WHEN DO OPTIMIZATION DUALS HELP REINFORCEMENT LEARNING?

Anonymous authors
Paper under double-blind review

ABSTRACT

In reinforcement learning with embedded optimization, inner solvers return Karush–Kuhn–Tucker (KKT) dual variables that provide structured abstractions of constraint and resource pressure induced by the current decision context, yet how such representations affect actor–critic learning remains unclear. We characterize when such representations improve critic learning and, when action-dependent, how they affect actor-gradient identifiability. Since the solver representation $Z = T(X)$ is a deterministic function of the complete optimization context X, explicitly providing it to the critic adds no Bayes information. Nevertheless, the added representation can expose target-relevant solver structure that is difficult for the baseline class to recover. For nested linear or frozen-feature classes, we derive an exact fixed-design finite-sample risk decomposition balancing captured structural signal against added estimation cost, with a separate Gaussian reference crossover and a bridge to fixed-policy value learning. On the actor side, we show that the detached action partial derivative is generally not identified by on-graph critic values, with its ambiguity confined to $\mathrm{Range}(J_T^\top)$. Matched total-derivative and score-function interfaces are instead extension-invariant. Experiments spanning controlled diagnostics, fixed-policy value learning, and full actor–critic control support the predicted critic-side learnability trade-offs and actor-side gradient-identifiability effects.

1 INTRODUCTION

Optimization modules are increasingly embedded in learned decision pipelines to enforce feasibility or solve structured subproblems Gould et al. (2022); Pan et al. (2024); Magoon et al. (2025). In control and reinforcement learning, differentiable approaches have largely focused on the resulting primal solutions and on propagating sensitivities of the solution map back to learned components Romero et al. (2025); Miao et al. (2025). Yet the same solve can also expose Karush–Kuhn–Tucker (KKT) dual variables, which encode local constraint pressure and, under standard sensitivity conditions, admit a marginal-value interpretation Boyd & Vandenberghe (2004); Bonnans & Shapiro (2013). These quantities are natural solver-derived representations of the current constrained decision problem, but their use in actor–critic learning raises a fundamental tension: because they are determined by the optimization context, any benefit must arise from how they reshape the learning problem rather than from new information; when they also depend on the action, the actor interface itself becomes part of the problem. These two perspectives motivate the critic- and actor-side questions developed below.

The first question concerns *same-information learnability*. Let $X = (s, a)$ denote the complete per-decision optimization context, where s collects the non-action information available at the decision point, and let $Z = T(X)$ be the corresponding solver representation. Since Z is deterministic given X, $\sigma(X, Z) = \sigma(X)$; hence, augmentation adds no Bayes information and cannot reduce the Bayes risk. It may nevertheless improve critic learning, since actor–critic methods rely on value estimates learned from finite data within restricted function classes Wang et al. (2024b); Tan et al. (2025). In this regime, the choice of representation can affect approximation accuracy, effective feature geometry, and statistical complexity Du et al. (2020); Le Lan et al. (2022); Garcin et al. (2025); Pavse et al. (2025). Solver-derived Z may expose target-relevant constraint pressure that is difficult for the baseline critic class to represent, but the approximation gain must outweigh the

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

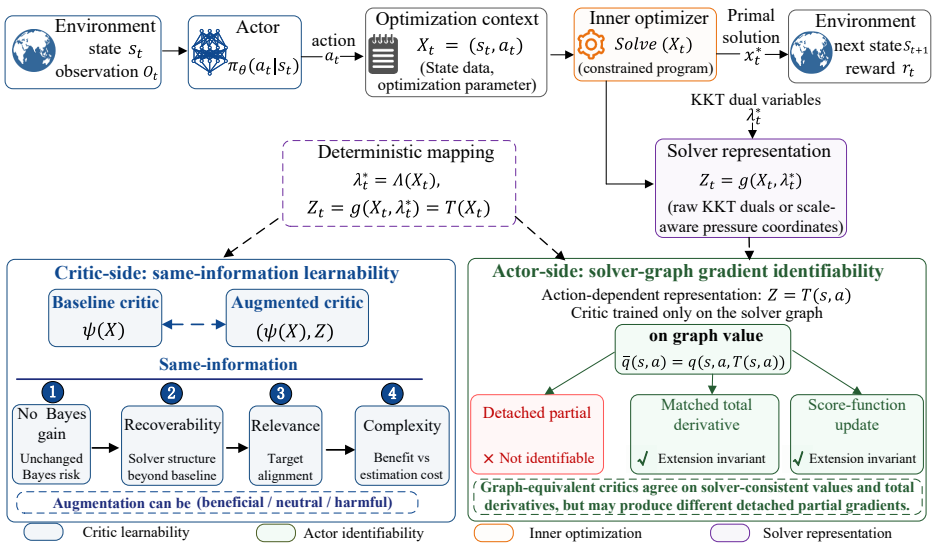


Figure 1: Solver-derived representations in actor–critic learning.

added estimation cost. Our first objective is to characterize this trade-off through recoverability, relevance, and complexity.

The second question concerns how action-dependent solver representations shape policy optimization and, empirically, how these effects translate into policy return. Deterministic and reparameterized actor–critic methods update the policy using the critic's sensitivity to the action Silver et al. (2014); Haarnoja et al. (2018); Pfau et al. (2025). Such a representation may help the critic capture action-dependent constraint pressure and solver-induced trade-offs, thereby providing a more structured signal for policy optimization. Accordingly, the actor update may depend not only on the critic's direct action derivative, but also on the solver-mediated dependence of $Z = T(X) = T(s, a)$ on a. The difficulty is that critic supervision is available only at solver-consistent inputs, which lie on the graph $\mathcal{M} = \{(s, a, z) : z = T(s, a)\}$. Thus, even with exact on-graph value fitting, the critic is constrained only through the composite function $\bar{q}(s, a) = q(s, a, T(s, a))$, which does not in general identify the detached partial derivative $\partial_a q(s, a, z)|_{z=T(s,a)}$. Two critics can therefore agree everywhere on $\mathcal{M}$ while inducing different detached actor gradients. This ambiguity is structural rather than a consequence of finite data or imperfect optimization: the critic loss does not constrain how q extends away from the solver graph. Our second objective is to characterize this ambiguity and identify actor-update interfaces that are invariant to such off-graph extensions.

Fig. 1 summarizes the two analysis paths considered in this work, from solver-derived representations to critic learnability and actor-side gradient behavior.

Our main contributions are summarized as follows. (1) On the critic side, we characterize when deterministic solver representations improve restricted-class learnability without adding Bayes information, quantifying the trade-off between target-aligned structural gain and finite-sample estimation cost. (2) On the actor side, we show that even exact on-graph critic values need not identify the detached action gradient, derive its exact ambiguity set for unrestricted smooth extensions, and identify extension-invariant total-derivative and score-function updates. (3) We evaluate these effects through controlled diagnostics, fixed-policy value learning, and full-RL experiments, demonstrating representation benefits across function classes, extension sensitivity in online actor–critic training, and control gains on replayed real workloads.

