SELECTIVE SHORTCUT VALUE LEARNING FOR LONG-HORIZON OFFLINE GOAL-CONDITIONED RL

Anonymous authors
Paper under double-blind review

ABSTRACT

One source of difficulty in long-horizon offline goal-conditioned reinforcement learning (GCRL) is that sparse rewards must propagate through many Bellman updates before primitive actions become well separated in value. We introduce Selective Shortcut Value Learning (SSVL), which shortens this propagation directly in primitive action-value learning using stopped trajectory segments and a single-discount shortcut continuation. A frozen shortcut-aligned reference selectively compresses segment rewards, while an auxiliary Shortcut Value Consistency (SVC) loss helps the separately learned IQL state value preserve shortcut relations; both are used only during training, and deployment remains a single primitive goal-conditioned policy. We analyze how shorter propagation strengthens progress-state separation and separately characterize the logged-continuation conditions under which shortcut regression targets preserve primitive-action preference. Across diverse state-based and visual OGBench tasks, SSVL is competitive with strong offline GCRL methods, including hierarchical approaches.

1 INTRODUCTION

Goal-conditioned reinforcement learning (GCRL) specifies behavior through desired outcomes rather than task-specific dense rewards, enabling reusable policies that can reach many goals (Schaul et al., 2015; Andrychowicz et al., 2017). Offline GCRL learns such behavior from previously collected trajectories using supervised, contrastive, and value-based objectives (Ghosh et al., 2021; Yang et al., 2022; Ma et al., 2022; Eysenbach et al., 2021; 2022). A persistent challenge is reliable primitive-action ordering for distant goals. Under sparse rewards, goal information must propagate through many discounted Bellman updates, which can weaken the value separation between candidate actions. Modest value-estimation errors can then corrupt local action ordering, echoing the sensitivity of greedy selection to small action gaps (Bellemare et al., 2016); Figure 1(A) illustrates this failure mode.

Existing approaches often simplify long-range decision making through intermediate states or subgoals. Planning methods connect distant states through waypoints (Nasiriany et al., 2019; Hoang et al., 2021; Eysenbach et al., 2019), while hierarchical GCRL decomposes control into subgoal selection and primitive execution (Levy et al., 2019; Nachum et al., 2018; Park et al., 2023). These approaches shorten the effective decision horizon, but may require additional planning or hierarchical structure.

We focus instead on the depth of value propagation from distant goals to primitive actions. Related long-to-short supervision has been introduced at other stages of learning: SAW transfers nearby-subgoal supervision to a flat policy through policy bootstrapping (Zhou & Kao, 2025), but retains one-step goal-conditioned value learning, whereas OTA shortens value propagation in the high-level state-value update of a hierarchical policy (Ahn et al., 2025). Our setting instead attaches shortcut supervision directly to primitive state–action pairs, where the endpoint reflects both the current action and the subsequent logged continuation. This creates a distinct offline attribution problem. We therefore ask: *how can value learning improve primitive-action ordering for distant goals while retaining a single primitive goal-conditioned policy?* See Appendix A for further discussion.

We introduce *Selective Shortcut Value Learning* (SSVL), which brings shortcut propagation directly into primitive state–action value learning. From logged trajectories, SSVL forms stopped segments and uses their observed endpoints to construct shortcut backups. Every shortcut segment uses a

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.

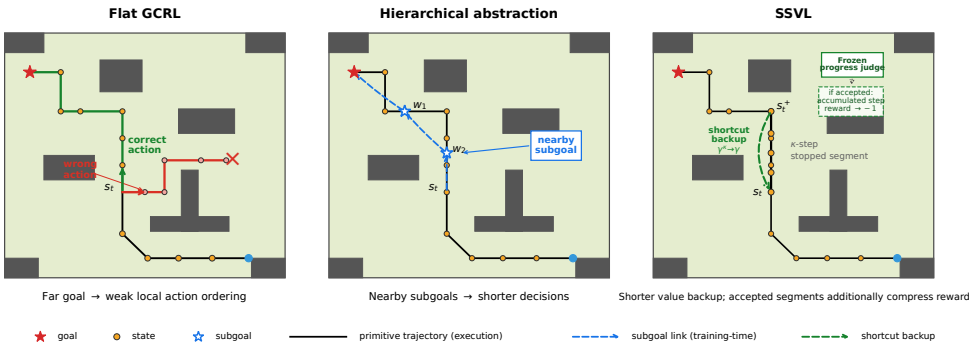

Figure 1: **Motivation and mechanism of SSVL. (A)** Distant goals can make flat primitive-action ordering fragile. **(B)** Hierarchical methods shorten control through nearby subgoals. **(C)** SSVL instead shortens training-time value propagation: a stopped κ-step segment backs up its endpoint with γ rather than γ^{κ}, while accepted segments additionally compress the accumulated reward to -1. Deployment remains a single primitive goal-conditioned policy.

single-discount continuation, so its endpoint value is discounted once rather than once per primitive step. A frozen shortcut-aligned reference provides a separate reward-side refinement: for accepted segments, it replaces the accumulated segment reward by a single-step reward, while rejected segments retain their realized segment reward. Shortcut Value Consistency (SVC) further introduces an auxiliary consistency relation for the separately learned state value. These shortcut mechanisms are used only during training, and deployment remains a single primitive goal-conditioned policy.

We analyze when shortcut value learning can improve primitive-action ranking by tracing how shorter Bellman propagation changes the value differences on which action comparisons depend. Along a fixed-length chain of states progressing toward a goal, shortening the propagation path increases the scale-normalized separation between successive states, making their ordering more robust to bounded normalized value-estimation error. We then characterize the logged-continuation condition needed for shortcut targets to preserve primitive-action preferences in offline data. A long-range target depends not only on the current action but also on the logged continuation that follows it, which can preserve, obscure, or even reverse the preference between two actions. Under a local alignment condition, reference-selective reward compression adds a further action margin and yields a sufficient condition for preserving the preferred ordering under bounded error. Across diverse state-based and visual OGBench tasks, SSVL is competitive with strong offline GCRL methods, including hierarchical approaches, while deploying a single primitive goal-conditioned policy, and controlled ablations support the distinct roles of shortcut propagation, reference selectivity, and value consistency.

Contributions. (i) Primitive-action value learning. We introduce SSVL, which shortens long-horizon value propagation directly in primitive action-value learning while retaining a single primitive goal-conditioned policy; reference selectivity and value consistency serve as training-only refinements. **(ii) Primitive-action ordering theory.** We characterize how shorter propagation strengthens progress-state separation, how logged continuation controls whether shortcut targets preserve primitive-action preferences, and when reference selectivity provides an additional action margin. **(iii) Broad empirical evaluation.** Across diverse state-based and visual OGBench tasks, SSVL is competitive with strong offline GCRL methods, including hierarchical approaches, while deploying a single primitive goal-conditioned policy.

2 Preliminaries

2.1 Offline Goal-Conditioned Reinforcement Learning

We consider discounted offline goal-conditioned reinforcement learning with state space $\mathcal{S}$, action space $\mathcal{A}$, goal space $\mathcal{G}$, goal-independent transition kernel $P(\cdot \mid s, a)$, initial-state distribution p_0,

and discount factor $\gamma \in (0, 1)$. The agent is conditioned on a goal $g \in \mathcal{G}$ and receives the sparse time-to-goal reward

$$r(s, g) = -\mathbf{1}\{s \neq g\}, \qquad V(g, g) = 0. \tag{1}$$

For notational simplicity, $s = g$ denotes goal attainment; when goals are represented by observations or learned features, it denotes the corresponding goal-matching event.

Learning is performed from a fixed offline dataset of trajectories

$$\mathcal{D} = \{\tau_i\}_{i=1}^{N}, \qquad \tau_i = (s_0, a_0, s_1, a_1, \ldots, s_{T_i}), \tag{2}$$

collected by unknown behavior policies. Training samples are formed by drawing transitions from $\mathcal{D}$ and assigning goals through the task-specific goal-sampling and relabeling procedure. These operations induce a joint training distribution $\nu_{\mathcal{D}}$ over relabeled one-step tuples (s, a, s', g), where the sampled goal need not be independent of the transition. The learning objective is a primitive goal-conditioned policy $\pi(a \mid s, g)$ that maximizes expected discounted return. For a policy π, we denote its goal-conditioned action and state values by $Q_\gamma^\pi(s, a, g)$ and $V_\gamma^\pi(s, g)$, respectively. We write $Q_\gamma^\star(s, a, g) := \sup_\pi Q_\gamma^\pi(s, a, g)$ for the optimal goal-conditioned action value.

2.2 Goal-Conditioned Implicit Q-Learning

SSVL uses goal-conditioned implicit Q-learning (GCIQL), based on Implicit Q-Learning (IQL) (Kostrikov et al., 2022; Park et al., 2023), as its value-learning backbone. IQL avoids maximizing the critic over out-of-distribution actions by fitting a separate state value through expectile regression. Let Q_ϕ and V_ψ denote the action and state values and let $\overline{Q}$ denote a slowly updated target critic. With expectile level $\tau_{\text{exp}} \in (1/2, 1)$ and

$$L_2^{\tau_{\text{exp}}}(u) := |\tau_{\text{exp}} - \mathbf{1}\{u < 0\}|\, u^2, \tag{3}$$

the state value is fit using

$$\mathcal{L}_{\text{IQL-}V}(\psi) = \mathbb{E}_{(s,a,s',g)\sim\nu_{\mathcal{D}}}\left[L_2^{\tau_{\text{exp}}}\big(\overline{Q}(s, a, g) - V_\psi(s, g)\big)\right]. \tag{4}$$

The corresponding primitive one-step critic target is

$$Y_t^{\text{1-step}} = r(s_t, g) + \gamma V_\psi(s_{t+1}, g), \qquad \mathcal{L}_{\text{IQL-}Q}(\phi) = \mathbb{E}_{(s_t,a_t,s_{t+1},g)\sim\nu_{\mathcal{D}}}\left[\big(Q_\phi(s_t, a_t, g) - Y_t^{\text{1-step}}\big)^2\right]. \tag{5}$$

SSVL modifies this critic target while retaining the same Q–V decomposition. The primitive policy is extracted from the learned values using the benchmark-specific offline policy-learning backend; our experiments use either advantage-weighted regression (AWR) (Peng et al., 2019) or DDPG+BC (Fujimoto & Gu, 2021).

Exact multi-step propagation. For comparison, consider a logged segment of realized length $\kappa \geq 1$ from (s_t, a_t) to the observed endpoint $s_t^+ = s_{t+\kappa}$. Standard multi-step propagation retains primitive-time discounting (Sutton & Barto, 2018):

$$Y_\kappa^{\text{exact}} = \sum_{\ell=0}^{\kappa-1} \gamma^\ell r(s_{t+\ell}, g) + \gamma^\kappa V_\psi(s_t^+, g) = -\sum_{\ell=0}^{\kappa-1} \gamma^\ell + \gamma^\kappa V_\psi(s_t^+, g), \tag{6}$$

where the second equality uses Eq. 1 for a segment before goal attainment, with $V_\psi(g, g) = 0$ for a goal-stopped endpoint. Looking farther ahead therefore does not itself shorten the primitive-time discounting: the endpoint remains weighted by γ^κ. Section 3 contrasts this exact relation with the single-discount shortcut propagation used by SSVL.

3 Shortcut Bellman Propagation and Primitive-Action Ordering

Under the sparse goal-reaching reward $r(s, g) = -\mathbf{1}\{s \neq g\}$, all nonterminal actions share the same immediate reward, so their relative preference is determined by continuation values. When competing action values are weakly separated, estimation error can change their ordering, echoing the classical action-gap perspective (Bellemare et al., 2016). In long-horizon GCRL, information about goal progress must propagate through long Bellman chains before inducing value differences between primitive actions. SSVL shortens this path through single-discount shortcut relations. We first analyze how single-discount propagation changes state-value separation and then separately characterize when shortcut targets preserve primitive-action preference under logged continuation.

3.1 Bellman Shortcuts Improve Progress-Order Robustness

We first isolate the fixed-length nonterminal branch of SSVL. For a logged trajectory, suppose the next K steps neither reach the goal nor cross a trajectory boundary, giving

$$\sigma_t^K = (s_t, a_t, s_{t+1}, a_{t+1}, \ldots, s_{t+K}),$$

with endpoint $s_t^+ := s_{t+K}$. The critic evaluates the current action a_t, while $a_{t+1}, \ldots, a_{t+K-1}$ form the logged continuation to s_t^+. Goal- or boundary-stopped segments with realized length $\kappa < K$ are introduced in Section 4.

For $\xi \in (0, 1)$, let

$$h_m(\xi) := \sum_{\ell=0}^{m-1} \xi^\ell = \frac{1-\xi^m}{1-\xi}. \tag{7}$$

Specializing Eq. 6 to this segment gives

$$Y_K^{\text{exact}} = -h_K(\gamma) + \gamma^K V(s_t^+, g), \qquad Y_K^{\text{sh}}(c_K) = -c_K + \gamma V(s_t^+, g), \quad c_K > 0. \tag{8}$$

Here $c_K = 1$ gives a unit-reward shortcut, while $c_K = h_K(\gamma)$ retains the accumulated segment reward and changes only the endpoint discount. The segment defines only a training-time Bellman relation; deployment still selects one primitive action per environment step.

To isolate propagation depth, consider the idealized shortcut chain

$$x_j \longrightarrow x_{j-1} \longrightarrow \cdots \longrightarrow x_0 = g, \tag{9}$$

where each edge represents one nonterminal K-step relation and x_{i-1} represents greater progress toward g than x_i. Repeated composition of Eq. 8 gives

$$V_{\text{exact}}(x_j, g) = -h_K(\gamma) h_j(\gamma^K), \qquad V_{\text{sh}}(x_j, g) = -c_K h_j(\gamma). \tag{10}$$

Both expressions in Eq. 10 are instances of the same chain recursion, differing only in their effective discount and per-relation reward. To compare their propagation behavior independently of this reward scale, consider

$$U_{\xi,c}(x_0, g) = 0, \qquad U_{\xi,c}(x_j, g) = -c + \xi U_{\xi,c}(x_{j-1}, g), \quad c > 0,$$

which gives $U_{\xi,c}(x_j, g) = -c\, h_j(\xi)$. We then measure the value separation between consecutive states relative to their value scale:

$$\rho_j(\xi) := \frac{U_{\xi,c}(x_{j-1}, g) - U_{\xi,c}(x_j, g)}{|U_{\xi,c}(x_j, g)|} = \frac{\xi^{j-1}}{h_j(\xi)}. \tag{11}$$

The exact and shortcut recursions correspond to $(\xi, c) = (\gamma^K, h_K(\gamma))$ and (γ, c_K), respectively, so the normalization isolates the effect of the effective discount from the per-relation reward scale.

Proposition 3.1 (Amplified ordering along the shortcut chain)**.** *For every $j \geq 2$ and $K > 1$,*

$$\rho_j(\gamma) > \rho_j(\gamma^K). \tag{12}$$

Moreover, if the normalized estimates of two consecutive states each have absolute error at most ε_{ord}, their order is preserved whenever $\rho_j(\xi) > 2\varepsilon_{\text{ord}}$.

The chain in Eq. 9 spans jK primitive transitions but only j shortcut relations. Replacing the block discount γ^K by γ therefore increases the value separation between consecutive shortcut states relative to their value scale. This is a propagation-depth effect rather than a claim about Euclidean goal distance; a standard multi-step target retaining $\gamma^K V(s_t^+, g)$ does not obtain the same amplification. See Appendix B.2 for the proof.

3.2 Logged Continuation and Primitive-Action Preference

Improved ordering between shortcut states does not by itself guarantee the correct ordering of primitive actions. In offline data, the endpoint of a K-step segment is jointly determined by the initial action and the subsequent logged continuation. We therefore characterize when the data-induced endpoint distribution remains informative about primitive-action preference.

Let $\mu_K^{\mathcal{D}}$ denote the population distribution of nonterminal shortcut tuples (S, A, G, S^+) induced jointly by the offline data and the segment–goal sampling procedure. For each supported (s, a, g) and every measurable $\mathcal{B} \subseteq \mathcal{S}$, define the conditional endpoint law

$$\mathsf{P}_K^{\mathcal{D}}(\mathcal{B} \mid s, a, g) := \Pr_{\mu_K^{\mathcal{D}}}\big(S^+ \in \mathcal{B} \mid S = s, A = a, G = g\big). \tag{13}$$

This law integrates over the subsequent actions in the logged segment and does not require the dataset to be generated by a single stationary behavior policy.

Fix (s, g) and two primitive actions, and for any shortcut target Y_K define

$$a^+ \succ_{\mathrm{prim}} a^- \quad \Longleftrightarrow \quad Q_\gamma^\star(s, a^+, g) > Q_\gamma^\star(s, a^-, g), \tag{14}$$

$$\Delta_K(Y) := \mathbb{E}[Y_K \mid s, a^+, g] - \mathbb{E}[Y_K \mid s, a^-, g]. \tag{15}$$

Because larger values are preferred under the negative step-reward convention, $\Delta_K(Y) > 0$ means that the shortcut objective agrees with $a^+ \succ_{\mathrm{prim}} a^-$. Let $V^{(K)}$ denote the population value entering the K-shortcut target, and define the behavior-conditioned shortcut signal

$$B_K(s; a^+, a^-, g) := \gamma\Big[\mathbb{E}_{\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a^+, g)} V^{(K)}(S^+, g) - \mathbb{E}_{\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a^-, g)} V^{(K)}(S^+, g)\Big]. \tag{16}$$

Proposition 3.2 (Uniform shortcut action margin)**.** *Suppose every nonterminal segment receives the same shortcut reward $-c_K$. Then*

$$\Delta_K^{\mathrm{uni}} = B_K. \tag{17}$$

Consequently, uniform shortcut propagation preserves $a^+ \succ_{\mathrm{prim}} a^-$ if and only if $B_K > 0$.

The quantity B_K makes the required transfer condition explicit. It is small when the logged continuation masks the effect of the initial action and negative when that continuation reverses the primitive preference. OTA uses the same unit-reward single-discount recursion at the high-level state-value layer, whereas the target analyzed here is attached directly to primitive state–action pairs (Ahn et al., 2025). See Appendix B.3 for the proof.

3.3 Reference-Selective Reward Compression

Uniform shortcut propagation relies on the mean endpoint-value difference captured by B_K. SSVL additionally uses a frozen shortcut-aligned reference to identify segments with larger estimated progress toward the goal. Let $V_{\mathrm{ref}}^{(K)}$ denote the population reference induced by shortcut-aligned pretraining and $\overline{V}_{\mathrm{ref}}$ its frozen learned approximation. Define

$$Z_K(s, S^+, g) = V_{\mathrm{ref}}^{(K)}(S^+, g) - V_{\mathrm{ref}}^{(K)}(s, g), \qquad \widehat{Z}_K(s, S^+, g) = \overline{V}_{\mathrm{ref}}(S^+, g) - \overline{V}_{\mathrm{ref}}(s, g). \tag{18}$$

The gate accepts a segment when $I_K = \mathbf{1}\{\widehat{Z}_K(s, S^+, g) \geq w\}$.

Although the score is state-based, its conditional distribution depends on the initial action through $\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a, g)$. For a threshold u, define

$$p_K^{\mathrm{prog}}(s, a, g; u) := \Pr\big(Z_K(s, S^+, g) \geq u \mid s, a, g\big). \tag{19}$$

Assumption 3.3 (Local primitive-to-shortcut preference alignment)**.** For $a^+ \succ_{\mathrm{prim}} a^-$, there exist $\delta_{\mathrm{sh}} > 0$ and $\eta_K > 0$ such that

$$p_K^{\mathrm{prog}}(s, a^+, g; w + \delta_{\mathrm{sh}}) - p_K^{\mathrm{prog}}(s, a^-, g; w - \delta_{\mathrm{sh}}) \geq \eta_K. \tag{20}$$

Assumption 3.3 requires the primitive preference to remain visible in the probability of producing a favorable K-step outcome under the logged continuation. It is local to the action pair and gate threshold and does not require primitive and shortcut values to be numerically equivalent.

For a nonterminal K-step segment, SSVL uses target

$$Y_K^{\mathrm{sel}} = -h_K(\gamma) + I_K\big(h_K(\gamma) - 1\big) + \gamma V^{(K)}(S^+, g). \tag{21}$$

Every segment uses the same single-discount bootstrap; the gate changes only the reward term, from $-h_K(\gamma)$ to -1 when accepted. Define

$$\begin{aligned} p_K^I(s, a, g) &:= \Pr(I_K = 1 \mid s, a, g), \\ M_K(s; a^+, a^-, g) &:= \big(h_K(\gamma) - 1\big)\big[p_K^I(s, a^+, g) - p_K^I(s, a^-, g)\big]. \end{aligned} \tag{22}$$

Write $\Delta_K^{\mathrm{sel}} := \Delta_K(Y_K^{\mathrm{sel}})$ and $M_K := M_K(s; a^+, a^-, g)$. Replacing $V^{(K)}$ by $\widehat{V}^{(K)}$ with the gate fixed gives the corresponding margin $\widehat{\Delta}_K^{\mathrm{sel}}$.

Theorem 3.4 (Reference-selective primitive-action target ordering)**.** *Suppose Assumption 3.3 holds and*

$$\|\overline{V}_{\mathrm{ref}} - V_{\mathrm{ref}}^{(K)}\|_\infty \le \varepsilon_{\mathrm{ref}}, \qquad 2\varepsilon_{\mathrm{ref}} \le \delta_{\mathrm{sh}}.$$

Then

$$p_K^I(s, a^+, g) - p_K^I(s, a^-, g) \ge \eta_K, \qquad \Delta_K^{\mathrm{sel}} = B_K + M_K \ge B_K + \big(h_K(\gamma) - 1\big)\eta_K. \tag{23}$$

Moreover, if $\|\widehat{V}^{(K)} - V^{(K)}\|_\infty \le \varepsilon_V$*, then*

$$\widehat{\Delta}_K^{\mathrm{sel}} \ge B_K + \big(h_K(\gamma) - 1\big)\eta_K - 2\gamma\varepsilon_V. \tag{24}$$

Hence the estimated selective shortcut target preserves the primitive preference $a^+ \succ_{\mathrm{prim}} a^-$ *whenever the right-hand side of Eq. 24 is positive.*

This is a target-ordering result, not a guarantee on the learned critic or policy.

For rejected segments, SSVL retains $-h_K(\gamma)$ but replaces the primitive-time bootstrap $\gamma^K V^{(K)}(S^+, g)$ with $\gamma V^{(K)}(S^+, g)$. Thus the single-discount continuation is universal, whereas reward compression is selective; under Assumption 3.3, the latter contributes the additional action margin M_K. These local conditions weaken when the current action has little influence on the endpoint, when logged continuation reverses that influence, or when the reference cannot resolve the threshold-level preference. See Appendix B.5 for the proof.

4 Selective Shortcut Value Learning

Section 3 analyzed how single-discount shortcut relations affect progress-state separation and primitive-action preference on fixed-length chains. We now instantiate these mechanisms in the full SSVL algorithm using stopped trajectory segments, a frozen reference that controls reward compression, and an auxiliary consistency loss for shortcut-connected state values.

