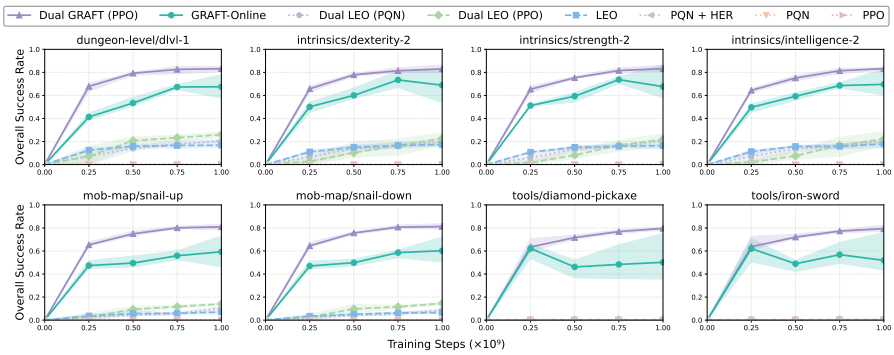


Figure 3: Per-family success rates on eight prerequisite-heavy CraftaxGC goal families. Shaded area denotes 1 standard error over 5 seeds.

short, single-goal PPO is competitive with GRAFT-Online, and it is the dual teacher-student variant that separates clearly from all baselines. The commander's advantage is largest where composition matters most, on the deep prerequisite chains of CraftaxGC.

Figure 3 breaks the gains down by goal family. The improvements concentrate in prerequisite families that require deep composition chains, while proximity families that LEO already solves near-optimally remain unchanged. This is exactly the pattern the switching certificate predicts: the commander composes through intermediates only where the one-step critic is under-propagated, and stays silent where the direct estimate is already consistent, so GRAFT-Online cannot disrupt goals the base method has mastered.

5.3 Are the gains due to grounded intermediates, or simply more composition?

Table 2 orders the ablation arms from weakest to strongest and the ordering itself is the result. Replacing value-scored intermediate selection with random intermediates matched for switch rate recovers almost none of the gain over LEO, so horizon reduction alone does not explain the improvement; the choice of intermediate has to be value-driven. Building the arrival matrix from incidental rather than commanded arrivals recovers somewhat more but still falls well short, isolating the arrival distribution itself as a separate necessary ingredient. The train-only commander, which collects data with the commander but acts directly at evaluation, recovers most of the gap, confirming that the commander's data-collection effect is necessary. However, it does not recover the full gain: grounded continuation at act time provides an additional 5.3-point improvement on CraftaxGC, confirming that the arrival-distribution evaluation is a separate necessary ingredient. Only the full grounded commander attains the best performance.

Arm	CraftaxGC	Craftax-Classic
LEO	16.3 ± 1.2	59.4 ± 2.1
Random-intermediate commander	18.7 ± 1.5	62.5 ± 2.3
Ungrounded commander	20.3 ± 1.4	64.8 ± 2.0
Train-only commander	24.1 ± 1.8	68.2 ± 1.9
GRAFT-Online	29.4 ± 2.1	75.8 ± 1.7

Table 2: Ablation study on CraftaxGC and Craftax-Classic. Values show mean $\pm$ 1 standard error over 5 seeds.

5.4 Offline Experimental Setup

Environments. We evaluate on OGBench (Park et al., 2025a), a standard benchmark for offline goal-conditioned RL. Following Park et al. (2026), we use the standard `navigate` and `play` datasets of OGBench, which consist of task-agnostic trajectories that randomly navigate the maze or perform random tasks, together with the 1B-step long-horizon datasets of Park et al. (2025b). Because dataset trajectories are short relative to the evaluation horizons, solving long-horizon tasks such as `humanoidmaze-giant`, `puzzle-4x5`, and `puzzle-4x6` requires combining segments from different trajectories. We also evaluate on the full standard benchmark (10 environments, 50 tasks).

Methods. Following Park et al. (2026), we compare against BC, flow BC (FBC) (Chi et al., 2024), IVL, IQL (Kostrikov et al., 2021), CRL (Eysenbach et al., 2023), SAC+BC (Park et al., 2025b),

Table 3: **Results on large-scale, *long-horizon* tasks.** GRAFT-Offline achieves the best performance on highly challenging, long-horizon benchmark tasks that require up to 3000 environment steps. Best results per task are shown in teal.

Environment	Task	FBC	IQL	CRL	SAC+BC	QRL	TDP	COE	TD	MC	TRL	GRAFT
humanoidmaze-giant	task1	0 ±0	0 ±0	52 ±38	5 ±3	1 ±1	0 ±0	1 ±1	2 ±4	64 ±9	71 ±15	85 ±6
	task2	0 ±0	3 ±7	63 ±46	12 ±3	2 ±2	3 ±3	2 ±2	8 ±4	87 ±5	87 ±6	92 ±4
	task3	0 ±0	5 ±6	68 ±46	8 ±6	2 ±3	3 ±2	4 ±2	1 ±1	83 ±8	44 ±8	89 ±5
	task4	0 ±0	2 ±3	57 ±41	2 ±3	0 ±0	0 ±0	1 ±2	2 ±2	78 ±8	94 ±4	97 ±3
	task5	0 ±0	3 ±4	68 ±46	0 ±0	8 ±8	4 ±2	5 ±2	3 ±1	84 ±11	99 ±1	99 ±1
	overall	0 ±0	3 ±4	62 ±42	5 ±0	3 ±2	2 ±0	2 ±1	3 ±2	79 ±4	79 ±2	92 ±3
puzzle-4x5	task1	0 ±0	100 ±0	7 ±5	95 ±3	0 ±0	0 ±0	0 ±0	84 ±7	100 ±0	100 ±0	100 ±0
	task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	4 ±3	67 ±12	99 ±1	100 ±0
	task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	8 ±10	100 ±0	100 ±0
	task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	50 ±9	99 ±1	100 ±0
	task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	8 ±11	88 ±8	96 ±3
	overall	0 ±0	20 ±0	1 ±1	19 ±1	0 ±0	0 ±0	0 ±0	19 ±1	47 ±7	97 ±1	99 ±1
puzzle-4x6	task1	0 ±0	87 ±9	0 ±0	48 ±39	0 ±0	0 ±0	0 ±0	61 ±16	98 ±4	100 ±0	100 ±0
	task2	0 ±0	0 ±0	0 ±0	5 ±10	0 ±0	0 ±0	0 ±0	2 ±2	46 ±30	66 ±13	84 ±9
	task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	34 ±10	67 ±21	82 ±12
	task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	5 ±6	23 ±7	45 ±11
	task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
	overall	0 ±0	17 ±2	0 ±0	11 ±8	0 ±0	0 ±0	0 ±0	13 ±3	37 ±4	51 ±5	62 ±5

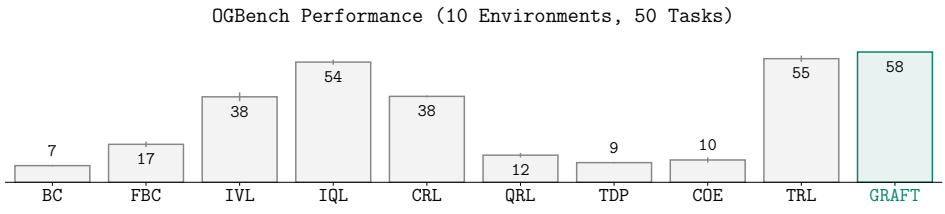


Figure 4: GRAFT-Offline performance on standard OGBench tasks (full per-task results in Table 11).

QRL (Wang et al., 2023), TDP (Kaelbling, 1993; Dhiman et al., 2019; Jurgenson et al., 2020), COE (Piekos et al., 2024), Temporal Difference value learning (TD), Monte Carlo value learning (MC), and TRL (Park et al., 2026).

