A Surrogate Perspective on Convergence of Fixed-Target DQN

Zichu Liu[1,2*] **Nneka Okolo**[3] **Ryan D'Orazio**[1,2]
Danilo Vucetic[1,2] **Ioannis Mitliagkas**[1,2,4] **Gauthier Gidel**[1,2,4]

[1]Université de Montréal [2]Mila – Quebec AI Institute
[3]Massachusetts Institute of Technology [4]Canada CIFAR AI Chair

Abstract

Deep Q-Networks (DQNs) approximate value iteration by freezing a target network, forming Bellman-optimality targets, and running an inner regression loop that minimizes a smooth *surrogate loss* (e.g., MSE/Huber loss) on data from a replay buffer. Yet, control is governed by the Bellman optimality operator's γ-contraction in ℓ_∞, whereas common objectives (MSE/Huber loss) measure progress in an averaged ℓ_2/ℓ_1 geometry; thus, improving the surrogate need not reduce the worst-case Bellman error that drives performance. To bridge this gap we connect the progress in the inner loop for each surrogate to the Bellman residual under the ℓ_∞ norm. Our analysis provides explicit thresholds denoted by α under which sufficient progress guarantees a ℓ_∞ contraction up to a standard approximation floor. This yields prediction-space convergence guarantees for DQN with fixed targets trained using MSE or Huber-type losses, and motivates a novel soft-ℓ_∞ surrogate that smoothly approximates the sup-norm and better matches the ℓ_∞ geometry resulting in the least stringent inner loop accuracy requirements. Across Atari benchmarks, soft-ℓ_∞ consistently reduces the worst-case Bellman residual and matches or improves returns relative to standard objectives, providing a practical, geometry-aligned recipe for stable DQN training.

1 Introduction

Deep Q-Networks (DQNs) mitigate the instabilities that arise from combining function approximation, bootstrapping, and off-policy learning, the deadly triad [36, 39], by using experience replay together with a target network that is frozen over many gradient updates [22]. This induces a two-loop structure: an outer loop freezes the target and forms Bellman-optimality backups, while an inner loop approximately fits the online network to these targets by minimizing a surrogate regression loss on replay data. Fixed-target DQN is therefore an inexact fitted Q-iteration procedure, where the Bellman update is implemented through finite surrogate-optimization steps rather than solved exactly. The two loops are governed by different geometries. The Bellman optimality operator contracts in the sup norm [5, 37], and sup-norm Bellman residuals directly control worst-case greedy-policy quality [41]. In contrast, the inner losses used in DQN, such as MSE and Huber losses, measure progress in replay-weighted ℓ_2 or ℓ_1 geometries. Thus, decreasing the surrogate loss does not *automatically* certify a decrease in the worst-case Bellman residual. A model may fit the Bellman targets well on average while leaving large pointwise errors on poorly covered state-action pairs, precisely the type of error that can break the control-relevant ℓ_∞ contraction.

Classical FQI analyses study how Bellman-update errors propagate, but treat these errors as given, either through an exact regression oracle or through an assumed bounded-error condition [23, 8].

*Corresponding author: zichu.liu@mila.quebec.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

They do not address the preceding optimization question: when does finite optimization of a replay-weighted surrogate produce a Bellman error small enough for the outer contraction to survive? This question is unavoidable in fixed-target DQN, where the update error is produced by a finite inner loop over a nonconvex parameterization; the surrogate viewpoint lets us state the required progress in prediction space rather than as a local parameter-space guarantee. We answer this question through a surrogate-specific α-condition that measures multiplicative inner-loop progress on the surrogate objective. We derive explicit admissible α-ranges under which the fixed-target update contracts in ℓ_∞ up to an approximation-error radius. These ranges depend sharply on how well the surrogate geometry aligns with the sup norm. The resulting analysis separates two effects that are often merged in standard error-propagation views: the inherent Bellman error determines the approximation floor, whereas the inner-loop progress parameter α determines whether finite optimization is accurate enough to preserve the Bellman contraction. The same analysis motivates a temperature-controlled soft-ℓ_∞ surrogate based on a log-sum-cosh aggregation of TD residuals. This loss interpolates between a quadratic regime and a worst-case-focused regime, and its gradients adaptively concentrate on the largest residuals. For MSE, soft-Huber, and soft-ℓ_∞, we derive different admissible α-thresholds and translate them into sufficient inner-loop step budgets through surrogate curvature. Among these losses, soft-ℓ_∞ yields the least stringent outer-loop accuracy requirement, removing the coverage dependence from the admissible α-threshold.

Our contributions are: (i) we prove a contraction guarantee for fixed-target DQN under inexact inner-loop optimization, the practical regime that existing FQI theory idealizes away by assuming exact regression or bounded Bellman errors; (ii) we show that approximation and optimization errors play distinct roles: the inherent Bellman error sets the approximation radius, while the inner-loop progress α determines whether contraction holds, with surrogate-specific thresholds that we translate into concrete step budgets; (iii) we show that soft-ℓ_∞ removes the coverage dependence from the admissible α-threshold among the surrogates considered, and validate this prediction across Atari benchmarks: it consistently achieves the smallest or tied-smallest worst-case Bellman residual and matches or improves returns in most games.

2 Background

2.1 Notation and Definitions

For a finite set $\mathcal{S}$, let $\Delta_{\mathcal{S}} := \{p \in \mathbb{R}^{|\mathcal{S}|} : p \geq 0,\ \sum_{s\in\mathcal{S}} p(s) = 1\}$. For $x \in \mathbb{R}^n$, let $\|x\|_\infty := \max_i |x_i|$. Given a norm $\|\cdot\|$, its dual norm is $\|y\|_* := \sup_{\|x\|\leq 1}\langle y, x\rangle$. A differentiable function $f : \mathcal{X} \to \mathbb{R}$ is L-smooth with respect to $\|\cdot\|$ if $\|\nabla f(x) - \nabla f(x')\|_* \leq L\|x - x'\|$ for all $x, x' \in \mathcal{X}$. It satisfies the Polyak-Łojasiewicz (PL) condition with parameter $\mu > 0$ if $\frac{1}{2}\|\nabla f(x)\|_*^2 \geq \mu(f(x) - f_\star)$ for all $x \in \mathcal{X}$, where $f_\star := \inf_{x\in\mathcal{X}} f(x)$. For complete definition, see Appendix A.

2.2 Reinforcement Learning

We consider an infinite-horizon discounted Markov decision process (MDP) $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$, with finite state space $\mathcal{S}$, finite action set $\mathcal{A}$, transition kernel $P(\cdot \mid s, a)$, bounded rewards $|r(s, a)| \leq R_{\max}$, and discount $\gamma \in (0, 1)$. A policy $\pi(\cdot \mid s) \in \Delta_{\mathcal{A}}$ induces the state-value function $V^\pi(s) = \mathbb{E}_\pi\Big[\sum_{h\geq 0} \gamma^h r(s_h, a_h) \mid s_0 = s\Big]$ and action-value function $Q^\pi(s, a) = \mathbb{E}_\pi\Big[\sum_{t\geq 0} \gamma^h r(s_h, a_h) \mid s_0 = s, a_0 = a\Big]$. Dynamic programming characterizes Q^π, and the optimal action-value function $Q^\star(s, a) = \max_\pi Q^\pi(s, a)$, as unique fixed points of their respective Bellman operators. For any policy π, the Bellman evaluation operator is denoted by T^π and the Bellman optimality operator T is$(TQ)(s, a) = r(s, a) + \gamma\, \mathbb{E}_{s'\sim P(\cdot|s,a)}\Big[\max_{a'\in\mathcal{A}} Q(s', a')\Big]$. Both T^π and T are γ-contractions in the sup norm $\|\cdot\|_\infty$; hence according to Banach's fixed-point theorem each admits a unique fixed point (respectively Q^π and $Q^\star$), and the value iteration algorithm which iterates $Q_{t+1} = TQ_t$ converges in $\|\cdot\|_\infty$ to $Q^\star$ [4, 6]. This contraction property motivates value based methods that implement Bellman updates via fitted regression to Bellman targets, including Neural Fitted Q-Iteration [31] and Deep Q-Networks (DQN) [21].

In DQN [21], at outer iteration t a target network $Q_{\hat{\theta}_t}$ is frozen and kept fixed for K_{target} updates of the online network. Throughout, h denotes the timesteps along a trajectory, t the outer iteration,

and k the inner update steps; when t is fixed we drop it and write θ_k (Section 3.2 and Appendix D). Given a transition (s, a, r, s') sampled from the replay buffer $\mathcal{D}$, the (one-step) Bellman target is $Y_t(s, a, s') = r + \gamma \max_{a' \in \mathcal{A}} Q_{\hat{\theta}_t}(s', a')$, and the inner-loop objective is

$$\ell_t(\theta) = \mathbb{E}_{(s,a,r,s') \sim \mathcal{D}}[\mathcal{L}(Q_\theta(s, a) - Y_t(s, a, s'))], \tag{1}$$

where $\mathcal{L}$ is an L-smooth surrogate loss (e.g., MSE, Huber). Despite the ℓ_∞-*contraction* property of T that governs control, minimizing ℓ_t is typically carried out in an averaged geometry induced by the surrogate, which does not automatically imply ℓ_∞-contraction. To bridge this gap, we formalize a sufficient condition on the surrogate decrease under which the outer iteration contracts in ℓ_∞. The next section introduces the surrogate loss optimization framework and the α-conditions that can help us translate per-step surrogate improvement into an explicit ℓ_∞-contraction guarantee.

2.3 Surrogate methods

Surrogate methods replace direct parameter-space optimization with a sequence of tractable auxiliary objectives [20]. In the prediction-space surrogate view of D'Orazio et al. [9], writing $g(\theta)$ for the model prediction, one constructs a target y_t from the current prediction $g(\theta_t)$ and fits the model by reducing a discrepancy $L(g(\theta) - y_t)$. This shift lets convergence to be stated toward a global prediction-space solution, while the parameterization enters through whether the optimizer achieves the required surrogate decrease.

D'Orazio et al. [9] formalize this perspective for hidden monotone variational inequalities. Under Euclidean geometry and strong monotonicity of the prediction-space update operator, they show that a multiplicative decrease condition on each subproblem, termed the α-descent condition, suffices for linear contraction of the prediction iterates even when the inner loop is only solved approximately.

Fixed-target DQN fits this template: the frozen network defines the Bellman target Y_t, and the inner loop approximately minimizes the surrogate $\ell_t(\theta)$ in Equation 1. However, the Bellman optimality operator contracts in ℓ_∞, not in the replay-weighted ℓ_2/ℓ_1 geometry of standard surrogates. Consequently, the Euclidean α-condition analysis does not apply, and reducing a regression surrogate does not by itself certify ℓ_∞ contraction. Our analysis bridges this gap by deriving, for each surrogate, the admissible α-range under which inner-loop surrogate decrease implies outer-loop ℓ_∞ contraction.

2.4 Surrogate Losses for Q learning

Motivated by the mismatch between the regression geometry and the control-relevant ℓ_∞ metric, we instantiate a family of surrogate losses on the Bellman residual. The empirical objective is naturally written in terms of sampled TD error $Q_\theta(s, a) - Y_t(s, a, s')$. To avoid complications that naturally arise from studying this sample-based objective, we replace Y_t by its conditional expectation and work with the corresponding population residual $\varepsilon_{\theta,t} := Q_\theta - TQ_{\hat{\theta}_t}$. We show in Section 5 that our performance claims are validated by experiments with the empirical error. Let w denote the replay-induced distribution over $\mathcal{S} \times \mathcal{A}$. We study three smooth surrogates with increasing emphasis on worst-case discrepancies: MSE, a soft-Huber proxy (log-cosh), and a soft-ℓ_∞ proxy (log-sum-cosh).

(a) MSE (Mean Squared Error). A common choice of surrogate loss in DQN and its variants [22, 36] is the squared TD error:

$$\mathcal{L}_{\mathrm{MSE}}(\varepsilon_{\theta,t}) \;=\; \tfrac{1}{2}\mathbb{E}_{(s,a)\sim w}\big[\varepsilon_{\theta,t}(s, a)^2\big]. \tag{2}$$

MSE measures Bellman residual in an ℓ_2 geometry and prioritizes average error reduction. While analytically convenient under standard light-tailed noise assumptions [17], its quadratic penalty disproportionately up-weights rare large TD errors, so a small number of outliers can dominate the updates. Practical implementations commonly temper squared-error updates by using a clipped TD-error objective [22]. This practice motivates robust surrogates such as Huber loss.

(b) Huber and Soft Huber (Log Cosh). A standard robust alternative to MSE is Huber loss with threshold $\delta > 0$:

$$\mathcal{L}_\delta(u) = \begin{cases} \frac{1}{2}\, u^2, & |u| \le \delta, \\ \delta\big(|u| - \frac{1}{2}\delta\big), & |u| > \delta, \end{cases}$$

which is quadratic near 0 and linear in the tails, thereby bounding the per-sample influence [16]. For theoretical clarity and smooth optimization, we use a log-cosh proxy (soft Huber) [33]:

$$\mathcal{L}_{\mathrm{LC}}(\varepsilon_{\theta,t}, \beta) = \tfrac{1}{\beta^2}\mathbb{E}_{(s,a)\sim w}\big[\,\mathrm{logcosh}\big(\beta\, \varepsilon_{\theta,t}(s, a)\big)\big], \tag{3}$$

where $\beta > 0$ is a temperature parameter, with an effective Huber threshold $\delta \approx 1/\beta$. This surrogate is convex and infinitely differentiable, its per-sample derivative with respect to the TD error is $\tanh(\beta(\varepsilon_{\theta,t}))/\beta$, which is bounded between $[-\frac{1}{\beta}, \frac{1}{\beta}]$; hence, large TD errors are naturally down-weighted. The temperature β controls the transition scale: smaller β yields a wider MSE-like (quadratic) regime, while larger β induces earlier saturation and more ℓ_1-like tail behavior.

(c) Soft ℓ_∞ (Log Sum Cosh). The Bellman optimality operator contracts in the non-smooth ℓ_∞ geometry, therefore directly optimizing an ℓ_∞ residual leads to a non-smooth objective and can be challenging for standard first-order methods. Following classical max-structure smoothing [24], we replace $\|\cdot\|_\infty$ with the smooth proxy:

$$\mathcal{L}_{\mathrm{LSC}}(\varepsilon_{\theta,t}, \beta) = \tfrac{1}{\beta} \log \mathbb{E}_{(s,a)\sim w}\big[\cosh\big(\beta \varepsilon_{\theta,t}(s,a)\big)\big]. \tag{4}$$

As $\beta \to \infty$, $\mathcal{L}_{\mathrm{LSC}}$ concentrates on the largest coordinates and approximates $\|\varepsilon_{\theta,t}\|_\infty$ up to an $O(1/\beta)$ additive term (See Lemma C.3). For $|\beta \varepsilon_{\theta,t}| \ll 1$, it is locally quadratic (MSE-like up to a β-dependent scaling), yielding a more uniform aggregation across coordinates. Thus, increasing β provides a smooth transition from a well-conditioned, uniformly-weighted regime to a worst-case focused, ℓ_∞-aligned regime that is directly tied to Bellman contraction. Moreover, $\mathcal{L}_{\mathrm{LSC}}$ adaptively emphasizes larger discrepancies. Its gradient reweights coordinates by $p_i \propto w_i \cosh(\beta(\varepsilon_{\theta,t})_i)$, so larger residuals receive greater weight. This is similar to prioritized experience replay [34], but implemented as a differentiable reweighting within the loss rather than a sampling heuristic.

3 Main Contribution

We analyze fixed-target DQN as an inexact fitted Q-iteration: each target refresh forms a Bellman-optimality backup, and the inner loop approximately fits this backup by reducing a surrogate loss. Section 3.1 derives surrogate-specific α-conditions under which finite inner-loop surrogate decrease implies outer-loop ℓ_∞ contraction; Section 3.2 converts these conditions into sufficient inner-loop step budgets; and Section 3.3 gives the resulting adaptive algorithm.

3.1 α-conditions for Surrogate Losses

We view each target-network refresh as an inexact fitted Q-iteration. The α-condition controls the inner-loop inexactness, while the inherent Bellman error captures function-class misspecification. A coverage assumption then lets us convert distribution-weighted surrogate progress into a uniform ℓ_∞ Bellman-residual bound.

Definition 3.1 (Inherent Bellman Error (IBE) [23])**.** Let $\mathcal{Z} = \{Q_\theta : \theta \in \Theta\}$ denote the action-value function class and T be the Bellman optimality operator. The inherent Bellman error:

$$\varepsilon_\infty \;:=\; \sup_{Q\in\mathcal{Z}} \inf_{Q'\in\mathcal{Z}} \big\|Q' - TQ\big\|_\infty. \tag{5}$$

The special case $\varepsilon_\infty = 0$ corresponds to *Bellman completeness*, i.e., $TQ \in \mathcal{Z}$ for all $Q \in \mathcal{Z}$. ε_∞ measures the worst-case ℓ_∞-approximation error incurred when representing a target TQ within $\mathcal{Z}$.

We additionally impose the following coverage assumption on the sampling distribution.

Assumption 3.2 (Full coverage)**.** Let w be the sampling distribution over $\mathcal{S} \times \mathcal{A}$ and define $\underline{w} := \min_{(s,a)\in\mathcal{S}\times\mathcal{A}} w(s,a)$. We assume $\underline{w} > 0$.

IBE is standard in fitted Q-iteration analyses [23, 25]. Under uniformly bounded rewards and function classes it is finite, and in Theorem 3.3 it affects only the asymptotic radius, not the admissible α range or the contraction rate. Assumption 3.2 is the intrinsic cost of certifying a uniform ℓ_∞ Bellman contraction from a distribution-weighted surrogate loss. Classical FQI analyses avoid global full coverage by proving distributional L_p error-propagation bounds rather than uniform ℓ_∞ contraction; any analysis that certifies such a uniform statement from a weighted surrogate requires pointwise coverage. See Remark 3.5 for relaxation of coverage assumption.

With misspecification quantified by ε_∞ and distribution mismatch controlled by Assumption 3.2, we can now state an α-condition that specifies how much inner-loop surrogate decrease is sufficient to guarantee ℓ_∞ improvement in the outer iteration. Fix a surrogate $\mathtt{sur} \in \{\mathrm{MSE}, \mathrm{LC}, \mathrm{LSC}\}$ at step

t, denote the minimum surrogate loss attainable by our function approximator as $\mathcal{L}^\star_{\mathtt{sur},t}$ we say the inner loop achieves reduction factor $\alpha_{\mathtt{sur}} \in [0,1)$ if it returns θ_{t+1} satisfying,

$$\mathcal{L}_{\mathtt{sur}}\big(\varepsilon_{\theta_{t+1},t},\beta\big) - \mathcal{L}^\star_{\mathtt{sur},t} < \alpha_{\mathtt{sur}}\big(\mathcal{L}_{\mathtt{sur}}\big(\varepsilon_{\theta_t,t},\beta\big) - \mathcal{L}^\star_{\mathtt{sur},t}\big). \tag{6}$$

We refer to Equation 6 as the α-*condition* for `sur` at outer step t. The following theorem gives surrogate-specific admissible values of α under which the outer iterates contract linearly up to a surrogate-dependent error floor.

