Dual-SAR: Primal–Dual Shift-Aware Rewards for Model-Based Offline RL without Hyperparameter Tuning

Liz Lemma Future Detective

January 20, 2026

Abstract

Model-based offline RL trains policies using an offline dataset and synthetic rollouts from a learned dynamics model, but performance degrades under distribution shift induced by both model bias (model vs environment dynamics) and policy shift (learned policy vs behavior policy). Recent work derives a shifts-aware reward (SAR) that augments the reward with log-likelihood ratio corrections for these shifts, but practical success depends on manually tuned coefficients controlling the strength of model-bias and policy-shift penalties. We propose Dual-SAR, which reframes SAR as the Lagrangian of an explicit constrained offline RL problem with mismatch budgets, and learns the SAR coefficients automatically as dual variables via primal–dual optimization. Dual-SAR alternates between (i) off-policy actor–critic updates on a corrected reward (drop-in for SAMBO-style pipelines) and (ii) stochastic dual updates that target user-specified budgets on dynamics mismatch and behavior deviation, using discriminative density-ratio estimators. We provide clean formulations and tight finite-sample guarantees in tabular/linear MDPs: Dual-SAR achieves near-feasibility and near-optimality among feasible policies with rates matching known offline RL lower bounds up to logarithmic factors, and we isolate the additional price of density-ratio error. Empirically, we recommend validating Dual-SAR on D4RL/NeoRL with stress tests where model error varies over training to demonstrate automatic conservatism and reduced tuning burden.

Table of Contents

methods; pessimism/PEVI bounds; domain adaptation reward correction (DARC).

3. 3. Preliminaries and Notation: MDP, occupancy measures, divergence budgets, density-ratio estimation via classifiers, effective horizon H and discounted-to-finite-horizon translation.

4. 4. Clean Problem Formulation (Constrained Shift-Robust Offline RL): define feasible set via model-bias and policy-shift constraints; discuss choices of D_f, D_g (KL/TV/IPM) and measurability from data.

5. 5. From Constraints to Shift-Aware Rewards: derive Lagrangian; show corrected reward is SAR with dual variables; connect to variational lower bounds (as in SAR) and to pessimism-style penalties.

6. 6. Dual-SAR Algorithm: MBPO-style rollouts + actor–critic on corrected reward; dual updates for (α, β) targeting budgets; stabilization (projection, stepsizes, clipping/certification of logits).

7. 7. Theory I (Tabular MDPs): feasibility and near-optimality bounds; decomposition into statistical error, optimization error, and ratio-estimation error; matching lower bound discussion.

8. 8. Theory II (Linear MDPs / Function Approximation): extend guarantees under realizability and concentrability; emphasize what breaks in general nonlinear settings.

9. 9. Complexity and Implementation Notes: compute/space costs; how to estimate constraint values from classifiers; practical knobs (budgets) vs tuned penalties; recommended diagnostics.

10. 10. Experimental Plan (Flagged as Strengthening): D4RL/NeoRL; ablations vs tuned SAR; automatic budget adaptation; stress tests with controlled model degradation and multi-stage training; calibration effects.

11. 11. Discussion and Limitations: dependence on density-ratio estimation; positivity/log transform; how to pick budgets; open problems (likelihood-free extensions, multi-source budgets).

1 Introduction

Offline reinforcement learning seeks to optimize a policy using a fixed dataset of interactions, without additional queries to the environment. In model-based offline RL, we additionally fit a dynamics model from the dataset and use it to generate synthetic transitions that enlarge the training distribution. This approach is attractive because it can reduce the variance of purely model-free offline updates and can improve sample efficiency by reusing the learned model to explore counterfactual futures. At the same time, it introduces a distinctive and often dominating failure mode: the policy we train is shaped by distributions that are not those of the true environment, but those induced by a learned model and by actions that may not be supported by the data. The resulting distribution shift manifests as a mismatch between the training objective (computed on data and model rollouts) and the deployment objective (real return in the environment).

We distinguish two sources of shift that must be controlled simultaneously. First, there is *model shift*: even if the policy remains close to the behavior policy that collected the data, multi-step rollouts under the learned model may deviate from the true environment, and the deviation typically compounds with rollout horizon. A short-horizon model may be accurate on the dataset distribution but still produce biased returns when the policy changes, since the state–action occupancy induced by the new policy differs from that of the dataset. Second, there is *policy shift*: even if we were to train purely on real data, optimizing an unconstrained policy can drive the learned policy toward regions of the action space that are underrepresented or absent in the dataset, where value estimates and model predictions are not statistically identifiable. In practice these effects interact: policy shift pushes the learner into regions where the model is weak, and model bias in turn supplies apparently favorable synthetic evidence that further encourages the shift.

Existing methods address these issues by imposing pessimism, restricting policy updates, or penalizing model rollouts. A particularly direct approach is to modify the reward of transitions by an additive correction term derived from a density ratio or a classifier logit, thereby discouraging synthetic transitions that appear unlikely under the real data and discouraging actions that appear unlikely under the behavior policy. Such "shift-aware reward" (SAR) schemes are operationally simple: one trains a discriminator to distinguish environment transitions from model transitions (or behavior actions from current-policy actions), interprets the discriminator output as an estimated log-ratio, and adds a penalty or bonus to the reward before performing standard off-policy actor–critic updates. However, in their basic form these methods require *manual tuning* of the penalty weights that trade off return maximization against conservatism. The appropriate magnitude depends on reward scale, discounting, model class, discriminator calibration, rollout horizon, and dataset coverage, and it can vary substantially across tasks. Consequently, a fixed penalty weight is either too small to prevent exploitation of model errors, or too large to prevent any improvement over the behavior policy.

We view this tuning problem as an algorithmic symptom of a missing *constraint-handling mechanism.* Penalized objectives are Lagrangian relaxations of constrained problems, but a fixed penalty corresponds to selecting dual variables a priori rather than solving for them. In offline RL, where the relevant divergences and ratios are only estimable from finite data (and often only on a restricted support), it is rarely clear how to select these multipliers in a way that both enforces safety-like budgets and permits improvement. This observation motivates our central design choice: we explicitly formulate model-based offline RL as a constrained optimization problem with two budgets, one controlling average model mismatch along the learned policy and one controlling average deviation of the learned policy from the behavior policy, and we solve the resulting saddle-point problem by stochastic primal–dual updates.

Concretely, we propose *Dual-SAR*, a shift-aware reward method in which the correction weights are *dual variables* updated online from estimated constraint violations. The algorithm alternates between (i) generating short synthetic rollouts from the learned model starting from states drawn from the offline dataset, (ii) training discriminators to estimate the relevant log-ratios between true and model dynamics and between current and behavior policies, (iii) performing off-policy RL updates on a mixed replay buffer using a corrected reward of the form

$$\tilde{r}_{\alpha,\beta}(s,a,s') \;=\; u(r(s,a)) \;+\; \alpha\,\widehat{\log\rho}_p(s,a,s') \;+\; \beta\,\widehat{\log\rho}_\pi(s,a),$$

with optional clipping for stability, and (iv) updating the dual variables by projected ascent so as to enforce the user-specified budgets. The presence of the utility transform u accommodates variational representations of divergences and ensures that the reward shaping aligns with the Lagrangian form; in the simplest instantiation one may take $u(r) = r$, while in settings where a logarithmic transformation is required we use the assumption that rewards are uniformly positive.

This primal–dual viewpoint yields three practical benefits. First, it replaces per-task penalty tuning by interpretable *budgets* on model mismatch and policy shift, which are typically easier to specify and can be monitored during training. Second, it decouples conservatism across the two failure modes: the dual variable associated with model mismatch increases only when model bias is detected along the current policy, and the dual variable associated with policy shift increases only when the policy departs too far from the behavior distribution. Third, the dual iterates provide diagnostics: persistent growth of a dual variable indicates that the corresponding budget is infeasible under the current policy class or that the estimators are unreliable in the visited region, which can be acted upon by shortening rollouts, improving the model, or restricting the policy class.

Our theoretical analysis formalizes these claims in a tabular (and, by standard extension, linear) setting under conventional coverage and realizability assumptions. We show that the corrected reward used by Dual-SAR is exactly the per-transition decomposition of the Lagrangian of the constrained problem for divergences admitting variational forms, justifying the algorithm as principled rather than heuristic reward shaping. We then establish that stochastic primal–dual updates converge to an approximate saddle point when applied in the occupancy-measure formulation, and we translate the resulting primal–dual gap into a bound on suboptimality relative to the best feasible policy in the class. Importantly, we isolate the contribution of discriminator error as an additive term proportional to the effective horizon, reflecting the fact that systematic log-ratio error accumulates through the discounted sum. Finally, we complement the upper bounds with matching-rate lower bounds showing that the statistical term is unimprovable (up to logarithmic factors) under the same information constraints, and we recall the fundamental impossibility of nontrivial guarantees without adequate support overlap.

In summary, Dual-SAR is designed to preserve the computational convenience of model-based synthetic rollouts and classifier-based correction, while providing an explicit mechanism to control the two dominant sources of distribution shift in offline model-based RL. The remainder of the paper develops the method and analysis, and situates it relative to prior pessimistic model-based approaches, ratio-estimation-based correction methods, and primal–dual constrained RL.

2 Related Work

Shift-aware reward shaping and SAMBO/SAR. A line of work most directly connected to our algorithmic template modifies the reward by additive terms derived from distribution shift signals, so that standard off-policy RL updates become conservative with respect to model error or dataset support. We refer to this family as *shift-aware reward* (SAR) methods: a discriminator is trained to separate environment transitions from model-generated transitions (or behavior actions from current-policy actions), and its calibrated logit is interpreted as an estimate of a log density ratio, which is then added to the reward with a user-chosen weight. Recent model-based offline variants (e.g., SAMBO and related approaches) instantiate this idea in an MBPO-style loop with short model rollouts and policy updates under a corrected reward **?**. Empirically, such methods can curb model exploitation and reduce the tendency of the policy to drift toward unsupported actions, but the choice of correction weights is typically task-dependent and sensitive to reward scaling, rollout horizon, and discriminator calibration. Our contribution is to treat these weights as dual variables associated with explicit budgets, thereby replacing hand-tuned penalties by a primal–dual mechanism that adapts conservatism online.