2 Related Work

Dual variables in constrained learning. Constrained RL typically formulates policy learning as a constrained Markov decision process with long-term cost or safety constraints Liu et al. (2022); Wachi et al. (2024); Gu et al. (2024). Lagrangian and primal–dual approaches introduce outer mul-

tipliers that evolve with the learning or execution process to enforce such constraints Hong et al. (2024); Wu et al. (2024); Chen et al. (2024). Several approaches further condition the policy, state representation, or value function on such dual variables, enabling adaptation across constraint conditions Calvo-Fullana et al. (2023); Lee et al. (2023); Uslu et al. (2026). Our dual variables have a different origin and role: they are KKT multipliers returned by an embedded optimization problem and recomputed for each context–action pair, rather than outer variables introduced to enforce an RL-level constraint. Instead of acting as adaptive constraint weights, they provide a local representation of the optimization structure induced by the current decision and can therefore be used as inputs to value learning and policy optimization.

Differentiable optimization and actor–critic learning. A growing literature studies how to differentiate through optimization-defined solution and control maps using KKT conditions, implicit differentiation, or optimal-control equations Xu et al. (2023); Banker & Mesbah (2025); Bian et al. (2026). These works primarily address how to compute derivatives of primal solution or control maps. Separately, recent studies show that representation structure and target relevance can materially affect value-function approximation and statistical efficiency Feng & Zhong (2024); Zhou (2024); Fujimoto et al. (2025). Related actor–critic analyses study how critic errors and value gradients affect policy optimization Mu & Klabjan (2024); Gao et al. (2024); Fang et al. (2025). Anders et al. (2020) establish that off-manifold gradient components can vary while on-manifold predictions remain unchanged. We specialize this geometry to the explicit solver graph ($z = T(s, a)$), characterize the resulting action-gradient ambiguity, and identify extension-invariant actor interfaces.

3 METHODOLOGY

This section develops the two learning interfaces induced by per-decision solver representations. We first introduce the notation needed to define the comparison, and then turn to the critic-side and actor-side analyses.

3.1 PROBLEM SETTING AND SOLVER REPRESENTATION

Using the notation introduced above, let $X_t = (s_t, a_t)$ denote the complete per-decision optimization context, where s_t collects the non-action information required to specify the embedded optimization problem and need not coincide with a compressed or partial agent observation. The embedded solver returns

$$(x_t^\star, \lambda_t^\star) = \mathrm{Solve}(X_t), \tag{1}$$

where $x_t^\star$ is the implemented primal solution and $\lambda_t^\star$ is the KKT dual vector selected by the solver. We assume that, for every X_t in the support of interest, the embedded problem admits at least one optimal primal–dual KKT pair. If the primal or dual solution is nonunique, Solve incorporates the deterministic selection rule used by the solver.

We construct the solver-derived critic representation

$$Z_t = g(X_t, \lambda_t^\star) \equiv T(X_t) \tag{2}$$

where $\lambda_t^\star = \Lambda(X_t)$ and g is a deterministic, possibly context-dependent transformation of the dual vector. Further details on the interpretation and scaling of the dual representation are provided in Appendix A.

Because T is measurable and $Z_t = T(X_t)$,

$$\sigma(X_t, Z_t) = \sigma(X_t). \tag{3}$$

Thus, augmentation introduces no information beyond the current optimization context. Throughout the following critic-side analysis, Z denotes the dual-derived representation in Eq. 2. The resulting characterizations apply more broadly to deterministic solver representations.

3.2 SAME-INFORMATION CRITIC LEARNABILITY

Suppressing the time index, let Y denote the scalar critic-learning target. By Eq. 3,

$$\mathbb{E}[Y \mid X, Z] = \mathbb{E}[Y \mid X] \quad \text{a.s.} \tag{4}$$

Hence, under squared loss, solver augmentation does not change the Bayes regression function or reduce the Bayes risk. Any benefit must therefore arise from approximation, estimation, or optimization effects within the learning procedure. We first characterize the approximation–estimation trade-off for nested linear or frozen-feature classes.

Let $\Psi = \psi(X) \in \mathbb{R}^p$ be the baseline feature representation and $Z \in \mathbb{R}^{d_z}$ the solver representation. For the strict same-information comparison, we assume $\sigma(\Psi, Z) = \sigma(\Psi)$. Thus, Z is recoverable from Ψ in the measure-theoretic sense, but need not be linearly recoverable from it. Assuming centered variables, or that a constant feature is included in Ψ, consider

$$\mathcal{F} = \{w^\top \Psi : w \in \mathbb{R}^p\}, \qquad \mathcal{H} = \{w^\top \Psi + \beta^\top Z : w \in \mathbb{R}^p,\, \beta \in \mathbb{R}^{d_z}\}. \tag{5}$$

Consequently, augmentation may reduce approximation error relative to $\mathcal{F}$ despite adding no information.

3.2.1 Target-aligned solver residual

Under the convention above, let $\Sigma_{\Psi\Psi} = \mathbb{E}[\Psi\Psi^\top] \succ 0$ and $\Sigma_{Z\Psi} = \mathbb{E}[Z\Psi^\top]$. Define

$$B = \Sigma_{Z\Psi}\Sigma_{\Psi\Psi}^{-1}, \qquad R = Z - B\Psi, \qquad \mathcal{S}_R = \{\gamma^\top R : \gamma \in \mathbb{R}^{d_z}\}. \tag{6}$$

Here, R is the residual of the population linear projection of Z onto the baseline features Ψ, and $\mathcal{S}_R$ is the linear subspace spanned by these residual solver coordinates. By construction, $\mathbb{E}[R\Psi^\top] = 0$; hence, since $Z = B\Psi + R$, $\mathcal{H} = \mathcal{F} \oplus \mathcal{S}_R$, where the sum is orthogonal in L_2. Let $m(X) = \mathbb{E}[Y \mid X]$ denote the regression function. For $\mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}$, let $P_\mathcal{G}$ denote the L_2-orthogonal projection and define the population approximation risk

$$\mathcal{R}^\star_\mathcal{G} = \|m - P_\mathcal{G} m\|^2_{L_2}. \tag{7}$$

Proposition 1 (Population structural gain)**.** *The population approximation gain from exposing the solver representation is*

$$\boxed{\Delta_{\text{struct}} := \mathcal{R}^\star_\mathcal{F} - \mathcal{R}^\star_\mathcal{H} = \|P_{\mathcal{S}_R} m\|^2_{L_2}.} \tag{8}$$

Since $P_{\mathcal{S}_R} m \in \mathcal{S}_R$, there exists $\gamma \in \mathbb{R}^{d_z}$ such that

$$P_{\mathcal{S}_R} m = \gamma^\top R. \tag{9}$$

Consequently,

$$\Delta_{\text{struct}} = \left\|\gamma^\top R\right\|^2_{L_2} = \gamma^\top \Sigma_R \gamma, \qquad \Sigma_R = \mathbb{E}[RR^\top]. \tag{10}$$

Here, γ captures the target alignment within the residual solver subspace, while Δ_{struct} measures the resulting structural gain. Hence, $\Delta_{\text{struct}} > 0$ if and only if $P_{\mathcal{S}_R} m \neq 0$ in L_2.