4.1 Selective Shortcut Critic Learning

Consider a logged trajectory $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T)$, sampled time t, and goal g. If $s_t = g$, we treat the relation as absorbing and use a zero target. Otherwise, starting from (s_t, a_t), SSVL follows the trajectory for at most K primitive steps and stops earlier at the goal or trajectory boundary. Let κ_t^g be the first positive offset at which g is reached, with $\kappa_t^g = \infty$ if this does not occur before the boundary. The realized length and endpoint are

$$\kappa_t := \min\{K, \kappa_t^g, T - t\}, \qquad s_t^+ := s_{t+\kappa_t}. \tag{25}$$

Thus $\kappa_t = K$ gives a full-K segment and $\kappa_t < K$ a goal- or boundary-stopped segment. The endpoint s_t^+ is an observed future state, not a subgoal chosen by the policy: the critic evaluates the primitive action a_t, while the logged continuation determines the realized endpoint.

SSVL pretrains a reference value $\overline{V}_{\mathrm{ref}}$ using the same shortcut length K as downstream learning and freezes it before downstream learning. For a stopped segment, the base gate threshold w is rescaled to its realized length,

$$\widehat{Z}_t := \overline{V}_{\mathrm{ref}}(s_t^+, g) - \overline{V}_{\mathrm{ref}}(s_t, g), \qquad w_t := w\frac{1 - \gamma^{\kappa_t/K}}{1 - \gamma}, \qquad I_t := \mathbf{1}\{\widehat{Z}_t \ge w_t\}. \tag{26}$$

A full-K segment uses threshold w, while shorter segments use the rescaled threshold above. The gate I_t controls only reward compression, not whether a shortcut continuation is used.

For $\Sigma_t = (s_t, a_t, g, s_t^+, \kappa_t)$ and h_m from Eq. 7, SSVL trains the primitive-action critic with

$$Y_t^{\mathrm{SSVL}} = -h_{\kappa_t}(\gamma) + I_t\big(h_{\kappa_t}(\gamma) - 1\big) + \gamma V_\psi(s_t^+, g). \tag{27}$$

Equivalently,

$$Y_t^{\mathrm{SSVL}} = \begin{cases} -1 + \gamma V_\psi(s_t^+, g), & I_t = 1, \\ -h_{\kappa_t}(\gamma) + \gamma V_\psi(s_t^+, g), & I_t = 0. \end{cases} \tag{28}$$

Algorithm 1 Selective Shortcut Value Learning (SSVL)

1: **Input:** dataset $\mathcal{D}$, shortcut length K, threshold w, SVC weight λ_{SVC}.
2: Pretrain $\overline{V}_{\mathrm{ref}}$ on shortcut relations and freeze it.
3: **for** each downstream gradient step **do**
4: Sample current tuples/goals; construct $\Sigma_t = (s_t, a_t, g, s_t^+, \kappa_t)$ by Eq. 25.
5: Compute I_t by Eq. 26 and Y_t^{SSVL} by Eq. 27.
6: Update Q_ϕ by Eq. 29 and V_ψ by Eq. 31.
7: Update π_η from current tuples using the policy-goal sampler and AWR or DDPG+BC.
8: **end for**
9: **Return:** primitive goal-conditioned policy $\pi_\eta(a \mid s, g)$.

Hence the single-discount continuation is universal: every segment uses the bootstrap $\gamma V_\psi(s_t^+, g)$. Reward compression is selective: acceptance replaces the discounted segment reward $-h_{\kappa_t}(\gamma)$ by -1, while rejection retains it. For a goal-stopped segment, $s_t^+ = g$ and $V_\psi(g, g) = 0$.

The critic is fit by squared Bellman regression,

$$\mathcal{L}_Q(\phi) = \mathbb{E}_{\Sigma_t}\left[\left(Q_\phi(s_t, a_t, g) - Y_t^{\mathrm{SSVL}}\right)^2\right]. \tag{29}$$

4.2 Shortcut Value Consistency

The critic and state value receive different shortcut supervision. Each segment directly trains $Q_\phi(s_t, a_t, g)$ with the endpoint target in Eq. 27, whereas the IQL expectile objective fits $V_\psi(s_t, g)$ locally from action values and does not directly couple it to $V_\psi(s_t^+, g)$. This asymmetry motivates an explicit cross-state relation for V.

Shortcut Value Consistency (SVC) adds an explicit cross-state relation. For each sampled nonterminal shortcut edge $s_t \to s_t^+$, define

$$e_t := V_\psi(s_t, g) + 1 - \gamma V_\psi(s_t^+, g), \qquad \mathcal{L}_{\mathrm{SVC}}(\psi) := \mathbb{E}[e_t^2]. \tag{30}$$

SVC therefore encourages the canonical unit-reward relation $V_\psi(s_t, g) = -1 + \gamma V_\psi(s_t^+, g)$. This supplies a common single-discount consistency relation across shortcut-connected states; $\mathcal{L}_{\mathrm{SVC}}$ remains auxiliary rather than the Bellman residual of the selective critic target.

SVC both couples shortcut-connected values to the single-discount recursion and backpropagates through source and successor evaluations. Appendix C formalizes these local consistency and two-sided optimization properties and compares the successor stop-gradient variant. These results do not remove state–action–goal coverage requirements or imply global neural-network convergence or policy improvement.

The complete state-value objective is

$$\mathcal{L}_V(\psi) = \mathcal{L}_{\mathrm{base}}(\psi) + \lambda_{\mathrm{SVC}}\mathcal{L}_{\mathrm{SVC}}(\psi), \tag{31}$$

where $\mathcal{L}_{\mathrm{base}}$ is the expectile loss in Eq. 4.

4.3 Primitive Policy Learning and Training Procedure

SSVL does not introduce a new primitive-policy objective. The critic and state value use stopped shortcut relations, while the policy is trained from current state–action tuples using the benchmark-specific policy-goal sampler. Algorithm 1 summarizes training after the reference is pretrained and frozen.

The frozen reference and shortcut endpoints are used only during training. At deployment, SSVL executes $\pi_\eta(a \mid s, g)$ directly, without reference evaluation or subgoal prediction.

5 Experiments

We evaluate SSVL on state-based and visual locomotion and manipulation tasks from OGBench (Park et al., 2025) to assess whether its value-learning changes support competitive perfor-

Table 1: **Representative OGBench results.** Binary success (%) averaged over five test goals. SSVL reports mean ± sd over six seeds; GCIQL/CRL/HIQL are official OGBench results, SAW/published OTA retain source-paper protocols, and † denotes our six-seed OTA reproduction. "–" denotes unavailable results; bold marks the highest mean.

Obs.	Domain	Task	GCIQL	CRL	HIQL	SAW	OTA	SSVL (ours)
State	Manipulation	Cube double-play	40 ± 5	10 ± 2	6 ± 2	40 ± 7	$3 \pm 1^{\dagger}$	$\mathbf{66 \pm 3}$
State	Manipulation	Scene play	51 ± 4	19 ± 2	38 ± 3	63 ± 6	$31 \pm 3^{\dagger}$	$\mathbf{85 \pm 4}$
State	PointMaze	giant-stitch	0 ± 0	0 ± 0	0 ± 0	0 ± 0	52 ± 7	$\mathbf{59 \pm 6}$
State	AntMaze	large-navigate	34 ± 4	83 ± 4	91 ± 2	90 ± 3	92 ± 1	$\mathbf{94 \pm 3}$
State	AntMaze	medium-explore	13 ± 2	3 ± 2	37 ± 10	25 ± 4	$\mathbf{94 \pm 3}$	$\mathbf{94 \pm 1}$
State	AntMaze	large-stitch	7 ± 2	11 ± 2	67 ± 5	66 ± 9	$\mathbf{84 \pm 3}$	23 ± 4
State	HumanoidMaze	giant-navigate	0 ± 0	3 ± 2	12 ± 4	35 ± 4	$\mathbf{92 \pm 1}$	67 ± 9
Visual	AntMaze	giant-navigate	0 ± 0	47 ± 2	6 ± 4	10 ± 2	$19 \pm 18^{\dagger}$	$\mathbf{62 \pm 5}$
Visual	Cube	double-noisy	22 ± 2	6 ± 3	59 ± 3	–	$\mathbf{65 \pm 2}$	24 ± 3

mance with primitive-policy deployment and how its main components contribute under controlled diagnostics.

5.1 Experimental Setup

We report binary goal-reaching success on OGBench (Park et al., 2025). Baselines include flat GCIQL (Park et al., 2025) and CRL (Eysenbach et al., 2022), hierarchical HIQL (Park et al., 2023) and OTA (Ahn et al., 2025), and flat policy-bootstrapping SAW (Zhou & Kao, 2025). SSVL uses the benchmark-specific AWR (Peng et al., 2019) or DDPG+BC (Fujimoto & Gu, 2021) actor backend. Table 1 gives representative comparisons and records result provenance; complete results and implementation details are in Appendices F, F.5, and D.1–D.4.

5.2 Performance Across State and Visual Tasks

Table 1 reveals distinct patterns across task families. SSVL is particularly strong on state-based manipulation, attaining the highest reported mean on the representative Cube and Scene tasks. It is also competitive on several maze settings, including AntMaze large-navigate and medium-explore and PointMaze giant-stitch. This strength extends to visual navigation: on Visual-AntMaze giant-navigate, SSVL attains the highest reported mean among the compared methods. Together, these results show that SSVL can support strong primitive-policy control on both state and visual tasks.

The gains are nevertheless task dependent. OTA remains substantially stronger on some AntMaze stitching and HumanoidMaze settings and on visual manipulation, whereas SSVL is markedly stronger on visual AntMaze navigation. The split therefore cannot be attributed to horizon length, trajectory stitching, or visual observations alone: SSVL succeeds on some tasks with each of these characteristics and fails on others. This heterogeneity suggests that shorter value propagation is helpful but not sufficient; the informativeness of supervision for the current primitive action may also be important.

A plausible remaining bottleneck is direct primitive-action supervision for the final goal. SSVL learns $(s, g_{\text{final}}) \mapsto a$, whereas hierarchical policies can factor control through $(s, g_{\text{final}}) \mapsto w$ and the more local $(s, w) \mapsto a$. Under stitching and exploration samplers, distant or cross-trajectory goals may provide weak supervision for the logged current action. SAW offers complementary evidence that nearby-subgoal supervision can improve this state–action–goal pairing. Accordingly, SSVL's failures are consistent with a supervision bottleneck and do not by themselves imply an intrinsic ceiling of primitive-policy deployment. Improving goal–action pairing may provide further headroom, although we do not claim that it explains every failure; see Appendices D.1 and F.5.

5.3 What Drives SSVL?

Figure 2 diagnoses single-discount continuation, reference-selective reward compression, and the IQL-specific SVC auxiliary; full analyses are in Appendices E.1–E.3.

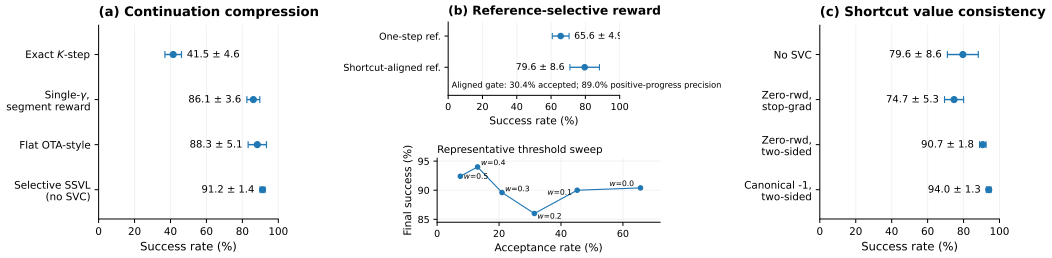

Figure 2: **Mechanism analysis of SSVL. (a)** AntMaze-large-navigate target ablations separate single-discount continuation and selective reward compression; "Flat OTA-style" is a matched flat control, not hierarchical OTA. **(b)** AntMaze-medium-explore compares one-step and shortcut-aligned references and diagnoses gate selectivity. **(c)** Matched SVC controls compare no SVC, successor stop-gradient, two-sided optimization, and the canonical unit-cost relation. Error bars denote standard deviations where applicable.

Continuation compression. Figure 2(a) shows that shortening the shortcut continuation accounts for the largest target-level gain on AntMaze large-navigate. Replacing the exact γ^K bootstrap by a single γ, while retaining the same accumulated segment reward and disabling SVC, raises success from 41.5 to 86.1. A matched comparison on Visual-AntMaze giant-navigate shows the same qualitative effect, increasing final success from 22.4 ± 2.0 to 60.3 ± 0.6, with improvements on all five evaluation goals. Thus, the continuation-compression effect is not confined to the state-based diagnostic. The contribution of reward compression is more task dependent, as examined next and in Appendix E.1.

Reference-selective reward. Figure 2(b) shows that the benefit of selectivity depends on how the frozen reference is constructed. Replacing the one-step reference by a shortcut-aligned reference improves no-SVC AntMaze medium-explore from 65.6 to 79.6 while changing only the reward-compression gate. The per-goal decomposition further shows that selectivity can be consequential where uniform shortcut supervision is weak: on the hardest goal, the matched Flat OTA-style control reaches only 3.3, whereas the shortcut-aligned selective target reaches 45.3. At the default threshold, 89.0% of accepted full-K relations exhibit positive planar progress, and a separate threshold sweep shows no monotone relationship between acceptance rate and success. Together, these results indicate that the gate provides structured, task-dependent gains rather than simply from accepting more segments. See Appendix E.2.

Shortcut value consistency. Figure 2(c) isolates the optimization-side role of two-sided endpoint coupling. Under the matched zero-reward relation, replacing successor stop-gradient by two-sided optimization increases success from 74.7 to 90.7, while the canonical unit-cost relation further reaches 94.0. Within these matched controls, two-sided coupling accounts for the larger improvement, with the reward convention providing a smaller refinement. See Appendix E.3.

Mechanism synthesis. Together, these diagnostics support SSVL's mechanism decomposition. Single-discount continuation is the primary propagation mechanism, while reference selectivity provides task-dependent refinement through selective reward compression. SVC serves a different, auxiliary role under the IQL backbone by helping the separately learned state value preserve shortcut relations.

6 Conclusion

We studied long-horizon offline GCRL through primitive-action ordering. SSVL shortens Bellman propagation directly in primitive action-value learning using stopped single-discount shortcuts, with reference-selective reward compression and SVC as training-time refinements. Theory isolates the state-ordering gain from shorter propagation and characterizes when logged continuation and reference selectivity yield primitive-consistent shortcut target ordering, while experiments show competitive results across state-based and visual OGBench tasks. Remaining limitations concern logged continuation and state–action–goal supervision.

AI USE STATEMENT

In this work, we used generative AI tools to assist with prose refinement, manuscript-level consistency and presentation checks, and review of mathematical derivations and proofs for potential errors or inconsistencies. Generative AI tools were additionally used for code review and debugging. The research ideas, conceptual framework, theoretical results, methodology, and experimental design were developed by the authors, and generative AI tools were not used to conduct the experiments or generate experimental results. AI-assisted suggestions were not accepted automatically: the authors manually reviewed each proposed revision, independently checked all mathematical arguments and claims, and verified the final prose against the underlying theory, experimental evidence, and cited sources. We take responsibility for the final content of this work, including all text, claims, proofs, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The appendix provides the complete theoretical proofs, goal-sampling and relabeling procedures, implementation details, hyperparameters, mechanism ablations, and full per-goal results used in this paper. SSVL is implemented on top of the same benchmark-specific policy backends used in our experiments, and all reported training-time components, including the frozen shortcut reference and SVC variants, are specified in the accompanying experimental documentation.

REFERENCES

Hongjoon Ahn, Heewoong Choi, Jisu Han, and Taesup Moon. Option-aware temporally abstracted value for offline goal-conditioned reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3337.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), 2017. doi: 10.1609/aaai.v31i1.10916.

Marc G. Bellemare, Georg Ostrovski, Arthur Guez, Philip S. Thomas, and Rémi Munos. Increasing the action gap: New operators for reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), 2016. doi: 10.1609/aaai.v30i1.10303.

Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. Goal-conditioned reinforcement learning with imagined subgoals. In *International conference on machine learning*, pp. 1430–1440. PMLR, 2021.

Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12875–12884, 2020.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymyr Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1407–1416. PMLR, 2018.

Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. Search on the replay buffer: Bridging planning and reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. C-learning: Learning to achieve goals via recursive classification. In *International Conference on Learning Representations*, 2021.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. *Advances in Neural Information Processing Systems*, 35:35603–35620, 2022.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems*, 34:20132–20145, 2021.

Dibya Ghosh, Abhishek Gupta, Ashwin Reddy, Justin Fu, Coline Manon Devin, Benjamin Eysenbach, and Sergey Levine. Learning to reach goals via iterated supervised learning. In *International Conference on Learning Representations*, 2021.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

Christopher Hoang, Sungryull Sohn, Jongwook Choi, Wilka Carvalho, and Honglak Lee. Successor feature landmarks for long-horizon goal-conditioned reinforcement learning. *Advances in neural information processing systems*, 34:26963–26975, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit Q-learning. In *International Conference on Learning Representations*, 2022.

Andrew Levy, George Konidaris, Robert Platt, and Kate Saenko. Learning multi-level hierarchies with hindsight. In *International Conference on Learning Representations*, 2019.

Corey Lynch, Mohi Khansari, Ted Xiao, Vikash Kumar, Jonathan Tompson, Sergey Levine, and Pierre Sermanet. Learning latent plans from play. In *Conference on robot learning*, pp. 1113–1132. PMLR, 2020.

Yecheng Jason Ma, Jason Yan, Dinesh Jayaraman, and Osbert Bastani. Offline goal-conditioned reinforcement learning via f-advantage regression. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Ofir Nachum, Shixiang Shane Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

Ofir Nachum, Haoran Tang, Xingyu Lu, Shixiang Gu, Honglak Lee, and Sergey Levine. Why does hierarchy (sometimes) work so well in reinforcement learning? *arXiv preprint arXiv:1909.10618*, 2019.

Soroush Nasiriany, Vitchyr Pong, Steven Lin, and Sergey Levine. Planning with goal-conditioned policies. *Advances in neural information processing systems*, 32, 2019.

Seohong Park, Dibya Ghosh, Benjamin Eysenbach, and Sergey Levine. HIQL: Offline goal-conditioned RL with latent states as actions. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-1515.

Seohong Park, Tobias Kreiman, and Sergey Levine. Foundation policies with Hilbert representations. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39737–39761. PMLR, 2024.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. OGBench: Benchmarking offline goal-conditioned RL. In *International Conference on Learning Representations*, 2025.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Doina Precup. *Temporal Abstraction in Reinforcement Learning*. PhD thesis, University of Massachusetts Amherst, 2000.

Zhizhou Ren, Kefan Dong, Yuan Zhou, Qiang Liu, and Jian Peng. Exploration via hindsight goal generation. *Advances in Neural Information Processing Systems*, 32, 2019.

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

Harshit Sikchi, Rohan Chitnis, Ahmed Touati, Alborz Geramifard, Amy Zhang, and Scott Niekum. Score models for offline goal-conditioned reinforcement learning. In *International Conference on Learning Representations*, 2024.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 2 edition, 2018.

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. doi: 10.1016/S0004-3702(99)00052-1.

Tongzhou Wang, Antonio Torralba, Phillip Isola, and Amy Zhang. Optimal goal-reaching reinforcement learning via quasimetric learning. In *International Conference on Machine Learning*, pp. 36411–36430. PMLR, 2023.

Rui Yang, Yiming Lu, Wenzhe Li, Hao Sun, Meng Fang, Yali Du, Xiu Li, Lei Han, and Chongjie Zhang. Rethinking goal-conditioned supervised learning and its connection to offline RL. In *International Conference on Learning Representations*, 2022.

Rui Yang, Lin Yong, Xiaoteng Ma, Hao Hu, Chongjie Zhang, and Tong Zhang. What is essential for unseen goal generalization of offline goal-conditioned rl? In *International Conference on Machine Learning*, pp. 39543–39571. PMLR, 2023.

Sirui Zheng, Chenjia Bai, Zhuoran Yang, and Zhaoran Wang. How does goal relabeling improve sample efficiency? In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 61246–61266. PMLR, 2024.

John L. Zhou and Jonathan C. Kao. Flattening hierarchies with policy bootstrapping. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2781.

A Related Work

Offline goal-conditioned reinforcement learning. Goal-conditioned reinforcement learning (GCRL) learns reusable behavior by conditioning policies or value functions on desired outcomes (Schaul et al., 2015; Andrychowicz et al., 2017). Offline GCRL extends this objective to fixed trajectory datasets, where sparse goal-reaching supervision is commonly strengthened through hindsight and goal relabeling (Andrychowicz et al., 2017; Ren et al., 2019; Yang et al., 2023; Zheng et al., 2024). Supervised and imitation-style approaches include GCSL and its subsequent analysis (Ghosh et al., 2021; Yang et al., 2022), while other methods derive offline objectives through distribution or occupancy matching, including GoFAR (Ma et al., 2022). Goal-conditioned value learning has also been formulated through recursive classification and contrastive objectives (Eysenbach et al., 2021; 2022), learned geometric or temporal structure (Wang et al., 2023; Park et al., 2024), and more recent score-based formulations (Sikchi et al., 2024). OGBench consolidates many of these directions into a common suite of long-horizon navigation and manipulation tasks with state and visual observations (Park et al., 2025). SSVL is complementary to these approaches: it does not introduce a new goal-relabeling or representation objective, but instead studies how long-range goal information is propagated through a primitive action-conditioned value objective once a training state–action–goal relation has been formed.

Long-horizon subgoal supervision and flat deployment. A prominent response to distant-goal difficulty is to introduce intermediate states through planning, landmarks, or hierarchical control (Nasiriany et al., 2019; Hoang et al., 2021; Eysenbach et al., 2019; Chaplot et al., 2020; Levy et al., 2019; Nachum et al., 2018). Hierarchical GCRL exploits the recursive structure of goal reaching by training a high-level subgoal policy and a low-level primitive controller; HIQL is a strong offline instance that conditions primitive actions on nearby subgoals while extracting both policy levels from a shared goal-conditioned value function (Park et al., 2023). Prior analyses of hierarchy

have also argued that some of its empirical benefits can arise from mechanisms such as improved exploration or multi-step training rather than from hierarchical execution alone (Nachum et al., 2019). More closely related to flat deployment, RIS learns imagined intermediate subgoals and regresses a full-goal policy toward a subgoal-conditioned target policy (Chane-Sane et al., 2021). SAW extends this policy-bootstrapping perspective to offline GCRL without a generative subgoal model: it samples in-trajectory waypoints, trains a subgoal-conditioned target policy, and advantage-weights the regression of a single final-goal policy toward that target (Zhou & Kao, 2025). Importantly, SAW retains one-step GCIVL for value learning, so its long-horizon shortcut in supervision occurs on the *policy* side. SSVL instead modifies the propagation structure of the primitive action-conditioned critic itself. Thus HIQL, RIS/SAW, and SSVL exploit related long-to-short structure at different points in the learning pipeline: hierarchical control, policy bootstrapping, and value propagation, respectively.