Classifier-based density-ratio estimation. The use of discriminators to obtain density ratios is classical and admits a precise variational interpretation. For many divergences (including KL and broader f-divergences), one can represent divergence functionals via a supremum over test functions, yielding an optimal discriminator whose logit recovers a log ratio **??**. This observation has been exploited in off-policy evaluation, domain adaptation under covariate shift, and model-based RL, where the goal is to quantify mismatch between the model rollout distribution and the real data distribution. Our setting requires two ratios of distinct types: a transition ratio p/m to diagnose model mismatch along the current occupancy, and a policy ratio π/π_b to diagnose action shift. The latter is more delicate offline because π_b is unknown and must be inferred indirectly from data, often via behavior cloning proxies or action discriminators. We emphasize that our method does not require the ratios to be perfectly estimated; rather, our analysis isolates the effect of bounded log-ratio error as an additive term in the feasibility and near-optimality bounds, consistent with the fact that systematic ratio bias accumulates over the effective horizon.

Conservative model-based offline RL: MOPO, MOReL, COMBO, MOBILE. A second closely related literature addresses model bias in offline model-based RL via *pessimism.* MOPO penalizes model rollouts according to an uncertainty estimate (often ensemble disagreement), discouraging trajectories that are likely to be out-of-distribution for the learned model **?**. MOReL constructs an explicit "unknown" absorbing state and trains policies that avoid leaving the trusted region of the model, yielding conservative improvement guarantees under suitable conditions **?**. COMBO combines conservative value regularization with model-based rollouts, coupling a pessimistic Q-learning objective with synthetic data generation **?**. MOBILE and related methods similarly incorporate uncertainty-aware penalties or pessimistic objectives to prevent exploitation of model errors. These approaches share the principle that safe model usage requires down-weighting or penalizing uncertain rollouts; they differ in the proxy used (uncertainty, conservative Q-regularization, absorbing states) and in whether they explicitly constrain policy shift relative to the behavior distribution. Our formulation separates *model mismatch* and *policy shift* as two constraints, each with its own budget and dual variable, which allows the algorithm to be conservative only along the failure mode that is empirically active.

Constrained RL and primal–dual methods. Primal–dual methods are standard for constrained RL in online settings, where one optimizes return subject to costs, safety constraints, or divergence constraints **????**. The algorithmic pattern—a Lagrangian with dual ascent on constraint violations—is well understood, and convergence can be shown in tabular or convex settings via occupancy-measure formulations and stochastic approximation. Our work reuses this classical mechanism but adapts it to the offline, model-based regime where constraints must be *estimated* from fixed data and model rollouts, and where the relevant constraints are not costs observed in the environment but divergences quantifying distribution shift. In particular, the transition constraint depends on the mismatch between p and m along d^{π}, and the policy constraint depends on π_b, which is unknown. Thus, the key technical issue is not the primal–dual update per se, but the interaction between dual dynamics and imperfect classifier-based estimators, which we make explicit in our feasibility and near-optimality guarantees.

Pessimism, confidence bounds, and PEVI-style analyses. From a theoretical perspective, offline RL guarantees are frequently obtained by constructing pessimistic value estimates (or lower confidence bounds) that hold uniformly over a function class, often via variants of pessimistic value iteration (PEVI) and concentrability assumptions. Such analyses yield minimax-optimal statistical rates (up to logarithmic factors) and clarify the necessity of coverage/overlap conditions for any nontrivial guarantee. Our bounds are consistent with this literature in that the dominant statistical term scales as $\widetilde{O}(H/\sqrt{n})$ in the finite-horizon view, and we explicitly include an additional $O(H\delta)$ term capturing discriminator log-ratio error. Rather than designing an explicit confidence interval, we impose budgets on divergences that operationalize a similar principle: avoid regions where either the model or the policy extrapolates beyond what the dataset can support.

Domain adaptation and reward correction (DARC). Finally, our transition-ratio correction is conceptually related to domain adaptation methods that reweight samples by density ratios, as well as to reward-correction approaches such as DARC, which adjust rewards using classifier-based estimates of mismatch between source and target dynamics **?**. DARC can be interpreted as shaping reward so that optimizing in an approximate dynamics model better matches performance in the true environment under certain assumptions. Our approach differs in two ways. First, we simultaneously address dynamics mismatch and policy shift, since in offline RL the learned policy itself induces the shift that renders model error consequential. Second, we treat correction magnitudes as dual variables driven by explicit budgets, rather than fixed hyperparameters, which aligns the correction with the Lagrangian of a constrained optimization problem and provides a mechanism for automatic conservatism adjustment.

3 Clean Problem Formulation: Constrained Shift-Robust Offline RL

We formalize the objective of learning a deployable policy from a fixed offline dataset while controlling two distinct failure modes: (i) exploitation of dynamics model bias when using synthetic rollouts, and (ii) extrapolation to actions insufficiently supported by the behavior data. Throughout, we consider a discounted MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p, r, \mu_0, \gamma)$ with bounded rewards $r(s,a) \in [r_{\min}, r_{\max}]$ and $r_{\min} > 0$, an offline dataset $D_{\text{env}} = \{(s_i, a_i, r_i, s'_i)\}_{i=1}^n$ collected under an unknown behavior policy π_b, and a learned dynamics model $m(\cdot \mid s,a)$ trained on D_{env}.

Discounted occupancy and evaluation target. For a policy π, let p^π denote the trajectory distribution induced by p and π, and define the discounted occupancy measure over state–action pairs by

$$d^\pi(s,a) \;:=\; (1-\gamma) \sum_{t \geq 0} \gamma^t \Pr_{p^\pi}(s_t = s, a_t = a).$$

Our true performance objective is the discounted return

$$J_{\mathcal{M}}(\pi) \;:=\; \mathbb{E}_{\tau \sim p^\pi}\Big[\sum_{t \geq 0} \gamma^t r(s_t, a_t)\Big].$$

We will occasionally use the effective horizon $H \approx (1-\gamma)^{-1}$ as a shorthand for the scale at which per-step perturbations (e.g., reward shaping or ratio errors) accumulate into value differences.

Two constraints: dynamics mismatch and policy shift. We restrict attention to policies that remain within user-specified budgets for (a) model mismatch and (b) policy shift. Fix divergences D_f and D_g acting on conditional distributions, and define

$$c_m(\pi) := \mathbb{E}_{(s,a)\sim d^\pi}\Big[D_f\big(p(\cdot \mid s,a) \,\|\, m(\cdot \mid s,a)\big)\Big],$$
$$c_\pi(\pi) := \mathbb{E}_{s \sim d^\pi}\Big[D_g\big(\pi(\cdot \mid s) \,\|\, \pi_b(\cdot \mid s)\big)\Big].$$

Given budgets $\varepsilon_m \geq 0$ and $\varepsilon_\pi \geq 0$, we define the feasible set

$$\Pi_{\text{feas}}(\varepsilon_m, \varepsilon_\pi) \;:=\; \Big\{\pi \in \Pi : \; c_m(\pi) \leq \varepsilon_m, \;\; c_\pi(\pi) \leq \varepsilon_\pi\Big\},$$

and the constrained offline RL problem

$$\max_{\pi \in \Pi_{\text{feas}}(\varepsilon_m, \varepsilon_\pi)} J_{\mathcal{M}}(\pi).$$

The first constraint is explicitly *shift-weighted*: it measures model error along the occupancy induced by π in the *true* environment, rather than along the behavior occupancy. This choice is deliberate, since model exploitation is only harmful insofar as π visits state–action pairs where m is inaccurate. The second constraint similarly measures shift where it matters for deployment, namely under d^{π}.

Choice of divergences and interpretation. The formulation permits several natural instantiations. For $D_f = D_g = \mathrm{KL}$, the constraints become expectations of conditional KL divergences, which admit variational and density-ratio interpretations and connect directly to additive log-ratio reward shaping. If one instead uses total variation, $D(\nu\|\mu) = \|\nu-\mu\|_{\mathrm{TV}}$, then $c_m(\pi)$ upper-bounds discrepancies in one-step transition expectations of bounded test functions, and $c_\pi(\pi)$ enforces a form of action support overlap. More generally, one may take D_f or D_g to be an integral probability metric (IPM) induced by a critic class $\mathcal{F}$, yielding constraints of the form

$$\sup_{f\in\mathcal{F}} \mathbb{E}_{x\sim\nu}[f(x)] - \mathbb{E}_{x\sim\mu}[f(x)].$$

IPM choices are attractive when likelihood ratios are ill-conditioned, whereas KL is attractive when calibrated log-ratios are available and one seeks a direct Lagrangian decomposition into per-transition penalties.

Measurability from offline data: what can be estimated. The constrained problem is stated in terms of p and π_b, neither of which is known. Our access is limited to (i) samples from the joint distribution induced by π_b and p (namely D_{env}), and (ii) samples from model rollouts under m and the current policy π (namely D_m). Consequently, we require divergence choices that admit estimators from these sample sources.