Proposition 1 separates linear recoverability from target relevance. The residual R captures solver structure outside the linear span of the baseline representation, but this structure improves approximation only through its component aligned with m. Thus, the magnitude of the solver residual alone is insufficient to imply a representation benefit. The proof is given in Appendix B.1.

3.2.2 Finite-sample augmentation margin

Condition on $X_{1:n} = (X_1, \ldots, X_n)$. Let F and H be the design matrices induced by the baseline features Ψ and augmented features (Ψ, Z), respectively, so that $\mathrm{col}(F) \subseteq \mathrm{col}(H)$, and let P_F, P_H be their orthogonal projectors. With $\mathbf{y} = (Y_1, \ldots, Y_n)^\top$ and $\mathbf{m} = (m(X_1), \ldots, m(X_n))^\top$, assume

$$\mathbf{y} = \mathbf{m} + \boldsymbol{\varepsilon}, \qquad \mathbb{E}[\boldsymbol{\varepsilon} \mid X_{1:n}] = 0, \qquad \Sigma_\varepsilon = \mathrm{Cov}(\boldsymbol{\varepsilon} \mid X_{1:n}). \tag{11}$$

Here $\boldsymbol{\varepsilon}$ denotes the residual variation of the critic targets around the regression function at the observed design points. The conditional mean-zero assumption ensures that the residual has zero conditional mean given $X_{1:n}$, while Σ_ε describes its conditional covariance structure.

For $G \in \{F, H\}$, define

$$\widehat{\mathbf{m}}_G = P_G \mathbf{y}, \qquad \mathcal{E}^{(n)}_G = \frac{1}{n}\mathbb{E}\left[\|\widehat{\mathbf{m}}_G - \mathbf{m}\|_2^2 \mid X_{1:n}\right], \tag{12}$$

where $\widehat{\mathbf{m}}_G$ denotes the fitted values under design space G, and $\mathcal{E}^{(n)}_G$ is the corresponding conditional in-design function-estimation risk.

Theorem 1 (Fixed-design solver-augmentation margin). *For each $G \in \{F, H\}$, the conditional in-design function-estimation risk decomposes as*

$$\mathcal{E}_G^{(n)} = \frac{\|(I_n - P_G)\mathbf{m}\|_2^2}{n} + \frac{\mathrm{tr}(P_G \Sigma_\varepsilon)}{n}. \tag{13}$$

Moreover, the finite-sample augmentation margin is

$$\boxed{\Delta_{\mathrm{FS}}^{(n)} := \mathcal{E}_F^{(n)} - \mathcal{E}_H^{(n)} = \frac{\|(P_H - P_F)\mathbf{m}\|_2^2}{n} - \frac{\mathrm{tr}[(P_H - P_F)\Sigma_\varepsilon]}{n}.} \tag{14}$$

Hence, the augmented estimator has lower conditional in-design function-estimation risk if and only if $\Delta_{\mathrm{FS}}^{(n)} > 0$.

Since the design spaces are nested, $P_H - P_F$ projects onto the additional directions introduced by augmentation. Thus, the two terms in Eq. 14 quantify, respectively, the target-aligned signal captured by these directions and their additional estimation cost. When the residual noise has constant variance and is uncorrelated across samples, $\Sigma_\varepsilon = \sigma_\varepsilon^2 I_n$, and the estimation-cost term reduces to $\sigma_\varepsilon^2[\mathrm{rank}(H) - \mathrm{rank}(F)]/n$. Therefore, augmentation helps exactly when the added signal exceeds this estimation cost. The result applies exactly to nested linear regression and linear readouts over fixed representations; its implications for sequential value learning and downstream control are examined in the subsequent analysis and experiments. The proof is given in Appendix B.2.

3.2.3 Gaussian reference crossover

We next illustrate the approximation–estimation trade-off using a Gaussian reference model, relaxing $\sigma(\Psi, Z) = \sigma(\Psi)$ for this calculation only. Assume that the n training samples are i.i.d., that (Ψ, R) is centered jointly Gaussian, and that the augmented model is realizable, so that

$$m(X) = \theta^\top \Psi + \gamma^\top R. \tag{15}$$

In addition, assume that the target residual $\varepsilon = Y - m(X)$ is independent of (Ψ, R) and satisfies $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Since $\mathbb{E}[R\Psi^\top] = 0$ by construction, joint Gaussianity implies that R and Ψ are independent. Let $r = \mathrm{rank}(\Sigma_R)$, and restrict R to its r-dimensional effective subspace, on which its covariance is nonsingular.

For $\mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}$, let $\widehat{m}_\mathcal{G}$ denote the corresponding ordinary least-squares (OLS) predictor, whose evaluations on the training sample yield the fitted vector $\widehat{\mathbf{m}}_G$ defined above. The baseline predictor uses Ψ, whereas the augmented predictor uses (Ψ, R). Since $Z = B\Psi + R$, the feature sets (Ψ, R) and (Ψ, Z) span the same linear space, so the latter is equivalent to fitting the augmented model with (Ψ, Z). For an independent test context X', define

$$\mathfrak{R}_\mathcal{G}^{\mathrm{out}} = \mathbb{E}\left[\left(\widehat{m}_\mathcal{G}(X') - m(X')\right)^2\right], \qquad \mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}, \tag{16}$$

where the expectation is over the training sample and the independent test context; the irreducible variance of a noisy test target is excluded.

Corollary 1 (Exact Gaussian crossover). *If $n > p + r + 1$, then*

$$\mathfrak{R}_\mathcal{F}^{\mathrm{out}} = \Delta_{\mathrm{struct}} + \left(\sigma^2 + \Delta_{\mathrm{struct}}\right) \frac{p}{n - p - 1}, \tag{17}$$

$$\mathfrak{R}_\mathcal{H}^{\mathrm{out}} = \sigma^2 \frac{p + r}{n - p - r - 1}. \tag{18}$$

Consequently,

$$\boxed{\mathfrak{R}_\mathcal{F}^{\mathrm{out}} - \mathfrak{R}_\mathcal{H}^{\mathrm{out}} = \frac{n - 1}{n - p - 1}\left[\Delta_{\mathrm{struct}} - \frac{\sigma^2 r}{n - p - r - 1}\right].} \tag{19}$$

Hence, solver augmentation has lower expected out-of-sample function-estimation risk if and only if

$$\Delta_{\mathrm{struct}} > \frac{\sigma^2 r}{n - p - r - 1}. \tag{20}$$

For $\Delta_{\mathrm{struct}} > 0$, Eq. 20 is equivalently

$$n > p + r + 1 + \frac{\sigma^2 r}{\Delta_{\mathrm{struct}}}. \tag{21}$$

In this reference model, augmentation is favored by stronger target-aligned residual signal and more data, and penalized by larger noise variance and residual rank. The proof is given in Appendix B.3.

Under the strict same-information condition, the fixed-design comparison can also be transferred to population risk under a uniform empirical–population norm-equivalence condition on $\mathcal{H}$. Under augmented realizability, a sufficiently large positive fixed-design margin then certifies lower conditional population risk. The formal statement and proof are given in Appendix B.4.