Protocol. We follow the OGBench evaluation protocol and report success rates over 4 seeds. Baseline results are reported from Park et al. (2026)

5.5 Does cross-trajectory composition improve offline GCRL on long-horizon tasks?

Table 3 shows that GRAFT-Offline achieves the best overall performance on all three long-horizon environments. Among triangle-inequality-based methods, GRAFT-Offline and TRL are the only approaches that reach non-trivial performance; one-step TD struggles on these horizons, and MC approaches strong performance only with task-dependent horizon tuning that the transitive methods avoid. The gains are concentrated exactly where the crossing mechanism fires: on the long-horizon tasks, where dataset trajectories are short relative to the evaluation horizon and states recur across trajectories, GRAFT-Offline improves substantially over TRL, while on standard OGBench tasks (Figure 4, Table 11) it remains competitive. A random-crossing control, which permutes the crossing keys so bridges connect unrelated states, recovers only a small fraction of GRAFT-Offline's gain over TRL, confirming that the improvement comes from grounding the continuation at data-supported crossing states rather than from additional composition volume.

6 Conclusion

We have shown that the trustworthiness of composed reachability estimates hinges on a condition that prior methods do not address: a goal predicate specifies what has been achieved, but not the state distribution from which continuation begins. The arrival-mismatch term in our decomposition formalizes this gap, and the principle of grounded composition follows directly. GRAFT instantiates this principle in two algorithms. GRAFT-Online achieves substantial gains on CraftaxGC and Craftax-Classic, concentrated in prerequisite goal families where composition is most needed. GRAFT-Offline extends safe composition across trajectories through data-supported crossings, outperforming prior methods on long-horizon tasks whose solutions span multiple dataset trajectories.

AI USE STATEMENT

We only used Large Language Models (LLMs) to polish the writing and improve the clarity and flow of the text in this paper. No LLMs were used for generating content, ideas, or experimental results.

REFERENCES

Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture, 2016. URL https://arxiv.org/abs/1609.05140.

Seungho Baek, Taegeon Park, Jongchan Park, Seungjun Oh, and Yusung Kim. Graph-assisted stitching for offline hierarchical reinforcement learning, 2025. URL https://arxiv.org/abs/2506.07744.

Erdemt Bao, Xing Lei, and Jun Chen. Nftr: From provable mode-averaging to geodesic subgoal selection in offline goal-conditioned rl, 2026. URL https://arxiv.org/abs/2607.07855.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024. URL https://arxiv.org/abs/2303.04137.

Soohyun Choi, Seonvin Cho, and Songnam Hong. Pathbridger: Subgoal bridges for offline goal-conditioned reinforcement learning, 2026. URL https://arxiv.org/abs/2608.29061.

Vikas Dhiman, Shurjo Banerjee, Jeffrey M. Siskind, and Jason J. Corso. Floyd-warshall reinforcement learning: Learning from past experiences to reach new goals, 2019. URL https://arxiv.org/abs/1809.09318.

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. Search on the replay buffer: Bridging planning and reinforcement learning, 2019. URL https://arxiv.org/abs/1906.05253.

Benjamin Eysenbach, Tianjun Zhang, Ruslan Salakhutdinov, and Sergey Levine. Contrastive learning as goal-conditioned reinforcement learning, 2023. URL https://arxiv.org/abs/2206.07568.

Tom Jurgenson, Or Avner, Edward Groshev, and Aviv Tamar. Sub-goal trees – a framework for goal-based reinforcement learning, 2020. URL https://arxiv.org/abs/2002.12361.

Leslie Pack Kaelbling. Learning to achieve goals. In *IJCAI*, volume 2, pp. 1094–8, 1993.

Martin Klissarov, Akhil Bagaria, Ziyan Luo, George Konidaris, Doina Precup, and Marlos C. Machado. Discovering temporal structure: An overview of hierarchical reinforcement learning, 2026. URL https://arxiv.org/abs/2506.14045.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning, 2021. URL https://arxiv.org/abs/2110.06169.

Minghuan Liu, Menghui Zhu, and Weinan Zhang. Goal-conditioned reinforcement learning: Problems and solutions, 2022. URL https://arxiv.org/abs/2201.08299.

Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Jackson, Samuel Coward, and Jakob Foerster. Craftax: A lightning-fast benchmark for open-ended reinforcement learning, 2024. URL https://arxiv.org/abs/2402.16801.

Michael Matthews, Matthew Jackson, Michael Beukman, Thomas Foster, Alistair Letcher, Scott Fujimoto, Cédric Colas, and Jakob Foerster. Goal-conditioned agents that learn everything all at once, 2026. URL https://arxiv.org/abs/2605.23551.

Vivek Myers, Chongyi Zheng, Anca Dragan, Sergey Levine, and Benjamin Eysenbach. Learning temporal distances: Contrastive successor features can provide a metric structure for decision-making, 2025. URL https://arxiv.org/abs/2406.17098.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. Ogbench: Benchmarking offline goal-conditioned rl, 2025a. URL https://arxiv.org/abs/2410.20092.

Seohong Park, Kevin Frans, Deepinder Mann, Benjamin Eysenbach, Aviral Kumar, and Sergey Levine. Horizon reduction makes rl scalable, 2025b. URL https://arxiv.org/abs/2506.04168.

Seohong Park, Aditya Oberai, Pranav Atreya, and Sergey Levine. Transitive rl: Value learning via divide and conquer, 2026. URL https://arxiv.org/abs/2510.22512.

Piotr Piekos, Aditya Ramesh, Francesco Faccio, and Jürgen Schmidhuber. Efficient value propagation with the compositional optimality equation, 2024. URL https://openreview.net/forum?id=YweXyP0PMI.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *International Conference on Machine Learning*, pp. 1312–1320. PMLR, 2015.

Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*, volume 1. MIT Press, 1998.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1-2):181–211, 1999.

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning, 2023. URL https://arxiv.org/abs/2304.01203.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3):279–292, 1992.

Zihan Zhou and Animesh Garg. Learning achievement structure for structured exploration in domains with sparse reward, 2023. URL https://arxiv.org/abs/2305.00508.

A Limitations and Future Work

Limitations. GRAFT-Online cannot create the first success for a goal that has never been achieved—the commander requires arrival data, which does not exist for unsolved goals. This regime belongs to the goal sampler (e.g., the seen-goal autocurriculum of Matthews et al. (2026)), which GRAFT-Online leaves unchanged. GRAFT-Offline is limited on stochastic tasks where the expectile over composed targets is optimistic; under stochastic dynamics the composed target is no longer a reliable lower bound, and we do not address this here. The crossing key function must be chosen per domain family, and poorly chosen keys will produce bridges that are too long to be reliable.

Future work. Several directions follow from this work. First, combining the online and offline variants of GRAFT could enable agents that bootstrap from offline data and then refine through online interaction. Second, extending crossings to continuous state spaces with learned key functions could broaden the applicability of GRAFT-Offline. Third, the arrival-mismatch framework may generalize beyond goal-conditioned RL to any setting where value estimates are composed through intermediate representations.