Theorem 3.3 (Surrogate α-conditions and ℓ_∞ contraction to a radius)**.** *Suppose Assumption 3.2 holds. Fix a surrogate* $\mathtt{sur} \in \{\mathrm{MSE}, \mathrm{LC}, \mathrm{LSC}\}$*, and let* $(Q_t)_{t\geq 0} \subset \mathcal{Z}$ *be the iterates produced by the outer loop. Suppose the inner loop at outer step* t *satisfies the* α*-condition Equation 6 for some* $\alpha \in [0,1)$*. Define* $E_t := \|Q_t - Q^\star\|_\infty$*. A sufficient condition for the outer update to be contractive in* ℓ_∞ *is that* α *lies in the surrogate-dependent range* $\alpha_{\mathrm{MSE}} < \underline{w}\Big(\frac{1-\gamma}{1+\gamma}\Big)^2$, $\alpha_{\mathrm{LC}} < \underline{w}\cdot\frac{1-\gamma}{1+\gamma}$, $\alpha_{\mathrm{LSC}} < \frac{1-\gamma}{1+\gamma}$. *In particular, choosing* $\alpha_{\mathrm{MSE}} = \frac{\underline{w}}{4}\left(\frac{1-\gamma}{1+\gamma}\right)^2$, $\alpha_{\mathrm{LC}} = \frac{\underline{w}}{2}\cdot\frac{1-\gamma}{1+\gamma}$, *and* $\alpha_{\mathrm{LSC}} = \frac{1-\gamma}{2+2\gamma}$ *yields the explicit bound* $E_t \leq \left(\frac{1+\gamma}{2}\right)^t E_0 + R_{\mathtt{sur}}$*, where the surrogate-dependent radii are*

$$R_{\mathtt{sur}} = \begin{cases} \dfrac{2}{1-\gamma}\sqrt{\dfrac{1}{\underline{w}} - \dfrac{(1-\gamma)^2}{4(1+\gamma)^2}}\,\varepsilon_\infty, & \mathtt{sur} = \mathrm{MSE}, \\[2ex] \dfrac{2}{1-\gamma}\left(\left(\dfrac{1}{\underline{w}} - \dfrac{1-\gamma}{2(1+\gamma)}\right)\varepsilon_\infty + \dfrac{\log 2}{\beta}\right), & \mathtt{sur} = \mathrm{LC}, \\[2ex] \dfrac{1+3\gamma}{1-\gamma^2}\,\varepsilon_\infty + \dfrac{2\log\frac{2}{\underline{w}}}{\beta(1-\gamma)}, & \mathtt{sur} = \mathrm{LSC}. \end{cases}$$

When $\varepsilon_\infty = 0$ (i.e., $\mathcal{Z}$ is *Bellman complete*), the approximation component of the radius disappears: $R_{\mathrm{MSE}} = 0$, and the remaining radii R_{LC} and R_{LSC} are due to smoothing, scaling as $O(1/\beta)$.

Remark 3.4 (Uniform, bias-free α-condition)**.** In addition to the surrogate-specific admissible ranges in Theorem 3.3, one may enforce the more stringent MSE-style requirement $\alpha < \underline{w}\Big(\frac{1-\gamma}{1+\gamma}\Big)^2$. This yields the same ℓ_∞ contraction rate as MSE and removes the surrogate-smoothing bias for LC/LSC. Consequently the steady-state radius becomes approximation-driven and $R_{\mathrm{sur}}(\varepsilon_\infty) \to 0$ as $\varepsilon_\infty \to 0$.

Remark 3.5 (Relaxing full coverage)**.** Full coverage can be relaxed to restricted coverage. The same proof applies on any region $\mathcal{R} \subseteq \mathcal{S}\times\mathcal{A}$ that is closed under one-step Bellman optimality backup: whenever $(s,a) \in \mathcal{R}$ and $P(s'|s,a) > 0$, we have $(s',a') \in R$ for every $a' \in \mathcal{A}$. Let $C^{\sup}_{\mathcal{R}} := \sup_{(s,a)\in\mathcal{R}} \frac{1}{w(s,a)}$. Theorem 3.3 holds with $\|f\|_{\infty,\mathcal{R}} := \sup_{(s,a)\in\mathcal{R}} |f(s,a)|$, $1/\underline{w}$ replaced by $C^{\sup}_{\mathcal{R}}$, and ε_∞ replaced by its restricted analogue $\varepsilon_{\infty,\mathcal{R}}$. See Appendix C.5 for formal statements.

Theorem 3.3 links *optimization progress* in the fitted regression step to *control-relevant* ℓ_∞ progress in the outer iteration. Two structural quantities govern this translation. First, the inherent Bellman error ε_∞ measures the worst-case ℓ_∞ approximation incurred when representing Bellman backups TQ within $\mathcal{Z}$; when $\varepsilon_\infty = 0$ the class is Bellman complete (closed under T) and the approximation-driven component of the radius vanishes. Second, the coverage parameter $\underline{w}$ controls how surrogate error measured *on average under* the replay distribution transfers to a *uniform* ℓ_∞ bound; smaller $\underline{w}$ corresponds to weaker coverage and amplifies the constants in the ℓ_∞ guarantee.

The admissible α thresholds reflect the mismatch between the surrogate geometry and the ℓ_∞ control geometry. For MSE and LC (soft-Huber), converting an averaged regression guarantee into a uniform bound introduces a dependence on $\underline{w}$, hence the stricter conditions $\alpha_{\mathrm{MSE}} = O\big(\underline{w}((1-\gamma)/(1+\gamma))^2\big)$ and $\alpha_{\mathrm{LC}} = O\big(\underline{w}(1-\gamma)/(1+\gamma)\big)$. In contrast, the LSC (soft-ℓ_∞) surrogate is a smooth proxy for a max-residual objective and therefore aligns more directly with ℓ_∞ control: its admissible range does not depend on $\underline{w}$. Finally, LC and LSC incur an additive $O(1/\beta)$ bias because a smooth max only approximates the nonsmooth ℓ_∞ residual; this term can be removed by imposing the stronger uniform α-condition (Remark 3.4), at the cost of a sharper objective that can require more inner-loop work which will be quantified in the next section.

Within the admissible ranges, the radius $R_{\texttt{sur}}$ increases monotonically with α, tighter inner-loop optimization yields a smaller steady-state ℓ_∞ error ball, while looser inner loops inflate the radius and eventually violate contractivity once α exceeds the surrogate-specific threshold.

3.2 Inner Loop Analysis

Having established α-conditions that guarantee an outer-loop ℓ_∞ contraction under sufficient surrogate decrease, we next quantify the number of gradient steps needed to achieve the α-condition for each surrogate, with $R_t := \max_{0\le k\le K} \|Q_{\theta_{t,k}} - TQ_{\hat{\theta}_t}\|_\infty$ denoting the largest frozen-target Bellman residual encountered in the inner loop at outer step t[2]. This follows from the curvature each surrogate induces, which controls the inner-loop optimization complexity. Throughout this subsection we study the complexity in the *tabular* setting, i.e., we analyze updates directly on Q-values to obtain a clean illustrative proxy for step complexity. In contrast, with parameterized models the nonlinear map couples the optimization geometry with the parametrization, making comparable analyses substantially more delicate. We analyze the *normalized sign* update, the steepest-descent step in $\|\cdot\|_\infty$. For an L_∞-smooth objective f, the update is

$$\theta_{k+1} = \theta_k - \frac{\|\nabla f(\theta_k)\|_1}{L_\infty}\,\mathrm{sign}\big(\nabla f(\theta_k)\big). \tag{7}$$

This update is an idealized proxy for adaptive optimizers such as Adam/AdamW, whose coordinate-wise rescaling often yields sign-like effective directions [2, 26]. The step complexity is bounded by combining ℓ_∞-smoothness with an ℓ_1-PL inequality that holds on an invariant sublevel set (see Lemma D.6). We state the step complexity below, using the specific α values from Theorem 3.3.

Proposition 3.6. *Fix an outer step t and let $\varepsilon_{t,k} := Q_{\theta_{t,k}} - TQ_{\hat{\theta}_t}$ denote the error at inner step k. Let $\phi(x) := \tanh^2(x)/\log\cosh(x)$ and $R_t := \max_{0\le k\le K}\|\varepsilon_{t,k}\|_\infty$. For any $\alpha \in (0,1)$, the α-condition holds after K iterations of Equation 7 whenever*

$$K \;\ge\; \frac{L_{\infty,\texttt{sur}}}{\mu_{\infty,\texttt{sur}}}\,\log\frac{1}{\alpha},$$

where the ratio of the smoothness and PL constants of the surrogate, $L_{\infty,\texttt{sur}}/\mu_{\infty,\texttt{sur}}$, equals $1/\underline{w}$ for MSE, $2/(\underline{w}\,\phi(\beta R_t))$ for LC, and $4/(\underline{p}_t\,\phi(\beta R_t))$ for LSC. Substituting the α choices suggested by Theorem 3.3, the number of iterations sufficient to achieve them satisfies

$$K_{\texttt{sur}} \;\ge\; \begin{cases} \dfrac{1}{\underline{w}}\left(\log\dfrac{1}{\underline{w}} + 2\log\dfrac{1+\gamma}{1-\gamma} + \log 4\right), & \texttt{sur} = \mathrm{MSE},\\[2ex] \dfrac{2}{\underline{w}\,\phi(\beta R_t)}\left(\log\dfrac{1}{\underline{w}} + \log\dfrac{1+\gamma}{1-\gamma} + \log 2\right), & \texttt{sur} = \mathrm{LC},\\[2ex] \dfrac{4}{\underline{p}_t\,\phi(\beta R_t)}\left(\log\dfrac{1+\gamma}{1-\gamma} + \log 2\right), & \texttt{sur} = \mathrm{LSC}, \end{cases}$$

where $p_i(\varepsilon) := \frac{w_i\cosh(\beta\varepsilon_i)}{\sum_j w_j\cosh(\beta\varepsilon_j)}$ and $\underline{p}_t := \min_{0\le k\le K}\min_i p_i(\varepsilon_{t,k})$. $\phi(x)$ is decreasing with $\phi(0) = 2$ and $\phi(x)\approx 1/x$ as $x\to\infty$.

Proposition 3.6 gives a clean decomposition of inner-loop complexity into two factors. First, the *outer geometry* fixes how much surrogate decrease is needed per target refresh, captured entirely by the α-condition through a $\log(1/\alpha)$ requirement. Second, the *inner curvature* determines the *true optimization geometry*; it controls the constant multiplying $\log(1/\alpha)$, i.e., the per-step progress achievable by steepest descent in $\|\cdot\|_\infty$. This separation clarifies how different surrogates behave. For MSE and soft-Huber, both the admissible α and the curvature coefficient inherit an unfavorable dependence on worst-case replay coverage through $\underline{w}$. In contrast, soft-ℓ_∞ *decouples the outer requirement from $\underline{w}$* (via the w-independent α range in Theorem 3.3) and its inner curvature is *adaptively reweighted* through $p(\cdot)$, concentrating optimization effort on the largest residual coordinates. When $\beta R_t = O(1)$, $\phi(\beta R_t)$ remains a stable constant and the reweighting factor $p(\cdot)$ stays well-behaved on

[2] If $|r|\le R_{\max}$, then both $\|Q\|_\infty$ and $\|TQ\|_\infty$ are bounded by the same constant $R_{\max}/(1-\gamma)$, so $R_t \le 2R_{\max}/(1-\gamma)$.

the bounded region that contains the iterates, yielding a more favorable and predictable optimization geometry than MSE/Huber while simultaneously avoiding worst-case coverage constants in the outer loop. This also motivates a practical rule: schedule β to keep $\beta R_t \approx O(1)$, preventing curvature degradation and avoiding overly peaked reweighting. Thus, Proposition 3.6 provides an idealized sign-normalized model of the inner optimizer, predicting the qualitative dependence of inner-loop difficulty on $\log(1/\alpha)$ and surrogate curvature; our experiments test this prediction under AdamW.

3.3 Algorithms

Theorem 3.3 and Proposition 3.6 together motivate an *adaptive* schedule for the temperature factor β to meet the required α-condition while maintaining good inner-loop conditioning. For soft Huber and soft ℓ_∞, this policy lets early iterations use greater effective curvature (small β), while later iteration increase ℓ_∞ alignment and reduce the radius of the asymptotic neighborhood. The resulting adaptive surrogate method is summarized in Algorithm 1.

Algorithm 1 Deep Q-Learning with Adaptive Surrogate

Input: outer iterations T; inner iterations K_{target}; discount γ; exploration ϵ; stepsize η; surrogate loss $\mathcal{L}(\cdot;\beta)$ (MSE / soft-Huber / soft-ℓ_∞); `use_adaptive_beta`$\in\{0,1\}$; controller $C>0$; clip range $[\beta_{\min},\beta_{\max}]$; EMA rate $\rho\in(0,1)$

Initialize parameters θ; set target parameters $\hat{\theta}\leftarrow\theta$; initialize an empty replay buffer $\mathcal{D}$
Initialize $\beta\leftarrow\beta_{\min}$; $\bar{E}\leftarrow 1$; observe initial state s
for $t=1$ **to** T **do**
  for $k=1$ **to** K_{target} **do**
    Select action a via ϵ-greedy from $Q_\theta(s,\cdot)$
    Execute a; observe r, s', and the termination flag $d\in\{0,1\}$ ($d=1$ if s' is terminal); store (s,a,r,s',d) in $\mathcal{D}$; $s\leftarrow s'$
    if $d=1$ **then**
      Reset environment; observe new initial state s
    end if
    Sample minibatch $\{(s_i,a_i,r_i,s_i',d_i)\}_{i=1}^m\sim\mathcal{D}$
    For each $i=1,\dots,m$
    Set $Y_i\leftarrow r_i+\gamma(1-d_i)\max_{a'\in\mathcal{A}}Q_{\hat{\theta}}(s_i',a')$
    Set $Q_i\leftarrow Q_\theta(s_i,a_i)$ and $\varepsilon_i\leftarrow Q_i-Y_i$
    if `use_adaptive_beta` $=1$ **then**
      $E\leftarrow\max_{i=1,\dots,m}|\varepsilon_i|$ (i.e., $E=\|\varepsilon\|_\infty$)
      $\bar{E}\leftarrow(1-\rho)\bar{E}+\rho E$; $\beta\leftarrow\text{clip}\left(\frac{C}{\bar{E}},[\beta_{\min},\beta_{\max}]\right)$
    end if
    $\theta\leftarrow\theta-\eta\,\nabla_\theta\,\mathcal{L}(\varepsilon;\beta)$
  end for
  Synchronize target network: $\hat{\theta}\leftarrow\theta$
end for
Output: Q_θ and greedy policy $\pi(s)=\arg\max_{a\in\mathcal{A}}Q_\theta(s,a)$

4 Related Works

Classical fitted value/policy iteration analyses bound how controlled Bellman-update errors propagate through dynamic-programming iterations, often under distributional L_p, concentrability, or representation assumptions [23, 1, 12, 8]. These works either rely on exact updates, or assume a bounded update-error sequence and analyze its propagation. Such assumptions are appropriate for error propagation, but they leave implicit how the update error is produced by finite inner-loop optimization. Specifically, an exact fitted-update oracle is the strongest case ($\alpha=0$); a bounded-error assumption similarly takes as input that the inner loop already reaches the accuracy needed for propagation, and realizing it with a finite inner loop imposes a target-dependent multiplicative-decrease requirement with the same coverage and contraction-gap dependence as Theorem 3.3. Our α-condition makes this hidden inner-loop requirement explicit and surrogate-dependent; see Appendix C.5 for details.

A complementary line of work proves convergence or finite-sample rates for value-based deep RL by exploiting specific parameterizations, sampling models, or overparameterized training regimes. Fan et al. [10] analyze a simplified DQN with target networks and experience replay, proving geometric

decay of the outer FQI error up to finite-sample and neural-approximation errors for deep ReLU classes; Xu & Gu [42] prove an $O(1/\sqrt{T})$ rate for sufficiently overparameterized neural Q-learning under non-i.i.d. Markovian data; and Cai et al. [7] prove convergence of neural TD to MSPBE global optima for *policy evaluation* in an NTK regime. These are strong global value-learning guarantees, but their convergence mechanisms are tied to the chosen architecture, sampling model, or overparameterized training dynamics. They do not isolate how the choice of regression-loss geometry affects the required inner-loop accuracy under fixed targets. Our analysis is complementary and parameterization-agnostic in a conditional sense: once the inherent Bellman error ε_∞ and the achieved inner-loop decrease α are specified, the theorem applies at the prediction level. This separates model capacity, which controls the approximation radius, from finite surrogate optimization, whose admissible accuracy depends sharply on the loss geometry.

A separate line of work studies optimization of robust Bellman objectives in the gradient-TD regime: Patterson et al. [28] derive saddlepoint reformulations and gradient-based algorithms for Huber and absolute Bellman errors without target networks. This studies a different algorithmic regime than the fixed-target two-loop setting analyzed here, and our α-thresholds give the corresponding finite-step contraction result for MSE, soft-Huber, and soft-ℓ_∞.

Recent empirical work studies how value-based agents scale with model capacity, update-to-data ratio, batch size, and compute [35, 32, 13], identifying Pareto frontiers between data and compute, compute-optimal allocations, and TD-overfitting effects. Our analysis gives a complementary mechanism: capacity enters through ε_∞, inner-loop compute governs the achievable α, and surrogate geometry governs the admissible α. Scaling capacity or updates helps preserve the outer ℓ_∞ contraction guarantee only when the achieved α stays within the surrogate-specific admissible range.

5 Experiment

Basic hyperparameter setup. We train each Q-learning agent for 15M frames per Atari game [38]. Updates begin after 80k steps and occur every 4 environment steps with batch size 32 sampled from a 1M transition uniform replay buffer; the discount is $\gamma = 0.99$. We employ Double-DQN [14] targets and synchronize the target network every $K_{\text{target}} = 500$ updates with $\tau = 1.0$ (hard copy, no smoothing). Optimization uses AdamW [19] with learning rate 5×10^{-5} and default PyTorch [27] hyperparameters (betas $0.9, 0.999$; $\epsilon = 10^{-8}$; weight decay 10^{-2}). Exploration follows a linear ϵ-greedy schedule from 1.0 to 0.01 over the first 10% of training, after which $\epsilon = 0.01$ is held fixed. Our network follows the standard Atari DQN design of Mnih et al. [22], comprising a three-layer CNN followed by a 512-unit fully connected layer and a linear action head; full architectural details (kernel sizes, strides, channels) are given in the Appendix. We compare MSE, soft Huber, and soft ℓ_∞. For soft Huber and soft ℓ_∞ we use an adaptive temperature $\beta_t = \text{clip}\Big(\frac{C}{\bar{E}_t},\, [\beta_{\min},\, \beta_{\max}]\Big)$, where $\bar{E}_t$ is an exponential moving average of the 0.95-quantile of error computed on the training minibatch, and $[\beta_{\min}, \beta_{\max}] = [0.1,\ 20]$. We set $C = 1$ for both soft ℓ_∞ and soft Huber. The adaptive schedule keeps $\beta_t \bar{E}_t \approx C$, matching the curvature heuristic from Proposition 3.6: early training uses a smoother quadratic regime, while later training becomes more ℓ_∞-aligned as residuals shrink. At the beginning of training, network outputs are near zero and $\bar{E}$ can be trivially small. To prevent the large β_t at initialization we apply a warm-up: $\beta_t = 0.5$ during the first 20% of training frames, after which the controller above is enabled. Every 50k steps we perform greedy ($\epsilon = 0$) evaluation for 5 episodes in a non-clipped evaluation environment. All curves report mean $\pm$ std over 5 seeds. Our implementation builds on CleanRL [15], including its Atari wrappers and replay buffer utilities.