3.2.4 Fixed-policy value-learning bridge

Fix a policy π with stationary state–action distribution μ^π. Let $\widetilde{r}^\pi$ denote the expected one-step reward signal, define $P^\pi Q(s,a) = \mathbb{E}[Q(S',A') \mid S = s, A = a]$ with $A' \sim \pi(\cdot \mid S')$, and let $\mathcal{T}^\pi Q = \widetilde{r}^\pi + \gamma P^\pi Q$ for $0 \le \gamma < 1$. Stationarity and conditional Jensen's inequality imply that P^π is nonexpansive in $L_2(\mu^\pi)$, and hence $\mathcal{T}^\pi$ is a γ-contraction.

For a closed linear value space $\mathcal{G}$ with orthogonal projector $\Pi_{\mathcal{G}}$, let $Q_{\mathcal{G}}$ be the unique projected fixed point satisfying $Q_{\mathcal{G}} = \Pi_{\mathcal{G}} \mathcal{T}^\pi Q_{\mathcal{G}}$, and define $a_{\mathcal{G}} = \|(I - \Pi_{\mathcal{G}}) Q^\pi\|_{2,\mu^\pi}$ Tsitsiklis & Van Roy (1996); Lim et al. (2026). Then

$$a_{\mathcal{G}} \le \|Q_{\mathcal{G}} - Q^\pi\|_{2,\mu^\pi} \le \frac{a_{\mathcal{G}}}{\sqrt{1-\gamma^2}}. \tag{22}$$

For fitted Q evaluation (FQE) Wang et al. (2024a), write $\widehat{Q}_{k+1} = \Pi_{\mathcal{G}} \mathcal{T}^\pi \widehat{Q}_k + e_k$, where e_k is the regression error relative to the exact projected Bellman update. The contraction argument yields

$$\|\widehat{Q}_K - Q^\pi\|_{2,\mu^\pi} \le \gamma^K \|\widehat{Q}_0 - Q_{\mathcal{G}}\|_{2,\mu^\pi} + \sum_{j=0}^{K-1} \gamma^{K-1-j} \|e_j\|_{2,\mu^\pi} + \frac{a_{\mathcal{G}}}{\sqrt{1-\gamma^2}}. \tag{23}$$

Solver augmentation may reduce $a_{\mathcal{G}}$ while increasing finite-sample regression error through the enlarged representation. Conditionally on the current iterate and realized regression design, Theorem 1 characterizes this one-step regression trade-off when the Bellman-target residual has zero conditional mean. Eq. 23 therefore connects the preceding representation trade-off to fixed-policy value estimation. The complete derivation and sampling conditions are given in Appendix B.5.

3.3 Solver-graph actor-gradient identifiability

For the actor-side analysis, we work pointwise in the state variable. For any fixed s, we suppress the state argument and write $T(a) := T(s,a)$. Let $\mathcal{U} \subseteq \mathbb{R}^{d_a}$ be open and let $T : \mathcal{U} \to \mathbb{R}^{d_z}$ be continuously differentiable, with Jacobian

$$J_T(a) := \left[\frac{\partial T_i(a)}{\partial a_j} \right]_{i=1,\dots,d_z;\, j=1,\dots,d_a} \in \mathbb{R}^{d_z \times d_a}. \tag{24}$$

Let $q, \widetilde{q} : \mathcal{U} \times \mathbb{R}^{d_z} \to \mathbb{R}$ be continuously differentiable. They are *solver-graph equivalent*, written $q \sim_T \widetilde{q}$, if

$$q(a, T(a)) = \widetilde{q}(a, T(a)), \qquad \forall a \in \mathcal{U}. \tag{25}$$

For a reference critic q, define the detached-gradient ambiguity set

$$\mathcal{A}_{\mathrm{semi}}(a; q) := \left\{ (\nabla_a \widetilde{q}(a,z) - \nabla_a q(a,z))|_{z=T(a)} : \widetilde{q} \sim_T q \right\}, \tag{26}$$

where ∇_a denotes differentiation with respect to the first argument while holding z fixed.

Theorem 2 (Solver-graph semi-gradient non-identifiability)**.** *For any solver-graph-equivalent critics and every $a \in \mathcal{U}$,*

$$(\nabla_a \widetilde{q}(a,z) - \nabla_a q(a,z))|_{z=T(a)} \in \mathrm{Range}\big(J_T(a)^\top\big). \tag{27}$$

For the unrestricted class of continuously differentiable critic extensions, the complete ambiguity set is

$$\boxed{\mathcal{A}_{\text{semi}}(a;q) = \text{Range}\big(J_T(a)^\top\big)\,.} \tag{28}$$

Equality is attained by setting $\widetilde{q} = q + h_v$, where

$$h_v(a', z) = v^\top (z - T(a'))\,, \qquad v \in \mathbb{R}^{d_z}, \tag{29}$$

since h_v vanishes on the solver graph and changes the detached gradient by $-J_T(a')^\top v$. By contrast, graph equivalence uniquely determines the matched total derivative:

$$\begin{aligned} &\nabla_a q(a,z)|_{z=T(a)} + J_T(a)^\top \nabla_z q(a, T(a)) \\ &\quad = \nabla_a \widetilde{q}(a,z)|_{z=T(a)} + J_T(a)^\top \nabla_z \widetilde{q}(a, T(a)). \end{aligned} \tag{30}$$

Thus, agreement on the solver graph identifies the derivative of the composed critic $a \mapsto q(a, T(a))$, but not the detached action partial derivative. The ambiguity is structural rather than a numerical differentiation error. For a restricted critic class, Eq. 27 continues to hold, whereas equality in Eq. 28 requires the admissible difference class to realize the graph-vanishing directions in Eq. 29. The result extends locally to smooth regions of piecewise-C^1 solver maps; see Appendix B.6.

Corollary 2 (Extension-invariant actor interfaces)**.** *Solver-graph-equivalent critics induce identical total-derivative and score-function updates, the latter using detached on-graph values and a common action-independent baseline.*

For a reparameterized action A_φ and $A \sim \pi_\varphi(\cdot \mid S)$, we consider

$$\texttt{dual_semigrad}: \quad \mathbb{E}[\nabla_\varphi q(S, A_\varphi, \text{sg}[T(S, A_\varphi)])]\,, \tag{31}$$

$$\texttt{dual_totalgrad}: \quad \mathbb{E}[\nabla_\varphi q(S, A_\varphi, T(S, A_\varphi))]\,, \tag{32}$$

$$\texttt{dual_score}: \quad \mathbb{E}[\nabla_\varphi \log \pi_\varphi(A \mid S)\ \text{sg}[q(S, A, T(S, A)) - b(S)]]\,. \tag{33}$$

For the extension audit, define

$$q_\kappa(s, a, z) = q(s, a, z) + \kappa\, g(s, a)^\top \big(z - T(s, a)\big), \tag{34}$$

$$(\nabla_a q_\kappa - \nabla_a q)|_{z=T(s,a)} = -\kappa J_{T,a}(s, a)^\top g(s, a), \tag{35}$$