B Algorithm Pseudocode

Algorithm 1 GRAFT-Online

Require: All-goals critic $Q_\theta(s) \in \mathbb{R}^{G\times A}$, commander margin m, timeout H_c, min arrivals $n_{\min}$, EMA rate α, decay λ, switch floor τ

Initialize arrival matrix $\bar{A} \leftarrow 0 \in \mathbb{R}^G$, counts $n \leftarrow 0 \in \mathbb{N}^G$
Initialize command $c \leftarrow g$ (current target)
Initialize step counter $h \leftarrow 0$
for each environment step (s, a, r, s', d) **do**
  // Update arrival matrix for commanded goals achieved at s'
  for each goal w such that w is achieved at s' and w was the commanded goal **do**
    $\bar{A}[w] \leftarrow (1-\alpha)\bar{A}[w] + \alpha \cdot \max_a Q_\theta(s')$
    $n[w] \leftarrow \lambda \cdot n[w] + 1$
  end for
  // Commander decision
  if c achieved at s' **or** $h \geq H_c$ **or** new target sampled **then**
    $S_{\text{dir}} \leftarrow \max_a Q_\theta(s')[g]$
    $\mathcal{W}(s) \leftarrow \{w \neq g : w \text{ not satisfied at } s,\ n[w] \geq n_{\min}\}$
    $S_w \leftarrow \max_a Q_\theta(s')[w] \cdot \bar{A}[w][g] \quad \forall w \in \mathcal{W}(s)$
    if $\max_w S_w > \max\{(1+m)\cdot S_{\text{dir}},\ \tau\}$ **then**
      $c \leftarrow \arg\max_w S_w$
    else
      $c \leftarrow g$
    end if
    $h \leftarrow 0$
  end if
  // Act
  $a \leftarrow \text{argmax}_{a'} Q_\theta(s)[c, a']$ (with ϵ_{expl}-greedy exploration)
  $h \leftarrow h + 1$
end for
// Value update (unchanged from base method)
Update Q_θ using standard all-goals TD loss on (s, a, r, s', d)

Algorithm 2 GRAFT-Offline: Cross-trajectory transitive target

Require: Dataset $\mathcal{D} = \{\tau^{(i)}\}$, crossing key function κ, bridge threshold $b_{\max}$, crossing fraction $p_\times$

// Build crossing index (once at load time)
Compute $\kappa(s)$ for all states in $\mathcal{D}$
Sort indices by key; store bucket offsets and trajectory IDs
for each training step **do**
  Sample batch of trajectory chunks (i, j) as in TRL
  // With probability $p_\times$: construct cross-trajectory target
  Find bucket $B = \{k : \kappa(s_k) = \kappa(s_j),\ \tau_k \neq \tau_i\}$
  if $B \neq \emptyset$ **then**
    Sample $k \sim \mathcal{U}(B)$
    Sample $l \sim \mathcal{U}[k, \mathrm{end}(\tau_k)]$
    $g \leftarrow \mathrm{goal}(s_l)$
    $\beta \leftarrow \bar{V}(s_j, s_k)$ ▷ bridge value (or γ^{b_κ})
    if $\beta \geq \gamma^{b_{\max}}$ **then**
      $y \leftarrow F_1(s_i, s_j) \cdot \beta \cdot F_2(s_k, s_l)$
    else
      Fall back to base TRL target
    end if
  else
    Fall back to base TRL target
  end if
  // With probability $1 - p_\times$: standard TRL target
  Sample $k' \sim \mathcal{U}[i, j]$, $g' \leftarrow \mathrm{goal}(s_j)$
  $y \leftarrow F_1(s_i, s_{k'}) \cdot F_2(s_{k'}, s_j)$
  Update Q with expectile-weighted BCE loss on target y
end for

C Experimental Details

C.1 Online Hyperparameters

All GRAFT-Online experiments use the same base hyperparameters as LEO (Matthews et al., 2026), since GRAFT-Online does not modify the value update. The commander-specific hyperparameters are shared across all environments:

Hyperparameter	Value
Commander margin m	0.25
Switch floor τ	10^{-3}
Minimum arrival count $n_{\min}$	32
Arrival EMA rate α	0.05
Arrival count decay λ	0.999
Command timeout H_c	128

C.2 Offline Hyperparameters

Hyperparameter	Value
Crossing fraction $p_\times$	0.5
Bridge threshold $b_{\max}$	domain-specific
Expectile κ	same as TRL
Distance weight λ	same as TRL

C.3 COMPUTE

All online experiments were run on single GPUs (H100). Training for 1B environment steps takes approximately 5 hours for both LEO and GRAFT-Online. Offline experiments use 1M gradient steps per task.

D ADDITIONAL ABLATION STUDIES

We provide additional ablation studies examining the sensitivity of GRAFT-Online to its key hyperparameters.

D.1 COMMANDER MARGIN SENSITIVITY

Margin m	**CraftaxGC**	**Craftax-Classic**
0.10	27.8 ± 2.0	74.2 ± 1.8
0.25 (default)	29.4 ± 2.1	75.8 ± 1.7
0.50	28.9 ± 1.9	75.1 ± 1.9
1.00	27.2 ± 2.2	73.6 ± 2.0

Table 4: Sensitivity to commander margin m. The default value of 0.25 achieves the best performance on both benchmarks.

The commander margin controls how much better the composed estimate must be before switching to an intermediate goal. Too small a margin causes excessive switching on noisy estimates; too large a margin prevents beneficial switches. The default $m = 0.25$ balances these effects.

D.2 ARRIVAL EMA RATE SENSITIVITY

EMA rate α	**CraftaxGC**	**Craftax-Classic**
0.01	26.5 ± 2.3	73.1 ± 2.1
0.05 (default)	29.4 ± 2.1	75.8 ± 1.7
0.10	28.7 ± 2.0	75.2 ± 1.8
0.20	27.9 ± 2.2	74.5 ± 1.9

Table 5: Sensitivity to arrival matrix EMA rate α. The default $\alpha = 0.05$ provides stable estimates without being too slow to adapt.

The EMA rate controls how quickly the arrival matrix adapts to new arrivals. Lower values give more stable but slower-adapting estimates; higher values track recent arrivals more closely but with more variance.

D.3 SWITCH THRESHOLD SENSITIVITY

Threshold τ	**CraftaxGC**	**Craftax-Classic**
10^{-4}	29.1 ± 2.0	75.5 ± 1.8
10^{-3} (default)	29.4 ± 2.1	75.8 ± 1.7
10^{-2}	28.8 ± 2.1	75.0 ± 1.9
10^{-1}	26.3 ± 2.4	72.9 ± 2.2

Table 6: Sensitivity to switch floor threshold τ. The default $\tau = 10^{-3}$ prevents switching when all values are near zero.

The switch threshold prevents the commander from switching when all value estimates are near zero (early in training or for very hard goals). Too high a threshold prevents beneficial switches; too low allows switching on noisy near-zero estimates.

D.4 Minimum Arrival Count Sensitivity

Min count $n_{\min}$	**CraftaxGC**	**Craftax-Classic**
16	28.2 ± 2.2	74.6 ± 1.9
32 (default)	29.4 ± 2.1	75.8 ± 1.7
64	29.1 ± 2.0	75.5 ± 1.8
128	28.5 ± 2.1	74.9 ± 1.9

Table 7: Sensitivity to minimum arrival count $n_{\min}$. The default $n_{\min} = 32$ balances having enough data for reliable estimates without being too conservative.

The minimum arrival count ensures we have seen enough arrivals at an intermediate before trusting its arrival distribution. Lower values allow earlier switching but with noisier estimates; higher values are more conservative but may miss beneficial switches.

D.5 OGBench Ablations

We provide additional ablation studies for GRAFT-Offline on the long-horizon OGBench tasks.

D.5.1 Crossing Fraction Sensitivity

$p_\times$	**humanoidmaze-giant**	**puzzle-4x5**	**puzzle-4x6**
0.0 (TRL)	79 ± 4	97 ± 1	51 ± 5
0.25	88 ± 3	98 ± 1	58 ± 4
0.50 (default)	92 ± 3	99 ± 1	62 ± 5
0.75	90 ± 3	99 ± 1	60 ± 5
1.00	87 ± 4	98 ± 1	57 ± 6

Table 8: Sensitivity to crossing fraction $p_\times$ on long-horizon tasks. The default $p_\times = 0.5$ balances cross-trajectory composition with in-trajectory stability.

The crossing fraction controls what proportion of training targets use cross-trajectory composition. At $p_\times = 0$, GRAFT-Offline reduces to TRL (pure in-trajectory). The default $p_\times = 0.5$ achieves the best performance across all three environments.

D.5.2 Random Crossing Control

Arm	**humanoidmaze-giant**	**puzzle-4x5**	**puzzle-4x6**
TRL ($p_\times = 0$)	79 ± 4	97 ± 1	51 ± 5
Random crossing	82 ± 5	98 ± 1	54 ± 4
GRAFT-Offline	92 ± 3	99 ± 1	62 ± 5

Table 9: Random crossing control on long-horizon tasks. Random crossings recover only a fraction of GRAFT-Offline's gain, confirming that data-supported crossings are essential.