Experimental results. Figs. 1–2 summarize our results under the standard setup. We evaluate the three surrogate functions and report (i) episodic return of the greedy policy on evaluation environment and (ii) the maximum Bellman residual R_∞ estimated on a fixed held-out evaluation set $\mathcal{D}_{\text{hold}}$ sampled from the replay buffer: $R_\infty(t) = \max_{(s,a)\in\mathcal{D}_{\text{hold}}} \big| Q_{\theta_t}(s,a) - Y_t(s,a,s') \big|$. The residual ordering largely tracks final return: smaller R_∞ usually yields higher or comparable returns, and near-tied returns correspond to trajectories with similar residuals. Soft ℓ_∞ attains the smallest or tied-smallest R_∞ in every game and the highest or tied-highest return in five of eight (DemonAttack, Breakout, Phoenix, Robotank, Assault); MSE is intermediate on R_∞ and best on SpaceInvaders and Seaquest; soft Huber has the largest R_∞ and lowest returns in most games, with MsPacman as the sole exception. This ordering reflects how each surrogate weights residual coordinates: the log-sum-cosh gradient $p_i \tanh(\beta\varepsilon_i)$ with $p_i \propto w_i \cosh(\beta\varepsilon_i)$ concentrates on the largest residuals;

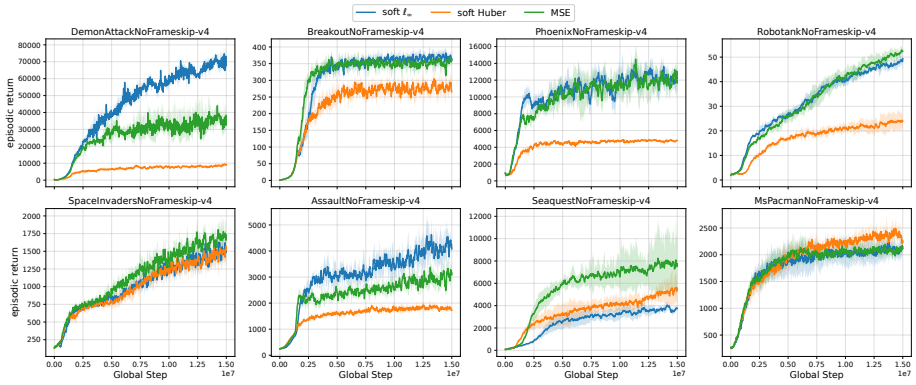

Figure 1: Learning curves across Atari games comparing soft ℓ_∞ (blue), soft Huber (orange), and MSE (green). *x-axis*: global training steps. *y-axis*: episodic return. Solid lines show the mean over 5 seeds; shaded bands denote the standard deviation.

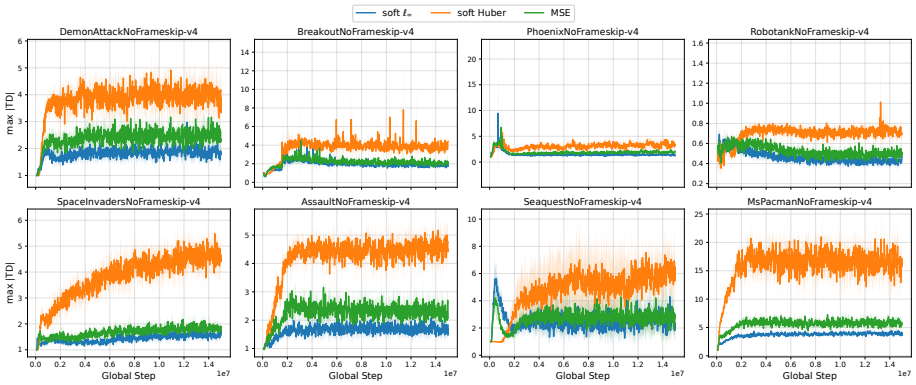

Figure 2: Max absolute Bellman error measured on a fixed held-out evaluation set. *x-axis*: global training steps. *y-axis*: max Bellman error $\|Q_{\theta_t} - Y_t\|_\infty$. Solid lines show the mean over 5 seeds; shaded areas denote standard deviation.

MSE is linear in $|\varepsilon|$ and balances average and worst-case reduction; soft Huber is sub-linear in the tail and down-weights exactly the residuals that drive R_∞. The MsPacman exception, where soft Huber wins despite a larger residual, is consistent with the action-gap phenomenon [11]: once the residual falls below the local action gap of the game, the greedy action is preserved and further reduction no longer changes the induced policy. Residual reduction thus matters up to a game-dependent threshold, beyond which different surrogates yield equivalent greedy decisions. The general connection to control performance follows from the classical bound for greedy policies [41]: for any Q and π greedy w.r.t. Q, V^π satisfies:$\|V^\star - V^\pi\|_\infty \leq \frac{2}{(1-\gamma)^2}\|TQ - Q\|_\infty$, so faster reduction of R_∞ yields a uniformly tighter suboptimality bound, correlating with better return. Additional ablations in Appendix F test whether the observed ordering is robust to optimization budget, model capacity, and architecture. Increasing the inner-loop budget K consistently improves convergence and final return, matching the role of K in achieving a smaller effective α. For capacity, following the width-scaling protocol of Schwarzer et al. [35], we scale the DQN network by $\tau \in \{0.5, 1, 2\}$; larger models typically improve returns once the inner loop is sufficiently optimized, consistent with capacity reducing the IBE rather than replacing the need for surrogate progress. Finally, Dueling-DQN [40] experiments show similar qualitative ordering, suggesting that the geometry-alignment effect is not specific to the vanilla DQN.

6 Conclusion and Limitations

We presented a surrogate-optimization view of fixed-target DQN that turns finite inner-loop progress into explicit ℓ_∞ contraction guarantees. The resulting α-thresholds separate approximation error, captured by ε_∞, from finite surrogate optimization, captured by α, and show that the soft-ℓ_∞ loss has the least stringent admissible accuracy requirement among the losses considered. Empirically, this geometry alignment consistently reduces max Bellman residuals and yields competitive or improved Atari returns. Our guarantees are conditional on the inner loop achieving the α-decrease; certifying this for finite-step AdamW on deep networks is an open problem requiring strong structural assumptions (e.g., NTK), which we view as complementary to our prediction-level analysis. The step-complexity result (Proposition 3.6) is tabular, and the uniform ℓ_∞ guarantee requires full coverage or its Bellman-closed analogue (Appendix C.5). Extending this framework to policy iteration and actor-critic, and combining soft-ℓ_∞ with PER, distributional RL, or larger-scale agents, are natural directions for future work.

Acknowledgments and Disclosure of Funding

This research was enabled in part by compute resources, software and technical help provided by Mila (mila.quebec) as well as by support provided by Calcul Québec (Calcul Québec) and the Digital Research Alliance of Canada (alliancecan.ca). G. Gidel and I. Mitliagkas are Canada CIFAR AI Chairs and are supported by Discovery Grants from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1): 89–129, 2008.

[2] Balles, L., Pedregosa, F., and Roux, N. L. The geometry of sign gradient descent. *arXiv preprint arXiv:2002.08056*, 2020.

[3] Beck, A. *First-order methods in optimization*. SIAM, 2017.

[4] Bellman, R. Dynamic programming, princeton univ. *Press Princeton, New Jersey*, 39, 1957.

[5] Bertsekas, D. and Tsitsiklis, J. N. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[6] Bertsekas, D. P. *Dynamic programming: deterministic and stochastic models*. Prentice-Hall, Inc., 1987.

[7] Cai, Q., Yang, Z., Lee, J. D., and Wang, Z. Neural temporal-difference learning converges to global optima. *Advances in Neural Information Processing Systems*, 32, 2019.

[8] Chen, J. and Jiang, N. Information-theoretic considerations in batch reinforcement learning. In *International conference on machine learning*, pp. 1042–1051. PMLR, 2019.

[9] D'Orazio, R., Vucetic, D., Liu, Z., Kim, J. L., Mitliagkas, I., and Gidel, G. Solving hidden monotone variational inequalities with surrogate losses. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4ZX2a3OKEV.

[10] Fan, J., Wang, Z., Xie, Y., and Yang, Z. A theoretical analysis of deep q-learning. In *Learning for dynamics and control*, pp. 486–489. PMLR, 2020.

[11] Farahmand, A.-m. Action-gap phenomenon in reinforcement learning. *Advances in neural information processing systems*, 24, 2011.

[12] Farahmand, A.-m., Szepesvári, C., and Munos, R. Error propagation for approximate policy and value iteration. *Advances in neural information processing systems*, 23, 2010.

[13] Fu, P., Rybkin, O., Zhou, Z., Nauman, M., Abbeel, P., Levine, S., and Kumar, A. Compute-optimal scaling for value-based deep rl. *arXiv preprint arXiv:2508.14881*, 2025.

[14] Hasselt, H. v., Guez, A., and Silver, D. Deep reinforcement learning with double q-learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 2094–2100, 2016.

[15] Huang, S., Dossa, R. F. J., Ye, C., Braga, J., Chakraborty, D., Mehta, K., and AraÃšjo, J. G. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022.

[16] Huber, P. J. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[17] Huber, P. J. Robust statistics. In *International encyclopedia of statistical science*, pp. 1248–1251. Springer, 2011.

[18] Karimi, H., Nutini, J., and Schmidt, M. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 795–811. Springer, 2016.

[19] Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[20] Mairal, J. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855, 2015.

[21] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[22] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[23] Munos, R. and Szepesvári, C. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(5), 2008.

[24] Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical programming*, 103 (1):127–152, 2005.

[25] Neu, G. and Okolo, N. Efficient global planning in large mdps via stochastic primal-dual optimization. In *International Conference on Algorithmic Learning Theory*, pp. 1101–1123. PMLR, 2023.

[26] Orvieto, A. and Gower, R. M. In search of adam's secret sauce. *arXiv preprint arXiv:2505.21829*, 2025.

[27] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[28] Patterson, A., Liao, V., and White, M. Robust losses for learning value functions. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):6157–6167, 2022.

[29] Pei, Z., Huang, Q., and Wang, S. Neuronal group communication for efficient neural representation. *arXiv preprint arXiv:2510.16851*, 2025.

[30] Pei, Z., Huang, Q., and Wang, S. When LLMs develop languages: Symbolic communication for efficient multi-agent reasoning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

[31] Riedmiller, M. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European conference on machine learning*, pp. 317–328. Springer, 2005.

[32] Rybkin, O., Nauman, M., Fu, P., Snell, C., Abbeel, P., Levine, S., and Kumar, A. Value-based deep rl scales predictably. *arXiv preprint arXiv:2502.04327*, 2025.

[33] Saleh, R. A. and Saleh, A. Statistical properties of the log-cosh loss function used in machine learning. *arXiv preprint arXiv:2208.04564*, 2022.

[34] Schaul, T., Quan, J., Antonoglou, I., and Silver, D. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

[35] Schwarzer, M., Ceron, J. S. O., Courville, A., Bellemare, M. G., Agarwal, R., and Castro, P. S. Bigger, better, faster: Human-level atari with human-level efficiency. In *International Conference on Machine Learning*, pp. 30365–30380. PMLR, 2023.

[36] Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 1998.

[37] Szepesvári, C. *Algorithms for reinforcement learning*. Springer nature, 2022.

[38] Towers, M., Kwiatkowski, A., Terry, J., Balis, J. U., De Cola, G., Deleu, T., Goulão, M., Kallinteris, A., Krimmel, M., KG, A., et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

[39] Van Hasselt, H., Doron, Y., Strub, F., Hessel, M., Sonnerat, N., and Modayil, J. Deep reinforcement learning and the deadly triad. *arXiv preprint arXiv:1812.02648*, 2018.

[40] Wang, Z., Schaul, T., Hessel, M., Hasselt, H., Lanctot, M., and Freitas, N. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pp. 1995–2003. PMLR, 2016.

[41] Williams, R. J. and Baird, L. C. Tight performance bounds on greedy policies based on imperfect value functions. In *In Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems*, 1993.

[42] Xu, P. and Gu, Q. A finite-time analysis of q-learning with neural network function approximation. In *International conference on machine learning*, pp. 10555–10565. PMLR, 2020.

[43] Xu, X., Li, X., Chen, N., Zhao, D., and Chen, C. Autonomous obstacle avoidance with improved deep reinforcement learning based on dynamic huber loss. *Applied Sciences (2076-3417)*, 15 (5), 2025.

A Definitions

Definition A.1 (Lipschitzness, Smoothness)**.** We say that a (differentiable and convex) function $f : \mathcal{X} \to \mathbb{R}$ is respectively L-Lipschitz, L-smooth if for any $x, x' \in \mathcal{X}$ it satisfies,

$$|f(x) - f(x')| \leq L \|x - x'\|, \tag{8}$$

$$\|\nabla f(x) - \nabla f(x')\|_* \leq L \|x - x'\|, \tag{9}$$

Definition A.2 (Polyak–Łojasiewicz (PL) condition [18])**.** Let $f : \mathcal{X} \to \mathbb{R}$ be differentiable on $\mathcal{X}$, and write $f_\star := \inf_{x \in \mathcal{X}} f(x)$. We say that f satisfies the PL condition with parameter $\mu > 0$ if, for all $x \in \mathcal{X}$,

$$\frac{1}{2} \|\nabla f(x)\|_*^2 \;\geq\; \mu\big(f(x) - f_\star\big).$$

B Connection between Huber and Soft Huber (log cosh)

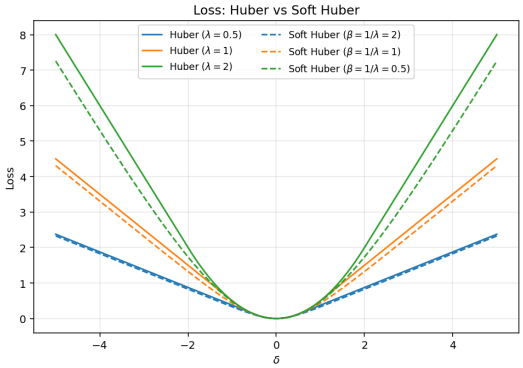

Figure 3: Huber vs. soft Huber (log–cosh) losses for matched pairs $(\lambda, \beta = 1/\lambda)$.

Definitions. For a residual $u \in \mathbb{R}$ and threshold $\lambda > 0$, the (symmetric) Huber loss is

$$\mathcal{L}_{\text{Huber}}(u; \lambda) = \begin{cases} \frac{1}{2} u^2, & |u| \leq \lambda, \\ \lambda\big(|u| - \frac{1}{2}\lambda\big), & |u| > \lambda. \end{cases} \tag{10}$$

Its gradient is $\nabla \mathcal{L}_{\text{Huber}}(u; \lambda) = u$ for $|u| \leq \lambda$ and $\lambda \operatorname{sign}(u)$ otherwise.

The *soft Huber* (log cosh) loss we use is the **scaled** form

$$\mathcal{L}_{\text{LC}}(u; \lambda) \;=\; \lambda^2 \log\cosh\Big(\frac{u}{\lambda}\Big) \;\equiv\; \frac{1}{\beta^2} \log\cosh(\beta\, u), \qquad \beta = \frac{1}{\lambda}. \tag{11}$$

Then $\nabla \mathcal{L}_{\text{LC}}(u; \lambda) = \lambda \tanh\big(u/\lambda\big)$ and $\nabla^2 \mathcal{L}_{\text{LC}}(u; \lambda) = \operatorname{sech}^2\big(u/\lambda\big) \in (0, 1]$.

Mapping between λ and β. Eq. equation 11 shows the one-to-one correspondence

$$\boxed{\beta = \tfrac{1}{\lambda}\,, \qquad \mathcal{L}_{\text{LC}}(u; \lambda) = \lambda^2 \log\cosh(u/\lambda) = \tfrac{1}{\beta^2} \log\cosh(\beta u)}$$

$\mathcal{L}_{\text{LC}}$ coincides with $\mathcal{L}_{\text{Huber}}$ up to a additive constant in the tails (the transition at $|u| = \lambda$ is smoothed).

Both losses interpolate between ℓ_2 (near zero) and ℓ_1 (for residuals greater than threshold), hence maintain bounded influence. Compared to L_{Huber}, $\mathcal{L}_{\text{LC}}$ is C^∞ (no kink at $|u| = \lambda$), has 1-Lipschitz gradient in the normalized coordinate.

Figure 3 compares the *loss* curves of Huber and soft Huber for three matched pairs $(\lambda,\ \beta = 1/\lambda)$, and Fig. 4 shows the corresponding *gradients*. The plots illustrate the quadratic core, linear tails with identical slope λ, and the smooth saturation of $\tanh(\cdot)$.

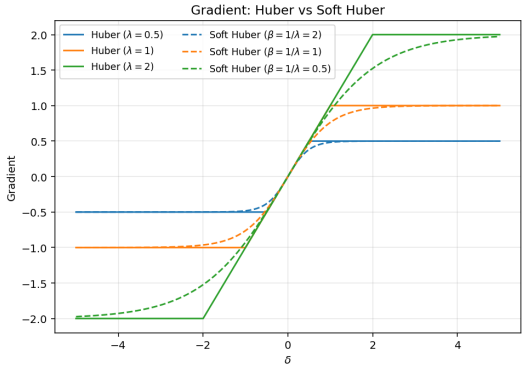

Figure 4: Gradients for Huber vs. soft Huber. Soft Huber replaces the hard plateau by the smooth $\tanh$ saturation.

C Proof of Theorem 3.3 and Remark 3.4

C.1 Setup and outer-loop recursion

We work with $Q_t \equiv Q_{\theta_t}$, the Bellman operator T, and the outer target

$$Z_t \;=\; (1-\eta)\, Q_t + \eta\, TQ_t, \qquad \eta \in (0,1].$$

Let $E_t := \|Q_t - Q^\star\|_\infty$ and let w be a probability vector over (s,a) with $w_{\min} := \min_{s,a} w(s,a) > 0$. The inner loop accepts θ_{t+1} when

$$\mathcal{L}_{\mathtt{sur}}\big(Q_{t+1} - Z_t, \beta\big) - \mathcal{L}^\star_{\mathtt{sur},t} \;<\; \alpha_{\mathtt{sur}}\big(\,\mathcal{L}_{\mathtt{sur}}\big(Q_t - Z_t, \beta\big) - \mathcal{L}^\star_{\mathtt{sur},t}\big).$$

for a surrogate loss ℓ_t defined below in each case. We will repeatedly use:

$$\|Q_t - TQ_t\|_\infty \;\le\; \|Q_t - Q^\star\|_\infty + \|TQ_t - TQ^\star\|_\infty \;\le\; (1+\gamma)\, E_t,$$

and the standard outer recursion (triangle inequality with $\|Q_{t+1} - Q^\star\|_\infty \le \|Q_{t+1} - Z_t\|_\infty + \|Z_t - Q^\star\|_\infty$ and $\|Z_t - Q^\star\|_\infty \le \|(1-\eta)Q_t + \eta TQ_t - Q^\star\| \le (1-\eta)E_t + \eta\gamma E_t \le (1-\eta(1-\gamma))E_t$):

$$E_{t+1} \;\le\; \underbrace{(1-\eta(1-\gamma))\, E_t}_{\text{error contraction}} \;+\; \underbrace{\|Q_{t+1} - Z_t\|_\infty}_{\text{inexactness}}. \tag{12}$$

Assumption 3.2 guarantees that the distribution w places nonzero weight on every state–action pair used by an optimal policy. As a result, any decrease in a weighted surrogate objective necessarily translates into a decrease of the largest Bellman error on the optimal actions. Consequently, once the inner loop achieves the stated α-reduction, the Bellman optimality update inherits a strict decrease in ℓ_∞ norm, and the outer iteration contracts and converges.