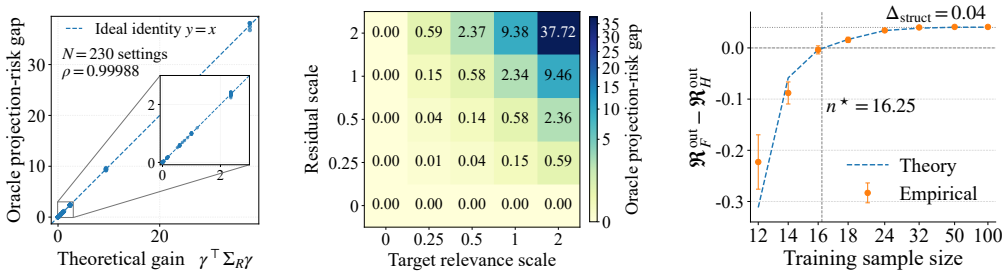
where $g(s, a) \in \mathbb{R}^{d_z}$ is fixed and continuously differentiable, and $J_{T,a}$ denotes the action Jacobian of T with the state argument restored. Thus, q_κ preserves all on-graph values while varying the detached action gradient, directly testing the non-identifiability in Theorem 2.

4 Experiments

Experimental setup. We match contexts, targets, data splits, normalization rules, optimization budgets, and, where applicable, architecture-search budgets. We compare the following representation variants. **Context** receives the complete context X with a zero-filled auxiliary block matching the width of Z, whereas **Exact dual** receives (X, Z). Additional controls include **Predicted**, **Shuffled**, **Random**, **Matched** (parameter-matched), and X**-only**, testing surrogate recovery, context–representation alignment, arbitrary random inputs, generic deterministic features, and auxiliary-branch effects, respectively. The controlled experiments vary residual structure and target relevance across 230 settings and compare the oracle projection-risk reduction with $\Delta_{\text{struct}} = \gamma^\top \Sigma_R \gamma$. For the Gaussian reference experiment, we set $p = 8$, $r = 1$, $\sigma^2 = 0.25$, and $\Delta_{\text{struct}} = 0.04$, using 1,000 replicates for $n \in \{12, 14, 16, 18, 24, 32, 50, 100\}$. We further evaluate linear, frozen-feature, and end-to-end neural learners, followed by neural FQE for fixed-policy value estimation.

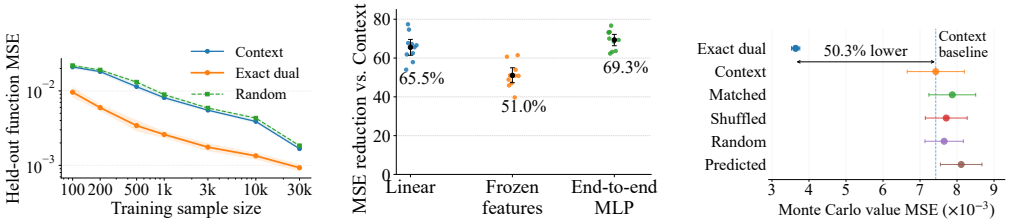
For the actor-side audit, we construct graph-equivalent critics that agree on $\mathcal{M}_s = \{(a, T(s, a))\}$ while varying their off-graph extensions through κ. We compare detached semi-gradient, total-derivative, and score-function interfaces in terms of gradient variation and policy sensitivity.

Finally, full-RL experiments use a controlled task and real Adaptive Charging Network (ACN) workload traces Lee et al. (2019); Mahmud et al. (2025), replayed on a fixed coupled-resource charging network. For the formal real-trace evaluation, the protocol and ten RL seeds are frozen before accessing the formal data, with a complementary 12-checkpoint analysis of learning dynamics. We report prediction/value MSE, gradient and policy sensitivity, paired return differences, and learning-curve AUC as appropriate. Detailed settings and control definitions are given in Appendix C.



(a) Structural-gain identity. (b) Residual–relevance interaction. (c) Finite-sample crossover.

Figure 2: Controlled validation of the residual–relevance characterization.



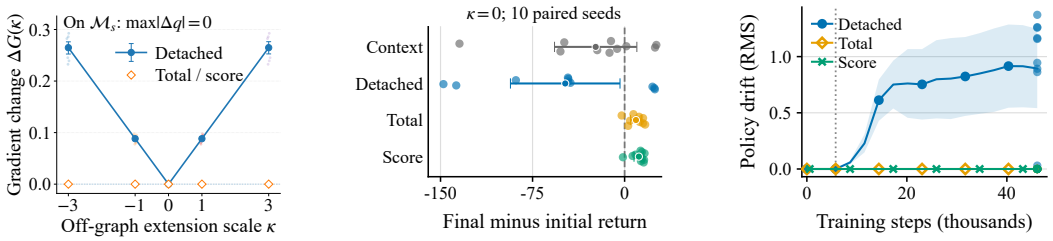
(a) Neural sample-size scaling. (b) Function-class robustness. (c) Fixed-policy value learning.

Figure 3: Learnability and fixed-policy value estimation with deterministic solver representations.

Controlled characterization. Fig. 2 examines the structural characterization, its mechanism, and the resulting finite-sample trade-off under controlled settings. Fig. 2a first tests whether the theoretical structural gain quantitatively captures the oracle projection-risk reduction across 230 settings. The independently evaluated oracle projection-risk gap closely matches $\gamma^\top \Sigma_R \gamma$ (Pearson $\rho = 0.99988$), with small deviations due to finite-sample risk estimation. Fig. 2b then tests the joint roles of residual structure and target relevance: the gain vanishes when either is absent and increases when both grow, consistent with $\Delta_{\text{struct}} = \gamma^\top \Sigma_R \gamma$. Together, these results show that deterministic representations can improve restricted-class learnability by exposing target-relevant structure already implicit in X, rather than by adding information. Finally, Fig. 2c tests the Gaussian reference crossover. The risk difference $\mathfrak{R}_F^{\text{out}} - \mathfrak{R}_H^{\text{out}}$ changes sign between $n = 16$ and $n = 18$, consistent with the predicted crossover $n^\star = 16.25$, and approaches Δ_{struct} as n increases.

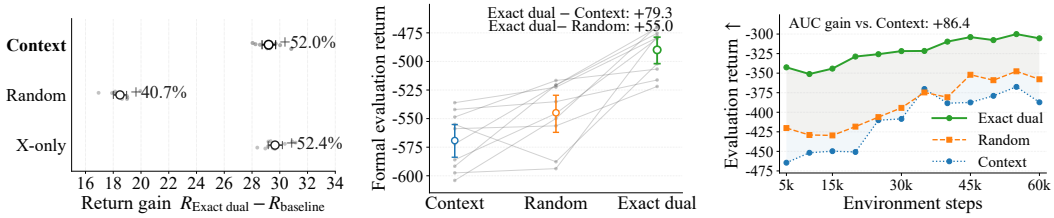
Representation and value learning. Fig. 3a tests sample-size scaling: Exact dual lowers held-out neural prediction error throughout the tested range, while dimension-matched random features track Context. Fig. 3b reports relative MSE reduction over Context, showing gains across linear regression, frozen nonlinear features, and end-to-end MLPs. Fig. 3c extends the comparison to fixed-policy neural FQE: Exact dual lowers Monte Carlo value-estimation MSE by approximately 50% relative to Context, improving in all $10/10$ paired seeds; predicted, shuffled, random, and parameter-matched controls do not reproduce the gain. We also predict optimal-cost responses to resource tightening using complete context and 20 paired seeds. At 2% tightening, Dual lowers test MSE by 58.8% relative to Full primal (Context augmented with the complete nominal primal solution and its summaries). Adding Dual to Full primal lowers its MSE by 61.8%, 37.5%, and 20.0% at 2%, 5%, and 10%, respectively, improving in every seed pair and outperforming Full primal augmented with shuffled dual features. Dual alone is worse than Full primal at 10%. Details appear in Appendix D.2. Additional fixed-policy FQE results on PowerGrid appear in Appendix D.3. Prediction gains alone do not establish higher control return.