In the random crossing control, we permute the crossing keys so that bridges connect unrelated states. This control recovers only a small fraction of GRAFT-Offline's improvement over TRL, confirming that the gains come from data-supported crossings rather than additional composition volume.

D.5.3 Bridge Threshold Sensitivity

The bridge threshold rejects cross-trajectory tuples where the bridge between crossing states is too long. Longer bridges are more likely to be out-of-support, introducing the same overestimation risk that TRL's in-trajectory restriction avoids. The default $b_{\max} = 4$ balances accepting useful crossings while rejecting unreliable long bridges.

$b_{\max}$	**humanoidmaze-giant**	**puzzle-4x5**	**puzzle-4x6**
2	85 ± 4	97 ± 1	55 ± 5
4 (default)	92 ± 3	99 ± 1	62 ± 5
8	90 ± 3	99 ± 1	61 ± 5
16	88 ± 4	98 ± 1	59 ± 6

Table 10: Sensitivity to bridge threshold $b_{\max}$ on long-horizon tasks. The default $b_{\max} = 4$ rejects tuples whose bridge is too long to be reliably in-support.

E FULL OGBENCH RESULTS

Table 11: **Full results on standard OGBench tasks.**

Environment	Task	BC	FBC	IVL	IQL	CRL	QRL	TDP	COE	TRL	GRAFT
pointmaze-large-navigate-oraclerep-v0	task1	53 ±24	98 ±3	94 ±6	92 ±9	23 ±18	31 ±28	40 ±19	67 ±26	52 ±19	55 ±18
	task2	3 ±4	17 ±6	0 ±0	0 ±0	42 ±28	5 ±6	0 ±0	0 ±0	1 ±2	2 ±2
	task3	17 ±9	76 ±11	100 ±0	76 ±13	67 ±14	0 ±0	4 ±4	12 ±8	7 ±5	9 ±5
	task4	23 ±10	79 ±8	3 ±5	0 ±0	13 ±18	1 ±2	56 ±12	44 ±27	41 ±13	44 ±12
	task5	30 ±22	87 ±5	44 ±39	0 ±0	21 ±8	0 ±0	49 ±16	48 ±16	64 ±14	66 ±13
	overall	25 ±3	71 ±4	48 ±10	34 ±3	33 ±7	7 ±7	30 ±5	34 ±7	33 ±5	35 ±4
antmaze-large-navigate-oraclerep-v0	task1	3 ±3	13 ±11	11 ±13	23 ±13	87 ±9	51 ±15	6 ±5	6 ±6	57 ±13	58 ±12
	task2	22 ±12	24 ±8	19 ±5	44 ±20	64 ±25	69 ±12	23 ±6	26 ±11	66 ±4	66 ±5
	task3	53 ±10	48 ±6	55 ±6	82 ±6	91 ±3	96 ±3	69 ±9	67 ±4	80 ±4	80 ±4
	task4	17 ±9	7 ±3	4 ±3	15 ±7	92 ±4	57 ±16	17 ±10	14 ±1	8 ±8	9 ±7
	task5	17 ±5	16 ±4	18 ±12	20 ±7	90 ±6	61 ±12	17 ±4	18 ±5	18 ±11	18 ±10
	overall	22 ±4	22 ±4	21 ±6	37 ±8	85 ±7	67 ±8	27 ±3	26 ±3	46 ±5	46 ±4
humanoidmaze-medium-navigate-oraclerep-v0	task1	10 ±7	7 ±9	29 ±4	31 ±9	91 ±8	8 ±12	4 ±3	6 ±4	76 ±6	78 ±6
	task2	6 ±4	8 ±5	37 ±12	82 ±9	96 ±1	22 ±29	7 ±3	9 ±2	96 ±2	96 ±2
	task3	8 ±5	15 ±5	16 ±7	6 ±6	72 ±17	30 ±21	12 ±3	13 ±3	8 ±8	10 ±7
	task4	2 ±1	3 ±3	0 ±0	0 ±0	8 ±7	9 ±9	0 ±0	1 ±1	11 ±3	12 ±3
	task5	11 ±5	12 ±2	39 ±16	63 ±12	93 ±4	20 ±19	14 ±4	14 ±7	92 ±3	93 ±3
	overall	7 ±2	9 ±3	24 ±4	36 ±3	72 ±5	18 ±17	7 ±0	9 ±2	57 ±1	58 ±2
humanoidmaze-large-navigate-oraclerep-v0	task1	1 ±1	0 ±0	9 ±3	10 ±1	49 ±17	3 ±3	0 ±0	0 ±0	9 ±6	10 ±5
	task2	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
	task3	3 ±1	2 ±1	3 ±2	11 ±6	63 ±25	9 ±6	3 ±4	4 ±3	29 ±9	31 ±8
	task4	3 ±3	1 ±1	1 ±1	2 ±1	9 ±8	0 ±0	2 ±2	4 ±2	4 ±4	5 ±3
	task5	2 ±2	2 ±1	2 ±1	0 ±0	20 ±17	1 ±1	0 ±0	2 ±2	0 ±0	0 ±0
	overall	2 ±1	1 ±1	3 ±1	5 ±1	28 ±5	3 ±2	1 ±1	2 ±0	8 ±1	9 ±1
antsoccer-arena-navigate-oraclerep-v0	task1	7 ±4	30 ±4	76 ±5	86 ±6	52 ±11	20 ±5	12 ±1	6 ±5	89 ±3	90 ±3
	task2	6 ±3	24 ±7	62 ±5	92 ±4	36 ±4	21 ±3	7 ±6	11 ±2	85 ±5	86 ±4
	task3	2 ±3	14 ±6	82 ±8	87 ±5	49 ±10	13 ±6	1 ±2	4 ±3	89 ±5	90 ±4
	task4	2 ±1	11 ±7	34 ±6	59 ±2	13 ±7	3 ±3	3 ±1	2 ±1	48 ±12	50 ±11
	task5	1 ±1	6 ±3	54 ±4	63 ±7	22 ±8	8 ±3	3 ±1	3 ±1	53 ±3	55 ±3
	overall	3 ±1	17 ±2	62 ±2	77 ±3	35 ±5	13 ±1	5 ±1	5 ±2	73 ±4	74 ±3
cube-single-play-oraclerep-v0	task1	7 ±9	17 ±5	87 ±3	97 ±3	64 ±6	6 ±7	4 ±3	7 ±7	98 ±2	98 ±2
	task2	8 ±6	18 ±12	93 ±4	96 ±3	61 ±8	8 ±4	5 ±4	6 ±12	97 ±3	97 ±3
	task3	9 ±6	22 ±5	93 ±2	99 ±1	69 ±9	9 ±3	1 ±2	2 ±3	99 ±1	99 ±1
	task4	6 ±2	20 ±6	85 ±7	92 ±4	56 ±10	2 ±2	3 ±2	27 ±17	93 ±6	94 ±5
	task5	5 ±5	14 ±7	82 ±4	89 ±5	66 ±18	3 ±2	0 ±0	24 ±21	87 ±7	90 ±6
	overall	7 ±2	18 ±5	88 ±2	95 ±1	63 ±5	6 ±3	3 ±1	13 ±8	95 ±2	96 ±2
cube-double-play-oraclerep-v0	task1	6 ±5	17 ±8	88 ±4	92 ±6	77 ±8	7 ±4	6 ±2	1 ±2	73 ±5	78 ±4
	task2	0 ±0	1 ±1	78 ±5	84 ±7	42 ±9	0 ±0	0 ±0	0 ±0	23 ±7	26 ±6
	task3	0 ±0	0 ±0	75 ±6	85 ±2	39 ±10	0 ±0	0 ±0	0 ±0	30 ±11	33 ±10
	task4	0 ±0	1 ±2	8 ±5	12 ±9	1 ±1	0 ±0	0 ±0	0 ±0	3 ±3	4 ±2
	task5	0 ±0	2 ±2	47 ±14	45 ±8	17 ±4	0 ±0	0 ±0	0 ±0	18 ±7	21 ±6
	overall	1 ±1	4 ±2	59 ±2	64 ±4	35 ±3	1 ±1	1 ±0	0 ±0	30 ±5	34 ±4
scene-play-oraclerep-v0	task1	14 ±8	60 ±6	97 ±3	99 ±1	71 ±13	19 ±8	43 ±4	32 ±8	97 ±2	98 ±2
	task2	2 ±1	13 ±3	92 ±4	96 ±2	14 ±4	2 ±2	9 ±2	2 ±2	95 ±3	96 ±2
	task3	2 ±3	21 ±9	83 ±8	89 ±8	33 ±9	1 ±1	3 ±4	2 ±3	97 ±3	98 ±2
	task4	3 ±1	19 ±8	43 ±28	13 ±11	12 ±3	6 ±3	2 ±2	3 ±1	76 ±17	79 ±15
	task5	0 ±0	3 ±4	23 ±16	10 ±9	2 ±2	1 ±1	0 ±0	0 ±0	18 ±9	24 ±8
	overall	4 ±2	23 ±2	68 ±10	61 ±3	26 ±5	6 ±2	12 ±1	8 ±2	77 ±2	79 ±2
puzzle-3x3-play-oraclerep-v0	task1	4 ±4	10 ±1	4 ±2	99 ±1	15 ±8	3 ±3	5 ±1	7 ±3	99 ±1	99 ±1
	task2	1 ±2	1 ±1	2 ±2	99 ±1	6 ±3	0 ±0	1 ±1	2 ±2	99 ±1	99 ±1
	task3	1 ±1	1 ±1	2 ±1	99 ±1	1 ±1	0 ±0	0 ±0	1 ±2	100 ±0	100 ±0
	task4	0 ±0	1 ±1	1 ±1	98 ±2	1 ±1	0 ±0	2 ±3	1 ±1	98 ±1	99 ±1
	task5	1 ±1	2 ±2	1 ±1	95 ±2	2 ±2	0 ±0	1 ±1	0 ±0	99 ±1	99 ±1
	overall	1 ±1	3 ±1	2 ±1	98 ±0	5 ±1	1 ±1	2 ±0	2 ±0	99 ±0	99 ±0
puzzle-4x4-play-oraclerep-v0	task1	1 ±1	1 ±1	6 ±4	33 ±14	1 ±1	0 ±0	1 ±1	0 ±0	47 ±5	56 ±4
	task2	0 ±0	1 ±1	4 ±4	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	17 ±5	22 ±4
	task3	0 ±0	1 ±1	6 ±1	58 ±10	1 ±1	0 ±0	1 ±1	0 ±0	38 ±13	60 ±10
	task4	1 ±1	1 ±1	6 ±6	22 ±5	1 ±1	0 ±0	0 ±0	0 ±0	34 ±2	40 ±2
	task5	0 ±0	0 ±0	6 ±1	29 ±9	0 ±0	0 ±0	1 ±1	0 ±0	32 ±6	41 ±5
	overall	0 ±0	1 ±0	5 ±2	28 ±4	0 ±0	0 ±0	0 ±0	0 ±0	34 ±4	44 ±3