Action separation under long-horizon value learning. Our focus on primitive-action ordering is also related to the classical action-gap perspective. The action gap—the difference between the value of the best action and competing actions—governs how robust greedy action selection is to approximation or estimation error; prior work has therefore studied Bellman operators that enlarge this gap while preserving optimal policies (Bellemare et al., 2016). Long-horizon offline GCRL raises a different but related problem. HIQL and SAW emphasize that sparse rewards and discounting can make primitive actions difficult to distinguish with respect to distant goals, motivating supervision through nearby subgoals (Park et al., 2023; Zhou & Kao, 2025). SSVL does not apply an optimality-preserving action-gap operator. Instead, Section 3 studies how shortening Bellman propagation changes progress-state separation and separately characterizes when shortcut targets preserve primitive-action preference under logged continuation. This second question is nontrivial in offline data because a shortcut endpoint is jointly determined by the current primitive action and the subsequent logged continuation. Our analysis therefore complements the action-gap literature by making the logged-continuation condition for primitive-action consistency explicit.

Multi-step and temporally abstracted value learning. Standard multi-step temporal-difference methods propagate information through several observed rewards before bootstrapping, but retain primitive-time discounting through an endpoint factor γ^n (Sutton & Barto, 2018). Temporal abstraction instead has classically been formalized through options and semi-Markov decision processes, where a temporally extended behavior has its own internal policy and termination rule (Sutton et al., 1999; Precup, 2000). Subsequent work developed end-to-end option learning and option discovery, including the Option-Critic architecture (Bacon et al., 2017). SSVL differs from both standard n-step learning and executable options: its stopped segment is an observed training relation, while deployment still chooses a primitive action at every environment step. The closest recent value-side comparison is OTA (Ahn et al., 2025), which introduces option-aware temporal abstraction into the *high-level state-value update* of a hierarchical goal-conditioned policy. OTA and SSVL share a useful single-discount recursion form but apply it at different control layers. OTA shortens value propagation for subgoal ordering, whereas SSVL attaches shortcut supervision directly to a primitive state–action pair and must therefore account for the logged-continuation dependence described above. Methodologically, SSVL further separates universal single-discount continuation from reference-selective segment-reward compression and uses SVC as an auxiliary consistency objective over shortcut-connected state values under the IQL backbone.

B Proofs for Section 3

This appendix establishes the action-ordering results in Section 3. The analysis follows the same progression as the main text. We first record the elementary identities for the idealized shortcut chain, which place the exact and single-discount recursions in a common discounted form and underlie Proposition 3.1. We then pass to the data-induced endpoint law to establish Proposition 3.2. Finally, we transfer the ideal shortcut-progress separation through the frozen reference gate and control shortcut-value estimation error to establish the selective and robust primitive-action margins in Theorem 3.4.

B.1 Proof Setup and Auxiliary Identities

We begin with the notation and elementary identities used throughout the proofs. Fix a discount factor $\gamma \in (0,1)$ and a shortcut length $K \geq 1$. Whenever the data-induced endpoint law is invoked, (s,a,g) is understood to be a supported tuple for which $\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s,a,g)$ in Eq. 13 is defined. For any measurable function $f : \mathcal{S} \to \mathbb{R}$, write

$$\mathbb{E}_{K,a}\big[f(S^+)\big] := \mathbb{E}_{\mathsf{P}_K^{\mathcal{D}}(\cdot|s,a,g)}\big[f(S^+)\big]. \tag{32}$$

When two primitive actions are compared, we fix $a^+ \succ_{\mathrm{prim}} a^-$ as in Eq. 14. We use the terminal convention $V(g,g) = 0$, and all sup norms are taken over the relevant state–goal domain. In the finite-error part of Theorem 3.4, replacing $V^{(K)}$ by $\widehat{V}^{(K)}$ leaves the frozen reference $\overline{V}_{\mathrm{ref}}$—and therefore the gate I_K—fixed.

We first record the finite-sum identities shared by the exact and single-discount chain recursions. Extend Eq. 7 to $m = 0$ by setting $h_0(\xi) := 0$.

Lemma B.1 (Finite geometric-sum identities)**.** *For every $\xi \in (0,1)$ and integer $m \geq 1$,*

$$h_m(\xi) = 1 + \xi h_{m-1}(\xi), \qquad h_m(\xi) - h_{m-1}(\xi) = \xi^{m-1}. \tag{33}$$

Moreover,

$$h_m(\xi) = \frac{1-\xi^m}{1-\xi}. \tag{34}$$

Proof. For $m = 1$, the first two identities follow directly from $h_0(\xi) = 0$ and $h_1(\xi) = 1$. For $m \geq 2$, separating the zeroth term and reindexing gives

$$h_m(\xi) = \sum_{\ell=0}^{m-1} \xi^\ell = 1 + \sum_{\ell=1}^{m-1} \xi^\ell = 1 + \xi \sum_{q=0}^{m-2} \xi^q = 1 + \xi h_{m-1}(\xi). \tag{35}$$

Subtracting the two consecutive finite sums leaves only the last term,

$$h_m(\xi) - h_{m-1}(\xi) = \sum_{\ell=0}^{m-1} \xi^\ell - \sum_{\ell=0}^{m-2} \xi^\ell = \xi^{m-1}. \tag{36}$$

Finally, multiplying the defining sum by $1-\xi$ gives the telescoping identity

$$(1-\xi)h_m(\xi) = \sum_{\ell=0}^{m-1} \xi^\ell - \sum_{\ell=1}^{m} \xi^\ell = 1 - \xi^m. \tag{37}$$

Since $1-\xi > 0$, division by $1-\xi$ yields Eq. 34. □

We next formalize the generic recursion used in Section 3.1. Let $c > 0$ denote the magnitude of the negative reward assigned to one relation of the idealized chain, and for $\xi \in (0,1)$ define

$$U_{\xi,c}(x_0, g) := 0, \qquad U_{\xi,c}(x_j, g) := -c + \xi U_{\xi,c}(x_{j-1}, g), \quad j \geq 1. \tag{38}$$

Lemma B.2 (Value and separation along a discounted chain)**.** *For every $\xi \in (0,1)$, $c > 0$, and integer $j \geq 1$, the recursion in Eq. 38 satisfies*

$$U_{\xi,c}(x_j, g) = -c\, h_j(\xi). \tag{39}$$

Consequently,

$$U_{\xi,c}(x_{j-1}, g) - U_{\xi,c}(x_j, g) = c\,\xi^{j-1}, \qquad \big|U_{\xi,c}(x_j, g)\big| = c\, h_j(\xi), \tag{40}$$

and hence

$$\frac{U_{\xi,c}(x_{j-1}, g) - U_{\xi,c}(x_j, g)}{|U_{\xi,c}(x_j, g)|} = \frac{\xi^{j-1}}{h_j(\xi)}. \tag{41}$$

Proof. We first establish Eq. 39 by induction. For $j = 1$,

$$\begin{aligned} U_{\xi,c}(x_1, g) &= -c + \xi U_{\xi,c}(x_0, g) \\ &= -c \\ &= -c\, h_1(\xi). \end{aligned} \tag{42}$$

Now let $j \geq 2$ and suppose

$$U_{\xi,c}(x_{j-1}, g) = -c\, h_{j-1}(\xi). \tag{43}$$

Substituting the induction hypothesis into the recursion and using Lemma B.1 gives

$$\begin{aligned} U_{\xi,c}(x_j, g) &= -c + \xi U_{\xi,c}(x_{j-1}, g) \\ &= -c - \xi c\, h_{j-1}(\xi) \\ &= -c\big[1 + \xi h_{j-1}(\xi)\big] \\ &= -c\, h_j(\xi). \end{aligned} \tag{44}$$

Thus Eq. 39 holds for every $j \geq 1$. The consecutive-state difference then satisfies

$$\begin{aligned} U_{\xi,c}(x_{j-1}, g) - U_{\xi,c}(x_j, g) &= -c\, h_{j-1}(\xi) + c\, h_j(\xi) \\ &= c\big[h_j(\xi) - h_{j-1}(\xi)\big] \\ &= c\, \xi^{j-1}, \end{aligned} \tag{45}$$

where the last equality again uses Lemma B.1. Since $c > 0$ and $h_j(\xi) > 0$, Eq. 39 also gives

$$|U_{\xi,c}(x_j, g)| = c\, h_j(\xi). \tag{46}$$

Dividing Eq. 45 by Eq. 46 proves Eq. 41. □

Lemma B.2 applies directly to the two recursions compared in Section 3.1. The exact K-step relation corresponds to $(\xi, c) = (\gamma^K, h_K(\gamma))$, whereas the single-discount shortcut relation corresponds to $(\xi, c) = (\gamma, c_K)$. Equation 39 therefore yields

$$\begin{aligned} V_{\text{exact}}(x_j, g) &= -h_K(\gamma) h_j(\gamma^K), \\ V_{\text{sh}}(x_j, g) &= -c_K h_j(\gamma), \end{aligned} \tag{47}$$

which is Eq. 10. Moreover, the positive scale c cancels from Eq. 41, so the normalized separation depends only on the effective discount and the chain index:

$$\rho_j(\xi) = \frac{\xi^{j-1}}{h_j(\xi)}.$$

This recovers Eq. 11 and completes the preparation for Proposition 3.1.

B.2 Proof of Proposition 3.1

Proof. We prove the two claims in turn. The first compares the normalized separation induced by the exact and single-discount chain recursions; the second shows how this separation controls robustness to bounded normalized value-estimation error.

Monotonicity of the normalized separation. By Lemma B.2, for every $\xi \in (0, 1)$ and $j \geq 1$,

$$\rho_j(\xi) = \frac{\xi^{j-1}}{h_j(\xi)}. \tag{48}$$

Dividing numerator and denominator by $\xi^{j-1} > 0$ gives

$$\rho_j(\xi) = \frac{1}{\sum_{\ell=0}^{j-1} \xi^{\ell-(j-1)}} = \frac{1}{\sum_{q=0}^{j-1} \xi^{-q}}, \tag{49}$$

where the second equality uses the reindexing $q = j - 1 - \ell$. Now fix $0 < \xi_1 < \xi_2 < 1$. For every integer $q \geq 1$,

$$\xi_1^{-q} > \xi_2^{-q}. \tag{50}$$

When $j \geq 2$, the two sums in Eq. 49 have the same $q = 0$ term and at least one term with $q \geq 1$. Hence

$$\sum_{q=0}^{j-1} \xi_1^{-q} > \sum_{q=0}^{j-1} \xi_2^{-q}. \tag{51}$$

Taking reciprocals of these positive quantities yields

$$\rho_j(\xi_1) < \rho_j(\xi_2), \tag{52}$$

so ρ_j is strictly increasing on $(0, 1)$ for every $j \geq 2$. Since $\gamma \in (0, 1)$ and $K > 1$,

$$\gamma^K = \gamma\, \gamma^{K-1} < \gamma, \tag{53}$$

because $\gamma^{K-1} < 1$. Applying Eq. 52 with $\xi_1 = \gamma^K$ and $\xi_2 = \gamma$ therefore gives

$$\rho_j(\gamma^K) < \rho_j(\gamma), \tag{54}$$

which is exactly Eq. 12. Thus replacing the block discount γ^K by the single discount γ strictly enlarges the scale-normalized separation between consecutive shortcut states whenever $j \geq 2$ and $K > 1$.

Robustness to normalized estimation error. Fix $\xi \in (0, 1)$, $c > 0$, and two consecutive states (x_{j-1}, x_j). Normalize both chain values by the same positive quantity $|U_{\xi,c}(x_j, g)|$ and write

$$\begin{aligned} \widetilde{U}_{j-1} &:= \frac{U_{\xi,c}(x_{j-1}, g)}{|U_{\xi,c}(x_j, g)|}, \\ \widetilde{U}_j &:= \frac{U_{\xi,c}(x_j, g)}{|U_{\xi,c}(x_j, g)|}. \end{aligned} \tag{55}$$

By Eq. 41,

$$\widetilde{U}_{j-1} - \widetilde{U}_j = \rho_j(\xi) > 0. \tag{56}$$

Let $\widehat{\widetilde{U}}_{j-1}$ and $\widehat{\widetilde{U}}_j$ denote normalized estimates satisfying

$$\left| \widehat{\widetilde{U}}_i - \widetilde{U}_i \right| \leq \varepsilon_{\mathrm{ord}}, \qquad i \in \{j-1, j\}. \tag{57}$$

Define

$$e_i := \widehat{\widetilde{U}}_i - \widetilde{U}_i, \qquad i \in \{j-1, j\}. \tag{58}$$

Then $|e_i| \leq \varepsilon_{\mathrm{ord}}$, and therefore

$$\begin{aligned} \widehat{\widetilde{U}}_{j-1} - \widehat{\widetilde{U}}_j &= \widetilde{U}_{j-1} - \widetilde{U}_j + e_{j-1} - e_j \\ &= \rho_j(\xi) + e_{j-1} - e_j \\ &\geq \rho_j(\xi) - |e_{j-1}| - |e_j| \\ &\geq \rho_j(\xi) - 2\varepsilon_{\mathrm{ord}}. \end{aligned} \tag{59}$$

Consequently, if

$$\rho_j(\xi) > 2\varepsilon_{\mathrm{ord}}, \tag{60}$$

then Eq. 59 implies

$$\widehat{\widetilde{U}}_{j-1} > \widehat{\widetilde{U}}_j. \tag{61}$$

Thus the ordering of the two consecutive states is preserved under the stated normalized error bound.

The strict comparison in the first part excludes only the degenerate cases. If $j = 1$, then

$$\rho_1(\xi) = \frac{\xi^0}{h_1(\xi)} = 1 \tag{62}$$

for every $\xi \in (0, 1)$, whereas $K = 1$ gives $\gamma^K = \gamma$. Hence neither case admits a strict separation between $\rho_j(\gamma)$ and $\rho_j(\gamma^K)$. This completes the proof. □

B.3 Proof of Proposition 3.2

Proof. Fix a supported state–goal pair (s, g) and primitive actions $a^+ \succ_{\mathrm{prim}} a^-$ as in Eq. 14. Under uniform shortcut propagation, every nonterminal K-step segment receives the same reward $-c_K$, independent of both the initial action and the sampled endpoint. The corresponding population target is

$$Y_K^{\mathrm{uni}} = -c_K + \gamma V^{(K)}(S^+, g). \tag{63}$$

Conditional on (s, a, g), the endpoint S^+ is distributed according to $\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a, g)$. Hence, using the shorthand in Eq. 32, for either $a \in \{a^+, a^-\}$,

$$\begin{aligned}\mathbb{E}\big[Y_K^{\mathrm{uni}} \mid s, a, g\big] &= \mathbb{E}_{K,a}\Big[-c_K + \gamma V^{(K)}(S^+, g)\Big] \\ &= -c_K + \gamma \mathbb{E}_{K,a}\Big[V^{(K)}(S^+, g)\Big]. \end{aligned} \tag{64}$$

Taking the difference between the two action-conditioned targets, the common reward cancels:

$$\begin{aligned}\Delta_K^{\mathrm{uni}} &= \mathbb{E}\big[Y_K^{\mathrm{uni}} \mid s, a^+, g\big] - \mathbb{E}\big[Y_K^{\mathrm{uni}} \mid s, a^-, g\big] \\ &= \gamma\left[\mathbb{E}_{K,a^+}\Big[V^{(K)}(S^+, g)\Big] - \mathbb{E}_{K,a^-}\Big[V^{(K)}(S^+, g)\Big]\right] \\ &= B_K(s; a^+, a^-, g). \end{aligned} \tag{65}$$

The last equality is Eq. 16. Thus the uniform reward contributes no action-dependent margin: primitive-action ordering is determined entirely by how the initial action changes the endpoint-value distribution through the logged continuation. This proves Eq. 17.

It remains only to interpret the sign of this margin. Under the negative step-reward convention, the shortcut objective ranks a^+ above a^- exactly when its conditional target is larger, and therefore

$$\Delta_K^{\mathrm{uni}} > 0 \quad \Longleftrightarrow \quad B_K > 0. \tag{66}$$

Accordingly, $B_K > 0$ preserves the strict primitive preference, $B_K = 0$ masks it, and $B_K < 0$ reverses it. This completes the proof. □

B.4 Reference-Score Perturbation and Gate Transfer

The selective-ordering result is stated in terms of the ideal shortcut-progress score Z_K, whereas the actual gate is computed from the frozen reference $\overline{V}_{\mathrm{ref}}$. We therefore first quantify how a uniform reference-value error perturbs the score and, in turn, the action-conditioned gate probabilities.

Lemma B.3 (Reference-score perturbation and gate transfer)**.** *Suppose*

$$\left\|\overline{V}_{\mathrm{ref}} - V_{\mathrm{ref}}^{(K)}\right\|_\infty \le \varepsilon_{\mathrm{ref}}. \tag{67}$$

Then, for every supported (s, a, g)*,*

$$p_K^{\mathrm{prog}}\big(s, a, g; w + 2\varepsilon_{\mathrm{ref}}\big) \le p_K^I(s, a, g) \le p_K^{\mathrm{prog}}\big(s, a, g; w - 2\varepsilon_{\mathrm{ref}}\big). \tag{68}$$

If, in addition, Assumption 3.3 holds and $2\varepsilon_{\mathrm{ref}} \le \delta_{\mathrm{sh}}$*, then*

$$p_K^I(s, a^+, g) - p_K^I(s, a^-, g) \ge \eta_K. \tag{69}$$

Proof. **Step 1: control the reference-score perturbation.** Fix a supported tuple (s, a, g) and an endpoint S^+ in the support of $\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a, g)$. By the definitions in Eq. 18,

$$\begin{aligned}\widehat{Z}_K(s, S^+, g) - Z_K(s, S^+, g) &= \overline{V}_{\mathrm{ref}}(S^+, g) - V_{\mathrm{ref}}^{(K)}(S^+, g) \\ &\quad - \Big[\overline{V}_{\mathrm{ref}}(s, g) - V_{\mathrm{ref}}^{(K)}(s, g)\Big]. \end{aligned} \tag{70}$$

The score compares the reference at two states, so the sup-norm error enters twice. The triangle inequality and Eq. 67 give

$$\begin{aligned}\left|\widehat{Z}_K(s, S^+, g) - Z_K(s, S^+, g)\right| &\le \left|\overline{V}_{\mathrm{ref}}(S^+, g) - V_{\mathrm{ref}}^{(K)}(S^+, g)\right| \\ &\quad + \left|\overline{V}_{\mathrm{ref}}(s, g) - V_{\mathrm{ref}}^{(K)}(s, g)\right| \\ &\le 2\varepsilon_{\mathrm{ref}}. \end{aligned} \tag{71}$$

Equivalently,

$$Z_K(s, S^+, g) - 2\varepsilon_{\mathrm{ref}} \le \widehat{Z}_K(s, S^+, g) \le Z_K(s, S^+, g) + 2\varepsilon_{\mathrm{ref}}. \tag{72}$$

Step 2: transfer score perturbation to gate probabilities. This pointwise interval immediately yields the two gate-event inclusions. If the ideal score exceeds the upper shifted threshold, then the learned score must cross the gate threshold:

$$\left\{ Z_K(s, S^+, g) \ge w + 2\varepsilon_{\mathrm{ref}} \right\} \subseteq \left\{ \widehat{Z}_K(s, S^+, g) \ge w \right\}. \tag{73}$$

Indeed,

$$\widehat{Z}_K(s, S^+, g) \ge Z_K(s, S^+, g) - 2\varepsilon_{\mathrm{ref}} \ge w. \tag{74}$$

Conversely, acceptance by the learned gate implies that the ideal score exceeds the lower shifted threshold:

$$\left\{ \widehat{Z}_K(s, S^+, g) \ge w \right\} \subseteq \left\{ Z_K(s, S^+, g) \ge w - 2\varepsilon_{\mathrm{ref}} \right\}, \tag{75}$$

because

$$Z_K(s, S^+, g) \ge \widehat{Z}_K(s, S^+, g) - 2\varepsilon_{\mathrm{ref}} \ge w - 2\varepsilon_{\mathrm{ref}}. \tag{76}$$

Hence, suppressing the common arguments (s, S^+, g),

$$\{ Z_K \ge w + 2\varepsilon_{\mathrm{ref}} \} \subseteq \left\{ \widehat{Z}_K \ge w \right\} \subseteq \{ Z_K \ge w - 2\varepsilon_{\mathrm{ref}} \}. \tag{77}$$

Taking conditional probabilities under $\mathsf{P}_K^{\mathcal{D}}(\cdot \mid s, a, g)$ and using the definitions of p_K^{prog} and p_K^I gives

$$\begin{aligned} p_K^{\mathrm{prog}}\big(s, a, g; w + 2\varepsilon_{\mathrm{ref}}\big) &\le \Pr\left(\widehat{Z}_K(s, S^+, g) \ge w \mid s, a, g \right) \\ &= p_K^I(s, a, g) \\ &\le p_K^{\mathrm{prog}}\big(s, a, g; w - 2\varepsilon_{\mathrm{ref}}\big), \end{aligned} \tag{78}$$

which proves Eq. 68.

Step 3: transfer the ideal action separation to the learned gate. It remains to transfer the ideal action-conditioned separation to the learned gate. Applying the lower side of the sandwich to a^+ and the upper side to a^- yields

$$\begin{aligned} p_K^I(s, a^+, g) - p_K^I(s, a^-, g) \ge\ & p_K^{\mathrm{prog}}\big(s, a^+, g; w + 2\varepsilon_{\mathrm{ref}}\big) \\ & - p_K^{\mathrm{prog}}\big(s, a^-, g; w - 2\varepsilon_{\mathrm{ref}}\big). \end{aligned} \tag{79}$$

For fixed (s, a, g), the tail probability $u \mapsto p_K^{\mathrm{prog}}(s, a, g; u)$ is nonincreasing. Since $2\varepsilon_{\mathrm{ref}} \le \delta_{\mathrm{sh}}$,

$$w + 2\varepsilon_{\mathrm{ref}} \le w + \delta_{\mathrm{sh}}, \qquad w - 2\varepsilon_{\mathrm{ref}} \ge w - \delta_{\mathrm{sh}},$$

and therefore

$$p_K^{\mathrm{prog}}\big(s, a^+, g; w + 2\varepsilon_{\mathrm{ref}}\big) \ge p_K^{\mathrm{prog}}\big(s, a^+, g; w + \delta_{\mathrm{sh}}\big), \tag{80}$$

$$p_K^{\mathrm{prog}}\big(s, a^-, g; w - 2\varepsilon_{\mathrm{ref}}\big) \le p_K^{\mathrm{prog}}\big(s, a^-, g; w - \delta_{\mathrm{sh}}\big). \tag{81}$$

Combining these inequalities with Eq. 79 gives

$$\begin{aligned} p_K^I(s, a^+, g) - p_K^I(s, a^-, g) &\ge p_K^{\mathrm{prog}}\big(s, a^+, g; w + \delta_{\mathrm{sh}}\big) \\ &\quad - p_K^{\mathrm{prog}}\big(s, a^-, g; w - \delta_{\mathrm{sh}}\big) \\ &\ge \eta_K, \end{aligned} \tag{82}$$

where the last inequality is Assumption 3.3. This proves Eq. 69. The argument uses only deterministic event inclusions and therefore requires no continuity or no-atom assumption at the gate threshold. □

B.5 Proof of Theorem 3.4

Proof. Fix a supported state–goal pair (s, g) and primitive actions $a^+ \succ_{\mathrm{prim}} a^-$ as in Eq. 14. The proof proceeds in four steps. We first transfer the ideal shortcut-progress separation in Assumption 3.3 to the gate induced by the frozen reference and obtain a lower bound on the gate-induced margin M_K. We then decompose the population selective action-preference margin into the endpoint contribution B_K and the gate-induced contribution M_K. Next, we control the effect of replacing the population shortcut value $V^{(K)}$ by an estimate $\widehat{V}^{(K)}$ while keeping the frozen gate fixed. Finally, we translate the resulting positive-margin condition into preservation of the primitive-action preference.