Actor-gradient identifiability. Fig. 4a tests the boundary in Theorem 2. Varying the extension coefficient κ changes the critic away from the solver graph while preserving its on-graph values. Nevertheless, the detached actor gradient changes, whereas the total-derivative and score-function interfaces remain invariant. We then examine whether this distinction persists when actor and critic are trained jointly on newly collected trajectories. Fig. 4c tracks RMS action differences from the corresponding $\kappa = 0$ policy on common probe states. Detached policies separate across extensions during training, whereas, within each interface and seed, Total and Score remain unchanged across



(a) Gradient ambiguity. (b) Return improvement. (c) Online policy sensitivity.

Figure 4: Actor-gradient identifiability and online learning.



(a) Controlled RL. (b) Real-trace evaluation. (c) Pre-formal learning dynamics.

Figure 5: Full-RL validation of solver representations.

extensions at all recorded checkpoints. Fig. 4b complements this mechanism test with held-out return improvement at $\kappa = 0$: Total and Score improve over their initial policies by 9.23 and 11.62 on average, respectively, with positive gains in nine of ten seed pairs for each interface (Table 9). Together, these results connect local gradient ambiguity to online policy sensitivity and document learning gains for the invariant interfaces in this task. They do not imply a universal return ordering. Learning curves, the complete extension sweep, and evaluation details appear in Appendix D.4.1.

Full-RL validation. We next test whether solver representations benefit sequential control. In the controlled task, Exact dual outperforms Context, Random, and X-only, with relative return gains of 52.0%, 40.7%, and 52.4%, respectively (Fig. 5a). On real ACN workloads replayed on a fixed coupled-resource charging network, Exact dual achieves mean returns 79.3 above Context and 55.0 above Random on the September test traces under the frozen formal protocol (Fig. 5b). Validation-set learning curves further show higher mean returns for Exact dual than both baselines at all 12 checkpoints, with an 86.4 AUC gain over Context (Fig. 5c). A complete-context MASAC comparison tests shared, invertible observation scaling across five paired seeds (Appendix D.6). Under this protocol, Dual augmentation improves late-window mean episodic return from -4483.50 for Context to 26.00 (Table 12). Together, these results demonstrate sequential-control benefits in controlled and real-trace settings, with additional support under shared observation scaling.

5 Conclusion

Optimization duals are neither universally beneficial features nor merely redundant deterministic outputs. Their critic-side utility depends on whether they expose target-aligned solver structure that is difficult for the learner to recover, and whether this structural benefit exceeds the estimation cost of the added representation. When the representation is action-dependent, critic values learned only on the solver graph do not identify the detached action partial derivative, whereas matched total-derivative and score-function updates remain invariant to graph-equivalent extensions. Together, these results provide a conditional design principle for optimization-augmented RL: assess recoverability, relevance, and complexity on the critic side, while accounting for gradient identifiability when solver representations depend on the action.

REFERENCES

Christopher J. Anders, Plamen Pasliev, Ann-Kathrin Dombrowski, Klaus-Robert Müller, and Pan Kessel. Fairwashing explanations with off-manifold detergent. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Thomas Banker and Ali Mesbah. Gradient-based framework for bilevel optimization of black-box functions: Synergizing model-free reinforcement learning and implicit function differentiation. *Industrial & Engineering Chemistry Research*, 64(5):2831, 2025.

Yuexin Bian, Jie Feng, and Yuanyuan Shi. Diffop: Reinforcement learning of optimization-based control policies via implicit policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19737–19745, 2026.

J Frédéric Bonnans and Alexander Shapiro. *Perturbation analysis of optimization problems*. Springer Science & Business Media, 2013.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*, volume 1. Cambridge university press Cambridge, 2004.

Miguel Calvo-Fullana, Santiago Paternain, Luiz FO Chamon, and Alejandro Ribeiro. State augmented constrained reinforcement learning: Overcoming the limitations of learning with rewards. *IEEE Transactions on Automatic Control*, 69(7):4275–4290, 2023.

Weiqin Chen, James Onyejizu, Long Vu, Lan Hoang, Dharmashankar Subramanian, Koushik Kar, Sandipan Mishra, and Santiago Paternain. Adaptive primal-dual method for safe reinforcement learning. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '24, pp. 326–334, Richland, SC, 2024.

Simon S. Du, Sham M. Kakade, Ruosong Wang, and Lin F. Yang. Is a good representation sufficient for sample efficient reinforcement learning? In *International Conference on Learning Representations*, 2020.

Linjiajie Fang, Ruoxue Liu, Jing Zhang, Wenjia Wang, and Bingyi Jing. Diffusion actor-critic: Formulating constrained policy iteration as diffusion noise regression for offline reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pp. 75743–75763, 2025.

Guhao Feng and Han Zhong. Rethinking model-based, policy-based, and value-based reinforcement learning via the lens of representation complexity. *Advances in Neural Information Processing Systems*, 37:8569–8611, 2024.

Scott Fujimoto, Pierluca D'Oro, Amy Zhang, Yuandong Tian, and Michael Rabbat. Towards general-purpose model-free reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pp. 83509–83552, 2025.

Feng Gao, Liangzhi Shi, Shenao Zhang, Zhaoran Wang, and Yi Wu. Adaptive-gradient policy optimization: Enhancing policy learning in non-smooth differentiable simulations. In *Forty-first International Conference on Machine Learning*, 2024.

Samuel Garcin, Trevor McInroe, Pablo Samuel Castro, Christopher Lucas, David Abel, Prakash Panangaden, and Stefano V Albrecht. Studying the interplay between the actor and critic representations in reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pp. 35590–35616, 2025.

Stephen Gould, Richard Hartley, and Dylan Campbell. Deep declarative networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):3988–4004, 2022. doi: 10.1109/TPAMI.2021.3059462.

Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, and Alois Knoll. A review of safe reinforcement learning: Methods, theories, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):11216–11235, 2024. doi: 10.1109/TPAMI.2024.3457538.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. PMLR, 2018.

Kihyuk Hong, Yuhang Li, and Ambuj Tewari. A primal-dual-critic algorithm for offline constrained reinforcement learning. In *International conference on artificial intelligence and statistics*, pp. 280–288. PMLR, 2024.