C.2 Surrogate-to-ℓ_∞ comparison lemmas

Lemma C.1 (Weighted ℓ_2 vs. ℓ_∞)**.** *For any x,*

$$\sqrt{w_{\min}}\, \|x\|_\infty \;\le\; \|x\|_w \;\le\; \|x\|_\infty.$$

Proof. *(Upper) Since $|x_i| \le \|x\|_\infty$ and $\sum w = 1$, $\|x\|_w^2 = \sum w\, x^2 \le \sum w\, \|x\|_\infty^2 = \|x\|_\infty^2$. (Lower) Let x_i achieve $\|x\|_\infty$. Then $\|x\|_w^2 = \sum w\, x^2 \ge w_i\, \|x\|_\infty^2 \ge w_{\min}\, \|x\|_\infty^2$. Taking square roots gives the claim.* □

Lemma C.2 (Soft-Huber vs. ℓ_∞)**.** *For any vector x,*

$$\frac{w_{\min}}{\beta}\, \|x\|_\infty - \frac{w_{\min}}{\beta^2} \log 2 \;\le\; \frac{1}{\beta^2} \sum_{s,a} w(s,a)\; \mathrm{logcosh}\big(\beta x(s,a)\big) \;\le\; \frac{1}{\beta}\, \|x\|_\infty.$$

Proof. *(Upper)* $\log\cosh(u) \le |u|$ *pointwise* $\Rightarrow \frac{1}{\beta^2}\sum_{s,a} w(s,a)\, \log\cosh(\beta x(s,a)) \le \frac{1}{\beta}\sum_{s,a} w(s,a)\, |x(s,a)| \le \frac{1}{\beta}\|x\|_\infty$ *since* $\sum w = 1$. *(Lower)* $\sum_{s,a} w(s,a)\, \log\cosh(\beta x(s,a)) \ge w_{\min} \log\cosh(\beta \|x\|_\infty)$ *and* $\log\cosh(z) \ge z - \log 2$ *for* $z \ge 0$ *yields* $\frac{1}{\beta^2}\sum_{s,a} w(s,a)\, \log\cosh(\beta x(s,a)) \ge \frac{w_{\min}}{\beta}\|x\|_\infty - \frac{w_{\min}}{\beta^2}\log 2$. □

Lemma C.3 (Soft-ℓ_∞ vs. ℓ_∞)**.** *For any vector x,*

$$\|x\|_\infty + \frac{1}{\beta}\log\frac{w_{\min}}{2} \;\le\; \frac{1}{\beta}\log\sum_{s,a} w(s,a)\cosh(\beta x(s,a)) \;\le\; \|x\|_\infty.$$

Proof. *(Upper)* $\cosh(\beta x(s,a)) \le e^{\beta\|x\|_\infty} \Rightarrow \sum_{s,a} w(s,a)\,\cosh(\beta x(s,a)) \le e^{\beta\|x\|_\infty}$*, hence the RHS. (Lower)* $\sum_{s,a} w(s,a)\,\cosh(\beta x(s,a)) \ge w_{\min}\cosh(\beta\|x\|_\infty) \ge w_{\min}e^{\beta\|x\|_\infty}/2$*, so the LHS follows by taking* $(1/\beta)\log(\cdot)$. □

C.3 Proof of Theorem 3.3

Recall the notation $E_t := \|Q_t - Q^\star\|_\infty$, the error recursion 12 gives, for all t,

$$E_{t+1} \;\le\; \gamma E_t + \|Q_{t+1} - TQ_t\|_\infty. \tag{13}$$

We also repeatedly use

$$\|Q_t - TQ_t\|_\infty \;\le\; (1+\gamma)\,E_t, \tag{14}$$

Since each $Q_t \in Z$, the intrinsic approximation radius $d_\infty := \inf_{Q\in Z}\|Q - Q^\star\|_\infty$ yields $d_\infty \le \liminf_{t\to\infty} E_t$, so it remains to upper bound $\limsup E_t$ for each surrogate. Define $Q_t^\infty := \arg\min_{Q\in Z}\|Q - TQ_t\|_\infty$, then by assumption 3.1 we have that:

$$\|Q_t^\infty - TQ_t\|_\infty \le \varepsilon_\infty, \quad \forall t$$

MSE surrogate. At iteration t, the MSE surrogate is

$$\mathcal{L}_{\mathrm{MSE},t}(Q) := \frac{1}{2}\sum_x w(x)\big(Q(x) - TQ_t(x)\big)^2.$$

Lemma B.1 gives, for any Q,

$$\frac{w_{\min}}{2}\,\|Q - TQ_t\|_\infty^2 \;\le\; \mathcal{L}_{\mathrm{MSE},t}(Q) \;\le\; \frac{1}{2}\,\|Q - TQ_t\|_\infty^2. \tag{15}$$

Let $Q_t^\star$ be a minimizer of $\mathcal{L}_{\mathrm{MSE},t}$ over Z. Then

$$\mathcal{L}_{\mathrm{MSE},t}(Q_t^\star) \;\le\; \mathcal{L}_{\mathrm{MSE},t}(Q_t^\infty) \;\le\; \tfrac{1}{2}\|Q_t^\infty - TQ_t\|_\infty^2 \;\le\; \tfrac{1}{2}\varepsilon_\infty^2.$$

The α-condition equation 6, together with equation 15, implies

$$\begin{aligned}
\frac{w_{\min}}{2}\,\|Q_{t+1} - TQ_t\|_\infty^2 &\le \mathcal{L}_{\mathrm{MSE},t}(Q_{t+1}) \\
&\le \alpha\,\mathcal{L}_{\mathrm{MSE},t}(Q_t) + (1-\alpha)\,\mathcal{L}_{\mathrm{MSE},t}(Q_t^\star) \\
&\le \frac{1}{2}\Big[\alpha\|Q_t - TQ_t\|_\infty^2 + (1-\alpha)\varepsilon_\infty^2\Big],
\end{aligned}$$

so

$$\|Q_{t+1} - TQ_t\|_\infty^2 \;\le\; \frac{\alpha}{w_{\min}}\|Q_t - TQ_t\|_\infty^2 + \frac{1-\alpha}{w_{\min}}\varepsilon_\infty^2.$$

Taking square roots and using $\sqrt{u+v} \le \sqrt{u} + \sqrt{v}$,

$$\|Q_{t+1} - TQ_t\|_\infty \;\le\; \sqrt{\frac{\alpha}{w_{\min}}}\,\|Q_t - TQ_t\|_\infty + \sqrt{\frac{1-\alpha}{w_{\min}}}\,\varepsilon_\infty.$$

Applying equation 14, $\|Q_t - TQ_t\|_\infty \le (1+\gamma)E_t$, we obtain

$$\|Q_{t+1} - TQ_t\|_\infty \;\le\; (1+\gamma)\sqrt{\frac{\alpha}{w_{\min}}}\,E_t + \sqrt{\frac{1-\alpha}{w_{\min}}}\,\varepsilon_\infty.$$

Substituting this into equation 13 yields

$$E_{t+1} \;\le\; \Big[\gamma + (1+\gamma)\sqrt{\tfrac{\alpha}{w_{\min}}}\Big]E_t + \sqrt{\tfrac{1-\alpha}{w_{\min}}}\,\varepsilon_\infty.$$

When $\alpha_{\mathrm{MSE}} < w_{\min}\left(\frac{1-\gamma}{1+\gamma}\right)^2$, i.e., $\gamma + (1+\gamma)\sqrt{\alpha/w_{\min}} < 1$, the linear recursion gives

$$\limsup_{t\to\infty} E_t \;\le\; \frac{\sqrt{\frac{1-\alpha}{w_{\min}}}\,\varepsilon_\infty}{1 - \gamma - (1+\gamma)\sqrt{\frac{\alpha}{w_{\min}}}}.$$

Soft Huber surrogate (LC)**.** Following Lemma C.2 we have that, for all Q,

$$\frac{w_{\min}}{\beta}\,\|Q - TQ_t\|_\infty - \frac{w_{\min}}{\beta^2}\log 2 \;\le\; \mathcal{L}_{\mathrm{LC},t}(Q) \;\le\; \frac{1}{\beta}\,\|Q - TQ_t\|_\infty. \tag{16}$$

Let $Q_t^\star$ minimize $\mathcal{L}_{\mathrm{LC},t}$ over Z, then

$$\mathcal{L}_{\mathrm{LC},t}(Q_t^\star) \;\le\; \mathcal{L}_{\mathrm{LC},t}(Q_t^\infty) \;\le\; \frac{1}{\beta}\,\varepsilon_\infty.$$

Using the α-condition equation 6 and equation 16,

$$\begin{aligned}\frac{w_{\min}}{\beta}\,\|Q_{t+1} - TQ_t\|_\infty - \frac{w_{\min}}{\beta^2}\log 2 &\le \mathcal{L}_{\mathrm{LC},t}(Q_{t+1}) \\ &\le \alpha\,\mathcal{L}_{\mathrm{LC},t}(Q_t) + (1-\alpha)\,\mathcal{L}_{\mathrm{LC},t}(Q_t^\star) \\ &\le \alpha\cdot\frac{1}{\beta}\,\|Q_t - TQ_t\|_\infty + (1-\alpha)\cdot\frac{1}{\beta}\,\varepsilon_\infty,\end{aligned}$$

so after multiplying by β,

$$w_{\min}\|Q_{t+1} - TQ_t\|_\infty \;\le\; \alpha\,\|Q_t - TQ_t\|_\infty + (1-\alpha)\,\varepsilon_\infty + \frac{w_{\min}}{\beta}\log 2.$$

Using equation 14,

$$w_{\min}\|Q_{t+1} - TQ_t\|_\infty \;\le\; \alpha(1+\gamma)E_t + (1-\alpha)\,\varepsilon_\infty + \frac{w_{\min}}{\beta}\log 2.$$

Substituting into equation 13,

$$E_{t+1} \;\le\; \big[\gamma + \frac{\alpha}{w_{\min}}(1+\gamma)\big]E_t + \frac{(1-\alpha)}{w_{\min}}\,\varepsilon_\infty + \frac{\log 2}{\beta}.$$

If $\alpha < w_{\min}\frac{1-\gamma}{1+\gamma}$, i.e.,the sequence E_t contracts as $t \to \infty$, we have

$$\limsup_{t\to\infty} E_t \;\le\; \frac{\frac{1-\alpha}{w_{\min}}\,\varepsilon_\infty + \frac{1}{\beta}\log 2}{1 - \gamma - \frac{\alpha}{w_{\min}}(1+\gamma)}.$$

For $\alpha = 0$, we will have

$$\limsup_{t\to\infty} E_t \;\le\; \frac{\frac{1}{w_{\min}}\,\varepsilon_\infty + \frac{1}{\beta}\log 2}{1-\gamma},$$

Soft ℓ_∞ surrogate (LSC)**.** For the soft-ℓ_∞ surrogate Lemma C.3 provides a constant $C_\beta = \frac{1}{\beta}\log\frac{2}{w_{\min}}$ such that, for all Q,

$$\|Q - TQ_t\|_\infty - C_\beta \;\le\; \mathcal{L}_{\mathrm{LSC},t}(Q) \;\le\; \|Q - TQ_t\|_\infty. \tag{17}$$

Let $Q_t^\star$ minimize $\mathcal{L}_{\mathrm{LSC},t}$ over Z, and let $Q_t^\infty \in Z$ satisfy $\|Q_t^\infty - TQ_t\|_\infty \le \varepsilon_\infty$ (approximate Bellman completeness). Then

$$\mathcal{L}_{\mathrm{LSC},t}(Q_t^\star) \;\le\; \mathcal{L}_{\mathrm{LSC},t}(Q_t^\infty) \;\le\; \varepsilon_\infty.$$

Using the α-condition equation 6 and equation 17,

$$\begin{aligned}\|Q_{t+1} - TQ_t\|_\infty - C_\beta &\le \mathcal{L}_{\mathrm{LSC},t}(Q_{t+1}) \\ &\le \alpha\,\mathcal{L}_{\mathrm{LSC},t}(Q_t) + (1-\alpha)\,\mathcal{L}_{\mathrm{LSC},t}(Q_t^\star) \\ &\le \alpha\,\|Q_t - TQ_t\|_\infty + (1-\alpha)\varepsilon_\infty,\end{aligned}$$

hence, after multiplying by β,

$$\|Q_{t+1} - TQ_t\|_\infty \;\le\; \alpha\,\|Q_t - TQ_t\|_\infty + (1-\alpha)\,\varepsilon_\infty + C_\beta. \tag{18}$$

Using equation 14, $\|Q_t - TQ_t\|_\infty \le (1+\gamma)E_t$ with $E_t := \|Q_t - Q^\star\|_\infty$, we obtain

$$\|Q_{t+1} - TQ_t\|_\infty \;\le\; \alpha(1+\gamma)E_t + (1-\alpha)\,\varepsilon_\infty + C_\beta. \tag{19}$$

Plugging equation 19 into the error–propagation inequality equation 13,

$$E_{t+1} \;\le\; \gamma E_t + \|Q_{t+1} - TQ_t\|_\infty,$$

gives

$$E_{t+1} \;\le\; \big[\gamma + \alpha(1+\gamma)\big] E_t + (1-\alpha)\,\varepsilon_\infty + C_\beta. \tag{20}$$

If

$$\gamma + \alpha(1+\gamma) < 1 \qquad \Longleftrightarrow \qquad \alpha < \frac{1-\gamma}{1+\gamma},$$

the linear recursion equation 20 yields

$$\limsup_{t\to\infty} E_t \;\le\; \frac{(1-\alpha)\,\varepsilon_\infty + C_\beta}{1-\gamma-\alpha(1+\gamma)}.$$

For $\alpha = 0$, equation 20 reduces to $E_{t+1} \le \gamma E_t + \varepsilon_\infty + \beta C_\beta$, so

$$\limsup_{t\to\infty} E_t \;\le\; \frac{\varepsilon_\infty + C_\beta}{1-\gamma}.$$

(up to absorbing β into C_β as in the main text).

This completes the proof of Theorem 3.3.

C.4 Proof for Remark 3.4

Lemma C.4 (Soft Huber vs. ℓ_∞)**.** *For any vector x,*

$$\frac{w_{\min}}{\beta^2} \log\cosh \beta\|x\|_\infty \;\le\; \frac{1}{\beta^2} \sum_{s,a} w(s,a)\; \mathrm{logcosh}\big(\beta x(s,a)\big) \;\le\; \frac{1}{\beta^2}\; \log\cosh \beta\|x\|_\infty.$$

Proof. $\frac{1}{\beta^2}\sum_{s,a} w(s,a)\; \mathrm{logcosh}\big(\beta x(s,a)\big) \;\le\; \frac{1}{\beta^2}\sum_{s,a} w(s,a)\; \log\cosh(\beta\|x\|_\infty) \;=\; \frac{1}{\beta^2}\log\cosh(\beta\|x\|_\infty)$ *(Lower)* $\sum_{s,a} w(s,a)\; \log\cosh(\beta x(s,a)) \ge w_{\min}\log\cosh(\beta\|x\|_\infty)$ □

Then we have, by Lemma C.4,

$$\frac{w_{\min}}{\beta^2} \log\cosh\big(\beta\|Q_{t+1} - Z_t\|_\infty\big) \;\le\; \mathcal{L}_{\mathrm{LC},t}(\theta_{t+1})$$

and

$$\mathcal{L}_{\mathrm{LC},t}(\theta_t) \;\le\; \frac{1}{\beta^2} \log\cosh\big(\beta\|Q_t - Z_t\|_\infty\big).$$

Now apply the α-condition (Eq. 6) including the approximation term:

$$\mathcal{L}_{\mathrm{LC},t}(\theta_{t+1}) \;\le\; \alpha_{\mathrm{LC}}\, \mathcal{L}_{\mathrm{LC},t}(\theta_t) + (1-\alpha_{\mathrm{LC}})\, \mathcal{L}^\star_{\mathrm{LC},t}. \tag{21}$$

To bound $\mathcal{L}^\star_{\mathrm{LC},t}$ using ε_∞, by Definition 3.1 there exists $\tilde{Q}_t \in \mathcal{Z}$ such that $\|\tilde{Q}_t - Z_t\|_\infty \le \varepsilon_\infty$, hence

$$\mathcal{L}^\star_{\mathrm{LC},t} = \min_{Q\in\mathcal{Z}} \mathcal{L}_{\mathrm{LC}}(Q - Z_t) \le \mathcal{L}_{\mathrm{LC}}(\tilde{Q}_t - Z_t) \le \frac{1}{\beta^2}\log\cosh(\beta\varepsilon_\infty),$$

where the last inequality uses Lemma C.4 (upper bound).

Combining the above displays yields

$$\frac{w_{\min}}{\beta^2} \log\cosh\big(\beta\|Q_{t+1} - Z_t\|_\infty\big) \;\le\; \frac{\alpha_{\mathrm{LC}}}{\beta^2} \log\cosh\big(\beta\|Q_t - Z_t\|_\infty\big) + \frac{1-\alpha_{\mathrm{LC}}}{\beta^2}\log\cosh(\beta\varepsilon_\infty). \tag{22}$$

Now let $r_t := \|Q_t - Z_t\|_\infty$ and define $u_t := \log\cosh(\beta r_t)$. Then equation 22 becomes

$$u_{t+1} \le \frac{\alpha_{\mathrm{LC}}}{w_{\min}} u_t + \frac{1-\alpha_{\mathrm{LC}}}{w_{\min}}\; \log\cosh(\beta\varepsilon_\infty).$$

Assuming $\alpha_{\mathrm{LC}} < w_{\min}$, this is a stable affine recursion, so (u_t) is bounded and

$\limsup_{t\to\infty} u_t \;\le\; C_{\alpha,w}\; \log\cosh(\beta\varepsilon_\infty)$ for some constant $C_{\alpha,w} < \infty$ depending only on $\alpha_{\mathrm{LC}}, w_{\min}$.

Since $\log\cosh$ is increasing on $[0,\infty)$, this implies

$$\limsup_{t\to\infty} r_t \;\le\; (\log\cosh)^{-1}\Big(C_{\alpha,w}\,\log\cosh(\beta\varepsilon_\infty)\Big)/\beta.$$

Finally, because $\log\cosh(0)=0$ and $(\log\cosh)^{-1}$ is continuous at 0, the right-hand side goes to 0 as $\varepsilon_\infty\to 0$. In particular, the steady-state residual scales with ε_∞ and vanishes under Bellman completeness.

Further more we can show in this case we have the same α-condition as in MSE: ignoring the approximation term in Eq. 6 (which we already see only contributes an additive steady-state floor handled separately), the multiplicative part of the α-condition gives:

$$\frac{w_{\min}}{\beta^2}\log\cosh\beta\|Q_{t+1}-Z_t\|_\infty \le \mathcal{L}_{\mathrm{LC},t}(\theta_{t+1}) \le \alpha_{\mathrm{LC}}\,\mathcal{L}_{\mathrm{LC},t}(\theta_t) \le \frac{\alpha_{\mathrm{LC}}}{\beta^2}\log\cosh\beta\|Q_t-Z_t\|_\infty \tag{23}$$

Now, using the fact that for $a>1$, and $x>0$ we have $a\log\cosh(x)\ge\log\cosh(\sqrt{a}x)$, we have that for $\alpha_{\mathrm{LC}}\le w_{\min}$,

$$\log\cosh\sqrt{\frac{w_{\min}\beta^2}{\alpha_{\mathrm{LC}}}}\|Q_{t+1}-Z_t\|_\infty \le \frac{w_{\min}}{\alpha_{\mathrm{LC}}}\log\cosh\beta\|Q_{t+1}-Z_t\|_\infty \le \log\cosh\beta\|Q_t-Z_t\|_\infty \tag{24}$$

and then because the function $\log\cosh^{-1}$ is increasing we have $\sqrt{\frac{w_{\min}}{\alpha_{\mathrm{LC}}}}\|Q_{t+1}-Z_t\|_\infty \le \|Q_t-Z_t\|_\infty$ which leads to

$$\|Q_{t+1}-Z_t\|_\infty \le \sqrt{\frac{\alpha_{\mathrm{LC}}}{w_{\min}}}\|Q_t-Z_t\|_\infty \le \sqrt{\frac{\alpha_{\mathrm{LC}}(1+\gamma)^2}{w_{\min}}}E_t \tag{25}$$

which leads to

$$E_{t+1} \;\le\; \Big[\,1-\eta(1-\gamma)\;+\;\eta(1+\gamma)\sqrt{\frac{\alpha_{\mathrm{LC}}}{w_{\min}}}\,\Big]\,E_t \tag{26}$$

Thus we get the slower but unbiased α condition: $\alpha_{\mathrm{LC}}\le w_{\min}(\frac{1-\gamma}{1+\gamma})^2$.