For the dynamics constraint, note that D_{env} provides samples of (s,a,s') distributed approximately as $(s,a) \sim d^{\pi_b}$ and $s' \sim p(\cdot \mid s,a)$, whereas D_m provides samples with $s' \sim m(\cdot \mid s,a)$ for (s,a) encountered along rollouts that start from dataset states and then follow π under m. When D_f is an f-divergence (including KL), $D_f(p(\cdot \mid s,a)\|m(\cdot \mid s,a))$ admits a variational representation that can be optimized by a classifier separating samples from $p(\cdot \mid s,a)$ and $m(\cdot \mid s,a)$. In the KL case, the optimal logit recovers $\log \rho_p(s,a,s') = \log \frac{p(s'|s,a)}{m(s'|s,a)}$ on the region where both densities are positive, enabling an empirical proxy for $c_m(\pi)$ by averaging a surrogate loss (or an explicit plug-in KL estimate) over (s,a) encountered under the current policy.

For the policy-shift constraint, $\pi_b(\cdot \mid s)$ is unknown, but D_{env} contains action samples drawn from it. When D_g is KL, we may rewrite

$$D_{\mathrm{KL}}\big(\pi(\cdot \mid s)\|\pi_b(\cdot \mid s)\big) = \mathbb{E}_{a\sim\pi(\cdot|s)}\Big[\log \frac{\pi(a \mid s)}{\pi_b(a \mid s)}\Big],$$

so that estimating $\log \rho_\pi(s,a) = \log \frac{\pi(a|s)}{\pi_b(a|s)}$ suffices. This log-ratio can be obtained via an action discriminator trained to distinguish (s,a) pairs produced by sampling $a \sim \pi(\cdot \mid s)$ (with s drawn from an appropriate state marginal) from those in D_{env}. For IPM-style D_g, an analogous critic can be trained to maximize discrepancy between $\pi(\cdot \mid s)$ samples and dataset actions.

Support and absolute continuity. Both constraints implicitly encode a requirement of overlap. For ratio-based instantiations (notably KL), ρ_p and ρ_π are defined only when p is absolutely continuous with respect to m (conditionally) and π is absolutely continuous with respect to π_b (conditionally). If $\pi_b(a \mid s) = 0$ while $\pi(a \mid s) > 0$, then $D_g(\pi(\cdot \mid s)\|\pi_b(\cdot \mid s)) = +\infty$ for KL, excluding such policies from Π_{feas} regardless of budget. This is not an artifact but a formal expression of the offline identifiability barrier: leaving dataset support cannot be certified without additional assumptions.

Discounted-to-finite-horizon translation. Finally, we remark that the above discounted constraints and objective admit the standard conversion to a finite-horizon viewpoint by interpreting $(1-\gamma)d^\pi$ as a normalized visitation distribution and using $H \approx (1-\gamma)^{-1}$ to track accumulation. In particular, per-step estimation errors in the discriminators that are uniformly bounded in logit translate into $O(H\delta)$ perturbations in the induced shaped objective and, correspondingly, into additive $O(H\delta)$ terms in feasibility and suboptimality bounds. This correspondence will be used implicitly when we state rates in either discounted or finite-horizon forms.

4 4. Clean Problem Formulation (Constrained Shift-Robust Offline RL): define feasible set via model-bias and policy-shift constraints; discuss choices of D_f, D_g (KL/TV/IPM) and measurability from data.

Beyond serving as a formal safety specification, the pair of constraints induces a *geometry* on the policy class that we will exploit algorithmically. In particular, ε_m and ε_π control two different directions in which offline optimization can fail: the first limits the degree to which the optimized policy may rely on regions where the learned simulator is inaccurate, while the second limits extrapolation in action space relative to the (unknown) data-generating policy. We emphasize that neither constraint is purely a property of the dataset; both are *policy dependent* through d^π, and hence must be enforced adaptively as π changes.

A basic modeling choice is whether D_f and D_g should be likelihood-based divergences (e.g. KL) or IPM-type discrepancies. When $D_f = D_g = \mathrm{KL}$, the constraints can be written as expectations of log density ratios:

$$c_m(\pi) = \mathbb{E}_{(s,a)\sim d^\pi}\Big[\mathbb{E}_{s'\sim p(\cdot|s,a)}\Big[\log \frac{p(s' \mid s,a)}{m(s' \mid s,a)}\Big]\Big], \tag{1}$$

$$c_\pi(\pi) = \mathbb{E}_{s\sim d^\pi}\Big[\mathbb{E}_{a\sim\pi(\cdot|s)}\Big[\log \frac{\pi(a \mid s)}{\pi_b(a \mid s)}\Big]\Big]. \tag{2}$$

The appeal of (1)–(2) is twofold: (i) the quantities admit direct estimation via calibrated classifiers, and (ii) they decompose into per-transition and per-decision additive terms, which will later allow us to rewrite the Lagrangian as an RL objective with a shaped reward. The drawback is the implicit absolute-continuity requirement: whenever $m(s' \mid s,a) = 0 < p(s' \mid s,a)$ or $\pi_b(a \mid s) = 0 < \pi(a \mid s)$, the corresponding KL is infinite. In offline RL this is conceptually appropriate—it excludes policies whose value is unidentifiable from the available support—but it also means that practical implementations must take care that policy parameterizations and constraint surrogates do not silently step outside support.

If one instead takes total variation (TV), then the constraints become

$$c_m(\pi) = \mathbb{E}_{(s,a)\sim d^\pi}\big[\|p(\cdot \mid s,a) - m(\cdot \mid s,a)\|_{\mathrm{TV}}\big], \qquad c_\pi(\pi) = \mathbb{E}_{s\sim d^\pi}\big[\|\pi(\cdot \mid s) - \pi_b(\cdot \mid s)\|_{\mathrm{TV}}\big],$$

which enforce a stronger form of distributional proximity that does not require density ratios to be well-conditioned. TV admits the variational representation

$$\|\nu - \mu\|_{\mathrm{TV}} = \sup_{\|f\|_\infty \le 1} \frac{1}{2}\Big(\mathbb{E}_{x\sim\nu}[f(x)] - \mathbb{E}_{x\sim\mu}[f(x)]\Big),$$

and hence can be estimated via a critic class that approximates the supremum. More generally, we may use an integral probability metric induced by a function class $\mathcal{F}$, giving

$$D_{\mathcal{F}}(\nu\|\mu) := \sup_{f\in\mathcal{F}} \mathbb{E}_\nu[f] - \mathbb{E}_\mu[f],$$

which interpolates between TV (via bounded $\mathcal{F}$) and weaker moment-matching constraints (via restricted $\mathcal{F}$). The price paid by IPM-style choices is that they typically do not yield an exact log-ratio decomposition; consequently, the corresponding Lagrangian penalties will be representable as learned critics rather than explicit ratio terms.

The estimability of $c_m(\pi)$ from (D_{env}, D_m) hinges on a sampleable contrast between p and m at comparable (s,a). Concretely, D_{env} provides conditionally real next-states $s' \sim p(\cdot \mid s,a)$ paired with the dataset (s,a), whereas D_m provides synthetic $s' \sim m(\cdot \mid s,a)$ along rollouts initiated from dataset states and then propagated under (m, π). This is not an innocuous detail: the distribution over (s, a) in D_m depends on both the rollout horizon and the policy, so any learned discriminator is implicitly trained on the *policy-induced* region that we aim to constrain. For f-divergences, and in particular KL, we may train a binary classifier $C_\phi(s, a, s') \in (0, 1)$ to distinguish labeled samples $(s, a, s') \sim D_{\text{env}}$ from $(s, a, s') \sim D_m$. Under standard calibration assumptions, the logit

$$\widehat{\ell}_\phi(s, a, s') := \log \frac{C_\phi(s, a, s')}{1 - C_\phi(s, a, s')}$$

approximates $\log \rho_p(s, a, s') = \log \frac{p(s'|s,a)}{m(s'|s,a)}$ on the support where both distributions place mass. Averaging $\widehat{\ell}_\phi$ over the *appropriate* (s, a, s') distribution then yields a computable surrogate for (1). The key point is that, because the constraint is weighted by d^π, the relevant average is not over D_{env} alone, but over the state–action pairs that π would visit. In practice, we approximate this weighting by sampling (s, a) from rollouts (under m) starting at dataset states; this corresponds to the standard MBPO-style approximation of d^π and will be the source of a controlled modeling bias term in later bounds.

An analogous construction applies to $c_\pi(\pi)$. Since π_b is unknown, we cannot evaluate $\log \pi_b(a \mid s)$ directly; however, we can estimate the ratio $\rho_\pi(s, a) = \pi(a \mid s)/\pi_b(a \mid s)$ by discriminating between (i) action samples drawn from the current policy and (ii) action samples in the dataset. Concretely, for a chosen state marginal ν (often taken to be the empirical state distribution from D_{env}, or a mixture of dataset states and model rollout states), we draw $(s, a) \sim (s \sim \nu,\ a \sim \pi(\cdot \mid s))$ and label them as policy-generated, and label (s, a) pairs from D_{env} as behavior-generated. Training a classifier $C_\psi(s, a)$ yields a logit $\widehat{\ell}_\psi(s, a)$ that approximates $\log \rho_\pi(s, a)$, and we may then estimate (2) by averaging $\widehat{\ell}_\psi$ over (s, a) sampled from $(s \sim \nu,\ a \sim \pi(\cdot \mid s))$. Here again the weighting matters: the constraint is stated under $s \sim d^\pi$, not under the dataset marginal, so the choice of ν is an approximation device whose adequacy depends on how well it tracks the deployed state distribution.

Finally, we record the operational role of the budgets. If ε_m is small, then any feasible policy must remain close to regions where the learned model is empirically indistinguishable from the environment under the discriminator family; if ε_π is small, then any feasible policy must remain close to the action choices supported by the dataset. These trade-offs will manifest in the dual variables of the Lagrangian: tightening a budget increases the marginal penalty for violating the associated constraint. In the next section we make this relationship explicit by passing from the constrained formulation to a shift-aware shaped reward whose coefficients are precisely the dual variables.