Charline Le Lan, Stephen Tu, Adam Oberman, Rishabh Agarwal, and Marc G. Bellemare. On the generalization of representations in reinforcement learning. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151, pp. 4132–4157. PMLR, Mar 2022.

Junseo Lee, Jaeseok Heo, Dohyeong Kim, Gunmin Lee, and Songhwai Oh. Dual variable actor-critic for adaptive safe reinforcement learning. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7568–7573. IEEE, 2023.

Zachary J Lee, Tongxin Li, and Steven H Low. Acn-data: Analysis and applications of an open ev charging dataset. In *Proceedings of the tenth ACM international conference on future energy systems*, pp. 139–149, 2019.

Han-Dong Lim, Hyunjun Na, Taeho Lee, and Donghwan Lee. Td-learning and q-learning: A survey of theory, analysis, and trends. *International Journal of Control, Automation, and Systems*, 24: 2181–2195, 2026.

Zuxin Liu, Zhepeng Cen, Vladislav Isenbaev, Wei Liu, Steven Wu, Bo Li, and Ding Zhao. Constrained variational policy optimization for safe reinforcement learning. In *International Conference on Machine Learning*, pp. 13644–13668. PMLR, 2022.

Connor Magoon, Fengyu Yang, Noam Aigerman, and Shahar Kovalsky. Differentiation through black-box quadratic programming solvers. *Advances in Neural Information Processing Systems*, 38:108486–108517, 2025.

Monowar Mahmud, Tarek Abedin, Md Mahfuzur Rahman, Shamiul Ashraf Shoishob, Tiong Sieh Kiong, and Mohammad Nur-E-Alam. Integrating demand forecasting and deep reinforcement learning for real-time electric vehicle charging price optimization. *Utilities Policy*, 96:102038, 2025. doi: https://doi.org/10.1016/j.jup.2025.102038.

Keyan Miao, Liqun Zhao, Han Wang, Konstantinos Gatsis, and Antonis Papachristodoulou. Optodenet: Neural ode controller design with differentiable optimization layers for safety and stability. In *Proceedings of the 7th Annual Learning for Dynamics & Control Conference*, volume 283, pp. 1217–1229. PMLR, Jun 2025.

Siqiao Mu and Diego Klabjan. On the second-order convergence of biased policy gradient algorithms. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 36455–36485, Vienna, Austria, 2024.

Jianming Pan, Zeqi Ye, Xiao Yang, Xu Yang, Weiqing Liu, Lewen Wang, and Jiang Bian. Bpqp: A differentiable convex optimization framework for efficient end-to-end learning. *Advances in Neural Information Processing Systems*, 37:77468–77493, 2024.

Brahma S Pavse, Yudong Chen, Qiaomin Xie, and Josiah P Hanna. Stable offline value function learning with bisimulation-based representations. In *International Conference on Machine Learning*, pp. 48488–48517. PMLR, 2025.

David Pfau, Ian Davies, Diana L Borsa, João Guilherme Madeira Araújo, Brendan Daniel Tracey, and Hado van Hasselt. Wasserstein policy optimization. In *Forty-second International Conference on Machine Learning*, 2025.

Angel Romero, Elie Aljalbout, Yunlong Song, and Davide Scaramuzza. Actor–critic model predictive control: Differentiable optimization meets reinforcement learning for agile flight. *IEEE Transactions on Robotics*, 42:673–692, 2025.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *International conference on machine learning*, pp. 387–395. PMLR, 2014.

Kevin Tan, Wei Fan, and Yuting Wei. Actor-critics can achieve optimal sample efficiency. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 58545–58590. PMLR, Jul 2025.

John Tsitsiklis and Benjamin Van Roy. Analysis of temporal-difference learning with function approximation. In M.C. Mozer, M. Jordan, and T. Petsche (eds.), *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1996.

Yiğit Berkay Uslu, Navid NaderiAlizadeh, Mark Eisen, and Alejandro Ribeiro. Fast state-augmented learning for wireless resource allocation with dual variable regression. *IEEE Transactions on Signal Processing*, 74:3553–3568, 2026. doi: 10.1109/TSP.2026.3717245.

Akifumi Wachi, Xun Shen, and Yanan Sui. A survey of constraint formulations in safe reinforcement learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, Jeju, Korea, 2024. doi: 10.24963/ijcai.2024/913.

Jiayi Wang, Zhengling Qi, and Raymond K. W. Wong. A fine-grained analysis of fitted q-evaluation: Beyond parametric models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 51273–51302. PMLR, Jul 2024a.

Yudan Wang, Yue Wang, Yi Zhou, and Shaofeng Zou. Non-asymptotic analysis for single-loop (Natural) actor-critic with compatible function approximation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 51771–51824. PMLR, Jul 2024b.

Zifan Wu, Bo Tang, Qian Lin, Chao Yu, Shangqin Mao, Qianlong Xie, Xingxing Wang, and Dong Wang. Off-policy primal-dual safe reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Ming Xu, Timothy L Molloy, and Stephen Gould. Revisiting implicit differentiation for learning problems in optimal control. *Advances in Neural Information Processing Systems*, 36:60060–60076, 2023.

Mertcan Yetkin, Brandon Augustino, Alberto J Lamadrid, and Lawrence V Snyder. Co-optimizing the smart grid and electric public transit bus system. *Optimization and Engineering*, 25(4):2425–2472, 2024.

Angela Zhou. Reward-relevance-filtered linear offline reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 3025–3033. PMLR, 2024.

Ray Daniel Zimmerman, Carlos Edmundo Murillo-Sánchez, and Robert John Thomas. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems*, 26(1):12–19, 2011. doi: 10.1109/TPWRS.2010.2051168.

APPENDIX OVERVIEW

The structure of the appendix is as follows.

- Appendix A provides additional analysis of KKT duals as constraint-pressure coordinates.
- Appendix B provides the proofs and supporting theoretical results for the critic- and actor-side analyses.
- Appendix C provides detailed protocols, control definitions, and implementation settings for the main-text experiments.
- Appendix D provides additional diagnostics, robustness analyses, and supplementary experimental results.
- Appendix E describes the use of large language models (LLMs).

A KKT DUALS AS CONSTRAINT-PRESSURE COORDINATES

Optimization assumptions. The embedded optimization problem has a proper closed convex objective, a nonempty closed convex feasible set, an attained finite optimum, and a constraint qualification such as Slater's condition. When primal or dual solutions are nonunique, the solver uses a deterministic measurable selection, so that the solver representation is a well-defined map $Z = T(X)$.

Consider the parametric convex program

$$V(\xi, b) = \min_{x \in \mathcal{X}(\xi)} f(x; \xi) \quad \text{subject to} \quad g_k(x; \xi) \le b_k, \quad k = 1, \dots, K, \tag{36}$$

where ξ collects the non-budget parameters and $b = (b_1, \dots, b_K)$ denotes the constraint budgets. Let $\lambda^\star \ge 0$ denote an optimal dual variable selected according to the deterministic rule above.

Pressure semantics. Under strong duality, $-\lambda^\star$ is a subgradient of the optimal value with respect to b. Hence, at differentiability points of $V(\xi, b)$,

$$\frac{\partial V(\xi, b)}{\partial b_k} = -\lambda_k^\star. \tag{37}$$

Thus, $\lambda_k^\star$ measures the local marginal value of relaxing the kth constraint.