Step 1: gate separation and reference-induced margin. The theorem assumes Assumption 3.3 together with

$$\left\|\overline{V}_{\mathrm{ref}} - V_{\mathrm{ref}}^{(K)}\right\|_\infty \le \varepsilon_{\mathrm{ref}}, \qquad 2\varepsilon_{\mathrm{ref}} \le \delta_{\mathrm{sh}}. \tag{83}$$

These are exactly the conditions required by Lemma B.3: the first bounds the perturbation of each reference value, while the second ensures that the resulting score error remains within the threshold slack appearing in Assumption 3.3. Invoking that lemma therefore gives

$$p_K^I(s, a^+, g) - p_K^I(s, a^-, g) \ge \eta_K. \tag{84}$$

Thus the preference encoded by the ideal shortcut-progress probabilities survives the bounded reference approximation and appears as a separation between the learned gate-acceptance probabilities.

We next convert this acceptance-probability separation into a reward-margin contribution. By Lemma B.1,

$$\begin{aligned} h_K(\gamma) - 1 &= \gamma h_{K-1}(\gamma) \\ &= \sum_{\ell=1}^{K-1} \gamma^\ell \ge 0. \end{aligned} \tag{85}$$

For $K = 1$, the sum is empty and the factor equals zero; for $K > 1$, it is strictly positive because $\gamma \in (0, 1)$. Hence multiplying Eq. 84 by $h_K(\gamma) - 1$ preserves the direction of the inequality. Using the definition of M_K in Eq. 22, we obtain

$$\begin{aligned} M_K &= \big(h_K(\gamma) - 1\big)\left[p_K^I(s, a^+, g) - p_K^I(s, a^-, g)\right] \\ &\ge \big(h_K(\gamma) - 1\big)\eta_K. \end{aligned} \tag{86}$$

Equations 84 and 86 establish the gate-separation and gate-induced margin bounds needed below. Notice that this step does not yet establish the full primitive-action margin: it only lower bounds the additional contribution created by selective reward compression.

Step 2: decomposition of the population selective margin. We now identify how the gate-induced term from Step 1 enters the primitive-action preference. Fix an action $a \in \{a^+, a^-\}$. The selective target in Eq. 21 is

$$Y_K^{\mathrm{sel}} = -h_K(\gamma) + I_K\big(h_K(\gamma) - 1\big) + \gamma V^{(K)}(S^+, g).$$

Taking the conditional expectation given (s, a, g) and using linearity of expectation yields

$$\begin{aligned} \mathbb{E}\big[Y_K^{\mathrm{sel}} \mid s, a, g\big] &= -h_K(\gamma) + \big(h_K(\gamma) - 1\big)\mathbb{E}[I_K \mid s, a, g] \\ &\quad + \gamma \mathbb{E}_{K,a}\Big[V^{(K)}(S^+, g)\Big] \\ &= -h_K(\gamma) + \big(h_K(\gamma) - 1\big)p_K^I(s, a, g) \\ &\quad + \gamma \mathbb{E}_{K,a}\Big[V^{(K)}(S^+, g)\Big]. \end{aligned} \tag{87}$$

The second equality follows from the definition $p_K^I(s, a, g) = \Pr(I_K = 1 \mid s, a, g)$ and the elementary identity $\mathbb{E}[I_K \mid s, a, g] = \Pr(I_K = 1 \mid s, a, g)$ for an indicator random variable.

By the definition of the conditional action-preference margin in Eq. 15, subtracting the conditional expectation for a^- from that for a^+ gives

$$\begin{aligned}\Delta_K^{\mathrm{sel}} &= \mathbb{E}\big[Y_K^{\mathrm{sel}} \mid s, a^+, g\big] - \mathbb{E}\big[Y_K^{\mathrm{sel}} \mid s, a^-, g\big] \\ &= \big(h_K(\gamma) - 1\big)\left[p_K^I(s, a^+, g) - p_K^I(s, a^-, g)\right] \\ &\quad + \gamma\left[\mathbb{E}_{K,a^+}\left[V^{(K)}(S^+, g)\right] - \mathbb{E}_{K,a^-}\left[V^{(K)}(S^+, g)\right]\right] \\ &= M_K + B_K. \end{aligned} \tag{88}$$

Here the common baseline reward $-h_K(\gamma)$ cancels because it is identical for the two action-conditioned targets. The indicator-dependent term does not cancel: its action-conditioned difference is exactly M_K by Eq. 22. The remaining endpoint-value difference is exactly B_K by Eq. 16. Hence Eq. 88 establishes the decomposition in Eq. 23.

Combining this decomposition with the lower bound from Step 1 gives

$$\Delta_K^{\mathrm{sel}} = B_K + M_K \geq B_K + \big(h_K(\gamma) - 1\big)\eta_K. \tag{89}$$

This makes explicit the two distinct sources of primitive-action separation analyzed in the main text: B_K is the action-conditioned endpoint signal inherited from the logged continuation, while the nonnegative term controlled by η_K is the additional separation supplied by reference-selective reward compression.

Step 3: robustness to shortcut-value estimation error. We next replace only the population shortcut value $V^{(K)}$ by an estimate $\widehat{V}^{(K)}$. As stipulated in the theorem, the frozen reference $\overline{V}_{\mathrm{ref}}$ and therefore the gate I_K are kept fixed. This distinction is important: the perturbation considered in this step affects the endpoint-value contribution, but it does not alter the acceptance probabilities or the margin term M_K established in Step 1.

Define the corresponding selective target by

$$\widehat{Y}_K^{\mathrm{sel}} := -h_K(\gamma) + I_K\big(h_K(\gamma) - 1\big) + \gamma\widehat{V}^{(K)}(S^+, g), \tag{90}$$

and define the resulting endpoint contribution by

$$\widehat{B}_K := \gamma\Big[\mathbb{E}_{K,a^+}\Big[\widehat{V}^{(K)}(S^+, g)\Big] - \mathbb{E}_{K,a^-}\Big[\widehat{V}^{(K)}(S^+, g)\Big]\Big]. \tag{91}$$

The corresponding learned action-preference margin is

$$\widehat{\Delta}_K^{\mathrm{sel}} := \mathbb{E}\Big[\widehat{Y}_K^{\mathrm{sel}} \mid s, a^+, g\Big] - \mathbb{E}\Big[\widehat{Y}_K^{\mathrm{sel}} \mid s, a^-, g\Big]. \tag{92}$$

Repeating the conditional-expectation calculation from Step 2 with $\widehat{V}^{(K)}$ in place of $V^{(K)}$ gives

$$\widehat{\Delta}_K^{\mathrm{sel}} = \widehat{B}_K + M_K. \tag{93}$$

The same M_K appears because the gate is unchanged; only the endpoint-value term is perturbed.

To control this perturbation, let

$$e_V(s', g) := \widehat{V}^{(K)}(s', g) - V^{(K)}(s', g). \tag{94}$$

The theorem assumes

$$\|e_V\|_\infty = \left\|\widehat{V}^{(K)} - V^{(K)}\right\|_\infty \leq \varepsilon_V. \tag{95}$$

Subtracting the definitions of $\widehat{B}_K$ and B_K and then substituting the error function gives

$$\widehat{B}_K - B_K = \gamma\left[\mathbb{E}_{K,a^+}\big[e_V(S^+, g)\big] - \mathbb{E}_{K,a^-}\big[e_V(S^+, g)\big]\right]. \tag{96}$$

Taking absolute values and applying the triangle inequality,

$$\begin{aligned}\left|\widehat{B}_K - B_K\right| &\leq \gamma\left|\mathbb{E}_{K,a^+}\big[e_V(S^+, g)\big]\right| + \gamma\left|\mathbb{E}_{K,a^-}\big[e_V(S^+, g)\big]\right| \\ &\leq \gamma\mathbb{E}_{K,a^+}\big[|e_V(S^+, g)|\big] + \gamma\mathbb{E}_{K,a^-}\big[|e_V(S^+, g)|\big] \\ &\leq 2\gamma\varepsilon_V. \end{aligned} \tag{97}$$

The second inequality uses $|\mathbb{E}[X]| \le \mathbb{E}[|X|]$, while the final inequality follows from the uniform bound in Eq. 95. Each action-conditioned expectation therefore contributes at most $\gamma\varepsilon_V$ to the perturbation of the margin, which explains the factor $2\gamma\varepsilon_V$. In particular,

$$\widehat{B}_K \ge B_K - 2\gamma\varepsilon_V. \tag{98}$$

Combining Eqs. 93, 98, and 86, we obtain

$$\begin{aligned}\widehat{\Delta}_K^{\mathrm{sel}} &= \widehat{B}_K + M_K \\ &\ge B_K - 2\gamma\varepsilon_V + \big(h_K(\gamma) - 1\big)\eta_K \\ &= B_K + \big(h_K(\gamma) - 1\big)\eta_K - 2\gamma\varepsilon_V. \end{aligned} \tag{99}$$

This is exactly the robustness bound in Eq. 24. Importantly, the reference-induced contribution is not charged a second estimation-error term here because the theorem holds the frozen gate fixed; reference approximation was already handled separately in Step 1 through Lemma B.3.

Step 4: preservation of primitive-action preference. It remains to translate the robust margin bound into an ordering statement. Under the negative step-reward convention, larger conditional target values are preferred, and by Eq. 15 a positive action-preference margin means that the shortcut objective ranks a^+ above a^-. If

$$B_K + \big(h_K(\gamma) - 1\big)\eta_K > 2\gamma\varepsilon_V, \tag{100}$$

then Eq. 99 implies

$$\widehat{\Delta}_K^{\mathrm{sel}} > 0. \tag{101}$$

Therefore the estimated selective shortcut target ranks a^+ above a^-. Since the pair was fixed with $a^+ \succ_{\mathrm{prim}} a^-$, this proves that the estimated shortcut regression target preserves the primitive preference under the stated sufficient condition, completing the target-ordering proof. □

C Auxiliary Analysis for Shortcut Value Consistency

This appendix establishes the two local properties of Shortcut Value Consistency (SVC) used in Section 4.2. We first show how local SVC residuals accumulate along an observed shortcut chain, yielding a direct bound on deviation from the canonical single-discount recursion. We then characterize the local optimization effect of differentiating through both shortcut endpoints and compare it with a successor stop-gradient update in the tabular, independently parameterized setting. These results concern local consistency and optimization only and do not imply global neural-network convergence or policy improvement.

Shortcut-chain consistency. For a shortcut edge, define the canonical residual

$$e(s, s^+, g) := V(s, g) + 1 - \gamma V(s^+, g).$$

A zero residual is exactly the single-discount shortcut relation $V(s, g) = -1 + \gamma V(s^+, g)$. The following lemma quantifies how nonzero local residuals accumulate when such relations are composed along an observed shortcut chain.

Lemma C.1 (Shortcut-chain consistency)**.** *Consider a shortcut chain*

$$x_j \to x_{j-1} \to \cdots \to x_0 = g$$

and define

$$e_i := V(x_i, g) + 1 - \gamma V(x_{i-1}, g), \qquad i = 1, \ldots, j.$$

Then

$$V(x_j, g) = -h_j(\gamma) + \sum_{i=1}^{j} \gamma^{j-i} e_i. \tag{102}$$

Consequently, if $|e_i| \le \varepsilon_{\mathrm{SVC}}$ for every edge, then

$$|V(x_j, g) + h_j(\gamma)| \le h_j(\gamma)\varepsilon_{\mathrm{SVC}}. \tag{103}$$

Proof. By the definition of e_i,

$$V(x_i, g) = -1 + \gamma V(x_{i-1}, g) + e_i.$$

Repeated substitution separates the contribution of the unit shortcut cost from that of the local residuals. Each traversed edge contributes one discounted unit cost, while a residual introduced at edge i is propagated through the remaining $j - i$ recursions and therefore receives the factor γ^{j-i}. Using $V(x_0, g) = V(g, g) = 0$ gives

$$\begin{aligned} V(x_j, g) &= -\sum_{\ell=0}^{j-1} \gamma^\ell + \sum_{i=1}^{j} \gamma^{j-i} e_i \\ &= -h_j(\gamma) + \sum_{i=1}^{j} \gamma^{j-i} e_i, \end{aligned}$$

where the second equality uses the definition of $h_j(\gamma)$. This proves Eq. 102.

If $|e_i| \le \varepsilon_{\mathrm{SVC}}$ for all i, Eq. 102 and the triangle inequality yield

$$\begin{aligned} |V(x_j, g) + h_j(\gamma)| &= \left| \sum_{i=1}^{j} \gamma^{j-i} e_i \right| \\ &\le \sum_{i=1}^{j} \gamma^{j-i} |e_i| \\ &\le \varepsilon_{\mathrm{SVC}} \sum_{i=1}^{j} \gamma^{j-i} \\ &= h_j(\gamma) \varepsilon_{\mathrm{SVC}}, \end{aligned}$$

where the last equality follows by reindexing the geometric sum. This proves Eq. 103. □

Lemma C.1 is a local consistency statement along observed shortcut relations. It shows that controlling the SVC residual on each observed edge controls deviation from the canonical single-discount recursion along that chain. It does not imply global value accuracy or remove the state–action–goal coverage requirement of the critic.

Two-sided SVC gradient. For one sampled edge, let

$$\delta_{\mathrm{SVC}} := V_\psi(s, g) + 1 - \gamma V_\psi(s^+, g).$$

Applying the chain rule to the squared residual gives

$$\nabla_\psi \delta_{\mathrm{SVC}}^2 = 2\delta_{\mathrm{SVC}} \left[\nabla_\psi V_\psi(s, g) - \gamma \nabla_\psi V_\psi(s^+, g) \right]. \tag{104}$$

Thus the full SVC objective propagates gradients through both endpoint evaluations, whereas a successor stop-gradient removes the second contribution. The following proposition isolates the resulting value-level correction when the two endpoint values are independently parameterized.

Proposition C.2 (Two-sided correction of a shortcut residual)**.** *Let $v = V(s, g)$ and $v^+ = V(s^+, g)$ be independently parameterized, and let*

$$\delta := v + 1 - \gamma v^+.$$

One gradient step of size α on δ^2 gives

$$v' = v - 2\alpha\delta, \qquad (v^+)' = v^+ + 2\alpha\gamma\delta, \tag{105}$$

and therefore

$$\delta' = \left[1 - 2\alpha(1 + \gamma^2)\right] \delta. \tag{106}$$

If the successor is treated as a stop-gradient target, then

$$\delta'_{\mathrm{semi}} = (1 - 2\alpha)\delta. \tag{107}$$

Proof. For independently parameterized endpoint values,

$$\frac{\partial \delta}{\partial v} = 1, \qquad \frac{\partial \delta}{\partial v^+} = -\gamma.$$

The chain rule therefore gives

$$\frac{\partial \delta^2}{\partial v} = 2\delta, \qquad \frac{\partial \delta^2}{\partial v^+} = -2\gamma\delta.$$

Gradient descent subtracts α times each derivative. Hence the source value moves by $-2\alpha\delta$, while the successor value moves by $+2\alpha\gamma\delta$, which gives Eq. 105. Substituting these two updates into the residual after one step yields

$$\begin{aligned}
\delta' &= v' + 1 - \gamma(v^+)' \\
&= (v - 2\alpha\delta) + 1 - \gamma\left(v^+ + 2\alpha\gamma\delta\right) \\
&= \delta - 2\alpha\delta - 2\alpha\gamma^2\delta \\
&= \left[1 - 2\alpha(1+\gamma^2)\right]\delta,
\end{aligned}$$

which proves Eq. 106. The factor $1 + \gamma^2$ arises because both endpoint corrections reduce the same residual in this independently parameterized calculation.

With a successor stop-gradient, v^+ is held fixed and only the source value is updated to $v - 2\alpha\delta$. Consequently,

$$\begin{aligned}
\delta'_{\text{semi}} &= (v - 2\alpha\delta) + 1 - \gamma v^+ \\
&= (1 - 2\alpha)\delta,
\end{aligned}$$

which proves Eq. 107. □

The update identities in Proposition C.2 hold algebraically for any α. Interpreting them as one-step residual contractions requires the corresponding multiplicative factor to have magnitude less than one. For the two-sided update,

$$\left|1 - 2\alpha(1+\gamma^2)\right| < 1 \quad \Longleftrightarrow \quad 0 < \alpha < \frac{1}{1+\gamma^2},$$

whereas for the successor stop-gradient update,

$$|1 - 2\alpha| < 1 \quad \Longleftrightarrow \quad 0 < \alpha < 1.$$

These are local tabular conditions only. Under shared neural-network parameters, Eq. 104 is the exact statement: both endpoint evaluations contribute to the parameter gradient, but their induced value changes can interact through the shared parameterization. Neither Proposition C.2 nor the contraction conditions imply global optimization convergence or improved policy performance.

D Experimental Setup and Implementation

D.1 Goal Sampling

Goal sampling enters SSVL at three distinct stages: frozen-reference pretraining, downstream value learning, and primitive-policy learning. To isolate the effects of shortcut propagation, reference alignment, and SVC, we keep the corresponding sampling rules fixed across all SSVL ablations reported in this paper.

During frozen-reference pretraining, goals are sampled from the current state, future states on the same trajectory, and random dataset states with probabilities $(0.2, 0.5, 0.3)$, respectively, following the goal-mixture convention used by HIQL (Park et al., 2023). Future goals are sampled along the same trajectory according to a $\mathrm{Geom}(1-\gamma)$ distribution, while random goals are sampled uniformly from the dataset state distribution. During downstream state-value learning, we use the mixture $(0.1, 0.6, 0.3)$ for current, future, and random goals. Future goals are sampled uniformly along the same trajectory by default; for visual tasks whose goal horizon exceeds 2000 steps, we instead use a near-to-far curriculum over future goals. The task-specific choice is reported with the remaining hyperparameters in Appendix D.4.

Primitive-policy training follows the task-dependent goal-sampling protocol of the corresponding OGBench/GCIQL setup (Park et al., 2025). For navigation-style locomotion and manipulation tasks, policy goals are sampled entirely from future states on the same trajectory. For stitching tasks, future and random goals are sampled with equal probability, 0.5 each. For exploration tasks, policy goals are sampled uniformly from the dataset state distribution. These policy-goal distributions are also held fixed across the SSVL ablations.

These policy samplers matter because the informativeness of primitive-action supervision depends on how the sampled final goal relates to the logged current action. In stitching and exploration settings, a sampled goal may lie far along a trajectory or come from a different trajectory, in which case the logged action at the current state can provide only weak supervision for choosing a primitive action toward that goal. SSVL addresses the depth of value propagation from long-range goals to primitive actions, but it does not by itself resolve this state–action–goal pairing problem. This distinction is consistent with the task-dependent performance in Section 5.2 and motivates improved goal sampling as a complementary direction rather than a component changed in the present experiments.

D.2 State–Goal Representations and Policy Extraction

To separate the effect of shortcut value learning from changes in representation or policy parameterization, SSVL retains the state–goal encoders and primitive-policy backends used by the corresponding OGBench/GCIQL implementations (Park et al., 2025). For pixel-based goal-reaching tasks, we use the OGBench joint state–goal representation with an IMPALA-style convolutional encoder (Espeholt et al., 2018). For state-based manipulation, the joint state–goal representation is implemented with a three-layer MLP of width $(512, 512, 512)$. The task-specific latent dimensions for these manipulation domains are reported with the remaining hyperparameters in Appendix D.4. Other benchmark-family-specific architecture choices follow the corresponding OGBench setup unless explicitly stated otherwise.

Consistent with the flat-control formulation in Section 4, SSVL also leaves policy extraction to the benchmark-specific primitive-action backend rather than introducing an additional high-level policy or subgoal predictor. The deployed policy is always a primitive goal-conditioned policy $\pi_\eta(a \mid s, g)$. Depending on the benchmark family, we use either advantage-weighted regression (AWR) (Peng et al., 2019) or DDPG+BC, following the corresponding OGBench implementation (Fujimoto & Gu, 2021; Park et al., 2025). For AWR-based extraction, the actor is optimized with the advantage-weighted log-likelihood objective

$$J_{\mathrm{AWR}}(\pi) = \mathbb{E}_{(s,a)\sim\mathcal{D},\, g\sim p^{\mathcal{D}}_{\mathrm{pol}}(g|s)} \left[\exp\big(\beta\big(Q(s,a,g) - V(s,g)\big)\big) \log \pi(a \mid s, g) \right], \tag{108}$$

where $p^{\mathcal{D}}_{\mathrm{pol}}(g \mid s)$ denotes the task-dependent policy-goal distribution described in Appendix D.1. For DDPG+BC-based extraction, we use

$$J_{\mathrm{DDPG+BC}}(\pi) = \mathbb{E}_{(s,a)\sim\mathcal{D},\, g\sim p^{\mathcal{D}}_{\mathrm{pol}}(g|s)} \left[Q(s, \pi^\mu(s,g), g) + \beta \log \pi(a \mid s, g) \right], \tag{109}$$

with $\pi^\mu(s,g) = \mathbb{E}_{a\sim\pi(\cdot|s,g)}[a]$. This is the paper-level DDPG+BC objective reported by OGBench (Park et al., 2025); following its implementation details, policy extraction uses double-Q learning and normalized Q values. The task-specific actor coefficients follow the corresponding OGBench settings and are summarized in Appendix D.4.

The downstream optimization order follows the IQL-style pipeline used by the implementation. Each update first fits the critic to the stopped-shortcut target, then updates the state value with its base objective together with SVC, and finally updates the primitive policy,

$$Q \;\longrightarrow\; V \;\longrightarrow\; \pi.$$

This ordering keeps the policy extraction stage downstream of the value-learning changes introduced by SSVL. Neither shortcut endpoints nor the frozen reference are required at test time; deployment consists only of evaluating the primitive goal-conditioned policy $\pi_\eta(a \mid s, g)$ at each environment step.

D.3 Frozen Shortcut Reference

The frozen reference provides the reward-side selectivity used by SSVL; it is trained separately from the downstream critic and is not part of the deployed policy. To align the gate with shortcut-

scale progress rather than primitive one-step changes, reference pretraining uses the same stopped-endpoint construction and shortcut length K as downstream SSVL. Starting from a sampled state, the reference endpoint advances by at most K steps, stops earlier when a sampled future goal on the same trajectory is reached, and is truncated at the trajectory boundary. The goal-sampling distribution used during this stage is given in Appendix D.1.