Lemma C.5 (Soft-ℓ_∞ vs. ℓ_∞)**.** *For any vector x,*

$$\frac{w_{\min}}{\beta}\log\cosh(\beta\|x\|_\infty) \le \frac{1}{\beta}\log\sum_{s,a} w(s,a)\cosh(\beta x(s,a)) \le \frac{1}{\beta}\log\cosh(\beta\|x\|_\infty)$$

Proof. *(Upper)* $\frac{1}{\beta}\log\sum_{s,a} w(s,a)\cosh(\beta x(s,a)) \le \frac{1}{\beta}\log\sum_{s,a} w(s,a)\cosh(\beta\|x\|_\infty) \le \frac{1}{\beta}\log\cosh(\beta\|x\|_\infty)$. *(Lower) By concavity of the* log, $\frac{1}{\beta}\log\sum_{s,a} w(s,a)\cosh(\beta x(s,a)) \ge \frac{1}{\beta}\sum_{s,a} w(s,a)\log\cosh(\beta x(s,a)) \ge \frac{w_{\min}}{\beta}\log\cosh(\beta\|x\|_\infty)$. □

Thus similarly as for the soft Huber we get

$$E_{t+1} \;\le\; \Big[\,1-\eta(1-\gamma)\;+\;(1+\gamma)\,\eta\,\sqrt{\frac{\alpha}{w_{\min}}}\,\Big]\,E_t \tag{27}$$

and the corresponding slower but unbiased α condition: $\alpha_{\mathrm{LSC}}\le w_{\min}(\frac{1-\gamma}{1+\gamma})^2$.

C.5 Coverage on Bellman-Closed Regions

Assumption 3.2 requires the sampling distribution to cover all of $\mathcal{S}\times\mathcal{A}$. The full coverage assumption can be relaxed to a restricted sup-norm. It is enough to choose a region $\mathcal{R}\subseteq\mathcal{S}\times\mathcal{A}$ that is closed under one-step Bellman optimality backup and require the sampling distribution to cover only this region:

$$\mathcal{R}\subseteq\mathrm{supp}(w).$$

One natural choice of such regions is the set of states reachable from an initial distribution ρ under the policy class of interest. If

$$\mathcal{S}_{\rho,\Pi} := \bigcup_{\pi\in\Pi}\mathrm{supp}(d_\rho^\pi),$$

then $R = \mathcal{S}_{\rho,\Pi} \times \mathcal{A}$ is a valid restricted region whenever it is closed under the one-step optimality backup. The product with $\mathcal{A}$ reflects the fact that the optimality operator evaluates all next actions through the maximization over $a' \in \mathcal{A}$.

Intuitively, $\mathcal{R}$ is a set of state-action pairs on which Bellman backups do not "look outside" the set: whenever the backup starts from $(s, a) \in \mathcal{R}$, every successor state s' that can be reached and every action $a' \in \mathcal{A}$ considered by the maximization also remain inside $\mathcal{R}$. Equivalently, in the directed graph with edges

$$(s, a) \longrightarrow (s', a') \quad \text{whenever} \quad P(s'|s, a) > 0,\ a' \in \mathcal{A},$$

the region $\mathcal{R}$ has no outgoing edges.

The support condition above implies the following coverage coefficient:

$$C_{\mathcal{R}}^{\sup} := \sup_{(s,a)\in\mathcal{R}} \frac{1}{w(s, a)} < \infty.$$

Thus the full-coverage requirement $\operatorname{supp}(w) = \mathcal{S} \times \mathcal{A}$ is replaced by coverage only on the Bellman-closed region where the guarantee is claimed.

The localized guarantee is then stated in the restricted sup norm

$$\|f\|_{\infty,\mathcal{R}} := \sup_{(s,a)\in\mathcal{R}} |f(s, a)|.$$

This localization replaces the two full-space ingredients in the proof of Theorem 3.3: the ordinary sup-norm Bellman contraction is replaced by a restricted contraction on Bellman-closed $\mathcal{R}$, and the full-space norm conversion using $1/\underline{w}$ is replaced by the localized conversion using $C_{\mathcal{R}}^{\sup}$.

Definition C.6 (Bellman-closed region)**.** A set $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{A}$ is *Bellman-closed* if, whenever $(s, a) \in \mathcal{R}$ and $P(s'|s, a) > 0$, we have $(s', a') \in \mathcal{R}$ for every $a' \in \mathcal{A}$.

The requirement includes every $a' \in \mathcal{A}$ because the standard Bellman optimality operator maximizes over all next actions:

$$(TQ)(s, a) = r(s, a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} \max_{a'\in\mathcal{A}} Q(s', a').$$

Exploration, such as an ϵ-greedy behavior policy with ϵ bounded away from zero, provides the corresponding action support in the sampling distribution: whenever a state is visited, all actions have positive sampling probability. Thus, transition dynamics provide the state closure while exploration provides the all-action closure required by the Bellman optimality backup.

Lemma C.7 (Restricted Bellman contraction)**.** *If $\mathcal{R}$ is Bellman-closed, then for all Q, Q',*

$$\|TQ - TQ'\|_{\infty,\mathcal{R}} \le \gamma \|Q - Q'\|_{\infty,\mathcal{R}}.$$

Proof. For any $(s, a) \in \mathcal{R}$,

$$|(TQ)(s, a) - (TQ')(s, a)| \le \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} \max_{a'\in\mathcal{A}} |Q(s', a') - Q'(s', a')|.$$

By Bellman-closedness, every pair (s', a') appearing in the maximization belongs to $\mathcal{R}$. Hence the right-hand side is at most $\gamma\|Q - Q'\|_{\infty,\mathcal{R}}$. Taking the supremum over $(s, a) \in \mathcal{R}$ completes the proof. □

For the localized theorem, surrogate losses are evaluated only on coordinates in $\mathcal{R}$. We use the original sampling weights restricted to $\mathcal{R}$:

$$L_{\mathrm{MSE},\mathcal{R}}(f) := \frac{1}{2} \sum_{(s,a)\in\mathcal{R}} w(s, a) f(s, a)^2,$$

$$L_{\mathrm{LC},\mathcal{R}}(f; \beta) := \frac{1}{\beta^2} \sum_{(s,a)\in\mathcal{R}} w(s, a) \log\cosh(\beta f(s, a)),$$

and

$$L_{\mathrm{LSC},\mathcal{R}}(f; \beta) := \frac{1}{\beta} \log \sum_{(s,a)\in\mathcal{R}} w(s, a) \cosh(\beta f(s, a)).$$

Lemma C.8 (Restricted surrogate comparisons)**.** *Suppose $C_{\mathcal{R}}^{\sup} < \infty$. Then for any f,*

$$\frac{1}{2C_{\mathcal{R}}^{\sup}}\|f\|_{\infty,\mathcal{R}}^2 \le L_{\mathrm{MSE},\mathcal{R}}(f) \le \frac{1}{2}\|f\|_{\infty,\mathcal{R}}^2,$$

$$\frac{1}{\beta C_{\mathcal{R}}^{\sup}}\|f\|_{\infty,\mathcal{R}} - \frac{\log 2}{\beta^2 C_{\mathcal{R}}^{\sup}} \le L_{\mathrm{LC},\mathcal{R}}(f;\beta) \le \frac{1}{\beta}\|f\|_{\infty,\mathcal{R}},$$

and

$$\|f\|_{\infty,\mathcal{R}} - \frac{1}{\beta}\log(2C_{\mathcal{R}}^{\sup}) \le L_{\mathrm{LSC},\mathcal{R}}(f;\beta) \le \|f\|_{\infty,\mathcal{R}}.$$

Proof. Let $(s^\star, a^\star) \in \mathcal{R}$ attain $\|f\|_{\infty,\mathcal{R}}$. Since $C_{\mathcal{R}}^{\sup} = \sup_{(s,a)\in\mathcal{R}} 1/w(s,a)$, we have $w(s^\star,a^\star) \ge 1/C_{\mathcal{R}}^{\sup}$. Therefore

$$L_{\mathrm{MSE},\mathcal{R}}(f) \ge \frac{1}{2}w(s^\star,a^\star)\|f\|_{\infty,\mathcal{R}}^2 \ge \frac{1}{2C_{\mathcal{R}}^{\sup}}\|f\|_{\infty,\mathcal{R}}^2.$$

The upper bound follows from $|f(s,a)| \le \|f\|_{\infty,\mathcal{R}}$ and $\sum_{(s,a)\in\mathcal{R}} w(s,a) \le 1$:

$$L_{\mathrm{MSE},\mathcal{R}}(f) \le \frac{1}{2}\|f\|_{\infty,\mathcal{R}}^2.$$

For the log-cosh loss, use the pointwise inequalities

$$\log\cosh z \le |z|, \qquad \log\cosh z \ge |z| - \log 2.$$

The upper bound follows from

$$L_{\mathrm{LC},\mathcal{R}}(f;\beta) \le \frac{1}{\beta^2}\sum_{(s,a)\in\mathcal{R}} w(s,a)\beta|f(s,a)| \le \frac{1}{\beta}\|f\|_{\infty,\mathcal{R}}.$$

The lower bound follows by applying the lower pointwise inequality at $(s^\star,a^\star)$:

$$\begin{aligned} L_{\mathrm{LC},\mathcal{R}}(f;\beta) &\ge \frac{1}{\beta^2}w(s^\star,a^\star)\big(\beta\|f\|_{\infty,\mathcal{R}} - \log 2\big) \\ &\ge \frac{1}{\beta C_{\mathcal{R}}^{\sup}}\|f\|_{\infty,\mathcal{R}} - \frac{\log 2}{\beta^2 C_{\mathcal{R}}^{\sup}}. \end{aligned}$$

For the log-sum-cosh loss, the upper bound follows from $\cosh(\beta f(s,a)) \le \exp(\beta\|f\|_{\infty,\mathcal{R}})$ and $\sum_{(s,a)\in\mathcal{R}} w(s,a) \le 1$:

$$\begin{aligned} L_{\mathrm{LSC},\mathcal{R}}(f;\beta) &\le \frac{1}{\beta}\log\left(\sum_{(s,a)\in\mathcal{R}} w(s,a)\exp(\beta\|f\|_{\infty,\mathcal{R}})\right) \\ &\le \|f\|_{\infty,\mathcal{R}}. \end{aligned}$$

For the lower bound, use $\cosh z \ge \frac{1}{2}\exp(|z|)$:

$$\begin{aligned} L_{\mathrm{LSC},\mathcal{R}}(f;\beta) &\ge \frac{1}{\beta}\log\left(w(s^\star,a^\star)\frac{1}{2}\exp(\beta\|f\|_{\infty,\mathcal{R}})\right) \\ &\ge \|f\|_{\infty,\mathcal{R}} - \frac{1}{\beta}\log(2C_{\mathcal{R}}^{\sup}). \end{aligned}$$

□

We now state the localized version of the main theorem. Let $e_t(Q)$ denote the residual used by the frozen-target surrogate subproblem at outer step t. The localized α-condition is imposed on the restriction of this residual to $\mathcal{R}$:

$$L_{\mathrm{sur},t,\mathcal{R}}(e_t(Q_{t+1})) - L_{\mathrm{sur},t,\mathcal{R}}^\star \le \alpha_{\mathrm{sur}}\left(L_{\mathrm{sur},t,\mathcal{R}}(e_t(Q_t)) - L_{\mathrm{sur},t,\mathcal{R}}^\star\right),$$

where $L_{\mathrm{sur},t,\mathcal{R}}^\star$ is the minimum localized surrogate loss attainable over Z. This is the exact analogue of the α-condition in the main text after restricting all residuals and losses to $\mathcal{R}$.

Corollary C.9 (Localized version of Theorem 3.3)**.** *Let $\mathcal{R}$ be Bellman-closed with $C_{\mathcal{R}}^{\sup} < \infty$, and define the localized inherent Bellman error*

$$\varepsilon_{\infty,\mathcal{R}} := \sup_{Q \in Z} \inf_{Q' \in Z} \|Q' - TQ\|_{\infty,\mathcal{R}}.$$

Then Theorem 3.3 holds with

$$\|\cdot\|_\infty \quad \textit{replaced by} \quad \|\cdot\|_{\infty,\mathcal{R}}, \qquad \frac{1}{\underline{w}} \quad \textit{replaced by} \quad C_{\mathcal{R}}^{\sup}, \qquad \varepsilon_\infty \quad \textit{replaced by} \quad \varepsilon_{\infty,\mathcal{R}}.$$

In particular, a sufficient admissible range is

$$\alpha_{\mathrm{MSE}} < \frac{1}{C_{\mathcal{R}}^{\sup}} \left(\frac{1-\gamma}{1+\gamma}\right)^2, \qquad \alpha_{\mathrm{LC}} < \frac{1}{C_{\mathcal{R}}^{\sup}} \frac{1-\gamma}{1+\gamma}, \qquad \alpha_{\mathrm{LSC}} < \frac{1-\gamma}{1+\gamma}.$$

Thus the admissible soft-ℓ_∞ threshold remains independent of the localized coverage coefficient.

Moreover, using the same choices of α as in Theorem 3.3, the localized iterates satisfy

$$\|Q_t - Q^\star\|_{\infty,\mathcal{R}} \le \left(\frac{1+\gamma}{2}\right)^t \|Q_0 - Q^\star\|_{\infty,\mathcal{R}} + R_{\mathrm{sur},\mathcal{R}},$$

where $R_{\mathrm{sur},\mathcal{R}}$ is obtained from the radius in Theorem 3.3 by replacing $1/\underline{w}$ with $C_{\mathcal{R}}^{\sup}$ and ε_∞ with $\varepsilon_{\infty,\mathcal{R}}$. Equivalently,

$$R_{\mathrm{MSE},\mathcal{R}} = \frac{2}{1-\gamma} \sqrt{C_{\mathcal{R}}^{\sup} - \frac{(1-\gamma)^2}{4(1+\gamma)^2}}\, \varepsilon_{\infty,\mathcal{R}},$$

$$R_{\mathrm{LC},\mathcal{R}} = \frac{2}{1-\gamma} \left[\left(C_{\mathcal{R}}^{\sup} - \frac{1-\gamma}{2(1+\gamma)} \right) \varepsilon_{\infty,\mathcal{R}} + \frac{\log 2}{\beta} \right],$$

and

$$R_{\mathrm{LSC},\mathcal{R}} = \frac{1+3\gamma}{(1-\gamma)^2} \varepsilon_{\infty,\mathcal{R}} + \frac{2}{\beta(1-\gamma)} \log(2C_{\mathcal{R}}^{\sup}).$$

Proof. The proof is identical to that of Theorem 3.3 after replacing the full-space ingredients by their localized analogues. Lemma C.7 replaces the ordinary sup-norm Bellman contraction, and Lemma C.8 replaces the full-space surrogate-to-sup-norm comparisons. The approximation term is measured by $\varepsilon_{\infty,\mathcal{R}}$ instead of ε_∞. All subsequent algebra carries over with the stated substitutions. □

Why ℓ_∞ and why coverage. A central limitation of weighted L_p control in value-based learning is that it does not directly certify the statewise action chosen by the greedy policy. The deployed policy is selected by the pointwise rule $\pi_Q(s) \in \arg\max_{a \in \mathcal{A}} Q(s,a)$, and whether this action is correct depends on the local action gap

$$\Delta(s) := Q^\star(s,a^\star) - \max_{a \neq a^\star} Q^\star(s,a)$$

relative to the pointwise estimation error [11]. If the error at a state is large compared with $\Delta(s)$, the greedy action can flip even when the weighted L_p error is small on average; conversely, large gaps can make the greedy action robust to nontrivial value error. Thus weighted L_p control and greedy-action correctness are related only through additional distributional arguments; the weighted norm alone does not certify the statewise $\arg\max$ decisions made by the policy. To turn a weighted L_p guarantee into a pointwise statement about greedy action correctness, one needs a conversion to a restricted sup-norm guarantee: over the full state-action space this incurs a **full-coverage** constant such as $1/\underline{w}$, while on a restricted Bellman-closed region it incurs $C_{\mathcal{R}}^{\sup}$.

Classical AVI/FQI analyses address a different question: they study how an already controlled Bellman error propagates. Exact-oracle analyses correspond to the limiting case in which the inner regression is solved exactly (i.e. $\alpha = 0$), while bounded-error analyses assume directly that

$$Q_{t+1} = TQ_t + e_t, \qquad \|e_t\| \le \varepsilon,$$

in the norm used for propagation. These assumptions are appropriate for error propagation, but they do not analyze how such an error is produced from finite inner loop optimization. In our notation, the bounded-error assumption hides an α-condition requirement. Decompose

$$e_t = e_t^{\mathrm{app}} + e_t^{\mathrm{opt}},$$

where $\|e_t^{\mathrm{app}}\|_\infty$ is controlled by the inherent Bellman error ε_∞. Then a sup-norm bounded-error assumption requires

$$\|e_t^{\mathrm{opt}}\|_\infty \le \varepsilon - \varepsilon_\infty.$$

For an inexact fitted Bellman update, the optimization residual is **endogenous**. In the MSE case, the proof shows that finite surrogate progress produces an optimization component scaling as

$$\|e_t^{\mathrm{opt}}\|_\infty = O\Big((1+\gamma)\sqrt{\alpha/\underline{w}}\,\|Q_t - Q^\star\|_\infty\Big),$$

with the approximation term contributing separately to the radius. Thus, if the AVI bounded-error condition is to be realized by finite inner-loop optimization, it implicitly requires the following sufficient condition:

$$\alpha \lesssim \underline{w}\left(\frac{\varepsilon - \varepsilon_\infty}{(1+\gamma)\|Q_t - Q^\star\|_\infty}\right)^2.$$

Our α-condition makes this hidden optimization requirement explicit and surrogate-specific.

The reason to work in the restricted sup norm is that strict Bellman contraction is an ℓ_∞ statement. The Bellman optimality operator is a γ-contraction in ℓ_∞, whereas it is not, in general, a contraction in a fixed weighted $L_p(\mu)$ norm. Therefore, any analysis that certifies that finite replay-weighted surrogate optimization preserves the Bellman contraction mechanism must convert weighted surrogate progress into a uniform residual on the region where contraction is claimed. Over the full state-action space this is the full-coverage assumption; on a Bellman-closed region $\mathcal{R}$, it is the localized coefficient

$$C_{\mathcal{R}}^{\sup} = \sup_{(s,a)\in\mathcal{R}} \frac{1}{w(s,a)}.$$

Thus $C_{\mathcal{R}}^{\sup}$ should not be viewed as a stronger assumption used to prove the same conclusion as standard AVI. Classical concentrability localizes distributional L_p-error propagation; our Bellman-closed coverage localizes sup-norm contraction analysis. The conclusion is correspondingly stronger and closer to the practical DQN mechanism: it identifies when the finite inner loop is accurate enough for the outer update to contract, how this requirement depends on the surrogate through α, and how the inherent Bellman error ε_∞ determines the limiting radius independently of the optimization accuracy.

D Proof for Proposition 3.6

Basic setup. We study the inner optimization over the residual vector $\delta = z - Y \in \mathbb{R}^m$, and let $R = \|\delta\|_\infty$; we are using the same network across all surrogates. The geometry we considered here is the ℓ_∞ norm, whose dual is ℓ_1. For a matrix H, we use the induced operator norm

$$\|H\|_{\infty,1} := \max_{\|u\|_\infty \le 1} \|Hu\|_1.$$

Throught the proof we consider the steepest-descent update Eq(7). We will bound smoothness by L_∞ for each surrogates (Step 1), invoke the descent lemma in this geometry (Step 2), use sublevel-set invariance to remain in $\mathcal{B}_\infty(R)$ (Step 3), and establish an ℓ_1–PL inequality on $\mathcal{B}_\infty(R)$ (Step 4), which together yield the claimed K (Step 5).