4.1 From constraints to shift-aware rewards

We pass from the constrained formulation to an unconstrained saddle problem by introducing dual variables $\alpha, \beta \geq 0$. Writing $c_m(\pi)$ and $c_\pi(\pi)$ for the two constraint functionals, we consider the Lagrangian

$$\mathcal{L}(\pi, \alpha, \beta) := J_{\mathcal{M}}(\pi) - \alpha\big(c_m(\pi) - \varepsilon_m\big) - \beta\big(c_\pi(\pi) - \varepsilon_\pi\big). \tag{3}$$

The constant offset $\alpha\varepsilon_m + \beta\varepsilon_\pi$ affects only the dual objective and will be ignored in the primal update; the essential term is the per-policy penalization $-\alpha c_m(\pi) - \beta c_\pi(\pi)$. When $D_f = D_g = \mathrm{KL}$, the KL expressions (1)–(2) suggest that $\mathcal{L}$ should admit a decomposition into additive contributions along trajectories. Our implementation, however, does not optimize $\mathcal{L}$ by explicitly computing inner expectations under $p(\cdot \mid s, a)$ and $\pi(\cdot \mid s)$; instead, we rewrite these terms so that they can be estimated from samples drawn from the two sources we actually have, namely model rollouts and behavior data. This is precisely where the "shift-aware reward" interpretation emerges.

To make the per-transition structure explicit, we introduce a utility transformation u, with the canonical choice $u(r) = \log r$, which is well-defined because $r_{\min} > 0$. We view the algorithm as optimizing the utility-return

$$J_u(\pi) := \mathbb{E}_{\tau\sim p^\pi}\Big[\sum_{t\geq 0} \gamma^t u(r(s_t, a_t))\Big],$$

and we later translate utility guarantees back to $J_{\mathcal{M}}(\pi)$ using standard inequalities (for instance, by Jensen and the bounds on r). The utility form is not cosmetic: it allows likelihood ratios to enter additively as log-ratios, matching the form produced by calibrated discriminators.

We now derive the corrected reward terms from a variational change-of-measure bound of the SAR type. Let q be any trajectory distribution absolutely continuous with respect to p^π. The Gibbs variational principle yields

$$\mathbb{E}_{\tau\sim p^\pi}\Big[F(\tau)\Big] \;\geq\; \mathbb{E}_{\tau\sim q}\Big[F(\tau)\Big] \;-\; \mathrm{KL}\big(q \,\|\, p^\pi\big), \tag{4}$$

for any bounded measurable F. Taking $F(\tau) = \sum_{t\geq 0} \gamma^t u(r(s_t, a_t))$ and choosing q as a distribution we can sample from yields a lower bound on $J_u(\pi)$ in which $\mathrm{KL}(q\|p^\pi)$ appears as a penalty. We choose q to be induced by the learned simulator m rather than the true dynamics, i.e., $q = m^\pi$ (the distribution of trajectories obtained by rolling out π under m, starting from μ_0 or, in practice, from dataset states). The log-likelihood ratio between m^π and p^π decomposes as a sum of one-step log ratios:

$$\log \frac{m^\pi(\tau)}{p^\pi(\tau)} = \sum_{t\geq 0} \log \frac{m(s_{t+1} \mid s_t, a_t)}{p(s_{t+1} \mid s_t, a_t)}.$$

Consequently,

$$\mathrm{KL}(m^{\pi}\|p^{\pi}) = \mathbb{E}_{\tau\sim m^{\pi}}\Big[\sum_{t\geq 0}\log\frac{m(s_{t+1}\mid s_t,a_t)}{p(s_{t+1}\mid s_t,a_t)}\Big] = -\mathbb{E}_{\tau\sim m^{\pi}}\Big[\sum_{t\geq 0}\log\rho_p(s_t,a_t,s_{t+1})\Big]. \tag{5}$$

Plugging (5) into (4) yields the SAR-style bound

$$J_u(\pi) \;\geq\; \mathbb{E}_{\tau\sim m^{\pi}}\Big[\sum_{t\geq 0}\gamma^t\Big(u(r(s_t,a_t)) + \alpha\,\log\rho_p(s_t,a_t,s_{t+1})\Big)\Big] \;-\; \alpha\cdot(\text{slack}), \tag{6}$$

where $\alpha \geq 0$ plays the role of a Lagrange multiplier weighting the mismatch term. Importantly, the expectation in (6) is under m^{π}, the distribution we can sample via synthetic rollouts, and the correction $\log\rho_p = \log\frac{p}{m}$ appears *additively* inside the sum. Since $\mathbb{E}_{m^{\pi}}[\log\rho_p] = -\mathrm{KL}(m^{\pi}\|p^{\pi}) \leq 0$, the added term acts as a pessimistic penalty whenever the simulator deviates from the environment.

A completely analogous device produces the policy-shift correction without requiring access to π_b. Consider the per-state action distributions. If we take q to be the behavior-induced action choice at visited states and p^{π} to be the current policy action choice, the log ratio $\log\rho_{\pi}(s,a) = \log\frac{\pi(a|s)}{\pi_b(a|s)}$ again enters additively. Operationally, we place the expectation over (s,a) under the behavior distribution (dataset samples), obtaining

$$\mathbb{E}_{(s,a)\sim d^{\pi_b}}\big[\log\rho_{\pi}(s,a)\big] = -\mathbb{E}_{s\sim d^{\pi_b}}\big[\mathrm{KL}(\pi_b(\cdot\mid s)\|\pi(\cdot\mid s))\big] \;\leq\; 0, \tag{7}$$

so that adding $+\beta\log\rho_{\pi}$ on dataset transitions also produces a conservative correction. This expectation placement is the reason we can treat the behavior samples as "anchors" while still penalizing action extrapolation: although $\mathrm{KL}(\pi(\cdot\mid s)\|\pi_b(\cdot\mid s))$ is the conceptual budget, the surrogate term we can stably estimate from data is the reverse-KL form induced by sampling from π_b, which suffices to discourage leaving the dataset support.

Collecting the two pieces, we obtain a shaped reward of the form

$$\tilde{r}_{\alpha,\beta}(s,a,s') = u(r(s,a)) \;+\; \alpha\,\widehat{\log\rho_p}(s,a,s') \;+\; \beta\,\widehat{\log\rho_{\pi}}(s,a), \tag{8}$$

with the understanding that $\widehat{\log\rho_p}$ is trained on (D_{env},D_m) and used on model-generated transitions, whereas $\widehat{\log\rho_{\pi}}$ is trained on policy-vs-behavior action samples and used on dataset transitions. Equation (8) is exactly the SAR correction, with coefficients α,β now interpreted as dual variables enforcing the two budgets. In particular, when the learned policy attempts to exploit regions where the discriminator suggests m is optimistic (small p/m), or selects actions rarely supported by the dataset (small π/π_b on behavior samples), the corresponding log-ratio becomes strongly negative and decreases $\tilde{r}_{\alpha,\beta}$, mimicking pessimism-style penalties used in model-based offline RL. The present derivation differs from ad hoc pessimism in that the penalty magnitudes are not tuned by hand: they are the multipliers of an explicit constrained problem and will be updated online to match the desired budgets.

Finally, we note that if one replaces KL by an IPM-type divergence, then $\widehat{\log \rho}$ in (8) is replaced by a learned critic $f \in \mathcal{F}$ realizing the variational supremum, and the same primal interpretation holds: for fixed dual variables, maximizing the Lagrangian reduces to standard RL on a modified per-transition reward. This is the precise sense in which the constraints induce shift-aware reward shaping, and it is the quantity on which we perform actor–critic updates in the algorithm that follows.

4.2 Dual-SAR: model rollouts, corrected-reward RL, and dual adaptation

We now specify the procedure by which we (i) construct training data that reflects the current policy while remaining anchored to the offline dataset, (ii) compute the shift-aware corrected reward used for policy improvement, and (iii) update the dual variables so as to target the budgets $(\varepsilon_m, \varepsilon_\pi)$ without manual tuning. The implementation is deliberately MBPO-style: we alternate between short synthetic rollouts under the learned model and off-policy actor–critic updates on a mixture of real and synthetic transitions.

Rollout generation (MBPO-style). At iteration k, we sample a minibatch of states from the empirical state marginal in D_{env} (equivalently, we sample transitions and take their s-components). From each sampled state s_0, we roll out the current policy π_k in the learned dynamics m for a short horizon h, producing synthetic transitions (s_t, a_t, r_t, s_{t+1}) where $a_t \sim \pi_k(\cdot \mid s_t)$ and $s_{t+1} \sim m(\cdot \mid s_t, a_t)$. We store these transitions in a synthetic replay buffer D_m. The restriction to short h is not merely computational: it limits compounding model bias and ensures that the distribution of D_m remains close to the dataset support in early iterations. In practice, we either fix h to a small constant or use an increasing schedule h_k (as in MBPO) once the discriminators indicate reduced mismatch.

Discriminators and log-ratio surrogates. We maintain two classifiers. The transition discriminator $C_\phi(s, a, s') \in (0, 1)$ is trained to distinguish environment transitions from model transitions, with binary cross-entropy objective

$$\max_{\phi} \ \mathbb{E}_{(s,a,s')\sim D_{\text{env}}}\big[\log C_\phi(s,a,s')\big] + \mathbb{E}_{(s,a,s')\sim D_m}\big[\log(1 - C_\phi(s,a,s'))\big].$$

When the discriminator is calibrated, the logit yields an estimate of the one-step log dynamics ratio (up to an additive constant that cancels in policy improvement),

$$\widehat{\log \rho}_p(s, a, s') \;\approx\; \log \frac{p(s' \mid s, a)}{m(s' \mid s, a)} = \log \frac{C_\phi(s, a, s')}{1 - C_\phi(s, a, s')}.$$

Similarly, the action discriminator $C_\psi(s, a) \in (0, 1)$ is trained to distinguish state–action pairs proposed by the current policy from those in the dataset:

$$\max_\psi \; \mathbb{E}_{(s,a)\sim D_{\pi_k}} \big[\log C_\psi(s, a)\big] + \mathbb{E}_{(s,a)\sim D_{\text{env}}} \big[\log(1 - C_\psi(s, a))\big],$$

where D_{π_k} denotes state–action samples obtained by pairing dataset states s with actions $a \sim \pi_k(\cdot \mid s)$. The calibrated logit yields $\widehat{\log \rho}_\pi(s, a) \approx \log \frac{\pi_k(a|s)}{\pi_b(a|s)}$. We emphasize that these estimators are only used on the distributions on which the corresponding discriminator is trained; this restriction is essential for both stability and the high-probability error control invoked later in the tabular analysis.