Let s_t^+ denote the resulting reference endpoint. Under the negative step-reward convention, reference pretraining assigns the endpoint sparse reward

$$r_{\mathrm{ref},t} = -\mathbf{1}\{s_t^+ \neq g\}, \tag{110}$$

and uses a single-discount continuation. Writing $m_t = \mathbf{1}\{s_t \neq g\}$ for the current-goal mask, the corresponding target is

$$Y_{\mathrm{ref},t} = r_{\mathrm{ref},t} + \gamma m_t V_{\mathrm{ref}}^-(s_t^+, g), \tag{111}$$

where V_{ref}^- denotes the target-network value used during pretraining. The reference is fitted with the same expectile-regression machinery as the goal-conditioned value pretraining pipeline,

$$\mathcal{L}_{\mathrm{ref}} = \mathbb{E}\left[L_2^{\tau_{\mathrm{ref}}}\left(Y_{\mathrm{ref},t} - V_{\mathrm{ref}}(s_t, g)\right)\right]. \tag{112}$$

For a goal-stopped relation, $s_t^+ = g$, so $r_{\mathrm{ref},t} = 0$ and the bootstrap also vanishes under $V_{\mathrm{ref}}(g, g) = 0$, giving a zero target. Reference pretraining does not use SVC or the selective downstream critic target; its role is to provide a shortcut-scale state-value signal from which the downstream reward-compression gate is constructed. Current-goal samples with $s_t = g$ follow the same absorbing zero-target convention.

After pretraining, the resulting checkpoint is frozen and denoted by $\overline{V}_{\mathrm{ref}}$. For a downstream stopped segment of realized length κ_t, SSVL evaluates the reference progress score and rescales the base threshold to the segment length,

$$\widehat{Z}_t = \overline{V}_{\mathrm{ref}}(s_t^+, g) - \overline{V}_{\mathrm{ref}}(s_t, g), \qquad w_t = w\frac{1-\gamma^{\kappa_t/K}}{1-\gamma}, \qquad I_t = \mathbf{1}\{\widehat{Z}_t \geq w_t\}. \tag{113}$$

Thus a full-K segment uses threshold w, while shorter stopped segments use a correspondingly smaller threshold, matching the gate definition in Eq. 26. The reference affects downstream learning only through this binary reward-compression decision. It does not select the shortcut endpoint, does not determine whether the single-discount continuation is used, and is not evaluated by the deployed primitive policy. In particular, every stopped segment used for critic learning receives the single-discount continuation, whereas the frozen reference controls only whether the accumulated segment reward is compressed. A one-step reference is retained only as an ablation comparator; Appendix E.2 examines how reference alignment and training maturity affect the resulting gate behavior and downstream performance.

D.4 Hyperparameters and Task-Specific Settings

To keep the comparison focused on shortcut value learning rather than changes in the underlying offline-RL backbone, SSVL inherits the optimization, architecture, and policy-extraction settings of the corresponding OGBench/GCIQL implementation unless stated otherwise (Park et al., 2025). Table 2 summarizes frozen-reference pretraining, and Table 3 gives the common downstream settings. The frozen reference uses the same shortcut length K as downstream SSVL. A one-step reference is considered only as an ablation comparator in Appendix E.2.

The remaining task-dependent choices concern goal sampling and shortcut scale rather than a change in the primitive-policy architecture. Downstream value learning uses the goal mixture $(0.1, 0.6, 0.3)$ for current, future, and random goals, respectively, while policy-goal sampling follows Appendix D.1. Future goals are sampled uniformly along the same trajectory by default; the near-to-far curriculum is used only for visual tasks whose goal horizon exceeds 2000 steps. For locomotion, $K = 100$ on HumanoidMaze and $K = 25$ by default on the remaining domains, with the single exception `antmaze-large-explore-v0`, where we use the matched setting $K = 20$ noted in Table 3. Task-specific manipulation and visual-domain shortcut lengths are listed in Table 4.

Table 2: Hyperparameters for frozen shortcut-reference pretraining. The reference uses the shortcut-aligned stopped-endpoint construction described in Appendix D.3.

Hyperparameter	**Value**
Batch size	256 (pixel), 1024 (state)
MLP architecture	$(512, 512, 512)$
Learning rate	3×10^{-4}
Optimizer	Adam (Kingma & Ba, 2015)
Nonlinearity	GELU (Hendrycks & Gimpel, 2016)
Gradient steps	500k (pixel), 1M (state)
Expectile	0.9
Discount γ	Same as corresponding downstream SSVL setting
Target update rate	0.005
Goal mixture $(p_{\text{cur}}, p_{\text{future}}, p_{\text{rand}})$	$(0.2, 0.5, 0.3)$
Shortcut length	Same K as downstream SSVL
SVC during reference pretraining	None

Table 3: Common hyperparameters for downstream SSVL training. Goal-sampling details are given in Appendix D.1; task-specific shortcut lengths and state-manipulation latent dimensions are given in Table 4.

Category	**Hyperparameter**	**Value**
Architecture	MLP	$(512, 512, 512)$
	Representation encoder	IMPALA-small (pixel); MLP (state manipulation)
Optimization	Batch size	256 (pixel), 1024 (state)
	Optimizer	Adam (Kingma & Ba, 2015)
	Learning rate	3×10^{-4}
	Nonlinearity	GELU (Hendrycks & Gimpel, 2016)
	Discount γ	0.995 for {`antmaze`,`pointmaze`}`-giant` and `humanoidmaze`; 0.99 otherwise
	Image augmentation probability	0.5 (pixel manipulation), 0 otherwise
	Target update rate	0.005
	Gradient steps	500k (pixel), 1M (state)
	Expectile	0.7 for {`antmaze`,`humanoidmaze`}`-`{`navigate`,`stitch`}; 0.9 otherwise
Policy	Actor backend / hyperparameters	Identical to the corresponding OGBench GCIQL setting
	Policy coefficient	Corresponding OGBench setting
	Evaluation episodes per goal	50
SSVL	Shortcut length K	100 (HumanoidMaze); 25 (other locomotion by default); Table 4 (others)
	Locomotion exception	$K = 20$ for `antmaze-large-explore-v0`[†]
	Gate threshold w	0.5
	SVC weight λ_{SVC}	0.3 (state locomotion), 0.01 (pixel locomotion), 0.03 (manipulation)
	SVC relation / gradient	Canonical unit-cost relation; two-sided endpoint update

[†] OTA uses task-specific hyperparameters and reports a subgoal horizon $k = 20$ on `antmaze-large-explore-v0` (its separate value-abstraction factor is $n = 15$) (Ahn et al., 2025). Motivated by this task-specific 20-step choice, we included $K = 20$ in a preliminary SSVL shortcut-length comparison; it outperformed the default $K = 25$, so $K = 20$ was fixed for the final six-seed benchmark on this task. The same K is used for reference pretraining and downstream SSVL.

OTA reproduction protocol. For the entries marked † in Table 1, our OTA reproduction uses six seeds and the same dataset preprocessing, goal-sampling rules, downstream gradient-step budget, and evaluation protocol as the corresponding SSVL run. OTA retains its own hierarchical policy structure and option-aware value-learning objective; the matched protocol is used to reduce experimental confounds rather than to turn OTA into a flat-policy control. These reproduced entries are distinct from the published OTA results reported elsewhere in the tables.

Wall-clock training time. We additionally compare wall-clock training time with our hierarchical OTA reproduction on `visual-antmaze-giant-navigate-v0`, where both methods use NVIDIA L40S GPUs and the same 500k downstream gradient-step budget. The three OTA timing runs take 6.14, 6.24, and 6.09 hours, averaging 6.16 hours, whereas the matched SSVL downstream

Table 4: Task-specific state-manipulation latent dimensions and shortcut lengths. A dash indicates that the state-manipulation latent-dimension setting is not applicable to the corresponding visual or Powderworld task.

Environment	Dataset type	Dataset	Latent dim.	K
cube	play	cube-single-play-v0	64	1
		cube-double-play-v0	64	1
		cube-triple-play-v0	64	3
		cube-quadruple-play-v0	64	5
	noisy	cube-single-noisy-v0	64	1
		cube-double-noisy-v0	64	1
		cube-triple-noisy-v0	64	3
		cube-quadruple-noisy-v0	64	5
scene	play	scene-play-v0	64	3
	noisy	scene-noisy-v0	64	3
puzzle	play	puzzle-3x3-play-v0	64	1
		puzzle-4x4-play-v0	160	1
		puzzle-4x5-play-v0	160	3
		puzzle-4x6-play-v0	160	5
	noisy	puzzle-3x3-noisy-v0	64	1
		puzzle-4x4-noisy-v0	160	1
		puzzle-4x5-noisy-v0	160	3
		puzzle-4x6-noisy-v0	160	5
visual-cube	play	visual-cube-single-play-v0	–	10
		visual-cube-double-play-v0	–	10
		visual-cube-triple-play-v0	–	10
		visual-cube-quadruple-play-v0	–	10
	noisy	visual-cube-single-noisy-v0	–	1
		visual-cube-double-noisy-v0	–	1
		visual-cube-triple-noisy-v0	–	1
		visual-cube-quadruple-noisy-v0	–	1
visual-scene	play	visual-scene-play-v0	–	10
	noisy	visual-scene-noisy-v0	–	1
visual-puzzle	play	visual-puzzle-3x3-play-v0	–	10
		visual-puzzle-4x4-play-v0	–	10
		visual-puzzle-4x5-play-v0	–	10
		visual-puzzle-4x6-play-v0	–	10
	noisy	visual-puzzle-3x3-noisy-v0	–	1
		visual-puzzle-4x4-noisy-v0	–	1
		visual-puzzle-4x5-noisy-v0	–	1
		visual-puzzle-4x6-noisy-v0	–	1
powderworld	play	powderworld-easy-play-v0	–	1
		powderworld-medium-play-v0	–	1
		powderworld-hard-play-v0	–	1

run takes 4.32 hours. Because the frozen reference is pretrained once and reused across the six downstream seeds, we amortize the full observed one-time 500k reference-pretraining cost of 1.27 hours, corresponding to 0.21 hours per downstream seed. The resulting effective SSVL cost is 4.53 hours

Table 5: Matched wall-clock comparison on `visual-antmaze-giant-navigate-v0`. Both methods use NVIDIA L40S GPUs and 500k downstream gradient steps. OTA timing is averaged over three runs. The SSVL reference charge amortizes the full observed one-time 500k reference-pretraining cost over six downstream seeds.

Method	Downstream	Ref. charge / seed	Effective total
OTA	6.16 h	–	6.16 h
SSVL	4.32 h	0.21 h	**4.53 h**

per seed, about 26.4% below the 6.16-hour OTA average. On the state-based runs we examined, SSVL and OTA had broadly comparable wall-clock training times.

E Additional Mechanism Experiments

The experiments in this section complement the mechanism analysis in Section 5.3 by separating the effects of shortcut continuation, reference-selective reward compression, and SVC under matched primitive-action learning pipelines. Unless stated otherwise, the endpoint construction, primitive-action backbone, and downstream optimization procedure are held fixed, so each comparison changes only the mechanism under examination. We first isolate the target-level effect of replacing primitive-time continuation by a single-discount shortcut relation, before turning to the reference gate and SVC in the subsequent subsections.

E.1 Shortcut-Target Decomposition

The first controlled study asks how much of the target-level improvement on `antmaze-large-navigate-v0` comes from shortening the bootstrap continuation, as opposed to changing the segment reward. We use $K = 25$, disable SVC, and keep the stopped segments and primitive-action backbone identical across all variants. For a stopped segment with realized length κ and endpoint s^+, define

$$h_\kappa(\gamma) = \sum_{\ell=0}^{\kappa-1} \gamma^\ell.$$

The exact stopped multi-step comparator preserves primitive-time discounting,

$$Y_{\text{exact}} = -h_\kappa(\gamma) + \gamma^\kappa V(s^+, g). \tag{114}$$

We compare it with three uniform single-discount controls,

$$Y_{\text{seg}} = -h_\kappa(\gamma) + \gamma V(s^+, g), \tag{115}$$

$$Y_{\text{all}} = -1 + \gamma V(s^+, g), \tag{116}$$

$$Y_{\text{OTA-flat}} = -\mathbf{1}\{s^+ \neq g\} + \gamma V(s^+, g), \tag{117}$$

and with the reference-selective SSVL target,

$$Y_{\text{SSVL}} = -h_\kappa(\gamma) + I\big(h_\kappa(\gamma) - 1\big) + \gamma V(s^+, g). \tag{118}$$

Equation 115 retains the realized segment reward for every stopped relation and therefore corresponds to the all-reject control. Equation 116 assigns the unit shortcut reward to every stopped relation. The flat OTA-style control in Eq. 117 applies the uniform single-discount reward convention associated with OTA to the same primitive-action backbone; it is an internal matched mechanism control rather than the hierarchical OTA algorithm (Ahn et al., 2025). Universal all-accept and flat OTA-style differ only on goal-stopped relations: when $s^+ = g$, the former assigns immediate reward -1, whereas the latter assigns 0.

Table 6 reports aggregate final success over three seeds. On this controlled task, replacing the primitive-time bootstrap γ^κ by a single γ while retaining the realized segment reward produces the largest change, increasing success from 41.5 ± 4.6 to 86.1 ± 3.6. Once the single-discount continuation is in place, the three uniform reward conventions fall in a much narrower range: $86.1 \pm$

Table 6: Shortcut-target decomposition on `antmaze-large-navigate-v0`. All results use the same primitive-action backbone, $K = 25$, three random seeds, and no SVC. Values are final binary success rate (%).

Target variant	Final success
Exact stopped K-step	41.5 ± 4.6
Single-γ, segment reward (all-reject)	86.1 ± 3.6
Universal all-accept	87.7 ± 5.0
Flat OTA-style	88.3 ± 5.1
Reference-selective SSVL, no SVC	$\mathbf{91.2 \pm 1.4}$

Table 7: Per-goal final success (%) for the single-discount target decomposition on `antmaze-large-navigate-v0`. Values are mean $\pm$ standard deviation over three seeds.

Target variant	Goal 1	Goal 2	Goal 3	Goal 4	Goal 5
Single-γ, segment reward (all-reject)	91.3 ± 2.5	87.3 ± 5.2	92.0 ± 3.3	79.3 ± 9.8	80.7 ± 5.0
Universal all-accept	92.0 ± 4.3	86.7 ± 5.2	94.7 ± 0.9	81.3 ± 11.1	84.0 ± 5.9
Flat OTA-style	95.3 ± 2.5	86.7 ± 0.9	89.3 ± 0.9	88.7 ± 8.2	81.3 ± 19.3
Reference-selective SSVL, no SVC	90.0 ± 4.3	89.3 ± 1.9	95.3 ± 0.9	88.7 ± 4.1	92.7 ± 2.5

3.6 for the segment reward, 87.7 ± 5.0 for universal all-accept, and 88.3 ± 5.1 for the flat OTA-style control. This decomposition therefore identifies $\gamma^\kappa \to \gamma$ as the dominant target-level change in this diagnostic setting, without implying that reward selectivity must be secondary on every task or goal.

The reference-selective target reaches 91.2 ± 1.4, providing an additional gain after the common single-discount continuation is already present. This comparison supports a further contribution from selective segment-reward compression, but it is not a matched-rate causal test of which individual segments should be accepted. The stronger conclusion is the separation between the two mechanisms: every stopped segment used for critic learning receives the single-discount continuation, whereas the frozen reference changes only whether the accumulated segment reward is compressed. As the reference analysis in Appendix E.2 shows, the magnitude of this selective-reward effect can vary substantially across tasks and goals.

The per-goal results in Table 7 show that the single-discount controls already achieve substantial success on several evaluation goals, while the selective target remains strong across all five. The decomposition is therefore consistent with the aggregate result: continuation compression explains the largest change on this task, while the additional effect of reward selectivity is smaller and goal dependent.

Cross-task visual validation. To test whether the continuation-compression effect is specific to the state-based diagnostic above, we repeat the matched comparison on `visual-antmaze-giant-navigate-v0`. All variants use $K = 25$, three downstream seeds, the same visual encoder and primitive-action learning pipeline, and no SVC. The exact and single-γ segment-reward variants differ only in the endpoint bootstrap, replacing $\gamma^\kappa V(s^+, g)$ by $\gamma V(s^+, g)$ while leaving the accumulated segment reward unchanged. The reference-selective variant uses the same single-discount continuation and additionally applies reward compression according to the frozen shortcut-aligned reference used by the corresponding benchmark configuration.

Replacing the primitive-time bootstrap by the single-discount bootstrap increases aggregate success from 22.4 ± 2.0 to 60.3 ± 0.6 while leaving the accumulated segment reward unchanged. The improvement is also positive on each of the five evaluation goals, indicating that the large continuation-compression effect observed above is not confined to state-based observations or to a single visual goal. The reference-selective no-SVC variant reaches 53.5 ± 15.1 and exhibits substantially greater seed variability, so selective reward compression does not provide a consistent additional gain on this task. Together with the reference analysis in Appendix E.2, these results support separating the two mechanisms: single-discount continuation provides the common shortcut propagation relation, whereas the incremental effect of reference-selective reward compression can depend on the task and goal supervision.

Table 8: Cross-task validation of shortcut-target decomposition on `visual-antmaze-giant-navigate-v0`. All variants use the same primitive-action backbone, $K = 25$, three random seeds, and no SVC. Values are final binary success rate (%), mean $\pm$ standard deviation over seeds.

Target variant	Overall	Goal 1	Goal 2	Goal 3	Goal 4	Goal 5
Exact stopped K-step	22.4 ± 2.0	12.0 ± 8.7	29.3 ± 16.3	5.3 ± 5.8	20.0 ± 3.5	45.3 ± 20.8
Single-γ, segment reward (all-reject)	$\mathbf{60.3 \pm 0.6}$	38.0 ± 5.3	75.3 ± 6.4	40.0 ± 13.9	62.7 ± 2.3	85.3 ± 7.0
Reference-selective SSVL, no SVC	53.5 ± 15.1	32.0 ± 21.2	77.3 ± 14.2	39.3 ± 15.0	40.0 ± 33.0	78.7 ± 21.4

Table 9: Effect of frozen-reference construction on `antmaze-medium-explore-v0`. SVC is disabled in both settings, so the comparison isolates the reference used by the reward-compression gate. Values are final binary success rate (%), mean $\pm$ standard deviation over three seeds.

Frozen reference	Final success
One-step reference	65.6 ± 4.9
Shortcut-aligned reference	$\mathbf{79.6 \pm 8.6}$

E.2 Reference Alignment and Gate Diagnostics

Having isolated the universal single-discount continuation in Appendix E.1, we now examine the separate reward-side refinement provided by the frozen reference. The analysis asks how reference construction and training maturity affect selective reward compression, whether the resulting accepted set is enriched for an independent proxy of goal progress, and whether downstream performance can be explained by acceptance frequency alone.

One-step versus shortcut-aligned reference. We first compare two frozen references on `antmaze-medium-explore-v0`, with SVC disabled in both cases. The one-step reference is pretrained with primitive one-step Bellman relations, whereas the shortcut-aligned reference is pretrained with the stopped single-discount relations described in Appendix D.3. The downstream shortcut construction is unchanged: every stopped segment used for critic learning still uses the same single-γ continuation, and the reference enters only through the reward-compression gate. Table 9 shows that replacing the one-step reference by the shortcut-aligned reference increases final success from 65.6 ± 4.9 to 79.6 ± 8.6 over three seeds.

Because the endpoint construction and single-discount downstream bootstrap are unchanged, this comparison places the observed gain within the selective-reward pathway rather than the temporal-compression pathway studied in Appendix E.1. We interpret the result as evidence that the gate benefits from a reference trained on the same shortcut scale used downstream. It does not show that the shortcut-aligned reference is uniformly more accurate as a value estimator, nor that reference alignment must produce the same gain on every task.

Sensitivity to reference maturity. Reference construction and training maturity are distinct. We therefore keep the shortcut-aligned pretraining objective fixed and vary only how long the frozen reference is trained. Using the same pretraining seed, we retain matched references after 1k, 5k, and 100k gradient updates. To quantify maturity without relying on Euclidean geometry, we sample 100,000 data-supported full-K triples (s_t, s_{t+K}, g) for which g occurs strictly after s_{t+K} in the same logged trajectory, and report the trajectory-order consistency $r_{\mathrm{ord}} := \Pr(\widehat{Z}_t > 0)$ together with the mean reference progress $\mathbb{E}[\widehat{Z}_t]$, where $\widehat{Z}_t$ is the frozen-reference progress score in Eq. 26. This diagnostic asks whether the reference orders the logged K-step successor ahead of the segment start with respect to a later goal; it is not an estimate of optimal-path distance or value-function error. For downstream learning, the canonical two-sided SVC objective is enabled and held fixed across all rows, so the frozen reference checkpoint is the only changed component.

As the reference matures, trajectory-order consistency and mean reference progress increase, the fixed-threshold gate admits a broader set of shortcut relations, and downstream success rises from 80.8 ± 2.8 to 89.3 ± 2.2. The 1k reference is therefore measurably under-trained for this gate, while

Table 10: Sensitivity to frozen-reference maturity on `antmaze-medium-explore-v0`. The shortcut-reference objective, pretraining seed, and downstream SSVL configuration are fixed; only the number of reference-pretraining updates changes. Reference diagnostics use 100,000 logged full-K triples. Final success is mean $\pm$ standard deviation over three downstream seeds.

Ref. steps	r_{ord}	$\mathbb{E}[\widehat{Z}]$	**Gate acc.**	**Final success**
1k	63.0%	0.132	12.4%	80.8 ± 2.8
5k	65.1%	0.205	27.3%	85.7 ± 1.3
100k	66.3%	0.231	30.0%	89.3 ± 2.2

much of the change has already occurred by 5k updates. Beyond 100k updates, trajectory-order consistency is essentially unchanged, remaining between 66.1% and 66.3% at 100k, 300k, and 1M updates, while final success increases more modestly from 89.3 ± 2.2 to 92.4 ± 2.0 and 93.8 ± 2.6, respectively. Thus sensitivity is concentrated in the under-trained regime rather than requiring continually longer reference optimization. Because r_{ord} and $\mathbb{E}[\widehat{Z}]$ are empirical ordering diagnostics rather than estimates of the uniform reference error appearing in Theorem 3.4, this experiment is a sensitivity study rather than a direct verification of the theorem's sufficient condition.

Planar progress diagnostic. Reference acceptance is defined by shortcut-value progress, not by Euclidean distance. To obtain an external, interpretable check on the accepted set, we therefore use the planar coordinates of AntMaze only as an auxiliary geometric proxy. Let $s^{XY} \in \mathbb{R}^2$, $(s^+)^{XY} \in \mathbb{R}^2$, and $g^{XY} \in \mathbb{R}^2$ denote the first two state coordinates of the segment start, stopped shortcut endpoint, and sampled goal, respectively, and define

$$\Delta_{\mathrm{XY}}(s, s^+, g) = \left\| s^{XY} - g^{XY} \right\|_2 - \left\| (s^+)^{XY} - g^{XY} \right\|_2 . \tag{119}$$

Thus $\Delta_{\mathrm{XY}} > 0$ means that the shortcut endpoint is closer to the sampled goal in straight-line planar distance. This quantity is deliberately external to the SSVL gate: it is neither the reference score nor the progress notion used in the theoretical analysis, and it does not measure feasible-path or maze-topological progress because Euclidean distance can disagree with movement around obstacles.

For an acceptance indicator I, we report

$$\mathrm{Prec}_{XY} = \Pr(\Delta_{\mathrm{XY}} > 0 \mid I = 1), \qquad \mathrm{Rec}_{XY} = \Pr(I = 1 \mid \Delta_{\mathrm{XY}} > 0). \tag{120}$$

The diagnostic uses 500,000 sampled relations from the shortcut-aligned reference on `antmaze-medium-explore-v0`, with $K = 25$, $\gamma = 0.99$, and the default full-K threshold $w = 0.5$. Table 11 shows that 30.364% of full-K relations are accepted, while 89.041% of the accepted set has positive planar progress. The accepted and rejected sets also differ in mean planar progress, with $\Delta_{\mathrm{XY}} = 0.753$ and -0.100, respectively.