Step 1: ℓ_∞-smoothness constants for the surrogates

Lemma D.1 (ℓ_∞-smoothness)**.** *Let $\delta \in \mathbb{R}^m$ index (s,a) and let $w_i > 0$ with $\sum_i w_i = 1$. Define*

$$\mathcal{L}_{\mathrm{MSE}}(\delta) = \tfrac{1}{2}\sum_{i=1}^m w_i\,\delta_i^2, \qquad \mathcal{L}_{\mathrm{LC}}(\delta) = \tfrac{1}{\beta^2}\sum_{i=1}^m w_i\,\log\cosh(\beta\delta_i), \qquad \mathcal{L}_{\mathrm{LSC}}(\delta) = \tfrac{1}{\beta}\log\Big(\sum_{i=1}^m w_i\,\cosh(\beta\delta_i)\Big).$$

Then each surrogate is ℓ_∞-smooth with the following constants:

$$\boxed{L_\infty^{\mathrm{MSE}} = 1, \qquad L_\infty^{\mathrm{LC}} \le 1, \qquad L_\infty^{\mathrm{LSC}} \le 2\beta.}$$

Proof. By Theorem 5.12 in Beck [3], for twice continuously differentiable S, L-smoothness w.r.t. a norm $\|\cdot\|$ is equivalent to the uniform Hessian bound $\sup_\delta \|\nabla^2 S(\delta)\|_{\|\cdot\|\to\|\cdot\|_*} \le L$, where here $\|\cdot\| = \|\cdot\|_\infty$, $\|\cdot\|_* = \|\cdot\|_1$, and the induced operator norm is $\|\cdot\|_{\infty,1}$.

(a) $\mathcal{L}_{\mathrm{MSE}}$**.** $\nabla^2\mathcal{L}_{\mathrm{MSE}}(\delta) = \operatorname{diag}(w)$ for all δ. For any u with $\|u\|_\infty \le 1$,

$$\big\|\nabla^2\mathcal{L}_{\mathrm{MSE}}\, u\big\|_1 = \sum_{i=1}^m w_i|u_i| \;\le\; \sum_{i=1}^m w_i \;=\; 1.$$

Taking the supremum over $\|u\|_\infty \le 1$ yields $\|\nabla^2\mathcal{L}_{\mathrm{MSE}}\|_{\infty,1} = 2$.

(b) $\mathcal{L}_{\mathrm{LC}}$**.** $\nabla^2\mathcal{L}_{\mathrm{LC}}(\delta) = \operatorname{diag}\big(w_i \operatorname{sech}^2(\beta\delta_i)\big)$ for all δ. Hence, for any $\|u\|_\infty \le 1$,

$$\big\|\nabla^2\mathcal{L}_{\mathrm{LC}}\, u\big\|_1 = \sum_{i=1}^m w_i \operatorname{sech}^2(\beta\delta_i)\,|u_i| \;\le\; \sum_{i=1}^m w_i \operatorname{sech}^2(\beta\delta_i) \;\le\; \sum_{i=1}^m w_i \;=\; 1,$$

since $|u_i| \le 1$ and $\operatorname{sech}^2(\cdot) \le 1$. Thus $\|\nabla^2\mathcal{L}_{\mathrm{LC}}\|_{\infty,1} \le 1$ globally.

(c) $\mathcal{L}_{\mathrm{LSC}}$**.** Let $\pi_i := \dfrac{w_i\cosh(\beta\delta_i)}{\sum_{j=1}^m w_j\cosh(\beta\delta_j)}$, $t_i := \tanh(\beta\delta_i)$, and $a_i := \pi_i t_i$. Then $\nabla\mathcal{L}_{\mathrm{LSC}}(\delta) = a$ and a direct differentiation gives the Hessian $\nabla^2\mathcal{L}_{\mathrm{LSC}}(\delta) = \beta\big(\operatorname{diag}(\pi) - aa^\top\big)$ for all δ. For any u with $\|u\|_\infty \le 1$,

$$\big\|\nabla^2\mathcal{L}_{\mathrm{LSC}}\, u\big\|_1 = \beta\,\big\|\operatorname{diag}(\pi)u - a(a^\top u)\big\|_1 \;\le\; \beta\Big(\|\operatorname{diag}(\pi)u\|_1 + \|a\|_1\,|a^\top u|\Big).$$

Since $\sum_i \pi_i = 1$ and $\|u\|_\infty \le 1$, we have $\|\operatorname{diag}(\pi)u\|_1 = \sum_i \pi_i|u_i| \le 1$. Also, $\|a\|_1 = \sum_i \pi_i|t_i| \le \sum_i \pi_i = 1$, and $|a^\top u| \le \|a\|_1\|u\|_\infty \le 1$. Consequently,

$$\big\|\nabla^2\mathcal{L}_{\mathrm{LSC}}\, u\big\|_1 \;\le\; \beta(1+1) \;=\; 2\beta,$$

uniformly over δ and u, hence $\|\nabla^2\mathcal{L}_{\mathrm{LSC}}\|_{\infty,1} \le 2\beta$. □

Step 2: Descent in ℓ_∞ geometry. We apply the standard descent inequality for the steepest-descent step (Eq. equation 7) in the ℓ_∞ geometry.

Lemma D.2 (Descent lemma in ℓ_∞ geometry (Lemma 2 of Balles et al. [2]))**.** *If a differentiable function $\mathcal{L}$ is L_∞-smooth (i.e., $\|\nabla^2\mathcal{L}\|_{\infty,1} \le L_\infty$), then the update*

$$\delta_{k+1} = \delta_k - \frac{\|\nabla\mathcal{L}(\delta_k)\|_1}{L_\infty}\operatorname{sign}\big(\nabla\mathcal{L}(\delta_k)\big)$$

satisfies

$$\mathcal{L}(\delta_{k+1}) \;\le\; \mathcal{L}(\delta_k) \;-\; \frac{1}{2L_\infty}\,\|\nabla\mathcal{L}(\delta_k)\|_1^2.$$

Specializing Lemma D.2 with the smoothness constants from Lemma D.1 gives

$$\begin{aligned}
\text{MSE:}\quad & \mathcal{L}_{\mathrm{MSE}}(\delta_{k+1}) \;\le\; \mathcal{L}_{\mathrm{MSE}}(\delta_k) \;-\; \tfrac{1}{2}\,\big\|\nabla\mathcal{L}_{\mathrm{MSE}}(\delta_k)\big\|_1^2,\\
\text{LC:}\quad & \mathcal{L}_{\mathrm{LC}}(\delta_{k+1}) \;\le\; \mathcal{L}_{\mathrm{LC}}(\delta_k) \;-\; \tfrac{1}{2}\,\big\|\nabla\mathcal{L}_{\mathrm{LC}}(\delta_k)\big\|_1^2,\\
\text{LSC:}\quad & \mathcal{L}_{\mathrm{LSC}}(\delta_{k+1}) \;\le\; \mathcal{L}_{\mathrm{LSC}}(\delta_k) \;-\; \tfrac{1}{4\beta}\,\big\|\nabla\mathcal{L}_{\mathrm{LSC}}(\delta_k)\big\|_1^2.
\end{aligned}$$

Step 3: Coercivity ⇒ bounded $\|\delta\|_\infty$ (region invariance) We show the surrogates are ℓ_∞-*coercive*, i.e., $\mathcal{L}(\delta) \to \infty$ whenever $\|\delta\|_\infty \to \infty$. This implies all sublevel sets $\{\mathcal{L} \le S_0\}$ are bounded in $\|\cdot\|_\infty$, so iterates produced by the descent step equation 7 remain in a compact region where the smoothness bounds of Step 1 apply.

Definition D.3 (ℓ_∞-coercivity)**.** A function $\mathcal{L} : \mathbb{R}^m \to \mathbb{R}$ is ℓ_∞-*coercive* if $\mathcal{L}(\delta) \to \infty$ whenever $\|\delta\|_\infty \to \infty$.

Lemma D.4 (Coercivity of $\mathcal{L}_{\mathrm{MSE}}$, $\mathcal{L}_{\mathrm{LC}}$, and $\mathcal{L}_{\mathrm{LSC}}$)**.** *Let $R := \|\delta\|_\infty$ and pick $i^\star$ such that $|\delta_{i^\star}| = E$. With $w_{\min} := \min_i w_i > 0$,*

$$\text{MSE:}\quad \mathcal{L}_{\mathrm{MSE}}(\delta) = \tfrac{1}{2}\sum_i w_i\,\delta_i^2 \;\geq\; \tfrac{w_{\min}}{2}\,R^2 \;\xrightarrow[R\to\infty]{}\; \infty.$$

$$\text{LC:}\quad \mathcal{L}_{\mathrm{LC}}(\delta) = \tfrac{1}{\beta^2}\sum_i w_i\,\log\cosh(\beta\delta_i) \;\geq\; \tfrac{w_{\min}}{\beta^2}\,\log\cosh(\beta R) \;\xrightarrow[R\to\infty]{}\; \infty.$$

$$\text{LSC:}\quad \mathcal{L}_{\mathrm{LSC}}(\delta) = \tfrac{1}{\beta}\log\Big(\sum_i w_i\,\cosh(\beta\delta_i)\Big) \;\geq\; \tfrac{1}{\beta}\log\big(w_{\min}\cosh(\beta R)\big) \;\xrightarrow[R\to\infty]{}\; \infty.$$

Thus all three surrogates are ℓ_∞-coercive.

Since coercivity means $\mathcal{L}(\delta)\to\infty$ whenever $\|\delta\|_\infty\to\infty$, no sublevel set $\{\delta : \mathcal{L}(\delta)\leq S_0\}$ can "reach infinity" in $\|\cdot\|_\infty$ (else it would contain a sequence with $\mathcal{L}\to\infty$), so it is bounded; monotone descent ($\mathcal{L}(\delta_{k+1})\leq\mathcal{L}(\delta_k)$) then keeps all iterates inside this bounded set:

Corollary D.5 (Sublevel-set radius bounds)**.** *Fix $S_0 > 0$ and suppose $\mathcal{L}(\delta)\leq S_0$. Then the maximal coordinate $E = \|\delta\|_\infty$ is bounded as*

$$\text{MSE:}\quad \mathcal{L}_{\mathrm{MSE}}(\delta)\leq S_0 \;\Rightarrow\; R \;\leq\; \sqrt{\frac{2S_0}{w_{\min}}}.$$

$$\text{LC:}\quad \mathcal{L}_{\mathrm{LC}}(\delta)\leq S_0 \;\Rightarrow\; \sum_i w_i\log\cosh(\beta\delta_i) \;\leq\; \beta^2 S_0 \;\Rightarrow\; \log\cosh(\beta R) \;\leq\; \frac{\beta^2 S_0}{w_{\min}}$$
$$\Rightarrow\; R \;\leq\; \tfrac{1}{\beta}\operatorname{arcosh}\Big(e^{\,\beta^2 S_0/w_{\min}}\Big).$$

$$\text{LSC:}\quad \mathcal{L}_{\mathrm{LSC}}(\delta)\leq S_0 \;\Rightarrow\; \sum_i w_i\cosh(\beta|\delta_i|) \;\leq\; e^{\beta S_0}$$
$$\Rightarrow\; \cosh(\beta R) \;\leq\; \frac{e^{\beta S_0}}{w_{\min}} \;\Rightarrow\; R \;\leq\; \tfrac{1}{\beta}\operatorname{arcosh}\Big(\tfrac{e^{\beta S_0}}{w_{\min}}\Big).$$

Starting from any δ_0 with finite $\mathcal{L}(\delta_0)$, the descent step equation 7 generates a nonincreasing sequence $\mathcal{L}(\delta_k)\leq\mathcal{L}(\delta_0)$; hence δ_k remains in the bounded sublevel set of $S_0 := \mathcal{L}(\delta_0)$. Therefore the region $\{\|\delta\|_\infty\leq R\}$ with R given by Corollary D.5 is *invariant* for the inner loop, and the smoothness constants from Lemma D.1 apply throughout.

Step 4: ℓ_1–PL inequalities on the population sublevel set. Fix $S_0 > 0$ and let R be any number such that the sublevel set $\{\delta : \mathcal{L}(\delta)\leq S_0\}$ satisfies $\|\delta\|_\infty\leq R$ (see Corollary D.5). Define

$$\phi(a) \;:=\; \frac{\tanh^2(a)}{\log\cosh(a)}, \qquad a\geq 0,$$

which is decreasing, with $\phi(0) = 2$ and $\phi(a)\sim 1/a$ as $a\to\infty$.

Lemma D.6 (ℓ_1–PL on $\{\mathcal{L}\leq S_0\}$)**.** *With population weights $w_i > 0$, $\sum_i w_i = 1$, and $w_{\min} := \min_i w_i$, the following inequalities hold for all δ with $\|\delta\|_\infty\leq R$:*

$$\boxed{\begin{aligned}&\frac{1}{2}\big\|\nabla\mathcal{L}_{\mathrm{MSE}}(\delta)\big\|_1^2 \;\geq\; w_{\min}\,\mathcal{L}_{\mathrm{MSE}}(\delta), \qquad \frac{1}{2}\big\|\nabla\mathcal{L}_{\mathrm{LC}}(\delta)\big\|_1^2 \;\geq\; \frac{w_{\min}}{2}\,\phi(\beta R)\,\mathcal{L}_{\mathrm{LC}}(\delta),\\ &\frac{1}{2}\big\|\nabla\mathcal{L}_{\mathrm{LSC}}(\delta)\big\|_1^2 \;\geq\; \frac{\beta}{2}\,p_{\min}(\delta)\,\phi(\beta R)\,\mathcal{L}_{\mathrm{LSC}}(\delta).\end{aligned}}$$

Here $p_i(\delta) := \dfrac{w_i\cosh(\beta\delta_i)}{\sum_j w_j\cosh(\beta\delta_j)}$ and $p_{\min}(\delta) := \min_i p_i(\delta)$.

Proof. MSE. $\mathcal{L}_{\mathrm{MSE}}(\delta) = \frac{1}{2}\sum_i w_i\delta_i^2$, so $\nabla\mathcal{L}_{\mathrm{MSE}} = (w_i\delta_i)_i$. Let $x_i := \sqrt{w_i}\,\delta_i$. Then $\|\nabla\mathcal{L}_{\mathrm{MSE}}\|_1 = \sum_i\sqrt{w_i}|x_i|\geq\sqrt{w_{\min}}\|x\|_2$, hence $\|\nabla\mathcal{L}_{\mathrm{MSE}}\|_1^2\geq 2w_{\min}\mathcal{L}_{\mathrm{MSE}}$.

LC. For $\mathcal{L}_{\mathrm{LC}}(\delta) = \beta^{-2} \sum_i w_i \log\cosh(\beta\delta_i)$,

$$\partial_i \mathcal{L}_{\mathrm{LC}}(\delta) = \frac{w_i}{\beta} \tanh(\beta\delta_i) \quad \Longrightarrow \quad \big(\partial_i \mathcal{L}_{\mathrm{LC}}(\delta)\big)^2 = \frac{w_i^2}{\beta^2} \tanh^2(\beta\delta_i).$$

Since $\phi(a) := \frac{\tanh^2 a}{\log\cosh a}$ is decreasing and $|\delta_i| \le E$ on the sublevel,

$$\tanh^2(\beta\delta_i) = \phi(\beta|\delta_i|)\, \log\cosh(\beta\delta_i) \;\ge\; \phi(\beta E)\, \log\cosh(\beta\delta_i).$$

Therefore

$$\sum_i \big(\partial_i \mathcal{L}_{\mathrm{LC}}(\delta)\big)^2 \;\ge\; \frac{\phi(\beta E)}{\beta^2} \sum_i w_i^2\, \log\cosh(\beta\delta_i) \;\ge\; \phi(\beta E)\, w_{\min}\, \mathcal{L}_{\mathrm{LC}}(\delta),$$

where we used $w_i^2 \ge w_{\min} w_i$ and the definition of $\mathcal{L}_{\mathrm{LC}}$. Finally, $(\sum_i |a_i|)^2 \ge \sum_i a_i^2$ gives $\|\nabla \mathcal{L}_{\mathrm{LC}}(\delta)\|_1^2 \ge w_{\min}\, \phi(\beta R)\, \mathcal{L}_{\mathrm{LC}}(\delta)$, and dividing by 2 yields the stated inequality.

LSC. Write $A(\delta) := \sum_j w_j \cosh(\beta\delta_j)$ and $p_i := \frac{w_i \cosh(\beta\delta_i)}{A}$. Then $\nabla \mathcal{L}_{\mathrm{LSC}}(\delta)_i = p_i \tanh(\beta\delta_i)$ and

$$\|\nabla \mathcal{L}_{\mathrm{LSC}}\|_1^2 \;\ge\; \sum_i p_i^2 \tanh^2(\beta\delta_i) \;\ge\; p_{\min} \sum_i p_i \tanh^2(\beta\delta_i).$$

As above, $\tanh^2(\beta\delta_i) = \phi(\beta|\delta_i|)\, \log\cosh(\beta\delta_i) \ge \phi(\beta R)\, \log\cosh(\beta\delta_i)$. Hence

$$\sum_i p_i \tanh^2(\beta\delta_i) \;\ge\; \phi(\beta R) \sum_i p_i \log\cosh(\beta\delta_i).$$

Using the log-sum-exp variational identity with $x_i := \log\cosh(\beta\delta_i)$,

$$\beta\, \mathcal{L}_{\mathrm{LSC}}(\delta) = \log \sum_i w_i e^{x_i} = \sum_i p_i x_i - \mathrm{KL}(p\|w) \;\le\; \sum_i p_i x_i,$$

so $\sum_i p_i \log\cosh(\beta\delta_i) \ge \beta\, \mathcal{L}_{\mathrm{LSC}}(\delta)$. Putting these together: $\|\nabla \mathcal{L}_{\mathrm{LSC}}\|_1^2 \ge \beta\, p_{\min}\, \phi(\beta R)\, \mathcal{L}_{\mathrm{LSC}}$. Divide by 2 to match the stated form. □

Remark D.7 (On $p_{\min}$ and shaping via β)**.** For LSC, $p(\delta)$ depends on β through $p_i \propto w_i \cosh(\beta\delta_i)$. More generally, choosing β to keep βR small (early in training) keeps p well-shaped relative to w; as R decreases along optimization, β can be increased while maintaining $\beta R = O(1)$.

Step 5: Conclusion—contraction and step budgets for the α-condition Combining the descent inequality (Lemma D.2) with the ℓ_1–PL bounds (Lemma D.6) gives a linear contraction on the sublevel set $\{\|\delta\|_\infty \le R\}$:

$$\mathcal{L}(\delta_{k+1}) \;\le\; \Big(1 - \frac{\mu_\infty}{L_\infty}\Big)\, \mathcal{L}(\delta_k), \qquad \Longrightarrow \qquad \mathcal{L}(\delta_K) \;\le\; \exp\Big(-\frac{\mu_\infty}{L_\infty}\, K\Big)\, \mathcal{L}(\delta_0).$$

Thus to reach an α-reduction, $\mathcal{L}(\delta_K) \le \alpha\, \mathcal{L}(\delta_0)$, it suffices to take

$$K \;\ge\; \frac{L_\infty}{\mu_\infty}\, \log\frac{1}{\alpha}.$$

Instantiating L_∞ from Lemma D.1 and μ_∞ from Lemma D.6 yields

$$\frac{L_\infty}{\mu_\infty} \;=\; \begin{cases} \dfrac{1}{w_{\min}}, & \text{MSE,} \\[2ex] \dfrac{2}{w_{\min}\, \phi(\beta R)}, & \text{LC,} \\[2ex] \dfrac{4}{p_{\min}(\delta)\, \phi(\beta R)}, & \text{LSC.} \end{cases}$$

Using the surrogate-specific α-conditions $\alpha_{\mathrm{MSE}} = w_{\min}\left(\frac{1-\gamma}{1+\gamma}\right)^2$, $\alpha_{\mathrm{LC}} = w_{\min}\left(\frac{1-\gamma}{1+\gamma}\right)$, $\alpha_{\mathrm{LSC}} = \left(\frac{1-\gamma}{1+\gamma}\right)$, we obtain the step budgets

$$\boxed{\begin{aligned} K_{\mathrm{MSE}} &\geq \frac{1}{w_{\min}} \log\frac{1}{\alpha_{\mathrm{MSE}}} = \frac{1}{w_{\min}}\Big(\log\frac{1}{w_{\min}} + 2\log\frac{1+\gamma}{1-\gamma}\Big), \\ K_{\mathrm{LC}} &\geq \frac{2}{w_{\min}\,\phi(\beta R)} \log\frac{1}{\alpha_{\mathrm{LC}}} = \frac{2}{w_{\min}\,\phi(\beta R)}\Big(\log\frac{1}{w_{\min}} + \log\frac{1+\gamma}{1-\gamma}\Big), \\ K_{\mathrm{LSC}} &\geq \frac{4}{p_{\min}(\delta)\,\phi(\beta R)} \log\frac{1}{\alpha_{\mathrm{LSC}}} = \frac{4}{p_{\min}(\delta)\,\phi(\beta R)} \log\frac{1+\gamma}{1-\gamma}. \end{aligned}}$$

By coercivity (Lemma D.4) and monotone descent, the iterates remain in the bounded sublevel set of $S_0 := \mathcal{L}(\delta_0)$, so these constants are valid along the entire inner-loop trajectory. In practice one can shape p via β so that with a "moderate" radius (e.g. $\beta R \approx 1$) and maintaining $\beta R = O(1)$ keeps p well-shaped relative to w.