Corrected reward and actor–critic update. Given the current dual variables (α_k, β_k), we label transitions with the corrected reward $\tilde{r}_{\alpha_k,\beta_k}$ as in (8), but applied in a source-aware manner: on synthetic transitions $(s, a, s') \in D_m$ we include the model-mismatch term $\alpha_k \widehat{\log \rho}_p(s, a, s')$, while on real transitions $(s, a) \in D_{\text{env}}$ we include the policy-shift term $\beta_k \widehat{\log \rho}_\pi(s, a)$. We then run an off-policy actor–critic update (e.g., SAC) on minibatches drawn from $D_{\text{env}} \cup D_m$, treating $\tilde{r}_{\alpha_k,\beta_k}$ as the reward. This step is a standard RL update with a nonstationary reward function; the only algorithmic novelty is that the reward is shaped by discriminator outputs and dual variables, and that the data distribution is a controlled mixture of real and model transitions.

Constraint monitoring and dual ascent. To adapt (α, β) to the user budgets, we compute empirical surrogates $\widehat{c}_m(\pi_{k+1})$ and $\widehat{c}_\pi(\pi_{k+1})$ from discriminator outputs, for instance by using sample averages of negative log-ratios (reverse-KL surrogates) or by using the corresponding variational divergences associated with the discriminator training objective. We then perform projected dual ascent,

$$\alpha_{k+1} = \big[\alpha_k + \eta_\alpha\big(\widehat{c}_m(\pi_{k+1}) - \varepsilon_m\big)\big]_+, \qquad \beta_{k+1} = \big[\beta_k + \eta_\beta\big(\widehat{c}_\pi(\pi_{k+1}) - \varepsilon_\pi\big)\big]_+.$$

Thus, persistent constraint violation increases the multiplier, thereby strengthening the pessimistic correction in subsequent policy updates; conversely, when the constraint is satisfied with slack, the multiplier decreases toward 0, reducing conservatism. In deployments where oscillations are undesirable, we may use a damped or proximal dual update (e.g., replacing $[\cdot]_+$ by projection onto $[0, \alpha_{\max}]$ and $[0, \beta_{\max}]$, or adding a quadratic penalty) without changing the conceptual role of the multipliers.

Stabilization: projection, clipping, and logit certification. Three practical mechanisms are required to make the above loop numerically stable. First, we always project dual iterates to enforce $\alpha_k, \beta_k \geq 0$, and we optionally cap them by known bounds to prevent excessively large reward perturbations. Second, because discriminator logits can be heavy-tailed early in training, we clip the additive correction terms:

$$\alpha_k \widehat{\log \rho_p} \in [-L, L], \qquad \beta_k \widehat{\log \rho_\pi} \in [-L, L],$$

so that Bellman backups remain well-conditioned and the corrected rewards are uniformly bounded. Third, we maintain a simple calibration/certification routine: we reserve held-out splits from D_{env} and D_m, and we monitor discriminator calibration error; when the held-out logit error exceeds a tolerance, we (temporarily) reduce rollout horizon h, increase discriminator training, or downweight the offending correction term. This operationally enforces the bounded log-ratio error regime assumed in the theory and prevents the algorithm from reacting to discriminator artifacts.

The output of Dual-SAR is either the final iterate π_K or an average of iterates, together with feasibility diagnostics reporting $\widehat{c}_m(\pi_K)$ and $\widehat{c}_\pi(\pi_K)$ and the corresponding multipliers. In the next section we analyze this procedure in the tabular setting by decomposing performance into (i) statistical error from finite D_{env}, (ii) optimization error from approximate actor–critic and finite iterations, and (iii) ratio-estimation error from imperfect discriminators.

4.3 Theory I (Tabular MDPs): feasibility and near-optimality with matching lower bounds

We analyze Dual-SAR in the finite (tabular) setting, where $|\mathcal{S}| < \infty$, $|\mathcal{A}| < \infty$, rewards are bounded with $r_{\min} > 0$, and policies are arbitrary distributions over $\mathcal{A}$ for each s. We adopt the discounted occupancy measure

$$d^\pi(s, a) \;:=\; (1 - \gamma) \sum_{t \geq 0} \gamma^t \Pr_{p,\pi}(s_t = s, a_t = a),$$

so that $J_{\mathcal{M}}(\pi) = \frac{1}{1-\gamma} \mathbb{E}_{(s,a)\sim d^\pi}[r(s, a)]$. The constraints are functionals of d^π and the one-step transition kernels, and in the tabular case the occupancy-measure formulation makes explicit the convex–concave saddle structure that motivates primal–dual updates.

Assumptions (coverage and bounded estimation error). We require standard overlap/coverage conditions ensuring that the feasible set is statistically identifiable from D_{env}. Concretely, we assume a concentrability-type bound: there exists $C < \infty$ such that for all π considered (in particular for

π^*_{feas} and the iterates of Dual-SAR),

$$\sup_{s,a} \frac{d^{\pi}(s,a)}{d^{\pi_b}(s,a)} \;\leq\; C \qquad \text{on the support where } d^{\pi_b}(s,a) > 0,$$

and the constraints are only enforced/evaluated on this support. We further assume discriminator-based log-ratio surrogates satisfy uniform (or high-probability) bounds on the training distributions:

$$\left|\widehat{\log \rho}_p(s,a,s') - \log \rho_p(s,a,s')\right| \leq \delta_p, \qquad \left|\widehat{\log \rho}_\pi(s,a) - \log \rho_\pi(s,a)\right| \leq \delta_\pi,$$

and we denote $\delta := \max(\delta_p, \delta_\pi)$. Finally, we isolate optimization error by allowing the primal RL update (actor–critic) and the discriminator updates to be approximate, summarized by a residual ε_{opt} (for instance, a bound on the achieved primal–dual gap relative to an ideal saddle point for the empirical problem).

From constrained control to a saddle-point problem. Using a variational representation for the chosen divergences (e.g., KL) and the occupancy-measure constraints, the constrained objective

$$\max_{\pi} \; J_{\mathcal{M}}(\pi) \quad \text{s.t.} \quad c_m(\pi) \leq \varepsilon_m, \;\; c_\pi(\pi) \leq \varepsilon_\pi$$

admits a Lagrangian $\mathcal{L}(\pi, \alpha, \beta)$ that is concave in (α, β) and (in the tabular occupancy parameterization) convex in the primal variable. The essential identity for Dual-SAR is that, up to constants independent of π, the Lagrangian corresponds to an RL problem with per-transition reward

$$\tilde{r}_{\alpha,\beta} \;=\; u(r(s,a)) + \alpha \log \frac{p(s' \mid s,a)}{m(s' \mid s,a)} + \beta \log \frac{\pi(a \mid s)}{\pi_b(a \mid s)},$$

with the understanding that in the algorithm we substitute discriminator logits for log-ratios and apply the terms on the sources where they are estimable. Thus, primal improvement corresponds to maximizing $\mathbb{E}[\sum_t \gamma^t \tilde{r}_{\alpha,\beta}]$, while dual ascent increases α or β when the associated constraint surrogate exceeds its budget.

Feasibility with imperfect ratios. Let $(\hat{\pi}, \hat{\alpha}, \hat{\beta})$ denote the output of Dual-SAR after T iterations (or an average iterate), and let $\widehat{c}_m, \widehat{c}_\pi$ denote the empirical constraint surrogates used for dual updates. Under bounded reward shaping (via clipping) and bounded logit error δ, the divergence functionals are Lipschitz with respect to log-ratio perturbations on the relevant supports. Consequently, if $\widehat{c}_m(\pi)$ and $\widehat{c}_\pi(\pi)$ are $\pm\Delta$ accurate uniformly over the iterates (with probability at least $1 - \xi$), then complementary slackness for an approximate saddle point yields

$$c_m(\hat{\pi}) \;\leq\; \varepsilon_m + O(\Delta + \delta_p), \qquad c_\pi(\hat{\pi}) \;\leq\; \varepsilon_\pi + O(\Delta + \delta_\pi),$$

with constants depending on the effective horizon $H \approx \frac{1}{1-\gamma}$ and on the clipping level used to keep $\tilde{r}_{\alpha,\beta}$ bounded. In particular, feasibility degrades additively with ratio-estimation error: even if the primal–dual loop converges perfectly on the empirical problem, the true constraints can only be satisfied up to the error with which we can estimate the relevant log-ratios on-distribution.

Near-optimality among feasible policies and an explicit error decomposition. Let π^*_{feas} be an optimal policy among those satisfying the true budgets. The performance bound we target has the form

$$J_{\mathcal{M}}(\pi^*_{\text{feas}}) - J_{\mathcal{M}}(\hat{\pi}) \le \underbrace{\widetilde{O}\left(\frac{H}{\sqrt{n}}\right)}_{\text{statistical}} + \underbrace{O(H\delta)}_{\text{ratio-estimation}} + \underbrace{\varepsilon_{\text{opt}}}_{\text{optimization}} . \quad (9)$$

The first term is the intrinsic offline statistical error: even with exact models and exact ratios, n transitions limit how well we can evaluate and optimize policies under distribution shift. The second term isolates the effect of imperfect discriminators (or, more generally, misspecified ratio estimators), which enters linearly in horizon due to the accumulation of per-step reward perturbations in $\sum_t \gamma^t \tilde{r}$. The third term captures finite-iteration and function-approximation artifacts in the actor–critic and discriminator training, and can be driven down with additional computation in the tabular case (e.g., exact dynamic programming as the primal update).