The accepted set is therefore strongly enriched for positive straight-line progress, providing an external check that the learned gate is not selecting an arbitrary subset of segments. At the same time, the score–Δ_{XY} Pearson correlation is only 0.261, so the reference score is not simply a rescaled version of this geometric proxy. These observations are diagnostic rather than a validation of the theoretical alignment assumption: they neither measure the action-conditioned acceptance gap in Theorem 3.4 nor establish that the reference has recovered maze-topological distance.

Threshold precision–recall tradeoff. Holding the sampled relations and frozen reference fixed, Table 12 varies only the gate threshold. Increasing w trades coverage for a more concentrated accepted set: acceptance falls from 59.56% at $w = 0$ to 11.97% at $w = 1$, XY-positive precision rises from 78.70% to 93.06%, and the corresponding recall falls from 79.88% to 18.99%. The default $w = 0.5$ therefore favors a comparatively precise accepted subset rather than maximizing reward-compression frequency.

A separate sweep on `antmaze-large-navigate-v0` tests whether downstream performance is explained simply by how often reward compression occurs. As Table 13 shows, varying w changes the full-K acceptance rate from 65.67% to 7.49%, yet final success remains high and is non-monotone in the acceptance fraction. Thus the mechanism is not well described as "accept more segments and perform better." This sweep is only a robustness check against that rate-only

Table 11: Gate diagnostics for the shortcut-aligned reference on `antmaze-medium-explore-v0` at $w = 0.5$. The analysis uses 500,000 sampled relations. "XY-positive" means $\Delta_{\mathrm{XY}} > 0$ according to Eq. 119.

Quantity	Value
Shortcut length K	25
Discount γ	0.99
Full-K fraction	76.929%
Full-K acceptance rate	30.364%
XY-positive fraction	58.675%
Accepted XY-positive precision	89.041%
XY-positive recall	46.077%
Mean accepted Δ_{XY}	0.753
Mean rejected Δ_{XY}	−0.100
Score–Δ_{XY} Pearson correlation	0.261
Implementation mismatch count	0

Table 12: Threshold sweep for the shortcut-aligned gate on `antmaze-medium-explore-v0`. The final column measures recall among relations in the top quartile of positive planar progress.

w	Acceptance	XY precision	XY recall	Top-Q recall
0.00	59.56%	78.70%	79.88%	92.51%
0.25	43.78%	84.64%	63.15%	83.33%
0.50	30.36%	89.04%	46.08%	69.36%
0.75	19.68%	91.82%	30.80%	52.18%
1.00	11.97%	93.06%	18.99%	34.85%

explanation; because changing the threshold also changes which segments are accepted, it is not a matched-rate causal test of gate identity.

Per-goal decomposition. Table 14 further shows that the effect of reference selectivity is goal dependent. On Goal 1 of `antmaze-medium-explore-v0`, the flat OTA-style uniform shortcut control and the one-step selective reference remain near zero at 3.3 ± 3.4 and 6.7 ± 3.8, whereas the shortcut-aligned selective reference reaches 45.3 ± 21.1. The remaining goals are already substantially easier under at least one control. Qualitatively, this pattern matches the distinction in Section 3.2: uniform shortcut propagation depends on the primitive-action signal retained by the logged endpoint distribution, while reference-selective reward compression can add separation when that signal is weak. The experiment does not directly measure B_K or the gate-induced margin M_K, so we treat this as supporting evidence rather than an empirical verification of the theorem's sufficient conditions.

E.3 SVC Variants and Sensitivity

Section 4.2 introduces SVC as an auxiliary cross-state relation for the separately learned state value, while Appendix C distinguishes its local consistency effect from the optimization effect of differentiating through both shortcut endpoints. We now test these ingredients empirically on `antmaze-medium-explore-v0`, keeping the shortcut-aligned frozen reference and selective critic target fixed throughout. The ablations separate successor-side gradient coupling from the auxiliary reward convention and then examine how strongly the consistency objective should be weighted relative to the base expectile loss.

Two-sided endpoint coupling. The canonical SVC relation couples shortcut-connected state values through

$$V(s, g) = -1 + \gamma V(s^+, g), \tag{121}$$

and therefore combines a unit-cost consistency convention with gradient contributions from both endpoint evaluations. To separate the latter optimization effect from the choice of auxiliary reward,

Table 13: Representative gate-threshold sweep on `antmaze-large-navigate-v0`. Changing w changes reward-compression frequency while preserving the common single-γ continuation.

w	**Full-K acceptance**	**Unit-reward rate**	**Final success**
0.0	65.67%	61.33%	90.4
0.1	45.23%	44.06%	90.0
0.2	31.40%	32.41%	86.0
0.3	20.95%	23.53%	89.6
0.4	13.06%	16.78%	**94.0**
0.5	7.49%	11.85%	92.4

Table 14: Per-goal final success (%) on `antmaze-medium-explore-v0`. SVC is disabled. Values are mean ± standard deviation over three seeds.

Setting	**Goal 1**	**Goal 2**	**Goal 3**	**Goal 4**	**Goal 5**
One-step reference	6.7 ± 3.8	74.0 ± 19.2	78.0 ± 25.5	72.7 ± 8.4	96.7 ± 0.9
Flat OTA-style	3.3 ± 3.4	76.0 ± 4.3	87.3 ± 12.7	80.7 ± 12.0	97.3 ± 2.5
Shortcut-aligned reference	$\mathbf{45.3 \pm 21.1}$	$\mathbf{86.7 \pm 6.8}$	76.7 ± 18.9	$\mathbf{90.0 \pm 5.9}$	$\mathbf{99.3 \pm 0.9}$

we use the zero-reward residual

$$\delta_0 = V(s, g) - \gamma V(s^+, g). \tag{122}$$

The two-sided diagnostic minimizes δ_0^2 with gradients through both $V(s, g)$ and $V(s^+, g)$. Its matched successor stop-gradient control uses the same numerical residual but removes the successor-side gradient contribution by minimizing

$$\big(V(s, g) - \mathrm{sg}\big[\gamma V(s^+, g)\big]\big)^2, \tag{123}$$

with respect to the sampled successor evaluation. This comparison targets the optimization-side distinction formalized locally in Proposition C.2; under shared neural-network parameters, it should be interpreted as a comparison of gradient pathways rather than as an independently parameterized tabular update.

Table 15 reports the resulting learning statistics. The shortcut-aligned selective model without SVC reaches 79.6 ± 8.6 final success. With the same zero-reward residual, retaining the successor-side gradient contribution raises final success to 90.7 ± 1.8, whereas the stop-gradient control reaches 74.7 ± 5.3; the same ordering appears in Last-3 performance, training AUC, and the best checkpoint. Because these two zero-reward variants share the same residual value and differ only in whether the successor evaluation contributes to the gradient, they provide the cleanest available ablation of the two-sided optimization component.

The 16.0-point final-success gap between the matched zero-reward variants supports the optimization role of two-sided shortcut consistency identified in Section 4.2: allowing the successor evaluation to participate in the gradient is substantially more effective on this task than treating it as a fixed target. Restoring the canonical unit-cost relation further increases final success from 90.7 ± 1.8 to 94.0 ± 1.3, indicating an additional contribution from the consistency convention itself. We therefore view two-sided coupling as the larger effect in this matched ablation, not as a claim that the auxiliary reward is unimportant in general.

Auxiliary reward mode. The preceding comparison separates the gradient pathway from the reward offset. We next ask whether the remaining benefit depends on a particular auxiliary reward convention. At $\lambda_{\mathrm{SVC}} = 0.3$, write the two-sided consistency residual for a generic auxiliary reward r_{SVC} as

$$\delta_{\mathrm{SVC}} = V(s, g) - \big(r_{\mathrm{SVC}} + \gamma V(s^+, g)\big). \tag{124}$$

We compare a constant positive reward, the canonical constant negative reward, and a normalized exponential-distance reward

$$r_{\exp} = \exp(-d_{\mathrm{norm}}), \qquad d_{\mathrm{norm}} \in [0, 1]. \tag{125}$$

Table 15: SVC coupling ablations on `antmaze-medium-explore-v0`. "Zero, two-sided" and "zero, stop-gradient" use the same zero-reward shortcut relation and differ only in whether gradients pass through the successor value.

Variant	Final	Last-3	AUC	Best
No SVC	79.6 ± 8.6	73.5 ± 6.8	65.1 ± 3.4	80.0 ± 8.3
Zero, stop-gradient	74.7 ± 5.3	75.5 ± 2.9	62.5 ± 1.8	80.3 ± 2.2
Zero, two-sided	90.7 ± 1.8	89.0 ± 1.9	81.7 ± 1.6	92.3 ± 2.7
Canonical -1, two-sided	$\mathbf{94.0 \pm 1.3}$	91.2 ± 0.3	83.1 ± 0.7	$\mathbf{95.1 \pm 0.8}$

Table 16: SVC auxiliary-reward robustness on `antmaze-medium-explore-v0`. All variants use two-sided endpoint optimization with $\lambda_{\mathrm{SVC}} = 0.3$.

SVC reward mode	Final	Last-3	AUC	Best
Constant positive	92.0 ± 2.1	91.4 ± 0.4	$\mathbf{84.7 \pm 1.0}$	94.5 ± 0.5
Canonical constant negative	94.0 ± 1.3	91.2 ± 0.3	83.1 ± 0.7	95.1 ± 0.8
Exponential distance, $\alpha = 1$	$\mathbf{94.4 \pm 2.0}$	$\mathbf{93.3 \pm 1.5}$	82.1 ± 1.7	$\mathbf{95.3 \pm 1.9}$

These choices modify only the auxiliary SVC relation; they do not change the environment reward, the stopped-segment critic target, or the frozen-reference gate.

All three two-sided reward conventions perform strongly, and no single choice dominates every metric: the exponential-distance variant has the highest Final, Last-3, and Best values, whereas the constant-positive variant has the highest AUC. We therefore retain the constant-negative relation in the default SSVL configuration because it matches the canonical unit-cost shortcut recursion in Eq. 121, not because it is empirically unique. Together with the matched zero-reward comparison, the result suggests that on this task the shared two-sided consistency structure is more stable across design choices than the precise auxiliary reward convention.

SVC coefficient sensitivity. Because SVC is auxiliary to the base expectile objective in Eq. 31, its weight should provide cross-state regularization without overwhelming local value fitting. We therefore vary λ_{SVC} under the canonical constant-negative relation and two-sided optimization. Table 17 exhibits an intermediate regime: final performance improves as the coefficient increases from 0 through 0.03, 0.10, and 0.30, but declines at $\lambda_{\mathrm{SVC}} = 1.0$. The two strongest intermediate settings differ only in which training summary they favor: $\lambda_{\mathrm{SVC}} = 0.10$ gives the highest Last-3 and AUC, whereas $\lambda_{\mathrm{SVC}} = 0.30$ gives the highest Final and Best values.

The sensitivity pattern is consistent with the intended role of SVC as a complementary state-value regularizer rather than a replacement for the base value-fitting objective. On this task, $\lambda_{\mathrm{SVC}} \in [0.1, 0.3]$ gives the strongest overall results, while assigning comparable or greater weight to the auxiliary term degrades performance. We use $\lambda_{\mathrm{SVC}} = 0.3$ for state-based locomotion in the default configuration; the task-family-specific coefficients used elsewhere are reported in Appendix D.4.

F Complete Benchmark Results

This section reports the complete OGBench benchmark results underlying the representative comparisons in Section 5. For each dataset, we report binary goal-reaching success (%) separately for the five standard test-time goals (`task1`–`task5`) together with the corresponding overall score. SSVL values are mean $\pm$ standard deviation over six seeds; controlled mechanism ablations elsewhere in the appendix use three seeds where explicitly stated. We include the six reference baselines benchmarked by OGBench: goal-conditioned behavioral cloning (GCBC) (Lynch et al., 2020; Ghosh et al., 2021), goal-conditioned implicit V-learning (GCIVL) and Q-learning (GCIQL) (Kostrikov et al., 2022; Park et al., 2023), quasimetric RL (QRL) (Wang et al., 2023), contrastive RL (CRL) (Eysenbach et al., 2022), and hierarchical implicit Q-learning (HIQL) (Park et al., 2023). The corresponding baseline columns retain the official OGBench results (Park et al., 2025), while the SSVL column reports our final results. The main text additionally compares SSVL with the more recent SAW and OTA methods on representative overlapping tasks; a dataset-level extended comparison

Table 17: Sensitivity to the SVC coefficient on `antmaze-medium-explore-v0`. The SVC relation uses the canonical constant-negative reward with two-sided endpoint optimization.

λ_{SVC}	**Final**	**Last-3**	**AUC**	**Best**
0	79.6 ± 8.6	73.5 ± 6.8	65.1 ± 3.4	80.0 ± 8.3
0.03	86.5 ± 0.8	83.7 ± 3.0	76.5 ± 3.1	88.1 ± 2.1
0.10	92.0 ± 1.5	$\mathbf{92.0 \pm 0.6}$	$\mathbf{83.9 \pm 1.6}$	94.3 ± 1.2
0.30	$\mathbf{94.0 \pm 1.3}$	91.2 ± 0.3	83.1 ± 0.7	$\mathbf{95.1 \pm 0.8}$
1.00	87.2 ± 0.8	83.2 ± 0.7	69.4 ± 0.1	89.2 ± 1.2

Table 18: **Complete results on PointMaze.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
pointmaze	navigate	pointmaze-medium-navigate-v0	task1	30 ±27	88 ±16	97 ±4	100 ±0	20 ±6	99 ±1	99 ±1
			task2	3 ±2	95 ±10	76 ±29	94 ±17	45 ±25	87 ±7	94 ±8
			task3	5 ±5	37 ±28	10 ±28	23 ±20	30 ±4	55 ±13	66 ±4
			task4	0 ±1	2 ±2	0 ±0	94 ±14	28 ±29	82 ±12	90 ±11
			task5	4 ±3	92 ±7	79 ±6	97 ±8	24 ±13	70 ±10	88 ±4
			overall	9 ±6	63 ±6	53 ±8	82 ±5	29 ±7	79 ±5	87 ±6
		pointmaze-large-navigate-v0	task1	63 ±11	76 ±23	86 ±14	95 ±8	42 ±27	83 ±13	70 ±19
			task2	1 ±2	0 ±0	0 ±0	100 ±0	31 ±24	2 ±7	87 ±5
			task3	10 ±7	98 ±5	83 ±8	40 ±50	78 ±7	88 ±10	88 ±13
			task4	20 ±18	0 ±0	0 ±0	96 ±7	24 ±14	72 ±19	82 ±10
			task5	52 ±17	53 ±20	0 ±0	96 ±7	20 ±10	46 ±16	62 ±30
			overall	29 ±6	45 ±5	34 ±3	86 ±9	39 ±7	58 ±5	78 ±15
		pointmaze-giant-navigate-v0	task1	1 ±3	0 ±0	0 ±0	98 ±7	6 ±15	0 ±0	100 ±0
			task2	1 ±4	0 ±0	0 ±0	92 ±16	28 ±10	72 ±17	79 ±14
			task3	0 ±0	0 ±1	0 ±0	68 ±27	9 ±5	32 ±11	42 ±14
			task4	0 ±0	0 ±0	0 ±0	66 ±20	64 ±17	60 ±22	97 ±7
			task5	5 ±12	0 ±0	0 ±0	19 ±32	29 ±28	66 ±20	65 ±19
			overall	1 ±2	0 ±0	0 ±0	68 ±7	27 ±10	46 ±9	77 ±5
		pointmaze-teleport-navigate-v0	task1	1 ±2	33 ±12	0 ±1	0 ±0	3 ±3	5 ±5	28 ±12
			task2	4 ±6	49 ±2	39 ±14	8 ±10	30 ±23	6 ±6	100 ±0
			task3	50 ±4	46 ±5	31 ±19	2 ±5	26 ±6	39 ±9	50 ±9
			task4	33 ±13	49 ±4	42 ±13	12 ±16	40 ±11	24 ±11	51 ±5
			task5	38 ±6	48 ±4	9 ±9	1 ±2	20 ±15	17 ±8	37 ±7
			overall	25 ±3	45 ±3	24 ±7	4 ±4	24 ±6	18 ±4	53 ±4
	stitch	pointmaze-medium-stitch-v0	task1	21 ±29	76 ±14	56 ±24	94 ±13	0 ±0	77 ±14	79 ±26
			task2	32 ±35	79 ±23	26 ±19	81 ±34	0 ±0	61 ±23	84 ±17
			task3	33 ±34	69 ±16	0 ±0	66 ±29	2 ±3	82 ±13	83 ±10
			task4	0 ±0	41 ±37	0 ±0	68 ±32	0 ±0	92 ±6	84 ±5
			task5	29 ±37	84 ±11	22 ±22	92 ±9	0 ±0	59 ±9	72 ±14
			overall	23 ±18	70 ±14	21 ±9	80 ±12	0 ±1	74 ±6	80 ±14
		pointmaze-large-stitch-v0	task1	8 ±13	0 ±1	56 ±11	100 ±1	0 ±0	3 ±5	82 ±20
			task2	0 ±0	0 ±0	0 ±0	74 ±37	0 ±0	0 ±0	65 ±10
			task3	26 ±28	60 ±29	98 ±4	74 ±23	0 ±0	59 ±25	99 ±2
			task4	0 ±0	0 ±0	0 ±0	88 ±32	0 ±0	1 ±4	53 ±18
			task5	0 ±0	0 ±0	0 ±0	85 ±22	0 ±0	0 ±0	42 ±12
			overall	7 ±5	12 ±6	31 ±2	84 ±15	0 ±0	13 ±6	68 ±12
		pointmaze-giant-stitch-v0	task1	0 ±0	0 ±0	0 ±0	99 ±2	0 ±0	0 ±0	59 ±9
			task2	0 ±0	0 ±0	0 ±0	80 ±27	0 ±0	0 ±0	43 ±8
			task3	0 ±0	0 ±0	0 ±0	3 ±5	0 ±0	0 ±0	58 ±14
			task4	0 ±0	0 ±0	0 ±0	63 ±23	0 ±0	0 ±0	70 ±19
			task5	0 ±0	0 ±0	0 ±0	4 ±8	0 ±0	0 ±0	67 ±21
			overall	0 ±0	0 ±0	0 ±0	50 ±8	0 ±0	0 ±0	59 ±6
		pointmaze-teleport-stitch-v0	task1	28 ±20	34 ±14	0 ±0	0 ±0	0 ±0	24 ±13	12 ±6
			task2	13 ±15	41 ±8	12 ±14	7 ±7	0 ±0	23 ±11	49 ±4
			task3	48 ±8	50 ±5	47 ±2	15 ±13	0 ±0	46 ±9	50 ±8
			task4	40 ±16	50 ±6	46 ±5	19 ±12	8 ±8	46 ±5	48 ±10
			task5	29 ±16	48 ±5	21 ±7	1 ±3	13 ±13	31 ±10	16 ±11
			overall	31 ±9	44 ±2	25 ±3	9 ±5	4 ±3	34 ±4	34 ±5

with these methods is reported separately because their published results do not provide the same complete per-goal coverage.

F.1 State-Based Locomotion Tasks

Tables 18–21 report the complete state-based locomotion results for PointMaze, AntMaze, HumanoidMaze, and AntSoccer. In addition to the dataset-level score, we retain the per-goal results to expose heterogeneity across evaluation goals that can be obscured by aggregate performance. This distinction is particularly relevant for long-horizon goal-conditioned control, where a method may learn reliable primitive-action behavior for some goals while remaining weak on others.

Table 19: **Complete results on AntMaze.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
antmaze	navigate	antmaze-medium-navigate-v0	task1	35 ±9	81 ±10	63 ±9	93 ±2	97 ±1	94 ±2	97 ±1
			task2	21 ±7	85 ±5	78 ±8	90 ±5	95 ±2	97 ±1	97 ±3
			task3	24 ±6	60 ±13	71 ±8	86 ±6	92 ±3	96 ±2	95 ±3
			task4	28 ±7	42 ±25	59 ±12	83 ±4	94 ±5	96 ±2	92 ±5
			task5	37 ±10	92 ±3	85 ±7	88 ±8	96 ±2	96 ±2	96 ±3
			overall	29 ±4	72 ±8	71 ±4	88 ±3	95 ±1	96 ±1	96 ±2
		antmaze-large-navigate-v0	task1	6 ±3	16 ±12	21 ±6	71 ±15	91 ±3	93 ±3	95 ±1
			task2	16 ±4	5 ±6	25 ±7	77 ±7	62 ±14	78 ±9	90 ±3
			task3	65 ±4	49 ±18	80 ±5	94 ±2	91 ±2	96 ±2	96 ±2
			task4	14 ±3	2 ±2	19 ±6	64 ±8	85 ±11	94 ±2	94 ±5
			task5	18 ±4	5 ±2	26 ±9	67 ±9	85 ±3	94 ±3	96 ±5
			overall	24 ±2	16 ±5	34 ±4	75 ±6	83 ±4	91 ±2	94 ±3
		antmaze-giant-navigate-v0	task1	0 ±0	0 ±0	0 ±0	1 ±2	2 ±2	47 ±10	72 ±5
			task2	0 ±0	0 ±0	0 ±0	17 ±5	21 ±10	74 ±5	80 ±6
			task3	0 ±0	0 ±0	0 ±0	14 ±8	5 ±5	55 ±7	65 ±9
			task4	0 ±0	0 ±0	0 ±0	18 ±6	35 ±9	69 ±5	85 ±4
			task5	1 ±1	1 ±1	1 ±1	18 ±5	16 ±10	82 ±4	77 ±8
			overall	0 ±0	0 ±0	0 ±0	14 ±3	16 ±3	65 ±5	76 ±6
		antmaze-teleport-navigate-v0	task1	17 ±5	35 ±5	26 ±5	31 ±6	35 ±5	37 ±5	42 ±5
			task2	51 ±5	41 ±5	58 ±8	47 ±22	92 ±3	66 ±8	93 ±4
			task3	22 ±3	36 ±8	31 ±5	35 ±6	47 ±4	37 ±5	55 ±4
			task4	25 ±5	45 ±3	33 ±5	33 ±6	50 ±2	30 ±2	60 ±2
			task5	14 ±6	38 ±6	26 ±9	28 ±8	44 ±3	41 ±8	40 ±5
			overall	26 ±3	39 ±3	35 ±5	35 ±5	53 ±2	42 ±3	58 ±4
	stitch	antmaze-medium-stitch-v0	task1	70 ±33	76 ±13	17 ±12	43 ±20	43 ±10	92 ±2	91 ±5
			task2	65 ±19	80 ±4	22 ±16	61 ±12	46 ±14	94 ±3	93 ±6
			task3	21 ±15	16 ±12	41 ±9	72 ±29	46 ±17	95 ±2	94 ±2
			task4	1 ±2	0 ±0	32 ±9	80 ±9	53 ±19	93 ±2	94 ±6
			task5	70 ±33	47 ±20	34 ±14	41 ±18	75 ±8	95 ±3	93 ±6
			overall	45 ±11	44 ±6	29 ±6	59 ±7	53 ±6	94 ±1	93 ±2
		antmaze-large-stitch-v0	task1	2 ±2	23 ±9	0 ±0	7 ±5	1 ±1	85 ±5	9 ±15
			task2	0 ±0	0 ±0	0 ±0	10 ±5	4 ±4	24 ±16	22 ±12
			task3	15 ±14	69 ±6	37 ±10	73 ±8	43 ±11	94 ±3	82 ±7
			task4	0 ±0	0 ±0	0 ±0	1 ±1	5 ±5	70 ±8	6 ±6
			task5	0 ±0	0 ±0	0 ±0	1 ±1	1 ±2	60 ±9	1 ±2
			overall	3 ±3	18 ±2	7 ±2	18 ±2	11 ±2	67 ±5	23 ±4
		antmaze-giant-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±1	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	5 ±5	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	3 ±3	0 ±0
			task5	0 ±0	0 ±0	0 ±0	2 ±2	0 ±0	0 ±1	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	0 ±0
		antmaze-teleport-stitch-v0	task1	21 ±13	39 ±7	12 ±4	22 ±6	30 ±6	44 ±5	35 ±11
			task2	39 ±12	44 ±6	18 ±7	22 ±6	30 ±4	42 ±3	57 ±12
			task3	34 ±12	36 ±8	18 ±4	25 ±7	23 ±11	26 ±4	46 ±4
			task4	46 ±6	44 ±4	18 ±5	24 ±9	38 ±4	26 ±4	49 ±4
			task5	16 ±14	33 ±6	17 ±6	26 ±5	32 ±7	40 ±6	35 ±8
			overall	31 ±6	39 ±3	17 ±2	24 ±5	31 ±4	36 ±2	44 ±8
	explore	antmaze-medium-explore-v0	task1	3 ±6	10 ±8	12 ±6	1 ±1	2 ±2	29 ±17	93 ±1
			task2	1 ±2	74 ±9	53 ±8	1 ±1	8 ±6	84 ±10	91 ±1
			task3	1 ±2	0 ±0	0 ±0	3 ±5	4 ±6	18 ±24	95 ±1
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	92 ±1
			task5	3 ±4	10 ±6	0 ±0	1 ±1	2 ±2	52 ±27	97 ±1
			overall	2 ±1	19 ±3	13 ±2	1 ±1	3 ±2	37 ±10	94 ±1
		antmaze-large-explore-v0	task1	0 ±0	37 ±12	1 ±1	0 ±0	0 ±1	1 ±3	72 ±1
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	38 ±1
			task3	0 ±0	12 ±6	1 ±1	0 ±0	1 ±1	18 ±24	77 ±1
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	65 ±1
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	75 ±1
			overall	0 ±0	10 ±3	0 ±0	0 ±0	0 ±0	4 ±5	66 ±1
		antmaze-teleport-explore-v0	task1	2 ±2	32 ±4	0 ±1	0 ±0	2 ±1	32 ±11	24 ±12
			task2	0 ±0	2 ±4	8 ±6	0 ±1	5 ±4	33 ±17	41 ±12
			task3	4 ±3	48 ±3	13 ±8	4 ±4	47 ±6	34 ±16	50 ±9
			task4	2 ±2	47 ±5	14 ±8	4 ±4	16 ±12	37 ±19	51 ±7
			task5	4 ±2	31 ±2	2 ±1	3 ±3	28 ±5	34 ±14	27 ±11
			overall	2 ±1	32 ±2	7 ±3	2 ±2	20 ±2	34 ±15	39 ±10