Remarks on the inner-loop analysis. Steps 2–5 can be read as a Lyapunov argument for the inner-loop update (7): the surrogate $\mathcal{L}$ is nonincreasing along the iterates, its sublevel sets are forward invariant, and on the invariant region $\{\|\delta\|_\infty \leq R\}$ it contracts linearly at rate $1 - \mu_\infty/L_\infty$. Lyapunov-style stability notions have also been used to analyze the internal dynamics of neural networks, e.g., the contraction of neuron activations toward stable patterns [29]. The resulting budget $K \geq (L_\infty/\mu_\infty)\log(1/\alpha)$ makes explicit how the accuracy demanded by the outer loop translates into inner-loop computation per target refresh; a similar trade-off between accuracy and computation is central to efficient LLM reasoning, where test-time frameworks adapt the token budget spent on each query [30].

E Q-Network Architectures: MLP (CartPole) and CNN (Atari)

Notes. The architectures are directly adopted from the CleanRL DQN implementation [15]; in Atari CNN the forward pass divides inputs by 255.0 (see code in the main text). The CartPole policy uses a 3-layer MLP on flattened observations with ReLU nonlinearities. CartPole result for different k are shown in Figure 5.

Compute. All experiments ran on a single NVIDIA GPU on a standard Linux server (e.g., A100, V100, RTX8000); no multi-GPU, TPU, or other specialized hardware was used.

Table 1: Observation preprocessing and input formatting used in our experiments. The Atari setup follows the common DQN-style pipeline; the CartPole setup uses raw vector observations flattened to a single dimension.

Hyper-parameter	**Atari (pixels)**	**CartPole (state vectors)**
Observation type	Stacked frames (4)	Low-dimensional state
Input shape to network	$(4, 84, 84)$	(d) (flattened)
Scaling / normalization	Divide by 255.0	None (raw values)
Frame stacking	4	N/A
Color space	Grayscale	N/A
Flatten before FC	After conv stack	Input already flat

F Additional Experiments

F.1 Ablation on Model Size

We study how function-approximation capacity interacts with surrogate choice following the same setup as Schwarzer et al. [35] by scaling the standard Nature-DQN convolutional architecture via

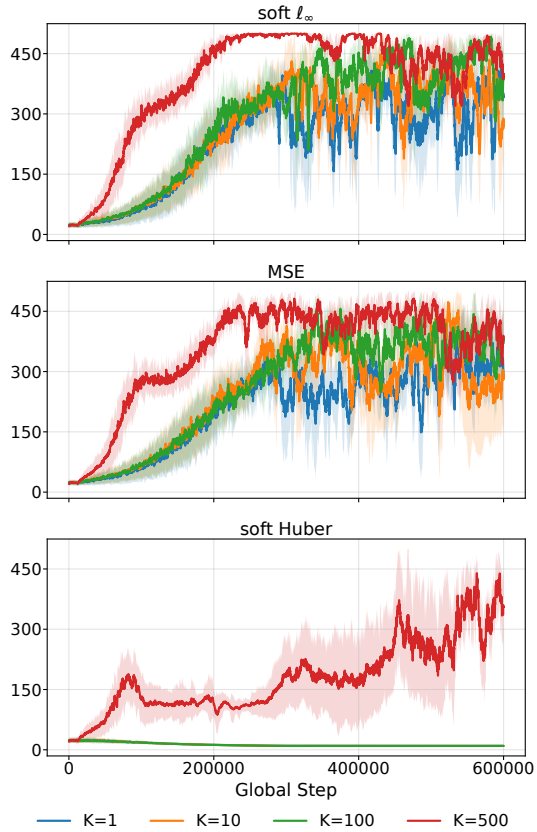

Figure 5: Episodic return on CartPole for each surrogate function with different number of inner steps k.

Table 2: Q-network architectures. Let $|A|$ denote the number of discrete actions and $d = \prod_i \text{shape}(s)_i$ the flattened CartPole observation dimension. All hidden layers use ReLU activations; the output heads are linear (no activation).

Domain	**Architecture (layer $\rightarrow$ units / kernel, stride)**
Atari (CNN)	Conv2d $(4 \rightarrow 32,\ 8 \times 8,\ \text{stride} = 4)$, ReLU Conv2d $(32 \rightarrow 64,\ 4 \times 4,\ \text{stride} = 2)$, ReLU Conv2d $(64 \rightarrow 64,\ 3 \times 3,\ \text{stride} = 1)$, ReLU Flatten Linear $(3136 \rightarrow 512)$, ReLU Linear $(512 \rightarrow \lvert A \rvert)$
CartPole (MLP)	Linear $(d \rightarrow 120)$, ReLU Linear $(120 \rightarrow 84)$, ReLU Linear $(84 \rightarrow \lvert A \rvert)$ (expects input of shape (B, d); outputs $(B, \lvert A \rvert)$)

Table 3: **Atari Q-network (Nature DQN) with width scaling.** We scale model capacity by a width multiplier $\tau \in \{0.5, 1, 2\}$. Channel and hidden widths are rounded to the nearest integer and clipped to be at least 1.

Component	Specification (layer $\rightarrow$ channels/units, kernel, stride)
Input	84×84 grayscale, 4 stacked frames
Conv1	Conv2d $(4 \rightarrow \lfloor 32\tau \rceil,\ 8 \times 8,\ \text{stride} = 4)$, ReLU
Conv2	Conv2d $(\lfloor 32\tau \rceil \rightarrow \lfloor 64\tau \rceil,\ 4 \times 4,\ \text{stride} = 2)$, ReLU
Conv3	Conv2d $(\lfloor 64\tau \rceil \rightarrow \lfloor 64\tau \rceil,\ 3 \times 3,\ \text{stride} = 1)$, ReLU
Flatten	7×7 spatial map $\Rightarrow$ features $= 7 \cdot 7 \cdot \lfloor 64\tau \rceil$
FC	Linear $(7 \cdot 7 \cdot \lfloor 64\tau \rceil \rightarrow \lfloor 512\tau \rceil)$, ReLU
Output	Linear $(\lfloor 512\tau \rceil \rightarrow \lvert A \rvert)$

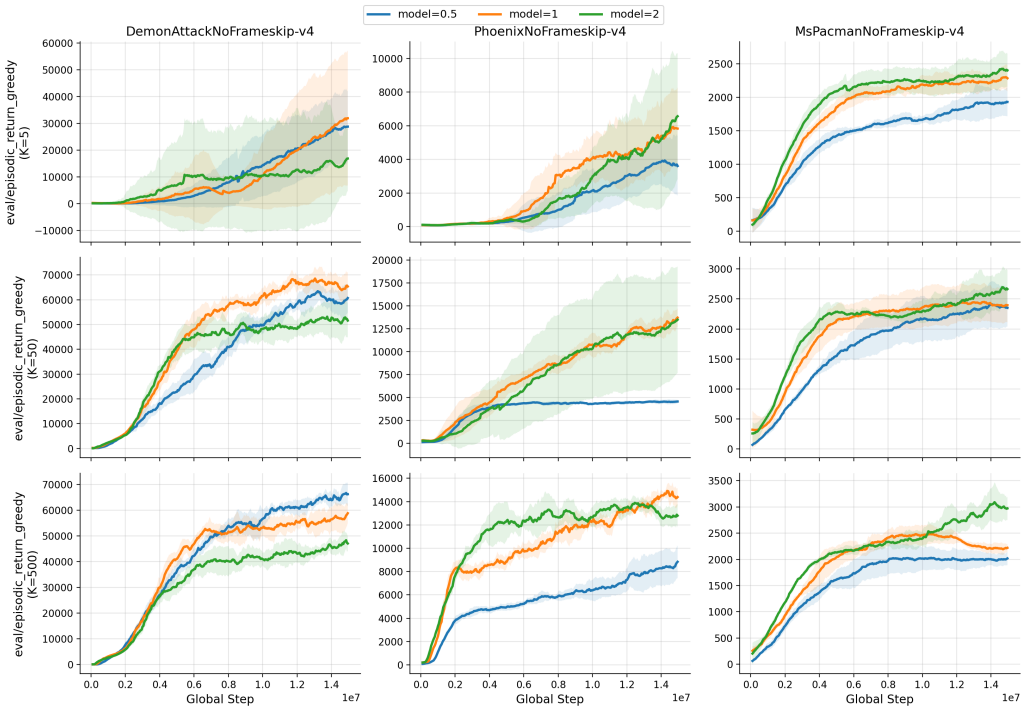

Figure 6: Episodic return of MSE with same number of inner steps and different model size. Shaded bands are std over seeds.

a width multiplier $\tau \in \{0.5, 1, 2\}$. Model architecture is presented in Table 3. Concretely, we multiply the convolutional channel counts $(32, 64, 64)$ and the fully-connected hidden width 512 by τ (rounded to integers, with a minimum of 1 channel/unit), yielding networks whose parameter count scales approximately as $\Theta(\tau^2)$. All other hyperparameters are kept fixed, and we evaluate on DemonAttack, Phoenix, and MsPacman using greedy evaluation returns.

Figure 6, 7, 8 shows that increased capacity typically improves performance across games, with the clearest gains appearing once K is moderate/large (50/500). This is consistent with the surrogate perspective: once the inner loop can make meaningful progress against a frozen target, performance becomes increasingly limited by approximation error rather than optimization. We also observe some non-monotonicity with increasing τ in a few configurations (notably DemonAttack), which is consistent with the fact that off-policy bootstrapping can amplify optimization noise and replay-induced distribution shift, so higher-capacity Q network do not always translate into higher returns under fixed hyperparameters.

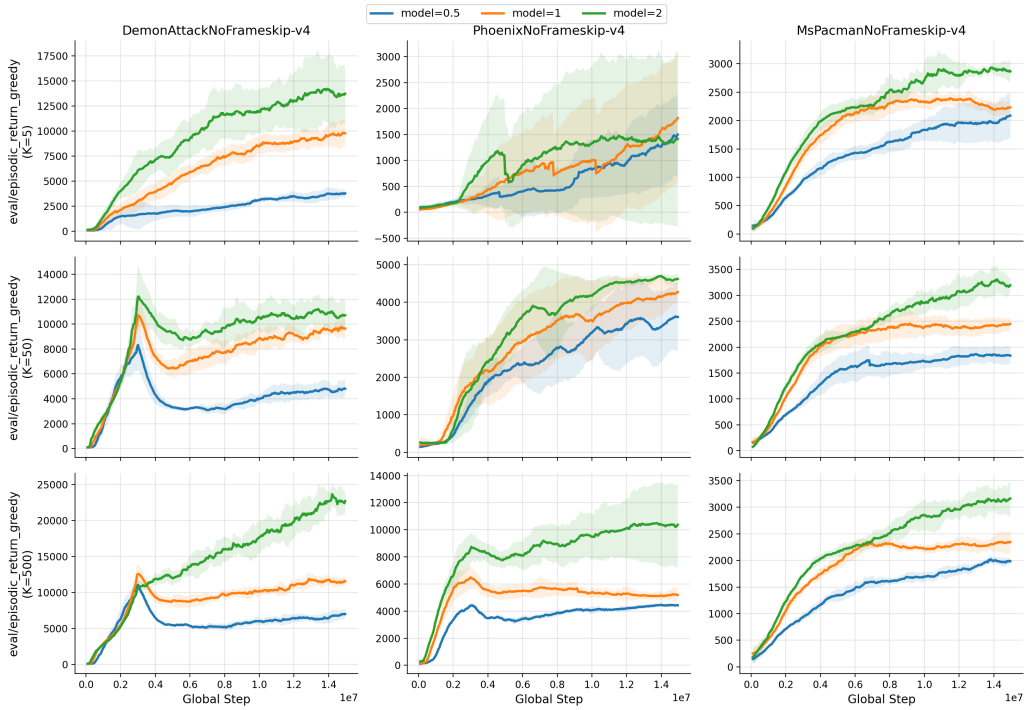

Figure 7: Episodic return of soft Huber with same number of inner steps and different model size. Shaded bands are std over seeds.

F.2 Ablation on Inner Steps

We next ablate the number of optimization steps performed per frozen target network, which corresponds to the inner-loop budget K_{target} in Algorithm 1 on the same three Atari environments.

Figure 9, 10, 11 demonstrates that increasing K_{target} is a primary driver of performance on DemonAttack and Phoenix, while MsPacman exhibits earlier saturation. This aligns with the role of the α-condition in Theorem 3.3: larger K_{target} gives the inner loop sufficient budget to reduce the surrogate loss against a fixed target before the next target refresh, thereby better realizing the predicted outer-loop improvement behavior. Across surrogates, the qualitative dependence on K_{target} is consistent: small K_{target} can lead to under-optimized regression against rapidly changing targets, whereas moderate/large K_{target} substantially stabilizes learning and improves returns.

F.3 Ablation on Fixed Temperature

Findings. Across all games, **soft-ℓ_∞ with adaptive** β is consistently competitive with, or superior to, any fixed choice of β, supporting the use of a controller that increases worst-case focus as training stabilizes. For **soft-Huber**, smaller β (e.g., $\beta = 1$) is mostly favored, which matches the MSE behavior: a smaller β enlarges the quadratic (Euclidean) region of the loss, yielding updates closer to ℓ_2 regression early on. These observations suggest that *adaptive β is particularly impactful for soft-ℓ_∞*, where alignment with the ℓ_∞ control geometry improves as β grows, while the necessity of adapting β for soft-Huber remains unclear in our Atari ablations, though concurrent work proposes dynamic Huber schedules as beneficial [43].

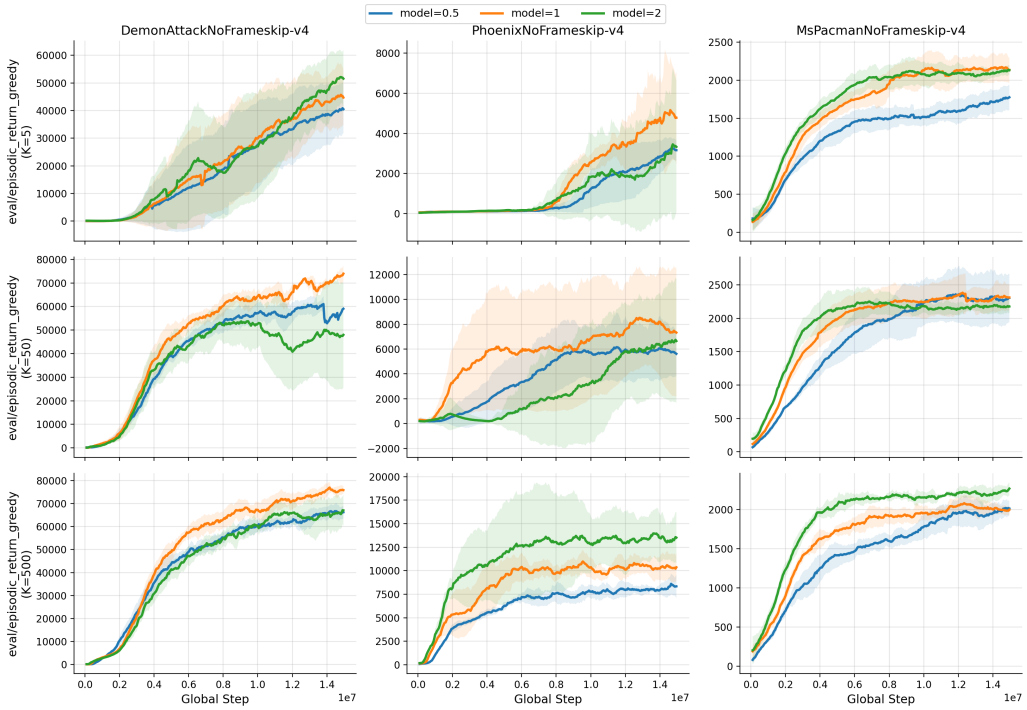

Figure 8: Episodic return of soft ℓ_∞ with same number of inner steps and different model size. Shaded bands are std over seeds.

F.4 Measuring the Inner-Loop Surrogate Reduction

Appendix F.2 shows how the return changes with the target period K_{target}, but not whether additional inner updates actually reduce the fixed-target surrogate, as the α-condition Eq.6 posits. We measure this directly on CartPole.

Setup. We train with the three surrogates (MSE, soft Huber (LC), and soft-ℓ_∞ (LSC)) under two target constructions: the max-backup target of Algorithm 1, which is the update analyzed in Theorem 3.3, and the Double-DQN target used in Section 5. Each of the six configurations is tested with five seeds for 400k environment steps, with $K_{\text{target}} = 500$ optimizer updates per target period. All runs use minibatches of 512 transitions, a replay buffer of 20k transitions, and AdamW with weight decay 10^{-3} and learning rate 2.5×10^{-4}, lowered to 5×10^{-5} after 300k steps; LC and LSC use the adaptive temperature of Section 3.3. In every run we keep ten target periods fixed in advance (those starting at environment steps 20k, 60k, …, 380k), for a total of $2 \times 3 \times 5 \times 10 = 300$ periods, all of which are reported.

At the start of a fixed period t, right after the hard target synchronization and before the first optimizer update, we sample 2,048 transitions from the replay buffer and freeze their target labels $r + \gamma(1-d)\max_{a'} Q_{\hat{\theta}_t}(s', a')$, together with the temperature and loss normalization of the surrogate. Let $\widehat{L}_t$ be the surrogate on this frozen set and $\theta_{t,k}$ the online parameters after k updates of the period. We track the ratio

$$U_t(k) \;=\; \frac{\widehat{L}_t(\theta_{t,k})}{\widehat{L}_t(\theta_{t,0})} \tag{28}$$

in two ways. In the *deployed* measurement, training proceeds unchanged and we evaluate the actual training iterates; $U_t(500)$ is read after the last update of the period, before the next synchronization. Double-DQN runs are scored on the same frozen labels: at the start of a period the online and target networks coincide, so the two constructions give identical labels there. In the *frozen control*, we clone

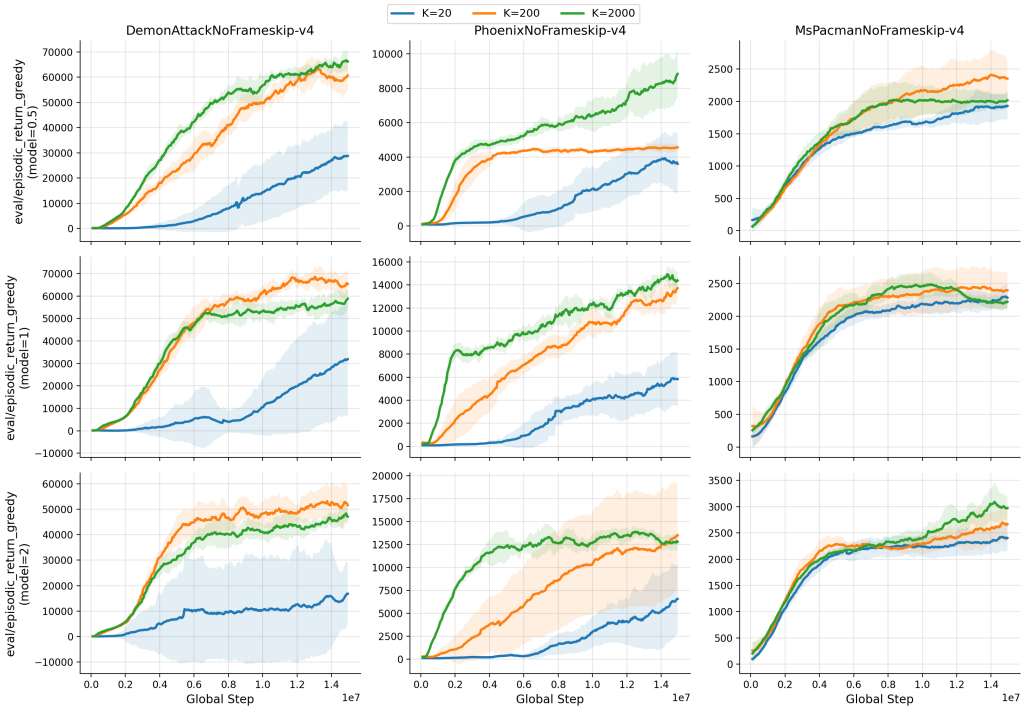

Figure 9: Episodic return of MSE with same models and different inner loop steps. Shaded bands are std over seeds.

the network and the AdamW state at the start of the period and run 500 updates on a fixed regression problem: its minibatches come from a fixed set of 16k transitions disjoint from the diagnostic set, and its labels and temperature are frozen at their period-start values. Each continuation is repeated three times. The deployed measurement shows what the actual run achieves. The control isolates the effect of giving one and the same inner problem more updates; it does not emulate complete runs with a different K_{target}, which would refresh their targets at different times.