A proof proceeds by (i) relating the achieved primal–dual gap to a value gap in the occupancy formulation, (ii) controlling empirical process error for the objectives and constraints via concentration (yielding the $\widetilde{O}(H/\sqrt{n})$ term under coverage), and (iii) perturbation analysis translating ℓ_∞ logit error bounds into additive deviations in the shaped return and in the constraint estimates (yielding $O(H\delta)$). The only role of model rollouts in this argument is algorithmic: they change the sampling distribution used by the primal update but do not bypass the information limit imposed by D_{env}.

Matching-rate lower bounds and the necessity of coverage. The rate $\widetilde{O}(H/\sqrt{n})$ in (9) is unimprovable in the worst case (up to logarithmic factors) under the same offline access model. A standard reduction embeds a contextual bandit as an $H = 1$ MDP; any algorithm achieving $o(1/\sqrt{n})$ suboptimality uniformly would contradict minimax lower bounds for offline policy optimization/off-policy evaluation with bounded rewards and limited overlap. Extending this reduction along a finite-horizon chain yields an $\Omega(H/\sqrt{n})$ lower bound.

Moreover, without overlap the problem is ill-posed: if a state–action pair has positive occupancy under π^*_{feas} but zero probability under the data distribution induced by π_b, then two environments can be constructed that agree on the dataset support yet assign arbitrarily different outcomes off-support. No algorithm using only D_{env} (and models trained on it) can then guarantee nontrivial improvement. In this sense, Dual-SAR does not eliminate the need for coverage assumptions; rather, the budgets $(\varepsilon_m, \varepsilon_\pi)$ and their dual adaptation provide an operational mechanism to remain within the regime where offline generalization is statistically controlled.

4.4 Theory II (Linear MDPs / Function Approximation): realizability, concentrability, and what fails beyond the linear regime

We next summarize how the preceding feasibility and near-optimality statements extend when we replace the tabular parameterization by a linear function class. The purpose of this section is not to optimize constants, but to separate (i) the statistical terms that follow from finite-sample identification under coverage and realizability from (ii) the genuinely algorithmic aspects of Dual-SAR (model rollouts, discriminators, and dual updates). Throughout we retain the discounted setting and write $H \asymp (1-\gamma)^{-1}$ for the effective horizon.

Linear MDP / linear value realizability. Fix a feature map $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ with $\|\phi(s,a)\|_2 \leq 1$. We assume the reward is linear,

$$r(s,a) = \langle \phi(s,a), \theta_r \rangle, \qquad \|\theta_r\|_2 \leq B_r,$$

and the transition kernel is linear in the standard "linear MDP" sense: there exist signed measures $\{P_i(\cdot)\}_{i=1}^d$ such that

$$p(\cdot \mid s,a) = \sum_{i=1}^d \phi_i(s,a)\, P_i(\cdot), \qquad \left\| \sum_{i=1}^d \phi_i(s,a)\, P_i(\cdot) \right\|_{\text{TV}} \leq 1.$$

Under these assumptions, for any policy π the Bellman equations admit a Q^π that is linear in ϕ (or, more generally, lies in a linear class that is Bellman-complete). This is the realizability condition that allows us to import known offline RL rates for fitted value iteration / least-squares temporal difference methods, and it is the precise point at which we depart from general nonlinear approximation.

Coverage in linear feature space. The tabular overlap condition is replaced by a feature-covariance condition on the data distribution induced by π_b. Let $\Sigma := \mathbb{E}_{(s,a)\sim d^{\pi_b}}[\phi(s,a)\phi(s,a)^\top]$. We assume a nondegeneracy condition

$$\lambda_{\min}(\Sigma) \;\geq\; \lambda_0 > 0,$$

and a concentrability-type bound ensuring that, for all policies π under consideration, feature second moments under d^{π} are controlled by those under d^{π_b}. One convenient sufficient form is: there exists $\kappa < \infty$ such that for all vectors $w \in \mathbb{R}^d$,

$$\mathbb{E}_{(s,a)\sim d^{\pi}}\left[\langle w, \phi(s,a)\rangle^2\right] \;\le\; \kappa\, \mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\langle w, \phi(s,a)\rangle^2\right].$$

In linear problems, such an assumption is the direct analogue of bounding density ratios in tabular problems; it is also what allows empirical least-squares objectives computed on D_{env} to control errors under the shifted occupancy d^{π}.

Near-optimality with linear statistical rates. Under linear realizability and the above covariance/concentrability assumptions, one obtains a value suboptimality guarantee whose statistical term scales with the feature dimension. In particular, for an appropriate implementation of the primal update (e.g., a pessimistic or regularized fitted Q-iteration in the linear class, driven by the corrected reward $\tilde{r}_{\alpha,\beta}$), we can bound the gap to the best feasible policy π^*_{feas} by

$$J_{\mathcal{M}}(\pi^*_{\text{feas}}) - J_{\mathcal{M}}(\hat{\pi}) \;\le\; \widetilde{O}\!\left(\frac{H\sqrt{d}}{\sqrt{n}}\right) \;+\; O(H\delta) \;+\; \varepsilon_{\text{opt}} \;+\; \varepsilon_{\text{approx}}. \tag{10}$$

Here $\varepsilon_{\text{approx}}$ is identically zero under exact linear realizability/Bellman completeness, but we keep it explicit because it is the term that dominates when the linear model is misspecified. The $\widetilde{O}(H\sqrt{d/n})$ rate is representative of least-squares-based offline control bounds in linear MDPs; alternative analyses yield $\widetilde{O}(H^2 d/n)$ or $\widetilde{O}(\sqrt{H^3 d/n})$ depending on the algorithmic template and on whether one targets evaluation or control. For our purposes, the salient point is that Dual-SAR does not change the dimension dependence: the role of the dual correction is to restrict optimization to a region where such linear generalization bounds apply (via the budgets), not to evade the need for coverage.

Feasibility bounds with function approximation. The feasibility argument carries over provided the constraint surrogates remain Lipschitz with respect to the estimated log-ratios on the relevant support and provided the policy/value learning procedure does not drive the iterates into regions where the discriminators are unconstrained. Concretely, under the same uniform logit error bounds $|\widehat{\log \rho_p} - \log \rho_p| \le \delta_p$ and $|\widehat{\log \rho_\pi} - \log \rho_\pi| \le \delta_\pi$ on the data and model rollout distributions, and assuming the empirical constraint estimates are accurate up to $\pm\Delta$, we obtain with probability at least $1-\xi$,

$$c_m(\hat{\pi}) \le \varepsilon_m + O(\Delta + \delta_p), \qquad c_\pi(\hat{\pi}) \le \varepsilon_\pi + O(\Delta + \delta_\pi),$$

with constants depending on H and the clipping used for $\tilde{r}_{\alpha,\beta}$. The only additional subtlety in the linear setting is that the primal update typically uses function approximation and thus introduces a controlled bias term; this bias is absorbed into ε_{opt} (algorithmic inexactness) and $\varepsilon_{\text{approx}}$ (model/class misspecification), and it impacts feasibility indirectly by changing the realized occupancy $d^{\hat{\pi}}$.

Why the analysis breaks in general nonlinear settings. Beyond linear/Bellman-complete classes, two distinct failure modes appear. First, *approximation bias* becomes structural: if the value class is not closed under the Bellman operator, then even with infinite data the primal update may converge to a fixed point with nonzero Bellman error, which manifests as a nonvanishing $\varepsilon_{\text{approx}}$ term in (10). In offline control, such bias interacts adversely with distribution shift because errors concentrate precisely on state–action regions that the learned policy visits but the dataset does not cover. Second, *optimization and saddle-point pathologies* emerge: the primal problem is no longer convex in an occupancy parameterization induced by the function class, and the discriminator/policy training objectives are nonconvex–nonconcave. In that regime, projected primal–dual updates need not converge to any meaningful saddle point, and complementary slackness cannot be invoked without additional regularity assumptions (e.g., two-time-scale stability, PL conditions, or specific network architectures). Thus, while Dual-SAR remains a sensible algorithmic template in nonlinear domains, the clean separation into statistical error $\widetilde{O}(\cdot)$, ratio-estimation error $O(H\delta)$, and optimization/approximation errors becomes only a heuristic unless one imposes stronger conditions (such as realizability of both critics and discriminators and a form of smoothness guaranteeing stable dual dynamics).

Takeaway. In the linear regime, the tabular theory transfers with the usual replacement "$|\mathcal{S}||\mathcal{A}|$" $\mapsto$ d and "exact dynamic programming" $\mapsto$ "least-squares Bellman methods", while the role of the budgets and dual variables is unchanged: they restrict the learned policy to remain in the identifiable region determined by D_{env} and by the accuracy of the ratio estimators. In the absence of linear realizability (or an alternative completeness property), the dominant obstruction is not the particular form of Dual-SAR but the unidentifiable nature of off-support generalization under nonlinear approximation.