F.2 Visual Locomotion Tasks

Tables 22 and 23 report the complete per-goal results for Visual AntMaze and Visual Humanoid-Maze. These benchmarks replace state observations with pixels while preserving the long-horizon navigation and stitching structure of the corresponding domains, providing a complementary test of whether improvements in primitive-action value learning extend beyond state-based inputs. The per-goal breakdown again exposes variation across target goals that can be hidden by the aggregate score.

F.3 State-Based Manipulation Tasks

Tables 24–26 report the complete results for the state-based Cube, Scene, and Puzzle manipulation benchmarks. These domains require goal-conditioned control over object configurations and there-

Table 20: **Complete results on HumanoidMaze.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
humanoidmaze	navigate	humanoidmaze-medium-navigate-v0	task1	4 ±1	22 ±5	23 ±6	12 ±7	84 ±3	95 ±2	96 ±2
			task2	8 ±4	42 ±8	49 ±6	25 ±8	80 ±5	96 ±2	95 ±1
			task3	12 ±3	15 ±3	12 ±6	25 ±10	43 ±11	79 ±6	96 ±1
			task4	2 ±1	0 ±0	1 ±0	16 ±7	5 ±5	75 ±6	89 ±3
			task5	12 ±4	40 ±8	51 ±8	29 ±12	87 ±7	97 ±1	97 ±3
			overall	8 ±2	24 ±2	27 ±2	21 ±8	60 ±4	89 ±2	95 ±2
		humanoidmaze-large-navigate-v0	task1	1 ±1	6 ±2	3 ±2	3 ±2	36 ±11	67 ±4	78 ±6
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±3	38 ±6
			task3	3 ±1	6 ±2	5 ±2	17 ±6	54 ±17	88 ±3	90 ±5
			task4	2 ±1	0 ±0	1 ±1	4 ±2	23 ±11	42 ±11	80 ±6
			task5	1 ±1	1 ±1	1 ±1	2 ±1	6 ±4	47 ±10	72 ±4
			overall	1 ±0	2 ±1	2 ±1	5 ±1	24 ±4	49 ±4	72 ±5
		humanoidmaze-giant-navigate-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	13 ±7	50 ±10
			task2	0 ±0	1 ±1	1 ±1	2 ±1	9 ±5	35 ±11	75 ±10
			task3	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	11 ±4	55 ±8
			task4	0 ±0	0 ±0	0 ±0	0 ±0	3 ±2	2 ±2	72 ±9
			task5	1 ±1	0 ±0	1 ±1	2 ±1	1 ±1	2 ±2	85 ±9
			overall	0 ±0	0 ±0	0 ±0	1 ±0	3 ±2	12 ±4	67 ±9
	stitch	humanoidmaze-medium-stitch-v0	task1	20 ±7	13 ±3	12 ±3	6 ±5	27 ±7	84 ±5	59 ±6
			task2	49 ±12	7 ±2	8 ±5	13 ±4	37 ±7	94 ±2	59 ±13
			task3	24 ±8	25 ±3	20 ±7	30 ±6	40 ±4	86 ±4	59 ±12
			task4	3 ±2	1 ±1	2 ±2	18 ±5	28 ±7	86 ±4	32 ±16
			task5	49 ±8	16 ±3	18 ±7	22 ±2	49 ±5	90 ±4	70 ±8
			overall	29 ±5	12 ±2	12 ±3	18 ±2	36 ±2	88 ±2	56 ±6
		humanoidmaze-large-stitch-v0	task1	3 ±4	2 ±1	1 ±1	0 ±0	0 ±0	21 ±5	8 ±6
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	5 ±2	3 ±4
			task3	20 ±11	3 ±2	1 ±1	16 ±7	13 ±3	84 ±4	48 ±13
			task4	2 ±1	1 ±1	0 ±1	1 ±1	4 ±1	19 ±4	20 ±10
			task5	2 ±2	1 ±1	0 ±0	0 ±0	3 ±1	12 ±2	14 ±16
			overall	6 ±3	1 ±1	0 ±0	3 ±1	4 ±1	28 ±3	18 ±6
		humanoidmaze-giant-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±2	0 ±0
			task2	0 ±0	1 ±1	0 ±0	1 ±1	0 ±0	12 ±6	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	2 ±2
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			task5	0 ±0	0 ±0	1 ±1	1 ±1	0 ±1	0 ±1	1 ±1
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	3 ±2	0 ±1

Table 21: **Complete results on AntSoccer.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
antsoccer	navigate	antsoccer-arena-navigate-v0	task1	7 ±3	61 ±4	60 ±6	12 ±2	32 ±5	67 ±4	85 ±4
			task2	10 ±3	45 ±5	56 ±4	8 ±4	27 ±4	59 ±4	71 ±10
			task3	3 ±2	62 ±6	63 ±6	12 ±2	28 ±5	76 ±4	62 ±13
			task4	3 ±1	23 ±5	28 ±6	3 ±2	11 ±2	30 ±3	35 ±10
			task5	3 ±1	42 ±6	42 ±7	3 ±3	15 ±4	56 ±4	61 ±5
			overall	5 ±1	47 ±3	50 ±2	8 ±2	23 ±2	58 ±2	63 ±8
		antsoccer-medium-navigate-v0	task1	9 ±3	17 ±5	29 ±5	8 ±8	13 ±4	45 ±6	35 ±10
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±1	1 ±1	0 ±1	2 ±1	2 ±1
			task4	0 ±0	0 ±0	1 ±1	1 ±1	1 ±1	3 ±1	2 ±1
			task5	1 ±1	3 ±1	4 ±3	1 ±2	2 ±1	13 ±5	8 ±4
			overall	2 ±0	4 ±1	7 ±1	2 ±2	3 ±1	13 ±2	9 ±3
	stitch	antsoccer-arena-stitch-v0	task1	73 ±5	37 ±4	6 ±2	2 ±1	2 ±2	24 ±2	10 ±4
			task2	36 ±19	13 ±4	2 ±1	2 ±2	1 ±0	14 ±4	5 ±5
			task3	6 ±15	34 ±9	1 ±1	1 ±1	0 ±0	20 ±3	4 ±6
			task4	7 ±12	11 ±3	0 ±0	0 ±0	0 ±0	7 ±2	4 ±3
			task5	0 ±0	12 ±3	1 ±1	0 ±0	0 ±0	12 ±5	5 ±4
			overall	24 ±8	21 ±3	2 ±0	1 ±1	1 ±0	15 ±1	6 ±4
		antsoccer-medium-stitch-v0	task1	10 ±7	4 ±2	0 ±0	0 ±0	0 ±0	21 ±6	10 ±8
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±1	0 ±0
			overall	2 ±1	1 ±0	0 ±0	0 ±0	0 ±0	4 ±1	2 ±6

fore test SSVL beyond the locomotion settings above, under substantially different state–action–goal structure. The per-goal results expose variation across manipulation targets that can be obscured by the aggregate score.

F.4 Visual Manipulation and Additional Domains

Tables 27–30 report the complete results for Visual Cube, Visual Scene, Visual Puzzle, and Powderworld. The first three benchmarks extend the corresponding state-based manipulation domains to pixel observations, while Powderworld broadens the evaluation to a distinct goal-conditioned control setting with different state–action–goal structure. The per-goal results again reveal substantial variation across target goals that can be obscured by aggregate performance.

Table 22: **Complete results on Visual AntMaze.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
visual-antmaze	navigate	visual-antmaze-medium-navigate-v0	task1	17 ±6	30 ±7	16 ±3	0 ±0	92 ±2	90 ±4	91 ±3
			task2	8 ±2	21 ±6	7 ±2	0 ±0	94 ±2	92 ±7	92 ±4
			task3	17 ±1	24 ±5	16 ±4	0 ±0	98 ±1	94 ±4	94 ±3
			task4	12 ±2	21 ±3	9 ±2	0 ±0	94 ±2	94 ±2	95 ±3
			task5	4 ±2	16 ±5	6 ±2	0 ±0	94 ±2	94 ±5	96 ±4
			overall	11 ±2	22 ±2	11 ±1	0 ±0	94 ±1	93 ±4	94 ±3
		visual-antmaze-large-navigate-v0	task1	3 ±1	7 ±2	4 ±3	0 ±0	78 ±5	60 ±10	85 ±4
			task2	4 ±3	4 ±1	2 ±1	0 ±0	80 ±3	28 ±9	72 ±8
			task3	4 ±2	6 ±2	4 ±1	1 ±1	90 ±3	85 ±10	94 ±4
			task4	4 ±2	5 ±3	6 ±1	0 ±1	88 ±3	46 ±7	91 ±6
			task5	4 ±2	5 ±1	4 ±2	0 ±0	83 ±2	44 ±10	85 ±2
			overall	4 ±0	5 ±1	4 ±1	0 ±0	84 ±1	53 ±9	85 ±4
		visual-antmaze-giant-navigate-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	17 ±2	2 ±1	49 ±5
			task2	1 ±1	2 ±1	1 ±1	0 ±0	73 ±9	12 ±8	73 ±6
			task3	0 ±0	0 ±0	0 ±0	0 ±0	22 ±6	2 ±3	36 ±4
			task4	0 ±1	0 ±1	0 ±0	0 ±0	47 ±5	4 ±2	59 ±6
			task5	1 ±1	2 ±3	1 ±1	0 ±1	77 ±5	13 ±11	92 ±4
			overall	0 ±0	1 ±1	0 ±0	0 ±0	47 ±2	6 ±4	62 ±5
		visual-antmaze-teleport-navigate-v0	task1	2 ±2	6 ±1	2 ±1	3 ±2	32 ±3	32 ±5	28 ±14
			task2	6 ±3	9 ±3	9 ±2	6 ±4	73 ±8	40 ±6	74 ±7
			task3	9 ±1	12 ±3	9 ±2	10 ±4	47 ±3	33 ±1	54 ±5
			task4	10 ±2	10 ±2	8 ±3	6 ±4	50 ±4	44 ±5	51 ±6
			task5	1 ±1	3 ±1	3 ±1	4 ±2	36 ±5	33 ±7	31 ±9
			overall	5 ±1	8 ±1	6 ±1	6 ±3	48 ±2	37 ±2	48 ±4
	stitch	visual-antmaze-medium-stitch-v0	task1	80 ±4	0 ±1	0 ±0	0 ±0	33 ±4	75 ±8	86 ±5
			task2	90 ±4	1 ±2	0 ±0	0 ±0	69 ±5	85 ±7	91 ±6
			task3	69 ±18	15 ±6	8 ±1	0 ±0	88 ±1	92 ±1	95 ±2
			task4	1 ±1	7 ±4	3 ±1	0 ±1	70 ±12	88 ±4	87 ±21
			task5	97 ±1	6 ±3	1 ±1	0 ±0	85 ±5	93 ±1	96 ±3
			overall	67 ±4	6 ±2	2 ±0	0 ±0	69 ±2	87 ±2	92 ±4
		visual-antmaze-large-stitch-v0	task1	26 ±11	0 ±0	0 ±0	0 ±0	6 ±1	36 ±5	4 ±2
			task2	0 ±0	0 ±0	0 ±0	0 ±0	2 ±1	3 ±2	0 ±1
			task3	73 ±14	3 ±2	0 ±0	2 ±2	36 ±10	87 ±6	34 ±21
			task4	7 ±5	1 ±1	0 ±0	1 ±1	8 ±1	7 ±4	3 ±2
			task5	11 ±5	0 ±0	0 ±0	0 ±0	5 ±2	6 ±1	3 ±2
			overall	24 ±3	1 ±1	0 ±0	1 ±1	11 ±3	28 ±2	9 ±5
		visual-antmaze-giant-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	1 ±2	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±1	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
		visual-antmaze-teleport-stitch-v0	task1	37 ±4	2 ±2	1 ±1	0 ±0	20 ±5	36 ±5	1 ±1
			task2	36 ±3	2 ±1	1 ±1	1 ±1	40 ±9	38 ±3	2 ±2
			task3	17 ±6	2 ±1	2 ±1	3 ±4	32 ±9	36 ±5	2 ±2
			task4	39 ±9	1 ±1	0 ±0	2 ±3	45 ±7	37 ±6	0 ±0
			task5	29 ±1	1 ±1	1 ±1	1 ±1	22 ±9	38 ±5	3 ±2
			overall	32 ±3	1 ±1	1 ±0	1 ±2	32 ±6	37 ±4	1 ±1
	explore	visual-antmaze-medium-explore-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±2	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
		visual-antmaze-large-explore-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
		visual-antmaze-teleport-explore-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±1	0 ±1	0 ±0	3 ±1	38 ±8	3 ±1
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	28 ±13	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	27 ±18	2 ±2
			overall	0 ±0	0 ±0	0 ±0	0 ±0	1 ±0	19 ±8	1 ±1

F.5 Extended Comparison with SAW and OTA

The complete tables above use the OGBench baseline suite because these methods share a common five-goal reporting format. We further compare SSVL on all exact-task overlaps available from the published results of SAW (Zhou & Kao, 2025) and OTA (Ahn et al., 2025), two recent approaches that address long-horizon offline GCRL through complementary forms of long-to-short supervision. SAW transfers nearby-subgoal supervision to a flat policy through policy bootstrapping, whereas OTA shortens value propagation in the high-level state-value update of a hierarchical policy. Because the reported results follow each paper's own evaluation and aggregation protocol, we preserve the published means and standard deviations rather than treating these comparisons as matched-run significance tests. SSVL entries are the final dataset-level averages from Tables 18–30; boldface marks the larger reported mean within each row, with ties bolded for both methods, and does not indicate statistical significance.

Table 23: **Complete results on Visual HumanoidMaze.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
visual-humanoidmaze	navigate	visual-humanoidmaze-medium-navigate-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			task3	0 ±0	0 ±0	0 ±0	0 ±0	2 ±1	0 ±1	5 ±5
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	3 ±2	0 ±1	0 ±1
			overall	0 ±0	0 ±0	0 ±0	0 ±0	1 ±0	0 ±0	1 ±1
		visual-humanoidmaze-large-navigate-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
		visual-humanoidmaze-giant-navigate-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
	stitch	visual-humanoidmaze-medium-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	3 ±1	0 ±0	0 ±0	0 ±0	3 ±2	1 ±2	3 ±2
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			task5	1 ±1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			overall	1 ±0	0 ±0	0 ±0	0 ±0	1 ±0	0 ±0	1 ±0
		visual-humanoidmaze-large-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
		visual-humanoidmaze-giant-stitch-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0

Across these exact-task overlaps, no single control structure dominates uniformly. SSVL is competitive with or stronger than OTA on several maze datasets and achieves nontrivial performance on `pointmaze-giant-stitch`, where SAW reports zero success, while OTA retains substantial advantages on several HumanoidMaze stitching settings and visual manipulation tasks. SAW is likewise stronger on some PointMaze and stitching settings, whereas SSVL performs substantially better on several state-based manipulation and long-horizon navigation tasks. This variation is consistent with the supervision-regime discussion in Section 5.2: long-to-short supervision can be effective at different stages of the learning pipeline, but the quality of supervision available for ordering primitive actions with respect to the final goal remains task dependent.

Table 24: **Complete results on Cube.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
cube	play	cube-single-play-v0	task1	7 ±3	57 ±6	71 ±9	6 ±2	20 ±6	15 ±5	79 ±11
			task2	5 ±2	51 ±6	71 ±6	5 ±2	20 ±4	16 ±5	81 ±10
			task3	7 ±3	55 ±6	70 ±6	4 ±1	21 ±6	16 ±3	84 ±9
			task4	4 ±2	50 ±4	61 ±8	4 ±2	16 ±3	14 ±5	71 ±11
			task5	4 ±2	52 ±6	67 ±7	4 ±3	15 ±3	13 ±4	80 ±6
			overall	6 ±2	53 ±4	68 ±6	5 ±1	19 ±2	15 ±3	79 ±9
		cube-double-play-v0	task1	6 ±3	58 ±5	74 ±8	6 ±3	30 ±7	22 ±6	93 ±1
			task2	0 ±0	51 ±6	55 ±11	0 ±0	9 ±2	4 ±3	86 ±7
			task3	0 ±0	42 ±7	45 ±7	0 ±0	6 ±1	3 ±2	85 ±5
			task4	0 ±0	7 ±2	4 ±3	0 ±0	0 ±0	1 ±1	25 ±12
			task5	0 ±0	21 ±1	23 ±6	0 ±0	3 ±1	2 ±1	45 ±6
			overall	1 ±1	36 ±3	40 ±5	1 ±0	10 ±2	6 ±2	66 ±3
		cube-triple-play-v0	task1	5 ±4	3 ±1	13 ±3	1 ±1	19 ±5	12 ±6	85 ±11
			task2	0 ±0	0 ±0	0 ±0	0 ±0	2 ±1	0 ±0	56 ±10
			task3	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0	57 ±16
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	15 ±5
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	17 ±6
			overall	1 ±1	1 ±0	3 ±1	0 ±0	4 ±1	3 ±1	45 ±8
		cube-quadruple-play-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	16 ±10
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	5 ±5
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	5 ±3
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	6 ±3
	noisy	cube-single-noisy-v0	task1	5 ±3	71 ±12	100 ±1	17 ±12	39 ±4	48 ±6	98 ±2
			task2	7 ±5	70 ±11	100 ±0	23 ±8	39 ±7	39 ±7	97 ±2
			task3	1 ±1	67 ±10	99 ±1	4 ±3	36 ±7	41 ±7	92 ±6
			task4	16 ±5	76 ±10	98 ±2	47 ±14	36 ±4	36 ±6	94 ±3
			task5	12 ±5	70 ±12	100 ±0	37 ±9	42 ±5	44 ±9	95 ±3
			overall	8 ±3	71 ±9	99 ±1	25 ±6	38 ±2	41 ±6	96 ±2
		cube-double-noisy-v0	task1	7 ±3	53 ±11	64 ±8	16 ±5	9 ±5	10 ±3	30 ±3
			task2	0 ±0	10 ±4	16 ±4	0 ±0	0 ±0	1 ±0	34 ±5
			task3	0 ±0	1 ±1	6 ±4	0 ±0	0 ±0	1 ±1	30 ±5
			task4	0 ±0	4 ±2	11 ±3	0 ±1	0 ±0	0 ±1	36 ±3
			task5	0 ±0	4 ±2	20 ±4	0 ±0	0 ±0	0 ±1	38 ±2
			overall	1 ±1	14 ±3	23 ±3	3 ±1	2 ±1	2 ±1	33 ±3
		cube-triple-noisy-v0	task1	6 ±3	44 ±7	8 ±2	5 ±2	13 ±6	8 ±3	57 ±12
			task2	0 ±0	0 ±0	1 ±1	0 ±0	0 ±0	0 ±0	6 ±2
			task3	0 ±0	2 ±1	0 ±0	0 ±0	0 ±0	0 ±0	9 ±7
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	5 ±5
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2
			overall	1 ±1	9 ±1	2 ±1	1 ±0	3 ±1	2 ±1	15 ±3
		cube-quadruple-noisy-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	3 ±6
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±3
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±2

Table 25: **Complete results on Scene.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
scene	play	scene-play-v0	task1	18 ±7	75 ±5	93 ±4	19 ±4	49 ±7	40 ±4	90 ±5
			task2	1 ±1	62 ±8	82 ±8	1 ±1	12 ±4	40 ±9	97 ±3
			task3	2 ±1	64 ±7	72 ±10	1 ±1	26 ±8	36 ±5	87 ±4
			task4	3 ±2	7 ±4	8 ±3	5 ±2	5 ±2	55 ±5	85 ±8
			task5	0 ±0	2 ±1	1 ±1	0 ±1	1 ±1	20 ±5	61 ±4
			overall	5 ±1	42 ±4	51 ±4	5 ±1	19 ±2	38 ±3	85 ±4
	noisy	scene-noisy-v0	task1	6 ±3	60 ±11	50 ±5	39 ±10	5 ±4	68 ±5	37 ±18
			task2	0 ±0	42 ±11	52 ±13	2 ±1	0 ±0	29 ±6	64 ±10
			task3	0 ±0	27 ±6	28 ±5	1 ±1	0 ±1	17 ±6	26 ±14
			task4	0 ±0	3 ±3	0 ±0	3 ±3	0 ±0	10 ±8	24 ±11
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	3 ±4
			overall	1 ±1	26 ±5	26 ±2	9 ±2	1 ±1	25 ±4	31 ±11