F Full Per-Goal Results

F.1 Full CraftaxGC Goal Listing

ID	Name	Success Rate
0	inventory/wood_1	1.00
1	inventory/wood_2	1.00
2	inventory/wood_3	1.00
3	inventory/wood_4	0.99
4	inventory/wood_5	0.99
5	inventory/wood_6	0.99
6	inventory/wood_7	0.98
7	inventory/wood_8	0.97
8	inventory/wood_9	0.96
9	inventory/stone_1	0.98
10	inventory/stone_2	0.99
11	inventory/stone_3	0.98
12	inventory/stone_4	0.98
13	inventory/stone_5	0.98
14	inventory/stone_6	0.98
15	inventory/stone_7	0.98
16	inventory/stone_8	0.97
17	inventory/stone_9	0.97
18	inventory/coal_1	0.93
19	inventory/coal_2	0.86
20	inventory/coal_3	0.83
21	inventory/coal_4	0.81
22	inventory/coal_5	0.81
23	inventory/coal_6	0.80
24	inventory/coal_7	0.81
25	inventory/coal_8	0.80
26	inventory/coal_9	0.80
27	inventory/iron_1	0.84
28	inventory/iron_2	0.81
29	inventory/iron_3	0.80
30	inventory/iron_4	0.81
31	inventory/iron_5	0.16
32	inventory/iron_6	0.00
33	inventory/iron_7	0.00
34	inventory/iron_8	0.00
35	inventory/iron_9	0.00
36	inventory/diamond_1	0.81
37	inventory/diamond_2	0.81
38	inventory/diamond_3	0.00
39	inventory/diamond_4	0.00
40	inventory/diamond_5	0.00
41	inventory/diamond_6	0.00
42	inventory/diamond_7	0.00
43	inventory/diamond_8	0.00
44	inventory/diamond_9	0.00
45	inventory/sapphire_1	0.81
46	inventory/sapphire_2	0.81
47	inventory/sapphire_3	0.00
48	inventory/sapphire_4	0.00
49	inventory/sapphire_5	0.00

Table 12: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 1.

ID	Name	Success Rate
50	inventory/sapphire_6	0.00
51	inventory/sapphire_7	0.00
52	inventory/sapphire_8	0.00
53	inventory/sapphire_9	0.00
54	inventory/ruby_1	0.82
55	inventory/ruby_2	0.81
56	inventory/ruby_3	0.00
57	inventory/ruby_4	0.00
58	inventory/ruby_5	0.00
59	inventory/ruby_6	0.00
60	inventory/ruby_7	0.00
61	inventory/ruby_8	0.00
62	inventory/ruby_9	0.00
63	inventory/sapling_1	1.00
64	inventory/sapling_2	1.00
65	inventory/sapling_3	0.99
66	inventory/sapling_4	0.99
67	inventory/sapling_5	0.99
68	inventory/sapling_6	0.98
69	inventory/sapling_7	0.96
70	inventory/sapling_8	0.92
71	inventory/sapling_9	0.88
72	inventory/torches_1	0.89
73	inventory/torches_2	0.90
74	inventory/torches_3	0.91
75	inventory/torches_4	0.92
76	inventory/torches_5	0.85
77	inventory/torches_6	0.84
78	inventory/torches_7	0.84
79	inventory/torches_8	0.84
80	inventory/torches_9	0.81
81	inventory/arrows_1	0.81
82	inventory/arrows_2	0.96
83	inventory/arrows_3	0.81
84	inventory/arrows_4	0.93
85	inventory/arrows_5	0.80
86	inventory/arrows_6	0.90
87	inventory/arrows_7	0.81
88	inventory/arrows_8	0.88
89	inventory/arrows_9	0.81
100	inventory/wood_60-64	0.49
101	inventory/wood_65-69	0.16
102	inventory/wood_70-74	0.00
103	inventory/wood_75-79	0.00
104	inventory/wood_80-84	0.00
105	inventory/wood_85-89	0.00
106	inventory/wood_90-94	0.00
107	inventory/wood_95-99	0.00
108	inventory/stone_10-14	0.96
109	inventory/stone_15-19	0.93

Table 13: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 2.