4.5 Complexity and Implementation Notes

Compute and memory costs. Dual-SAR adds two discriminator updates and two scalar dual updates to an otherwise standard model-based offline RL loop. Let b denote the number of model rollouts per iteration, h the model rollout horizon, U_d the number of discriminator gradient steps per iteration, U_{ac} the number of actor–critic gradient steps per iteration, and let C_m, C_d, C_{ac} denote the per-sample forward/backward costs of the dynamics model, discriminators, and actor–critic networks, respectively. Then the dominant per-iteration time complexity is

$$O(b\,h\,C_m) \;+\; O(U_d\,B\,C_d) \;+\; O(U_{ac}\,B\,C_{ac}),$$

where B is the minibatch size used for learning. The dual updates for (α, β) and the computation of corrected rewards are $O(B)$ and are negligible. Relative to MBPO-style training, Dual-SAR's overhead is essentially the cost of training the two classifiers (and optionally calibrating them). In our experience, this overhead is comparable to training an additional critic and is typically dominated by actor–critic updates when U_{ac} is large. The principal knob for controlling wall-clock time is h: increasing h increases synthetic data throughput linearly, but also increases the frequency with which the policy is exposed to long-horizon model errors, which interacts with ε_m.

Memory usage is $O(|D_{\text{env}}| + |D_m| + P)$, where P is the total parameter count of π, critics, m, and discriminators. We emphasize that D_m need not be stored indefinitely: one can maintain a sliding-window buffer of recent model rollouts (as in standard model-based RL) since the discriminators and the policy are trained on the current induced distributions. If an ensemble of models is used, memory grows linearly in the ensemble size, but this is optional for Dual-SAR.

Estimating constraint values from classifiers. The constraints involve divergences of the form $D_f(p(\cdot \mid s,a)\|m(\cdot \mid s,a))$ and $D_g(\pi(\cdot \mid s)\|\pi_b(\cdot \mid s))$, averaged under the discounted occupancy of the current policy. In practice we estimate these quantities through variational surrogates induced by binary classification, and we use the resulting scores both (i) as per-sample reward corrections and (ii) as empirical estimates $\widehat{c}_m(\pi)$, $\widehat{c}_\pi(\pi)$ for the dual updates.

For the model-mismatch constraint, we train a transition discriminator $C_\phi(s,a,s') \in (0,1)$ to separate real transitions (s,a,s') from D_{env} (label $y=1$) and synthetic transitions from D_m (label $y=0$) using logistic loss. Under standard density-ratio arguments, the logit

$$\widehat{\ell}_p(s,a,s') \;:=\; \log \frac{C_\phi(s,a,s')}{1 - C_\phi(s,a,s')}$$

approximates $\log \frac{q_{\text{env}}(s,a,s')}{q_m(s,a,s')}$, where q_{env} and q_m are the joint transition distributions induced by the respective replay buffers. When the two buffers share the same (s,a)-marginal (e.g., via conditioning on the same (s,a) samples or via careful rollout initialization), this recovers a conditional ratio estimate of $\log \frac{p(s'|s,a)}{m(s'|s,a)}$; otherwise it is best interpreted as an integral-probability-metric surrogate controlling the discrepancy between the two induced transition distributions, which is sufficient for the role of the dual update as a conservatism controller. We then form the empirical constraint estimate by averaging scores on the distribution induced by π, e.g.,

$$\widehat{c}_m(\pi) \;:=\; \frac{1}{|B_m|} \sum_{(s,a,s')\in B_m} \operatorname{clip}\big(\widehat{\ell}_p(s,a,s'), -L, L\big),$$

where B_m is a minibatch from D_m and clipping stabilizes the dual dynamics (and bounds the shaped reward).

For the policy-shift constraint, we train an action discriminator $C_\psi(s,a) \in (0,1)$ to separate actions drawn from the current policy at dataset states (label $y=1$) from actions appearing in D_{env} (label $y=0$). Concretely, we sample s from D_{env}, draw $a \sim \pi(\cdot \mid s)$, and treat (s,a) as a policy sample; we treat (s,a) from D_{env} as a behavior sample. The corresponding logit

$$\widehat{\ell}_\pi(s,a) \;:=\; \log \frac{C_\psi(s,a)}{1-C_\psi(s,a)}$$

approximates $\log \frac{\pi(a|s)}{\pi_b(a|s)}$ when the state marginals match by construction (which they do if both classes use the same s-samples). We estimate

$$\widehat{c}_\pi(\pi) \;:=\; \frac{1}{|B_\pi|} \sum_{(s,a)\in B_\pi} \operatorname{clip}\big(\widehat{\ell}_\pi(s,a), -L, L\big),$$

with B_π obtained by sampling s from D_{env} and $a \sim \pi(\cdot \mid s)$. If one prefers a specific divergence (e.g., KL), then one can convert classifier outputs into an explicit f-divergence estimate via the corresponding variational representation; however, for the purposes of dual control, the clipped logit average is often a robust surrogate.

Since the dual updates are sensitive to score scale, we recommend calibrating C_ϕ, C_ψ (e.g., temperature scaling on held-out splits) and monitoring calibration error; in particular, discriminator AUC near 1 coupled with poor calibration can cause unstable (α,β) growth without improving constraint satisfaction.

Budgets as knobs (as opposed to tuned penalties). The user-facing hyperparameters are ε_m and ε_π, which directly encode how much model mismatch and policy shift we are willing to tolerate under the learned occupancy. Unlike fixed penalty coefficients, these budgets have an operational meaning: they target the *average* allowed discrepancy (under the induced state visitation), not a particular reward scale. The dual variables (α,β) are then adjusted automatically to satisfy the budgets. In practice, we set ε_m and ε_π on a coarse grid (often logarithmic), and we find that performance is less brittle than tuning a single static penalty weight, because α and β adapt across training phases (e.g., early iterations may require large β to prevent leaving the dataset support, while later iterations may allow β to decrease once the policy stabilizes). Additional stability knobs include: (i) clipping level L for logits added to rewards, (ii) dual step sizes η_α, η_β, (iii) a delayed dual warm-up (updating (α, β) only after discriminators reach a minimum accuracy), and (iv) limiting model rollout horizon h early in training.

Recommended diagnostics. We recommend logging: (1) $\widehat{c}_m(\pi)$ and $\widehat{c}_\pi(\pi)$ together with their target budgets; (2) dual iterates α_k, β_k and their update increments; (3) summary statistics of correction terms $\alpha\,\widehat{\ell}_p$ on D_m and $\beta\,\widehat{\ell}_\pi$ on D_{env} (mean, quantiles, and fraction clipped); (4) discriminator separation metrics (AUC/accuracy) and calibration metrics on held-out splits; (5) an estimate of shift severity such as effective sample size under $\exp(\widehat{\ell}_\pi)$ (for debugging extreme policy drift); and (6) model-error proxies on a validation subset of D_{env} (e.g., negative log-likelihood or one-step prediction error), since rapidly worsening model fit under the induced policy is often visible before return collapses. These diagnostics typically suffice to distinguish (i) overly aggressive policy updates (rising $\widehat{c}_\pi$ and β), (ii) excessive reliance on model rollouts (rising $\widehat{c}_m$ and α), and (iii) discriminator misspecification (high training accuracy but unstable or non-monotone constraint estimates).

4.6 Experimental Plan

We evaluate Dual-SAR on standard offline benchmarks with heterogeneous behavior coverage and reward scales, with the goal of isolating (i) the benefit of dual adaptation relative to fixed penalties, (ii) the practical meaning of the budgets $(\varepsilon_m, \varepsilon_\pi)$, and (iii) failure modes stemming from ratio estimation and calibration. Our primary benchmark suite is D4RL (MuJoCo locomotion tasks across `random`, `medium`, `medium-replay`, `medium-expert`, and `expert` datasets), complemented by NeoRL, which provides multiple dataset generation regimes and permits controlled shifts in behavior policy and environment stochasticity. We report D4RL normalized scores as customary, but we additionally report feasibility diagnostics aligned with our constraints: empirical model-mismatch scores $\widehat{c}_m(\pi)$, policy-shift scores $\widehat{c}_\pi(\pi)$, the dual iterates (α, β), and the fraction of samples affected by logit clipping. We run all methods with multiple seeds and report mean and standard error; unless otherwise stated, we fix total gradient steps and wall-clock budgets so that comparisons are not confounded by additional compute.

Baselines and controlled comparisons. We compare against representative model-free offline RL methods (e.g., CQL, IQL, TD3+BC) and model-based offline RL methods (e.g., MOPO/MOReL-style pessimism, MBPO-style rollouts with conservative tuning), using published hyperparameter ranges and selecting via offline validation when possible. The key comparison, however, is to a tuned SAR-style penalty method in which the correction terms appear with fixed coefficients $(\bar{\alpha}, \bar{\beta})$ (or a single tuned penalty weight), chosen by an oracle sweep per task. This isolates the value of the primal–dual mechanism: Dual-SAR should match or exceed the performance of the best tuned penalty on average while eliminating per-task tuning. To avoid an unfair advantage, we allow the tuned baselines the same discriminator architecture, the same replay mixture of $D_{\text{env}} \cup D_m$, and the same clipping level L; the only difference is whether (α, β) are adapted online to satisfy budgets.

Budgets as interpretable control knobs. We design experiments to test whether $(\varepsilon_m, \varepsilon_\pi)$ behave monotonically and predictably. For each environment/dataset, we run a small grid over ε_m and ε_π (log-spaced), and we examine: (i) the resulting learned rollout horizon effectively used by the algorithm (as measured by the proportion of updates coming from D_m before constraints tighten), (ii) the induced dual variables (α, β), and (iii) the realized constraint values $\widehat{c}_m(\hat{\pi}), \widehat{c}_\pi(\hat{\pi})$ at convergence. We expect that tightening ε_π prevents destructive extrapolation on low-coverage datasets (e.g., `random`, `medium-replay`), while loosening ε_π enables gains on higher-coverage datasets. Analogously, tightening ε_m should reduce reliance on long-horizon synthetic rollouts, approaching a model-free offline algorithm in the limit. We explicitly check that the empirical violations track the budgets, i.e., $\widehat{c}_m(\pi) \approx \varepsilon_m$ and $\widehat{c}_\pi(\pi) \approx \varepsilon_\pi$ when the corresponding dual is active, consistent with complementary slackness at approximate saddle points.