The ratio is a conservative witness for the empirical counterpart of Eq.(6). Let $\widehat{L}_t^\star \geq 0$ be the smallest value of $\widehat{L}_t$ over the function class. Since the surrogates are nonnegative, whenever $U_t(k) < 1$,

$$\frac{\widehat{L}_t(\theta_{t,k}) - \widehat{L}_t^\star}{\widehat{L}_t(\theta_{t,0}) - \widehat{L}_t^\star} \;\leq\; U_t(k) \;<\; 1, \tag{29}$$

so the excess loss contracts by a factor of at most $U_t(k)$, and no knowledge of $\widehat{L}_t^\star$ is needed.

Results. Table 4 and Figure 13 summarize the results. In the deployed runs, 293 of the 300 periods end with at least a 1% reduction (145/150 under max-backup and 148/150 under Double-DQN), and 290 satisfy $U_t(500) < U_t(5)$, i.e., the frozen surrogate is lower after 500 updates than after 5. The median endpoint ratio lies between 0.23 and 0.62, depending on the configuration. The frozen control agrees: 857 of the 900 continuations end with at least a 1% reduction, and 833 satisfy $U_t(500) < U_t(5)$. For both measurements, the median of $\log U_t(500) - \log U_t(5)$ over the periods of a run is negative in all 30 target–surrogate–seed combinations.

Together with Appendix F.2, these measurements give direct finite-sample evidence that, on CartPole, additional inner updates produce the multiplicative surrogate decrease posited by Eq.(6), for the analyzed update as well as for the Double-DQN variant. They do not verify the population-level α-condition, the numerical thresholds of Theorem 3.3, or the outer ℓ_∞ contraction, because the population objective, its optimum, and the coverage quantities are not observed.

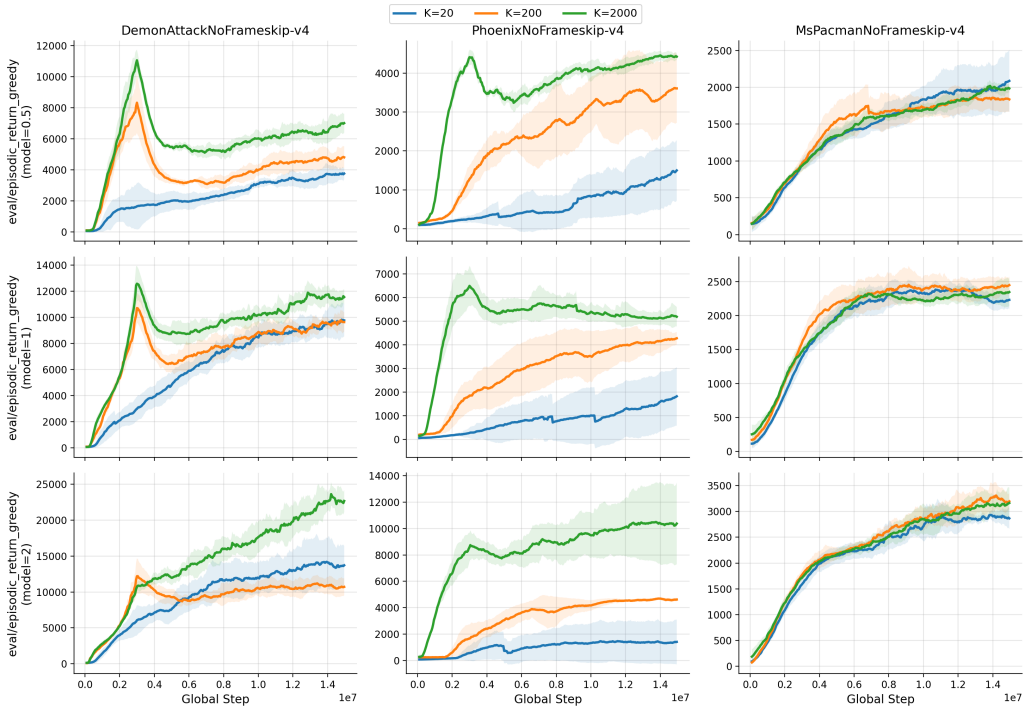

Figure 10: Episodic return of soft Huber with same models and different inner loop steps. Shaded bands are std over seeds.

Table 4: CartPole diagnostic: frozen-surrogate ratio after 500 inner updates. Each row covers 50 periods (5 seeds $\times$ 10 periods); the frozen control has three continuations per period. Columns: the median of $U_t(500)$, taken first over the periods of a seed and then across seeds; the number of periods (continuations) with $U_t(500) \leq 0.99$, i.e., at least a 1% reduction; and the number with $U_t(500) < U_t(5)$.

		Deployed			Frozen control	
Target	Surrogate	Median	≤ 0.99	$< U_t(5)$	≤ 0.99	$< U_t(5)$
Max-backup	MSE	0.231	50/50	49/50	146/150	142/150
	Soft Huber (LC)	0.587	50/50	50/50	150/150	150/150
	Soft-ℓ_∞ (LSC)	0.358	45/50	45/50	130/150	120/150
Double-DQN	MSE	0.295	50/50	50/50	149/150	145/150
	Soft Huber (LC)	0.625	50/50	50/50	150/150	150/150
	Soft-ℓ_∞ (LSC)	0.440	48/50	46/50	132/150	126/150
Total		–	293/300	290/300	857/900	833/900

F.5 Dueling DQN architecture

We additionally test whether the surrogate ordering observed in the main experiments is tied to the standard single-head DQN architecture. To this end, we repeat the Atari experiments with a Dueling DQN architecture [40], which decomposes the last layers into a state-value stream and an action-advantage stream while keeping the same underlying Q-learning update. All other training details are kept the same as in Section 5: the replay buffer, target-network update period, optimizer, exploration schedule, adaptive temperature rule, and evaluation protocol are unchanged. Thus, this ablation isolates whether the loss-geometry effect persists under a common architectural variant of DQN.

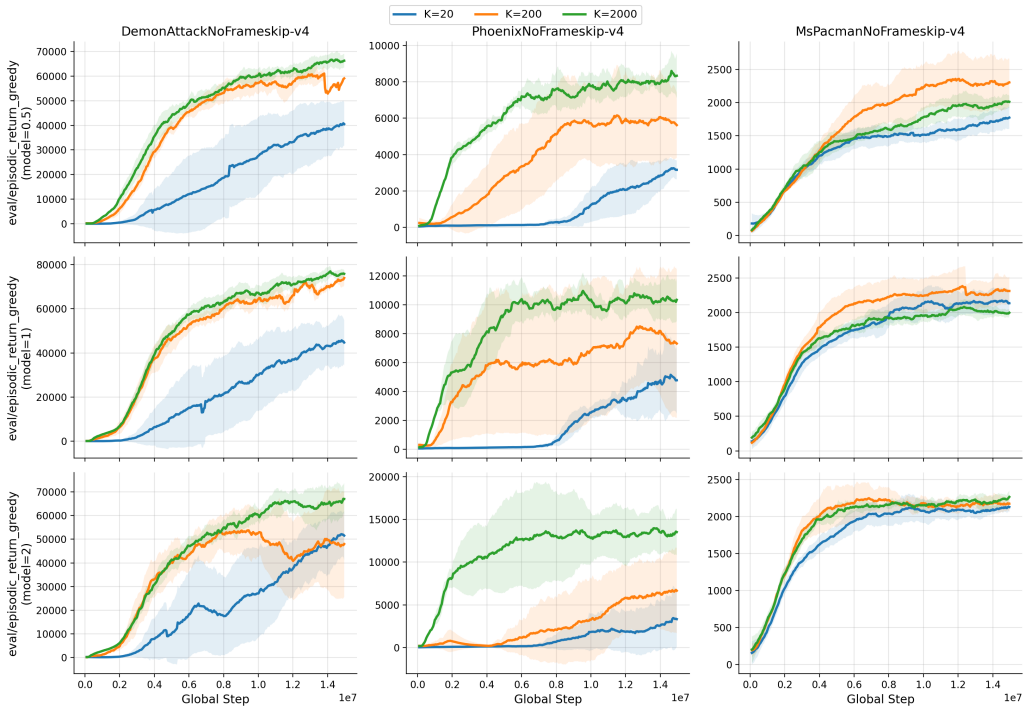

Figure 11: Episodic return of soft ℓ_∞ with same models and different inner loop steps. Shaded bands are std over seeds.

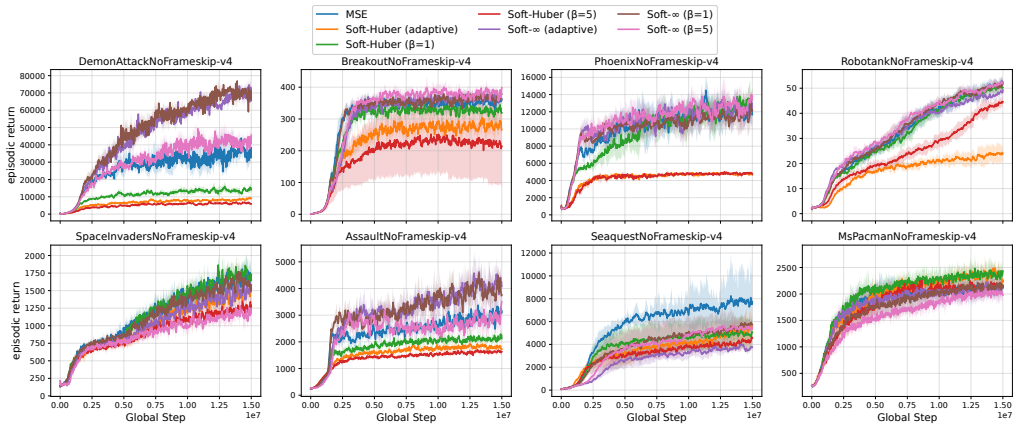

Figure 12: **Fixed-β ablation vs. adaptive β.** Episodic return on eight Atari games comparing soft-ℓ_∞ and soft-Huber with *fixed* temperatures ($\beta \in \{1, 5\}$) against their *adaptive* variants; MSE shown for reference. Shaded bands are std over seeds.

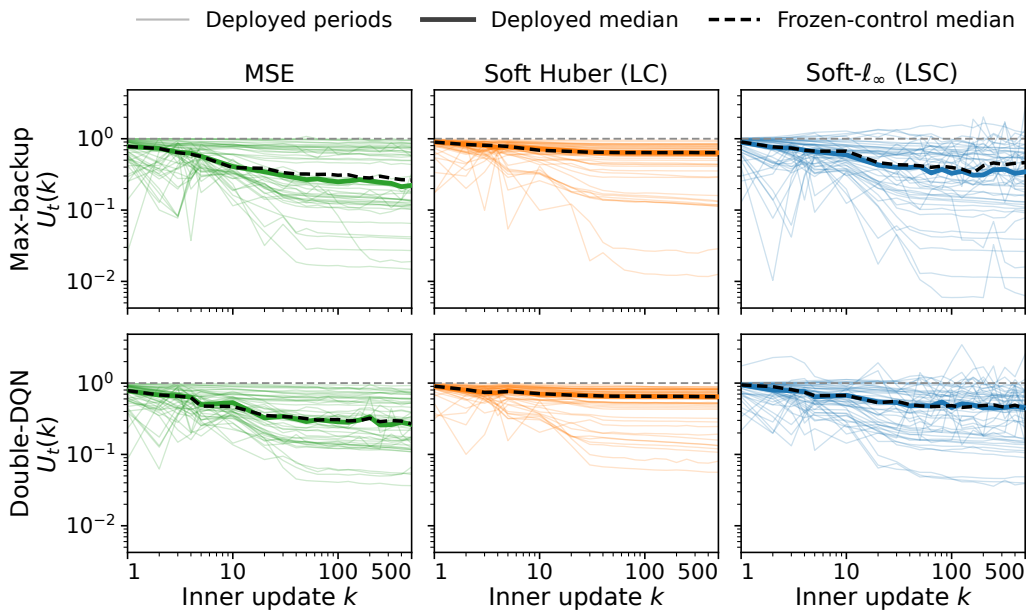

Figure 13: Frozen-surrogate ratio $U_t(k)$ against the number of inner updates k on CartPole (logarithmic axes). Thin lines: the 50 instrumented periods of each configuration in the deployed measurement. Solid line: median of the deployed measurement. Dashed line: median of the frozen control. Medians are computed as in Table 4. The horizontal line marks $U_t(k) = 1$.

Figure 14 reports episodic returns and Figure 15 reports the corresponding maximum TD-target residuals. The results show that the qualitative residual ordering persists under the dueling architecture. Across most games, soft-ℓ_∞ remains in the lowest-residual group, often matching or improving over MSE, while soft-Huber produces substantially larger maximum residuals. This is especially visible in DemonAttack, Breakout, SpaceInvaders, Assault, and MsPacman, where the soft-Huber max-TD curves remain consistently above the other two losses. The return curves are more environment-dependent, as expected from action-gap and policy-equivalence effects [11], but they remain broadly consistent with the residual picture: soft-ℓ_∞ gives strong or competitive returns in several games, including DemonAttack, Breakout, Phoenix, and Assault, while MSE is stronger in some games such as Robotank and SpaceInvaders. Overall, the dueling experiments support the main conclusion that the benefit of the soft-ℓ_∞ surrogate is not an artifact of the vanilla DQN head; it persists when the value function is represented through a value–advantage decomposition.

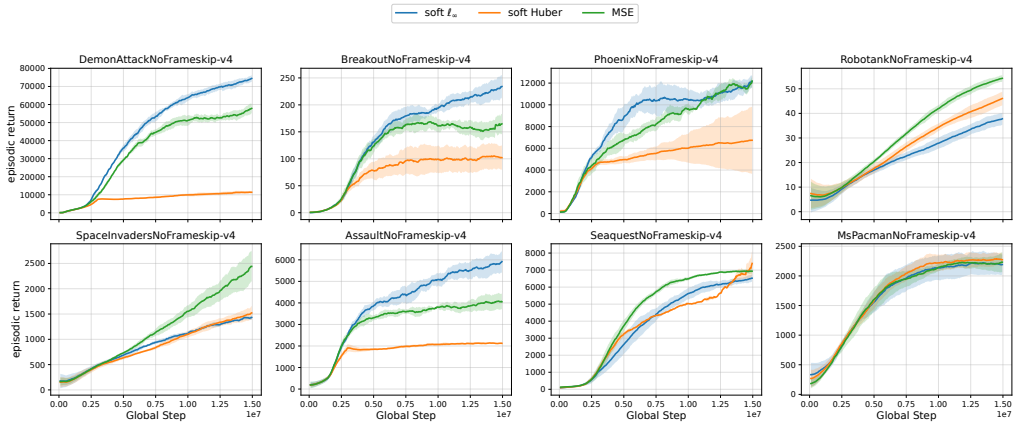

Figure 14: Dueling DQN episodic returns across Atari games. We compare soft-ℓ_∞, soft-Huber, and MSE using the same training protocol as in the main experiments, replacing only the standard DQN head by a dueling value–advantage head. Solid lines show the mean over seeds and shaded bands denote standard deviation.

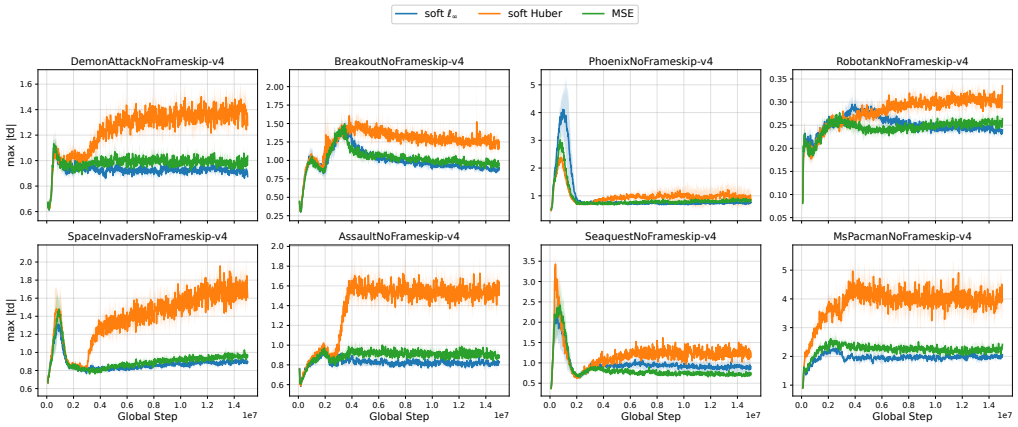

Figure 15: Dueling DQN maximum Bellman residuals across Atari games. The soft-ℓ_∞ surrogate remains in the lowest-residual group across most environments, whereas soft-Huber often leaves substantially larger maximum TD errors. This indicates that the residual-reduction effect of the soft-ℓ_∞ geometry is robust to the dueling architecture.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state three contributions: (i) a contraction guarantee for fixed-target DQN under inexact inner-loop optimization (Theorem 3.3, Section 3.1), (ii) surrogate-specific admissible α-thresholds with the soft-ℓ_∞ loss removing the coverage dependence from the admissible threshold (Theorem 3.3, Proposition 3.6,Appendix C.5), and (iii) empirical validation on Atari (Section 5). Each claim is supported by the corresponding theoretical or empirical result, and assumptions are stated explicitly.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 explicitly states that the guarantees are conditional on achieving the α-decrease, that certifying this for finite-step AdamW on deep networks is outside the theory, that Proposition 3.6 is tabular, and that the uniform ℓ_∞ guarantee requires full coverage or its Bellman-closed analogue.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All assumptions (Assumption 3.2) are stated alongside the main results. Full proofs of Theorem 3.3 and Proposition 3.6 are given in Appendix C and D, with all supporting lemmas numbered and cross-referenced.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification:Full hyperparameters in Section 5 and Appendix E; implementation builds on CleanRL.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Anonymized code in supplemental material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5and Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Mean $\pm$ std over 5 seeds; shaded bands in all figures.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification:Each Atari run trains for 15M frames and takes approximately 24 GPU-hours with small variation across A100/V100/RTX8000 machines. The main experiments use 8 games $\times$ 3 surrogates $\times$ 5 seeds, for approximately $8 \times 3 \times 5 \times 24 = 2880$ GPU-hours. Ablations use the same 15M-frame budget per run, with the same approximate per-run cost. We report the GPU types used in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Conforms.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: Foundational theoretical work on RL convergence; no direct societal impact.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification:No high-risk assets released.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: CleanRL, Gymnasium, PyTorch, AdamW cited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs not part of core methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.