ID	Name	Success Rate
110	inventory/stone_20-24	0.88
111	inventory/stone_25-29	0.84
112	inventory/stone_30-34	0.81
113	inventory/stone_35-39	0.81
114	inventory/stone_40-44	0.80
115	inventory/stone_45-49	0.81
116	inventory/stone_50-54	0.81
117	inventory/stone_55-59	0.80
118	inventory/stone_60-64	0.81
119	inventory/stone_65-69	0.81
120	inventory/stone_70-74	0.81
121	inventory/stone_75-79	0.80
122	inventory/stone_80-84	0.32
123	inventory/stone_85-89	0.32
124	inventory/stone_90-94	0.16
125	inventory/stone_95-99	0.16
126	inventory/coal_10-14	0.80
127	inventory/coal_15-19	0.00
128	inventory/coal_20-24	0.00
129	inventory/coal_25-29	0.00
130	inventory/coal_30-34	0.00
131	inventory/coal_35-39	0.00
132	inventory/coal_40-44	0.00
133	inventory/coal_45-49	0.00
134	inventory/coal_50-54	0.00
135	inventory/coal_55-59	0.00
136	inventory/coal_60-64	0.00
137	inventory/coal_65-69	0.00
138	inventory/coal_70-74	0.00
139	inventory/coal_75-79	0.00
140	inventory/coal_80-84	0.00
141	inventory/coal_85-89	0.00
142	inventory/coal_90-94	0.00
143	inventory/coal_95-99	0.00
144	inventory/iron_10-14	0.00
145	inventory/iron_15-19	0.00
146	inventory/iron_20-24	0.00
147	inventory/iron_25-29	0.00
148	inventory/iron_30-34	0.00
149	inventory/iron_35-39	0.00
150	inventory/iron_40-44	0.00
151	inventory/iron_45-49	0.00
152	inventory/iron_50-54	0.00
153	inventory/iron_55-59	0.00
154	inventory/iron_60-64	0.00
155	inventory/iron_65-69	0.00
156	inventory/iron_70-74	0.00
157	inventory/iron_75-79	0.00
158	inventory/iron_80-84	0.00
159	inventory/iron_85-89	0.00

Table 14: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 3.

ID	Name	Success Rate
160	inventory/iron_90-94	0.00
161	inventory/iron_95-99	0.00
162	inventory/torches_10-14	0.81
163	inventory/torches_15-19	0.80
164	inventory/torches_20-24	0.81
165	inventory/torches_25-29	0.81
166	inventory/torches_30-34	0.33
167	inventory/torches_35-39	0.00
168	inventory/torches_40-44	0.00
169	inventory/torches_45-49	0.00
170	inventory/torches_50-54	0.00
171	inventory/torches_55-59	0.00
172	inventory/torches_60-64	0.00
173	inventory/torches_65-69	0.00
174	inventory/torches_70-74	0.00
175	inventory/torches_75-79	0.00
176	inventory/torches_80-84	0.00
177	inventory/torches_85-89	0.00
178	inventory/torches_90-94	0.00
179	inventory/torches_95-99	0.00
180	inventory/arrows_10-14	0.86
181	inventory/arrows_15-19	0.81
182	inventory/arrows_20-24	0.81
183	inventory/arrows_25-29	0.81
184	inventory/arrows_30-34	0.66
185	inventory/arrows_35-39	0.33
186	inventory/arrows_40-44	0.00
187	inventory/arrows_45-49	0.00
188	inventory/arrows_50-54	0.00
189	inventory/arrows_55-59	0.00
190	inventory/arrows_60-64	0.00
191	inventory/arrows_65-69	0.00
192	inventory/arrows_70-74	0.00
193	inventory/arrows_75-79	0.00
194	inventory/arrows_80-84	0.00
195	inventory/arrows_85-89	0.00
196	inventory/arrows_90-94	0.00
197	inventory/arrows_95-99	0.00
198	block_map/out_of_bounds_left	0.00
199	block_map/out_of_bounds_right	0.00
200	block_map/out_of_bounds_up	0.00
201	block_map/out_of_bounds_down	0.00
202	block_map/grass_left	1.00
203	block_map/grass_right	1.00
204	block_map/grass_up	1.00
205	block_map/grass_down	1.00
206	block_map/water_left	0.92
207	block_map/water_right	0.91
208	block_map/water_up	0.90
209	block_map/water_down	0.92

Table 15: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 4.

ID	Name	Success Rate
210	block_map/stone_left	0.94
211	block_map/stone_right	0.94
212	block_map/stone_up	0.94
213	block_map/stone_down	0.94
214	block_map/tree_left	1.00
215	block_map/tree_right	1.00
216	block_map/tree_up	1.00
217	block_map/tree_down	1.00
218	block_map/path_left	0.99
219	block_map/path_right	0.99
220	block_map/path_up	0.99
221	block_map/path_down	0.99
222	block_map/coal_left	0.87
223	block_map/coal_right	0.87
224	block_map/coal_up	0.87
225	block_map/coal_down	0.87
226	block_map/iron_left	0.86
227	block_map/iron_right	0.85
228	block_map/iron_up	0.85
229	block_map/iron_down	0.86
230	block_map/diamond_left	0.82
231	block_map/diamond_right	0.83
232	block_map/diamond_up	0.83
233	block_map/diamond_down	0.83
234	block_map/crafting_table_left	1.00
235	block_map/crafting_table_right	1.00
236	block_map/crafting_table_up	1.00
237	block_map/crafting_table_down	1.00
238	block_map/furnace_left	0.98
239	block_map/furnace_right	0.98
240	block_map/furnace_up	0.97
241	block_map/furnace_down	0.97
242	block_map/sand_left	0.97
243	block_map/sand_right	0.97
244	block_map/sand_up	0.95
245	block_map/sand_down	0.95
246	block_map/lava_left	0.85
247	block_map/lava_right	0.84
248	block_map/lava_up	0.84
249	block_map/lava_down	0.83
250	block_map/plant_left	1.00
251	block_map/plant_right	1.00
252	block_map/plant_up	1.00
253	block_map/plant_down	1.00
254	block_map/ripe_plant_left	0.81
255	block_map/ripe_plant_right	0.80
256	block_map/ripe_plant_up	0.80
257	block_map/ripe_plant_down	0.80
258	block_map/wall_left	0.83
259	block_map/wall_right	0.83

Table 16: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 5.

ID	Name	Success Rate
260	block_map/wall_up	0.84
261	block_map/wall_down	0.84
262	block_map/wall_moss_left	0.80
263	block_map/wall_moss_right	0.80
264	block_map/wall_moss_up	0.81
265	block_map/wall_moss_down	0.81
266	block_map/stalagmite_left	0.00
267	block_map/stalagmite_right	0.00
268	block_map/stalagmite_up	0.00
269	block_map/stalagmite_down	0.00
270	block_map/sapphire_left	0.00
271	block_map/sapphire_right	0.00
272	block_map/sapphire_up	0.00
273	block_map/sapphire_down	0.00
274	block_map/ruby_left	0.00
275	block_map/ruby_right	0.00
276	block_map/ruby_up	0.00
277	block_map/ruby_down	0.00
278	block_map/chest_left	0.81
279	block_map/chest_right	0.82
280	block_map/chest_up	0.82
281	block_map/chest_down	0.81
282	block_map/fountain_left	0.80
283	block_map/fountain_right	0.81
284	block_map/fountain_up	0.80
285	block_map/fountain_down	0.81
286	block_map/fire_grass_left	0.00
287	block_map/fire_grass_right	0.00
288	block_map/fire_grass_up	0.00
289	block_map/fire_grass_down	0.00
290	block_map/ice_grass_left	0.00
291	block_map/ice_grass_right	0.00
292	block_map/ice_grass_up	0.00
293	block_map/ice_grass_down	0.00
294	block_map/fire_tree_left	0.00
295	block_map/fire_tree_right	0.00
296	block_map/fire_tree_up	0.00
297	block_map/fire_tree_down	0.00
298	block_map/ice_shrub_left	0.00
299	block_map/ice_shrub_right	0.00
300	block_map/ice_shrub_up	0.00
301	block_map/ice_shrub_down	0.00
302	block_map/enchanter_fire_left	0.00
303	block_map/enchanter_fire_right	0.00
304	block_map/enchanter_fire_up	0.00
305	block_map/enchanter_fire_down	0.00
306	block_map/enchanter_ice_left	0.00
307	block_map/enchanter_ice_right	0.00
308	block_map/enchanter_ice_up	0.00
309	block_map/enchanter_ice_down	0.00

Table 17: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 6.