Ablations. We perform ablations to identify which components are necessary for stability and which primarily affect sample efficiency:

- *Dual adaptation vs. fixed penalties:* replace the dual updates by fixed $(\bar{\alpha}, \bar{\beta})$ chosen either (a) by a global setting across tasks or (b) by per-task tuning. This quantifies the value of automatic adaptation and the extent to which a single penalty weight is brittle across tasks and datasets.

- *Model-rollout dependence:* vary rollout horizon h and the ratio of synthetic to real samples. We test whether Dual-SAR can safely leverage larger h when ε_m is sufficiently small (forcing α upward and penalizing mismatch), and whether it recovers MBPO-like benefits on well-modeled regimes.

- *Clipping and reward transform:* remove clipping or vary L, and optionally compare $u(r) = r$ versus $u(r) = \log r$ (when $r_{\min} > 0$) to test sensitivity of training dynamics to heavy-tailed logits and reward scaling.

- *Discriminator architecture and capacity:* reduce capacity to induce underfitting and increase capacity to induce near-separation, measuring how each affects feasibility and return.

Each ablation is evaluated not only by return but also by the evolution of (α, β) and the stability of $\widehat{c}_m, \widehat{c}_\pi$, since we view constraint tracking as the primary operational objective.

Stress tests via controlled model degradation. To probe robustness to model misspecification, we construct synthetic degradations of the learned dynamics: (i) injecting Gaussian noise into model outputs, (ii) training m on a reduced subset of D_{env} to simulate data scarcity, and (iii) using intentionally mis-specified architectures. For each degradation level, we rerun Dual-SAR and fixed-penalty baselines while holding all other hyperparameters constant. Our hypothesis is that dual adaptation will respond by increasing α (tightening the effective trust region in model space), reducing the contribution of D_m to policy improvement, and thereby degrading performance gracefully rather than catastrophically. We additionally test a multi-stage training regime in which we deliberately increase h over time (e.g., $h = 1 \to 5 \to 15$), mimicking common model-based practice; we check whether α rises at the stage transitions and whether feasibility diagnostics anticipate performance drops.

Calibration and ratio-estimation reliability. Because our feasibility and reward corrections depend on discriminator logits, we directly test calibration effects. Concretely, we compare (i) no calibration, (ii) temperature scaling on held-out splits, and (iii) isotonic regression (when feasible), and we report standard calibration diagnostics (e.g., expected calibration error) alongside RL outcomes. We also test whether enforcing mild regularization on logits (weight decay, label smoothing) improves dual stability by preventing spurious extreme $\widehat{\ell}$ values. Finally, we quantify "near-separation" regimes by reporting discriminator AUC and the empirical distribution of $\widehat{\ell}_p, \widehat{\ell}_\pi$; we expect that high AUC without calibration correlates with overly aggressive growth of (α, β) and overly conservative policies, whereas calibrated logits yield smoother dual dynamics and better budget tracking.

Reporting and reproducibility. For each benchmark, we provide (i) performance profiles versus $(\varepsilon_m, \varepsilon_\pi)$, (ii) learning curves with dual trajectories, and (iii) sensitivity to h and L. We also report the final replay mixture proportions and the realized constraint estimates, so that performance improvements can be interpreted as either true exploitation of model rollouts within budget or as artifacts of uncontrolled shift. This experimental plan is designed to make explicit whether Dual-SAR is functioning as intended: namely, as an automatically adapting conservatism controller that trades off synthetic rollouts and policy shift in a manner consistent with the specified budgets.

4.7 Discussion and Limitations

Dual-SAR reduces offline model-based RL to a constrained optimization whose Lagrangian induces the corrected reward $\tilde{r}_{\alpha,\beta}$. This formulation clarifies what the method *can* and *cannot* guarantee: when the constraints can be estimated reliably on the relevant occupancy measure, the dual mechanism provides an operational procedure for trading off model rollouts against distribution shift; when they cannot, the algorithm may either become overly conservative (duals diverge) or overly aggressive (constraints are spuriously underestimated). We summarize several limitations and open directions that are intrinsic to this approach rather than artifacts of a particular implementation.

Dependence on density-ratio (logit) estimation. Our guarantees and the practical behavior of Dual-SAR depend on estimating $\log \rho_p(s, a, s') = \log \frac{p(s'|s,a)}{m(s'|s,a)}$ and $\log \rho_\pi(s, a) = \log \frac{\pi(a|s)}{\pi_b(a|s)}$ from discriminators trained on samples from D_{env}, D_m, and D_π. This introduces several coupled failure modes. First, *support mismatch* remains decisive: if m or π induces transitions/actions outside the support of the corresponding training distributions, then neither ρ_p nor ρ_π is identifiable, and discriminator outputs can extrapolate arbitrarily. This is not merely a technicality; it is the operational content of impossibility results such as Theorem 5. Second, even on-support, ratio estimation is statistically hard in high dimension: small absolute classification error can translate into large log-ratio error on rare events, which then enters $\tilde{r}_{\alpha,\beta}$ additively and can dominate Bellman backups. Third, discriminator *misspecification* matters: if the discriminator class cannot represent the Bayes-optimal logit, then the induced surrogate divergences may underpenalize precisely the regions where model bias or policy shift is most harmful, producing apparent constraint satisfaction without real safety. Finally, *calibration* is essential because the corrected reward uses logits rather than only a ranking statistic; uncalibrated near-separation can produce extreme $\widehat{\log \rho}$ values, causing instability in both primal learning and dual ascent. Clipping alleviates this but turns the constraints into a truncated surrogate, which complicates interpretation.

Positivity and the role of the reward transform. We assumed $r_{\min} > 0$ to permit optional use of $u(r) = \log r$ in variational manipulations and to improve numerical conditioning when raw rewards vary widely. This assumption is restrictive for many benchmarks with sparse rewards, signed costs, or terminal bonuses. A straightforward workaround is to use $u(r) = r$ (which removes the positivity requirement), or to shift and scale rewards to enforce positivity. However, reward shifting interacts subtly with the Lagrangian shaping: while adding a constant per step does not change the optimal policy in an infinite-horizon discounted MDP (up to an additive constant in value), it does change the *relative magnitude* of the correction terms compared to the base reward during learning, and hence affects optimization dynamics for a fixed clipping level L. More generally, if one wishes to interpret $\tilde{r}_{\alpha,\beta}$ as arising from a utility-regularized objective, then the choice of u determines which risk/scale properties are being optimized. An open technical point is to develop a principled selection of u that preserves the variational identities used for divergences while accommodating common reward conventions (including zeros and negatives) without introducing brittle hyperparameters.

How should one pick budgets $(\varepsilon_m, \varepsilon_\pi)$**?** The budgets are intended to be *interpretable* trust-region radii: ε_π limits how far π may move from π_b in action space under the induced state distribution, and ε_m limits how much synthetic rollouts may rely on regions where m deviates from p. That said, selecting budgets remains a modeling choice that encodes the user's tolerance for extrapolation and model bias. We see three practical principles. (i) *Feasibility first:* since $c_\pi(\pi_b) = 0$ for standard divergences, any $\varepsilon_\pi \geq 0$ admits π_b; in contrast, ε_m should be large enough that at least a conservative policy (often near π_b) does not immediately violate the model mismatch constraint under its occupancy. (ii) *Data-driven anchoring:* one may estimate baseline mismatch statistics on D_{env} (e.g., using the transition discriminator on one-step samples) and choose ε_m as a quantile or multiple of this baseline, reflecting the desired allowance beyond observed transitions. (iii) *Stability-driven tuning without oracle sweeps:* because (α, β) adapt online, the budgets can often be tuned coarsely by monitoring whether the dual iterates saturate (budgets too tight) or collapse (budgets too loose), with the goal of achieving nontrivial but bounded dual values and consistent constraint tracking. A more principled alternative is to treat $(\varepsilon_m, \varepsilon_\pi)$ as high-level safety parameters set by domain constraints (e.g., maximum allowable policy KL per state) rather than by return-based selection.

Open problems and extensions. Several directions are natural and, in our view, necessary for broad applicability. *Likelihood-free or implicit-model settings:* when m is implicit (e.g., diffusion-based dynamics) or when p is only accessible through samples, the ratio ρ_p cannot be evaluated but may still be estimated adversarially. Extending the theory to integral probability metrics or f-divergences implemented purely via critics, with finite-sample calibration guarantees, would better match modern generative models. *Multi-source and multi-budget formulations:* in many offline regimes we have multiple datasets $D_{\text{env}}^{(j)}$ with different behavior policies $\pi_b^{(j)}$ and different reliability. One would like separate shift budgets $\varepsilon_\pi^{(j)}$ (and possibly model budgets conditioned on source), together with a mechanism that learns how much to rely on each source. This suggests vector-valued dual variables and a resource-allocation view of conservative offline RL. *Beyond average constraints:* our constraints are occupancy-weighted averages; they do not prevent rare but catastrophic violations. Risk-sensitive variants (e.g., CVaR constraints on per-trajectory mismatch) and state-wise constraints (e.g., $\mathrm{KL}(\pi(\cdot \mid s)\|\pi_b(\cdot \mid s)) \leq \varepsilon(s)$) are conceptually aligned with Dual-SAR but require new estimators and more delicate dual dynamics.

In summary, Dual-SAR provides a coherent mechanism for *controlling* model bias and policy shift via explicit budgets, but its reliability is fundamentally tied to (i) overlap/coverage, (ii) calibrated ratio estimation, and (iii) meaningful budget selection. The central theoretical bounds isolate these dependencies, and the most impactful future work is to weaken them without reverting to per-task tuning or untestable assumptions.