Table 26: **Complete results on Puzzle.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
puzzle	play	puzzle-3x3-play-v0	task1	5 ±1	17 ±4	99 ±2	3 ±2	11 ±3	29 ±4	100 ±0
			task2	2 ±1	4 ±2	96 ±3	0 ±0	2 ±1	11 ±3	99 ±1
			task3	1 ±1	3 ±1	95 ±1	0 ±0	1 ±1	7 ±3	99 ±1
			task4	1 ±1	3 ±1	91 ±3	0 ±0	2 ±1	5 ±1	99 ±1
			task5	1 ±0	2 ±1	94 ±2	0 ±0	2 ±1	8 ±3	100 ±0
			overall	2 ±0	6 ±1	95 ±1	1 ±0	3 ±1	12 ±2	99 ±1
		puzzle-4x4-play-v0	task1	0 ±0	17 ±4	42 ±7	1 ±1	1 ±1	10 ±3	65 ±9
			task2	0 ±0	13 ±5	2 ±1	0 ±0	0 ±0	9 ±4	0 ±1
			task3	0 ±0	12 ±3	40 ±5	0 ±0	0 ±0	7 ±3	49 ±6
			task4	0 ±0	11 ±3	23 ±5	0 ±0	0 ±1	6 ±2	29 ±4
			task5	0 ±0	10 ±4	23 ±5	0 ±0	0 ±0	5 ±2	35 ±9
			overall	0 ±0	13 ±2	26 ±3	0 ±0	0 ±0	7 ±2	35 ±6
		puzzle-4x5-play-v0	task1	1 ±1	33 ±6	71 ±5	0 ±1	6 ±2	17 ±5	99 ±1
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	7 ±1	14 ±1	0 ±0	1 ±0	4 ±1	20 ±0
		puzzle-4x6-play-v0	task1	0 ±0	43 ±8	52 ±6	0 ±0	12 ±5	12 ±5	93 ±6
			task2	0 ±0	7 ±4	5 ±3	0 ±0	7 ±3	2 ±1	53 ±17
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	10 ±2	12 ±1	0 ±0	4 ±1	3 ±1	29 ±6
	noisy	puzzle-3x3-noisy-v0	task1	4 ±2	89 ±8	100 ±0	1 ±1	76 ±8	67 ±10	100 ±0
			task2	0 ±0	42 ±29	88 ±9	0 ±0	26 ±9	54 ±11	100 ±0
			task3	0 ±0	26 ±20	99 ±1	0 ±0	15 ±9	43 ±12	100 ±0
			task4	0 ±0	23 ±20	94 ±3	0 ±0	12 ±9	41 ±13	100 ±0
			task5	0 ±0	31 ±19	88 ±5	0 ±0	18 ±6	47 ±15	100 ±0
			overall	1 ±0	42 ±19	94 ±3	0 ±0	30 ±6	51 ±11	100 ±0
		puzzle-4x4-noisy-v0	task1	0 ±0	51 ±10	49 ±9	0 ±0	0 ±0	19 ±5	96 ±4
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	16 ±5	21 ±18
			task3	0 ±0	34 ±4	61 ±14	0 ±0	0 ±0	17 ±6	96 ±3
			task4	0 ±0	9 ±3	23 ±10	0 ±0	0 ±0	14 ±5	81 ±12
			task5	0 ±0	8 ±4	14 ±9	0 ±0	0 ±0	12 ±4	83 ±7
			overall	0 ±0	20 ±3	29 ±7	0 ±0	0 ±0	16 ±4	76 ±7
		puzzle-4x5-noisy-v0	task1	0 ±0	97 ±1	97 ±2	0 ±0	16 ±9	21 ±5	100 ±0
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	19 ±0	19 ±0	0 ±0	3 ±2	5 ±1	20 ±0
		puzzle-4x6-noisy-v0	task1	0 ±0	80 ±8	86 ±7	0 ±0	28 ±13	8 ±4	99 ±1
			task2	0 ±0	3 ±2	1 ±1	0 ±0	1 ±1	1 ±1	6 ±7
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	0 ±0	17 ±2	18 ±2	0 ±0	6 ±3	2 ±1	20 ±1

Table 27: **Complete results on Visual Cube.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
visual-cube	play	visual-cube-single-play-v0	task1	12 ±4	70 ±3	42 ±12	68 ±9	47 ±20	93 ±1	81 ±4
			task2	4 ±4	65 ±9	44 ±11	35 ±34	40 ±20	93 ±3	48 ±8
			task3	6 ±5	49 ±2	24 ±9	41 ±27	33 ±17	84 ±5	83 ±8
			task4	0 ±0	60 ±13	21 ±6	32 ±10	18 ±16	84 ±5	73 ±8
			task5	0 ±0	55 ±6	20 ±7	30 ±8	16 ±10	88 ±2	64 ±8
			overall	5 ±1	60 ±5	30 ±5	41 ±15	31 ±15	89 ±0	70 ±3
		visual-cube-double-play-v0	task1	4 ±3	44 ±8	6 ±4	20 ±3	7 ±4	91 ±1	78 ±8
			task2	0 ±0	0 ±1	0 ±0	2 ±2	0 ±0	54 ±5	30 ±10
			task3	0 ±1	0 ±0	0 ±0	2 ±0	0 ±0	40 ±6	25 ±8
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			task5	0 ±0	4 ±2	0 ±0	0 ±0	0 ±0	11 ±4	25 ±15
			overall	1 ±1	10 ±2	1 ±1	5 ±0	2 ±1	39 ±2	32 ±6
		visual-cube-triple-play-v0	task1	73 ±8	68 ±8	76 ±6	81 ±3	85 ±12	98 ±1	92 ±5
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	7 ±3	4 ±3
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1
			overall	15 ±2	14 ±2	15 ±1	16 ±1	17 ±2	21 ±0	20 ±1
		visual-cube-quadruple-play-v0	task1	42 ±4	1 ±2	36 ±7	23 ±5	20 ±4	66 ±7	58 ±14
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	1 ±1
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	1 ±1	0 ±0
			overall	8 ±1	0 ±0	7 ±1	5 ±1	4 ±1	14 ±1	12 ±2
	noisy	visual-cube-single-noisy-v0	task1	12 ±4	89 ±4	81 ±6	18 ±29	43 ±41	100 ±1	82 ±5
			task2	17 ±6	48 ±14	0 ±0	0 ±0	28 ±13	100 ±1	0 ±0
			task3	6 ±2	77 ±4	90 ±3	28 ±27	42 ±41	99 ±1	92 ±2
			task4	14 ±2	82 ±2	35 ±11	2 ±1	43 ±35	100 ±1	49 ±13
			task5	20 ±10	77 ±2	33 ±7	2 ±1	37 ±27	99 ±1	37 ±15
			overall	14 ±3	75 ±3	48 ±3	10 ±5	39 ±30	99 ±0	52 ±7
		visual-cube-double-noisy-v0	task1	20 ±5	70 ±8	70 ±5	27 ±9	24 ±11	98 ±2	72 ±8
			task2	2 ±2	5 ±3	14 ±2	2 ±4	3 ±3	87 ±9	17 ±5
			task3	2 ±1	5 ±5	16 ±6	0 ±0	2 ±2	68 ±10	15 ±7
			task4	1 ±1	3 ±1	0 ±0	0 ±0	0 ±0	13 ±9	0 ±0
			task5	0 ±0	0 ±0	7 ±2	1 ±1	0 ±0	30 ±4	12 ±3
			overall	5 ±1	17 ±4	22 ±2	6 ±2	6 ±3	59 ±3	24 ±3
		visual-cube-triple-noisy-v0	task1	80 ±7	90 ±5	62 ±6	44 ±21	78 ±7	99 ±1	70 ±7
			task2	0 ±1	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	0 ±0
			task3	0 ±1	0 ±0	0 ±0	0 ±0	0 ±0	14 ±11	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	16 ±1	18 ±1	12 ±1	9 ±4	16 ±1	23 ±2	14 ±3
		visual-cube-quadruple-noisy-v0	task1	46 ±2	2 ±1	10 ±9	2 ±2	39 ±10	60 ±41	20 ±8
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	2 ±2	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	9 ±0	0 ±0	2 ±2	0 ±0	8 ±2	12 ±8	4 ±10

Table 28: **Complete results on Visual Scene.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
visual-scene	play	visual-scene-play-v0	task1	59 ±7	84 ±4	56 ±4	44 ±6	52 ±6	80 ±6	83 ±3
			task2	0 ±0	24 ±8	1 ±1	2 ±2	1 ±1	81 ±7	54 ±9
			task3	0 ±0	16 ±8	0 ±0	0 ±0	0 ±0	61 ±11	42 ±16
			task4	2 ±1	0 ±0	3 ±4	2 ±1	1 ±1	20 ±8	29 ±9
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	3 ±2	3 ±4
			overall	12 ±2	25 ±3	12 ±2	10 ±1	11 ±2	49 ±4	42 ±8
	noisy	visual-scene-noisy-v0	task1	64 ±9	76 ±6	49 ±22	8 ±2	70 ±9	91 ±4	29 ±15
			task2	0 ±0	14 ±7	2 ±2	0 ±0	0 ±0	69 ±5	6 ±3
			task3	0 ±0	24 ±5	7 ±2	0 ±0	2 ±2	82 ±6	5 ±4
			task4	0 ±1	0 ±0	0 ±0	0 ±0	1 ±1	8 ±5	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	13 ±2	23 ±2	12 ±4	2 ±0	15 ±2	50 ±1	7 ±3

Table 29: **Complete results on Visual Puzzle.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
visual-puzzle	play	visual-puzzle-3x3-play-v0	task1	1 ±1	97 ±2	5 ±9	3 ±3	2 ±1	98 ±1	97 ±2
			task2	0 ±0	1 ±1	0 ±0	0 ±0	0 ±0	73 ±10	83 ±12
			task3	0 ±0	0 ±1	0 ±0	0 ±0	0 ±0	64 ±9	70 ±12
			task4	0 ±0	2 ±3	0 ±0	0 ±0	0 ±0	63 ±9	60 ±13
			task5	0 ±0	3 ±2	0 ±0	0 ±0	0 ±0	66 ±12	74 ±10
			overall	0 ±0	21 ±1	1 ±2	1 ±1	0 ±0	73 ±8	76 ±9
		visual-puzzle-4x4-play-v0	task1	11 ±3	86 ±7	18 ±5	0 ±0	10 ±7	70 ±47	64 ±13
			task2	19 ±5	8 ±5	28 ±12	0 ±0	19 ±13	38 ±32	33 ±17
			task3	8 ±3	80 ±4	15 ±4	0 ±0	8 ±7	66 ±44	36 ±18
			task4	6 ±2	65 ±9	10 ±5	0 ±0	7 ±5	66 ±44	39 ±20
			task5	6 ±2	61 ±8	9 ±3	0 ±0	4 ±4	60 ±41	40 ±22
			overall	10 ±1	60 ±5	16 ±4	0 ±0	10 ±6	60 ±41	43 ±17
		visual-puzzle-4x5-play-v0	task1	22 ±8	86 ±4	31 ±8	0 ±0	31 ±6	66 ±44	81 ±9
			task2	1 ±1	0 ±0	1 ±1	0 ±0	0 ±1	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	5 ±2	17 ±1	7 ±2	0 ±0	6 ±1	13 ±9	16 ±2
		visual-puzzle-4x6-play-v0	task1	12 ±4	66 ±3	10 ±3	0 ±0	12 ±5	34 ±23	53 ±13
			task2	0 ±1	8 ±2	2 ±2	0 ±0	2 ±1	8 ±6	12 ±6
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	2 ±1	15 ±1	2 ±1	0 ±0	3 ±1	9 ±6	14 ±3
	noisy	visual-puzzle-3x3-noisy-v0	task1	4 ±6	100 ±1	98 ±2	0 ±0	7 ±7	100 ±0	99 ±1
			task2	0 ±0	0 ±1	5 ±7	0 ±0	0 ±0	64 ±11	81 ±18
			task3	0 ±0	0 ±0	2 ±2	0 ±0	0 ±0	55 ±3	80 ±10
			task4	0 ±0	0 ±0	5 ±4	0 ±0	0 ±0	61 ±8	72 ±9
			task5	0 ±0	0 ±0	19 ±10	0 ±0	0 ±0	71 ±13	83 ±6
			overall	1 ±1	20 ±0	26 ±4	0 ±0	1 ±1	70 ±6	82 ±10
		visual-puzzle-4x4-noisy-v0	task1	6 ±2	90 ±9	85 ±3	0 ±0	4 ±3	98 ±2	76 ±5
			task2	16 ±7	1 ±1	1 ±1	0 ±0	14 ±1	44 ±20	7 ±8
			task3	4 ±3	88 ±4	77 ±6	0 ±0	4 ±4	95 ±2	79 ±15
			task4	2 ±1	36 ±10	47 ±17	0 ±0	2 ±1	91 ±2	67 ±10
			task5	4 ±2	20 ±9	34 ±13	0 ±0	5 ±4	94 ±1	50 ±12
			overall	7 ±3	47 ±3	49 ±7	0 ±0	6 ±2	84 ±4	56 ±10
		visual-puzzle-4x5-noisy-v0	task1	30 ±6	72 ±48	96 ±2	0 ±0	33 ±5	72 ±48	99 ±1
			task2	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	6 ±1	14 ±10	19 ±0	0 ±0	7 ±1	14 ±10	20 ±0
		visual-puzzle-4x6-noisy-v0	task1	10 ±2	61 ±41	82 ±4	0 ±0	9 ±7	56 ±7	85 ±8
			task2	0 ±0	1 ±1	2 ±1	0 ±0	0 ±0	12 ±10	3 ±2
			task3	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task4	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0
			overall	2 ±1	12 ±8	17 ±1	0 ±0	2 ±1	14 ±2	18 ±2

Table 30: **Complete results on Powderworld.**

Environment Type	Dataset Type	Dataset	Task	GCBC	GCIVL	GCIQL	QRL	CRL	HIQL	SSVL (ours)
powderworld	play	powderworld-easy-play-v0	task1	1 ±1	99 ±1	95 ±4	25 ±8	40 ±6	62 ±5	100 ±0
			task2	0 ±0	96 ±4	93 ±3	12 ±4	43 ±10	54 ±10	98 ±2
			task3	0 ±0	100 ±0	94 ±8	8 ±7	15 ±5	41 ±26	99 ±1
			task4	0 ±0	100 ±0	92 ±8	8 ±5	10 ±3	8 ±6	97 ±5
			task5	0 ±0	100 ±1	93 ±5	7 ±3	2 ±3	2 ±2	99 ±1
			overall	0 ±0	99 ±1	93 ±5	12 ±2	22 ±5	33 ±9	99 ±1
		powderworld-medium-play-v0	task1	0 ±0	81 ±12	7 ±14	2 ±3	0 ±1	26 ±20	41 ±23
			task2	3 ±3	28 ±14	0 ±0	9 ±6	2 ±3	16 ±12	64 ±16
			task3	2 ±3	99 ±2	72 ±14	4 ±5	2 ±2	60 ±37	100 ±0
			task4	0 ±0	39 ±17	0 ±1	0 ±0	0 ±0	5 ±4	35 ±27
			task5	0 ±0	2 ±2	0 ±0	0 ±0	0 ±0	1 ±2	1 ±1
			overall	1 ±1	50 ±4	16 ±5	3 ±1	1 ±1	22 ±14	48 ±13
		powderworld-hard-play-v0	task1	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	32 ±16
			task2	0 ±0	4 ±3	0 ±0	0 ±0	0 ±0	1 ±2	0 ±0
			task3	0 ±0	4 ±4	0 ±0	0 ±0	0 ±0	2 ±2	22 ±12
			task4	0 ±0	12 ±14	0 ±0	0 ±0	0 ±0	0 ±0	14 ±8
			task5	0 ±0	0 ±0	0 ±0	0 ±0	0 ±0	0 ±1	0 ±0
			overall	0 ±0	4 ±3	0 ±0	0 ±0	0 ±0	1 ±1	13 ±4

Table 31: **Extended comparison with SAW on locomotion, stitching, and exploration datasets.** SAW values are reported in the original paper. The SAW paper averages binary success across the five test-time goals and uses eight seeds for these state-based tasks. SSVL values are taken from the complete benchmark tables in this appendix.

Domain	Dataset	SAW	SSVL (ours)
PointMaze	medium-navigate	$\mathbf{97 \pm 2}$	87 ± 6
PointMaze	large-navigate	$\mathbf{85 \pm 10}$	78 ± 15
PointMaze	giant-navigate	68 ± 8	$\mathbf{77 \pm 5}$
AntMaze	medium-navigate	$\mathbf{97 \pm 1}$	96 ± 2
AntMaze	large-navigate	90 ± 3	$\mathbf{94 \pm 3}$
AntMaze	giant-navigate	73 ± 4	$\mathbf{76 \pm 6}$
HumanoidMaze	medium-navigate	88 ± 3	$\mathbf{95 \pm 2}$
HumanoidMaze	large-navigate	46 ± 4	$\mathbf{72 \pm 5}$
HumanoidMaze	giant-navigate	35 ± 4	$\mathbf{67 \pm 9}$
PointMaze	medium-stitch	$\mathbf{87 \pm 4}$	80 ± 14
PointMaze	large-stitch	8 ± 9	$\mathbf{68 \pm 12}$
PointMaze	giant-stitch	0 ± 0	$\mathbf{59 \pm 6}$
AntMaze	medium-stitch	$\mathbf{96 \pm 1}$	93 ± 2
AntMaze	large-stitch	$\mathbf{66 \pm 9}$	23 ± 4
AntMaze	giant-stitch	$\mathbf{1 \pm 1}$	0 ± 0
AntMaze	medium-explore	25 ± 4	$\mathbf{94 \pm 1}$
AntMaze	large-explore	6 ± 4	$\mathbf{66 \pm 1}$
HumanoidMaze	medium-stitch	$\mathbf{75 \pm 5}$	56 ± 6
HumanoidMaze	large-stitch	$\mathbf{25 \pm 3}$	18 ± 6
HumanoidMaze	giant-stitch	$\mathbf{4 \pm 3}$	0 ± 1

Table 32: **Extended comparison with SAW on manipulation and visual datasets.** SAW values are reported in the original paper; its pixel-based results use four seeds. * marks the `cube-single-play` SAW result reported with the task-specific value-learning hyperparameters noted by the SAW paper. SSVL values are taken from the complete benchmark tables in this appendix.

Domain	Dataset	SAW	SSVL (ours)
Cube	single-play	$72 \pm 5^*$	$\mathbf{79 \pm 9}$
Cube	double-play	40 ± 7	$\mathbf{66 \pm 3}$
Cube	triple-play	4 ± 2	$\mathbf{45 \pm 8}$
Scene	play	63 ± 6	$\mathbf{85 \pm 4}$
Visual AntMaze	medium-navigate	$\mathbf{95 \pm 0}$	94 ± 3
Visual AntMaze	large-navigate	82 ± 4	$\mathbf{85 \pm 4}$
Visual AntMaze	giant-navigate	10 ± 2	$\mathbf{62 \pm 5}$
Visual Cube	single-play	$\mathbf{88 \pm 3}$	70 ± 3
Visual Cube	double-play	$\mathbf{40 \pm 3}$	32 ± 6
Visual Cube	triple-play	$\mathbf{20 \pm 1}$	$\mathbf{20 \pm 1}$
Visual Scene	play	$\mathbf{47 \pm 6}$	42 ± 8

Table 33: **Extended comparison with reported OTA results on maze datasets.** OTA values are the eight-seed per-environment results reported in the original OTA paper. SSVL values are taken from the complete benchmark tables in this appendix.

Domain	Dataset	OTA	SSVL (ours)
PointMaze	medium-navigate	86 ± 2	$\mathbf{87 \pm 6}$
PointMaze	large-navigate	$\mathbf{85 \pm 5}$	78 ± 15
PointMaze	giant-navigate	72 ± 6	$\mathbf{77 \pm 5}$
PointMaze	medium-stitch	75 ± 5	$\mathbf{80 \pm 14}$
PointMaze	large-stitch	66 ± 8	$\mathbf{68 \pm 12}$
PointMaze	giant-stitch	52 ± 7	$\mathbf{59 \pm 6}$
AntMaze	medium-navigate	$\mathbf{96 \pm 1}$	$\mathbf{96 \pm 2}$
AntMaze	large-navigate	92 ± 1	$\mathbf{94 \pm 3}$
AntMaze	giant-navigate	$\mathbf{77 \pm 4}$	76 ± 6
AntMaze	medium-stitch	$\mathbf{93 \pm 1}$	$\mathbf{93 \pm 2}$
AntMaze	large-stitch	$\mathbf{84 \pm 3}$	23 ± 4
AntMaze	giant-stitch	$\mathbf{37 \pm 6}$	0 ± 0
AntMaze	medium-explore	$\mathbf{94 \pm 3}$	$\mathbf{94 \pm 1}$
AntMaze	large-explore	$\mathbf{75 \pm 16}$	66 ± 1
HumanoidMaze	medium-navigate	94 ± 1	$\mathbf{95 \pm 2}$
HumanoidMaze	large-navigate	$\mathbf{83 \pm 2}$	72 ± 5
HumanoidMaze	giant-navigate	$\mathbf{92 \pm 1}$	67 ± 9
HumanoidMaze	medium-stitch	$\mathbf{88 \pm 2}$	56 ± 6
HumanoidMaze	large-stitch	$\mathbf{57 \pm 3}$	18 ± 6
HumanoidMaze	giant-stitch	$\mathbf{79 \pm 3}$	0 ± 1

Table 34: **Extended OTA comparison on visual manipulation and supplemental overlapping tasks.** The visual manipulation noisy results are the eight-seed values reported in the OTA paper. The visual-play entries are additional results reported by OTA under its fixed visual-play comparison protocol. † denotes our six-seed reproduction of the hierarchical OTA algorithm on exact tasks not reported by the OTA paper. These reproduced values are kept distinct from both reported OTA and the flat OTA-style mechanism control used in Section E.1.

Domain	Dataset	OTA	SSVL (ours)
Visual Cube	single-noisy	$\mathbf{99 \pm 0}$	52 ± 7
Visual Cube	double-noisy	$\mathbf{65 \pm 2}$	24 ± 3
Visual Cube	triple-noisy	$\mathbf{26 \pm 2}$	14 ± 3
Visual Scene	noisy	$\mathbf{54 \pm 2}$	7 ± 3
Visual Cube	double-play	$\mathbf{51 \pm 3}$	32 ± 6
Visual Cube	triple-play	$\mathbf{28 \pm 1}$	20 ± 1
Visual Scene	play	$\mathbf{56 \pm 5}$	42 ± 8
Cube	double-play	$3 \pm 1^{\dagger}$	$\mathbf{66 \pm 3}$
Scene	play	$31 \pm 3^{\dagger}$	$\mathbf{85 \pm 4}$
Visual AntMaze	giant-navigate	$19 \pm 18^{\dagger}$	$\mathbf{62 \pm 5}$