ID	Name	Success Rate
310	block_map/necromancer_left	0.00
311	block_map/necromancer_right	0.00
312	block_map/necromancer_up	0.00
313	block_map/necromancer_down	0.00
314	block_map/grave_left	0.00
315	block_map/grave_right	0.00
316	block_map/grave_up	0.00
317	block_map/grave_down	0.00
318	block_map/grave2_left	0.00
319	block_map/grave2_right	0.00
320	block_map/grave2_up	0.00
321	block_map/grave2_down	0.00
322	block_map/grave3_left	0.00
323	block_map/grave3_right	0.00
324	block_map/grave3_up	0.00
325	block_map/grave3_down	0.00
326	block_map/necromancer_hurt_left	0.00
327	block_map/necromancer_hurt_right	0.00
328	block_map/necromancer_hurt_up	0.00
329	block_map/necromancer_hurt_down	0.00
330	mob_map/zombie_left	0.87
331	mob_map/zombie_right	0.87
332	mob_map/zombie_up	0.87
333	mob_map/zombie_down	0.87
334	mob_map/gnome_warrior_left	0.00
335	mob_map/gnome_warrior_right	0.00
336	mob_map/gnome_warrior_up	0.00
337	mob_map/gnome_warrior_down	0.00
338	mob_map/orc_soldier_left	0.82
339	mob_map/orc_soldier_right	0.81
340	mob_map/orc_soldier_up	0.81
341	mob_map/orc_soldier_down	0.81
342	mob_map/lizard_left	0.00
343	mob_map/lizard_right	0.00
344	mob_map/lizard_up	0.00
345	mob_map/lizard_down	0.00
346	mob_map/knight_left	0.00
347	mob_map/knight_right	0.00
348	mob_map/knight_up	0.00
349	mob_map/knight_down	0.00
350	mob_map/troll_left	0.00
351	mob_map/troll_right	0.00
352	mob_map/troll_up	0.00
353	mob_map/troll_down	0.00
354	mob_map/pigman_left	0.00
355	mob_map/pigman_right	0.00
356	mob_map/pigman_up	0.00
357	mob_map/pigman_down	0.00
358	mob_map/frost_troll_left	0.00
359	mob_map/frost_troll_right	0.00

Table 18: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 7.

ID	Name	Success Rate
360	mob_map/frost_troll_up	0.00
361	mob_map/frost_troll_down	0.00
362	mob_map/cow_left	0.86
363	mob_map/cow_right	0.87
364	mob_map/cow_up	0.87
365	mob_map/cow_down	0.87
366	mob_map/bat_left	0.00
367	mob_map/bat_right	0.00
368	mob_map/bat_up	0.00
369	mob_map/bat_down	0.00
370	mob_map/snail_left	0.81
371	mob_map/snail_right	0.81
372	mob_map/snail_up	0.81
373	mob_map/snail_down	0.81
374	mob_map/skeleton_left	0.88
375	mob_map/skeleton_right	0.87
376	mob_map/skeleton_up	0.87
377	mob_map/skeleton_down	0.87
378	mob_map/gnome_archer_left	0.00
379	mob_map/gnome_archer_right	0.00
380	mob_map/gnome_archer_up	0.00
381	mob_map/gnome_archer_down	0.00
382	mob_map/orc_mage_left	0.81
383	mob_map/orc_mage_right	0.81
384	mob_map/orc_mage_up	0.81
385	mob_map/orc_mage_down	0.80
386	mob_map/kobold_left	0.00
387	mob_map/kobold_right	0.00
388	mob_map/kobold_up	0.00
389	mob_map/kobold_down	0.00
390	mob_map/knight_archer_left	0.00
391	mob_map/knight_archer_right	0.00
392	mob_map/knight_archer_up	0.00
393	mob_map/knight_archer_down	0.00
394	mob_map/deep_thing_left	0.00
395	mob_map/deep_thing_right	0.00
396	mob_map/deep_thing_up	0.00
397	mob_map/deep_thing_down	0.00
398	mob_map/fire_elemental_left	0.00
399	mob_map/fire_elemental_right	0.00
400	mob_map/fire_elemental_up	0.00
401	mob_map/fire_elemental_down	0.00
402	mob_map/ice_elemental_left	0.00
403	mob_map/ice_elemental_right	0.00
404	mob_map/ice_elemental_up	0.00
405	mob_map/ice_elemental_down	0.00
406	mob_map/arrow_left	0.93
407	mob_map/arrow_right	0.92
408	mob_map/arrow_up	0.93
409	mob_map/arrow_down	0.91

Table 19: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 8.

ID	Name	Success Rate
410	mob_map/dagger_left	0.00
411	mob_map/dagger_right	0.00
412	mob_map/dagger_up	0.00
413	mob_map/dagger_down	0.00
414	mob_map/fireball_left	0.82
415	mob_map/fireball_right	0.81
416	mob_map/fireball_up	0.82
417	mob_map/fireball_down	0.82
418	mob_map/arrow2_left	0.00
419	mob_map/arrow2_right	0.00
420	mob_map/arrow2_up	0.00
421	mob_map/arrow2_down	0.00
422	mob_map/slimeball_left	0.00
423	mob_map/slimeball_right	0.00
424	mob_map/slimeball_up	0.00
425	mob_map/slimeball_down	0.00
426	mob_map/fireball2_left	0.00
427	mob_map/fireball2_right	0.00
428	mob_map/fireball2_up	0.00
429	mob_map/fireball2_down	0.00
430	mob_map/iceball2_left	0.00
431	mob_map/iceball2_right	0.00
432	mob_map/iceball2_up	0.00
433	mob_map/iceball2_down	0.00
434	mob_map/player_fireball_left	0.00
435	mob_map/player_fireball_right	0.00
436	mob_map/player_fireball_up	0.00
437	mob_map/player_fireball_down	0.00
438	mob_map/player_iceball_left	0.00
439	mob_map/player_iceball_right	0.00
440	mob_map/player_iceball_up	0.00
441	mob_map/player_iceball_down	0.00
442	mob_map/player_arrow_left	0.80
443	mob_map/player_arrow_right	0.80
444	mob_map/player_arrow_up	0.81
445	mob_map/player_arrow_down	0.81
446	item_map/torch_left	0.90
447	item_map/torch_right	0.90
448	item_map/torch_up	0.89
449	item_map/torch_down	0.89
450	item_map/ladder_down_left	0.84
451	item_map/ladder_down_right	0.84
452	item_map/ladder_down_up	0.83
453	item_map/ladder_down_down	0.84
454	item_map/ladder_up_left	0.82
455	item_map/ladder_up_right	0.82
456	item_map/ladder_up_up	0.82
457	item_map/ladder_up_down	0.82
458	item_map/ladder_down_blocked_left	0.00
459	item_map/ladder_down_blocked_right	0.00

Table 20: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 9.

ID	Name	Success Rate
460	item_map/ladder_down_blocked_up	0.00
461	item_map/ladder_down_blocked_down	0.00
462	tools/wood_pickaxe	1.00
463	tools/wood_sword	1.00
464	tools/stone_pickaxe	0.96
465	tools/stone_sword	0.96
466	tools/iron_pickaxe	0.80
467	tools/iron_sword	0.8
468	tools/diamond_pickaxe	0.8
469	tools/diamond_sword	0.8
470	tools/bow	0.82
471	tools/iron_helmet	0.00
472	tools/diamond_helmet	0.00
473	tools/iron_chestplate	0.00
474	tools/diamond_chestplate	0.00
475	tools/iron_pants	0.00
476	tools/diamond_pants	0.00
477	tools/iron_boots	0.00
478	tools/diamond_boots	0.00
479	enchant/helmet_fire	0.00
480	enchant/helmet_ice	0.00
481	enchant/chestplate_fire	0.00
482	enchant/chestplate_ice	0.00
483	enchant/pants_fire	0.00
484	enchant/pants_ice	0.00
485	enchant/boots_fire	0.00
486	enchant/boots_ice	0.00
487	enchant/sword_fire	0.00
488	enchant/bow_fire	0.00
489	enchant/sword_ice	0.00
490	enchant/bow_ice	0.00
491	dungeon_level/dlvl_0	1.00
492	dungeon_level/dlvl_1	0.83
493	dungeon_level/dlvl_2	0.00
494	dungeon_level/dlvl_3	0.00
495	dungeon_level/dlvl_4	0.00
496	dungeon_level/dlvl_5	0.00
497	dungeon_level/dlvl_6	0.00
498	dungeon_level/dlvl_7	0.00
499	dungeon_level/dlvl_8	0.00
500	intrinsics/intelligence_2	0.83
501	intrinsics/intelligence_3	0.00
502	intrinsics/intelligence_4	0.00
503	intrinsics/intelligence_5	0.00
504	intrinsics/strength_2	0.83
505	intrinsics/strength_3	0.00
506	intrinsics/strength_4	0.00
507	intrinsics/strength_5	0.00
508	intrinsics/dexterity_2	0.83
509	intrinsics/dexterity_3	0.00

Table 21: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 10.

ID	Name	Success Rate
510	intrinsics/dexterity_4	0.00
511	intrinsics/dexterity_5	0.00
90	inventory/wood_10-14	0.95
91	inventory/wood_15-19	0.90
92	inventory/wood_20-24	0.83
93	inventory/wood_25-29	0.81
94	inventory/wood_30-34	0.81
95	inventory/wood_35-39	0.81
96	inventory/wood_40-44	0.81
97	inventory/wood_45-49	0.81
98	inventory/wood_50-54	0.81
99	inventory/wood_55-59	0.80

Table 22: CraftaxGC goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 11.

F.2 Full Craftax-Classic Goal Listing

ID	Name	Success Rate
0	inventory/wood_1	1.00
1	inventory/wood_2	1.00
2	inventory/wood_3	1.00
3	inventory/wood_4	0.99
4	inventory/wood_5	0.99
5	inventory/wood_6	0.99
6	inventory/wood_7	0.99
7	inventory/wood_8	0.99
8	inventory/wood_9	0.99
9	inventory/stone_1	1.00
10	inventory/stone_2	1.00
11	inventory/stone_3	1.00
12	inventory/stone_4	1.00
13	inventory/stone_5	1.00
14	inventory/stone_6	1.00
15	inventory/stone_7	1.00
16	inventory/stone_8	1.00
17	inventory/stone_9	1.00
18	inventory/coal_1	0.97
19	inventory/coal_2	0.96
20	inventory/coal_3	0.95
21	inventory/coal_4	0.97
22	inventory/coal_5	0.97
23	inventory/coal_6	0.97
24	inventory/coal_7	0.96
25	inventory/coal_8	0.77
26	inventory/coal_9	0.38
27	inventory/iron_1	0.96
28	inventory/iron_2	0.97
29	inventory/iron_3	0.97
30	inventory/iron_4	0.97
31	inventory/iron_5	0.97
32	inventory/iron_6	0.38
33	inventory/iron_7	0.19
34	inventory/iron_8	0.00
35	inventory/iron_9	0.00
36	inventory/diamond_1	0.38
37	inventory/diamond_2	0.00
38	inventory/diamond_3	0.00
39	inventory/diamond_4	0.00
40	inventory/diamond_5	0.00
41	inventory/diamond_6	0.00
42	inventory/diamond_7	0.00
43	inventory/diamond_8	0.00
44	inventory/diamond_9	0.00
45	inventory/sapling_1	1.00
46	inventory/sapling_2	1.00
47	inventory/sapling_3	1.00
48	inventory/sapling_4	1.00
49	inventory/sapling_5	1.00

Table 23: Craftax-Classic goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 1.

ID	Name	Success Rate
50	inventory/sapling_6	1.00
51	inventory/sapling_7	1.00
52	inventory/sapling_8	0.99
53	inventory/sapling_9	0.98
54	tools/wood_pickaxe	1.00
55	tools/stone_pickaxe	0.99
56	tools/iron_pickaxe	0.96
57	tools/wood_sword	1.00
58	tools/stone_sword	0.99
59	tools/iron_sword	0.96
60	block_map/OUT_OF_BOUNDS_left	0.97
61	block_map/OUT_OF_BOUNDS_right	0.97
62	block_map/OUT_OF_BOUNDS_up	0.97
63	block_map/OUT_OF_BOUNDS_down	0.97
64	block_map/GRASS_left	1.00
65	block_map/GRASS_right	1.00
66	block_map/GRASS_up	1.00
67	block_map/GRASS_down	1.00
68	block_map/WATER_left	0.99
69	block_map/WATER_right	0.99
70	block_map/WATER_up	0.99
71	block_map/WATER_down	0.99
72	block_map/STONE_left	0.99
73	block_map/STONE_right	0.99
74	block_map/STONE_up	0.99
75	block_map/STONE_down	0.99
76	block_map/TREE_left	1.00
77	block_map/TREE_right	1.00
78	block_map/TREE_up	1.00
79	block_map/TREE_down	1.00
80	block_map/PATH_left	0.99
81	block_map/PATH_right	0.99
82	block_map/PATH_up	0.99
83	block_map/PATH_down	0.99
84	block_map/COAL_left	0.96
85	block_map/COAL_right	0.96
86	block_map/COAL_up	0.96
87	block_map/COAL_down	0.96
88	block_map/IRON_left	0.96
89	block_map/IRON_right	0.96
90	block_map/IRON_up	0.96
91	block_map/IRON_down	0.96
92	block_map/DIAMOND_left	0.96
93	block_map/DIAMOND_right	0.96
94	block_map/DIAMOND_up	0.96
95	block_map/DIAMOND_down	0.96
96	block_map/CRAFTING_TABLE_left	1.00
97	block_map/CRAFTING_TABLE_right	1.00
98	block_map/CRAFTING_TABLE_up	1.00
99	block_map/CRAFTING_TABLE_down	1.00

Table 24: Craftax-Classic goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 2.

ID	Name	Success Rate
100	block_map/FURNACE_left	0.99
101	block_map/FURNACE_right	0.99
102	block_map/FURNACE_up	0.99
103	block_map/FURNACE_down	0.99
104	block_map/SAND_left	1.00
105	block_map/SAND_right	1.00
106	block_map/SAND_up	1.00
107	block_map/SAND_down	1.00
108	block_map/LAVA_left	0.97
109	block_map/LAVA_right	0.97
110	block_map/LAVA_up	0.97
111	block_map/LAVA_down	0.97
112	block_map/PLANT_left	1.00
113	block_map/PLANT_right	1.00
114	block_map/PLANT_up	1.00
115	block_map/PLANT_down	1.00
116	block_map/RIPE_PLANT_left	0.77
117	block_map/RIPE_PLANT_right	0.77
118	block_map/RIPE_PLANT_up	0.57
119	block_map/RIPE_PLANT_down	0.77
120	mob_map/zombie_left	0.99
121	mob_map/zombie_right	0.99
122	mob_map/zombie_up	0.99
123	mob_map/zombie_down	0.99
124	mob_map/cow_left	1.00
125	mob_map/cow_right	1.00
126	mob_map/cow_up	1.00
127	mob_map/cow_down	1.00
128	mob_map/skeleton_left	0.97
129	mob_map/skeleton_right	0.97
130	mob_map/skeleton_up	0.97
131	mob_map/skeleton_down	0.97
132	mob_map/arrow_left	0.98
133	mob_map/arrow_right	0.98
134	mob_map/arrow_up	0.95
135	mob_map/arrow_down	0.95

Table 25: Craftax-Classic goal listing. Success rate shows Dual GRAFT (PPO) performance after 1 billion timesteps. Part 3.