Raw multiplier magnitudes depend on constraint scaling. Under the joint row scaling $g_k \mapsto \alpha_k g_k$ and $b_k \mapsto \alpha_k b_k$, with $\alpha_k > 0$, the corresponding scaled multiplier satisfies

$$\widetilde{\lambda}_k^\star = \frac{\lambda_k^\star}{\alpha_k}. \tag{38}$$

Consequently, the budget-weighted pressure $b_k \lambda_k^\star$ is invariant under this joint scaling.

For linear coupling constraints $\mathbf{D}x \le b$, let $\mathbf{d}_i$ denote the ith column of $\mathbf{D}$. The aggregate dual pressure associated with component x_i is

$$\rho_i = \mathbf{d}_i^\top \lambda^\star. \tag{39}$$

Representation interpretation. Complementary slackness gives

$$\lambda_k^\star \left(g_k(x^\star; \xi) - b_k \right) = 0, \tag{40}$$

so a strictly inactive constraint has zero dual weight, while a nonzero multiplier identifies a binding constraint and, at differentiability points of the value function, a nonzero marginal sensitivity. Together, $\lambda^\star$, the budget-weighted pressures $b_k \lambda_k^\star$, and the component pressures ρ_i provide structured coordinates of constraint activity, marginal sensitivity, and coupled resource pressure.

Raw dual variables may nevertheless be nonunique or numerically sensitive near redundant constraints, active-set changes, or failures of regularity. We therefore audit both their KKT consistency and the stability of the derived pressure coordinates below.

Table 1: **Numerical audit of KKT pressure semantics and solver stability.** p95 denotes the empirical 95th percentile, not a confidence interval. Shadow-price, reconstruction, and pressure errors are relative.

Audit	Metric	Result
KKT conditions	Primal violation (p95)	4.04×10^{-8}
	Stationarity residual (p95)	1.06×10^{-4}
	Complementarity residual (p95)	5.57×10^{-6}
	Dual violation (max)	0
Shadow price	Relative error (median / p95)	$\mathbf{3.32 \times 10^{-5}}$ / $\mathbf{8.43 \times 10^{-4}}$
	Sign agreement	$\mathbf{100}\%$
Row scaling	Raw multiplier ratio	$\widetilde{\lambda}_k^\star / \lambda_k^\star \simeq 1/\alpha_k$
	Budget-weighted pressure ratio	$(\widetilde{b}_k \widetilde{\lambda}_k^\star)/(b_k \lambda_k^\star) \simeq \mathbf{1}$
Component pressure	Reconstruction error (p95), f / d	4.75×10^{-6} / 5.78×10^{-6}
Cross-solver	Pressure error (p95), CLARABEL / SCS	6.94×10^{-6} / 1.46×10^{-4}
	Active-set agreement	$\mathbf{100}\%$
Constraint order	Pressure error (p95)	3.29×10^{-6}
	Active-set agreement	$\mathbf{100}\%$

Numerical audit. Table 1 audits the solver-derived pressure coordinates using 200 nominal resource-allocation instances, with 60 instances used for solver and constraint-order comparisons. ECOS provides the reference solutions. The KKT rows report feasibility, stationarity, and complementarity residuals. The shadow-price audit compares active multipliers with negative finite-difference derivatives of the optimal value, pooling relative budget perturbations of 0.1%, 0.3%, and 1%.

Under joint positive rescaling of a constraint row and its budget, nonzero raw multipliers change as $\widetilde{\lambda}_k^\star \simeq \lambda_k^\star/\alpha_k$, while budget-weighted pressures remain approximately invariant. The component-pressure audit compares pressures computed directly from the duals with those reconstructed from the stationarity conditions for compute allocations f and downlink allocations d. Reconstruction errors are evaluated over the 673 non-padded requests in the nominal instances. Cross-solver errors compare budget-weighted pressures from CLARABEL and SCS with the ECOS reference; constraint-order errors compare interleaved and canonical constraint orderings under ECOS.

Relative errors divide the absolute difference, or Euclidean distance for vectors, by the larger compared magnitude, with denominator floors of 10^{-8} for scalar errors and 10^{-10} for vector errors. These audits support numerical consistency and pressure semantics on the tested instances; they do not establish a learning advantage.

B Proofs and Supporting Results

B.1 Proof of Proposition 1

Proof. Recall that

$$R = Z - B\Psi, \qquad B = \Sigma_{Z\Psi}\Sigma_{\Psi\Psi}^{-1}. \tag{41}$$

Since Ψ and Z are centered,

$$\mathbb{E}[R] = \mathbb{E}[Z] - B\mathbb{E}[\Psi] = 0. \tag{42}$$

Moreover,

$$\begin{aligned}\mathbb{E}\left[R\Psi^\top\right] &= \mathbb{E}\left[(Z - B\Psi)\Psi^\top\right] \\ &= \Sigma_{Z\Psi} - B\Sigma_{\Psi\Psi} \\ &= \Sigma_{Z\Psi} - \Sigma_{Z\Psi}\Sigma_{\Psi\Psi}^{-1}\Sigma_{\Psi\Psi} \\ &= 0.\end{aligned} \tag{43}$$

For arbitrary $w \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^{d_z}$, it follows that

$$\begin{aligned}\left\langle w^\top \Psi, \gamma^\top R \right\rangle_{L_2} &= \mathbb{E}\left[(w^\top \Psi)(\gamma^\top R)\right] \\ &= w^\top \mathbb{E}\left[\Psi R^\top\right] \gamma \\ &= 0. \end{aligned} \tag{44}$$

Therefore,

$$\mathcal{F} \perp \mathcal{S}_R. \tag{45}$$

Since $Z = B\Psi + R$, every element of $\mathrm{span}(\Psi, Z)$ can be expressed as the sum of an element of $\mathrm{span}(\Psi)$ and an element of $\mathcal{S}_R$. Conversely, $R = Z - B\Psi$ belongs to $\mathrm{span}(\Psi, Z)$. Hence,

$$\mathcal{H} = \mathrm{span}(\Psi, Z) = \mathcal{F} \oplus \mathcal{S}_R. \tag{46}$$

The orthogonal decomposition in Eq. 46 implies

$$P_{\mathcal{H}} m = P_{\mathcal{F}} m + P_{\mathcal{S}_R} m. \tag{47}$$

Consequently,

$$m - P_{\mathcal{F}} m = P_{\mathcal{S}_R} m + (m - P_{\mathcal{H}} m). \tag{48}$$

The two terms on the right-hand side are orthogonal because $P_{\mathcal{S}_R} m \in \mathcal{H}$, whereas $m - P_{\mathcal{H}} m \perp \mathcal{H}$. The Pythagorean identity therefore gives

$$\|m - P_{\mathcal{F}} m\|_{L_2}^2 = \|P_{\mathcal{S}_R} m\|_{L_2}^2 + \|m - P_{\mathcal{H}} m\|_{L_2}^2. \tag{49}$$

Using the definition of the best-in-class risks,

$$\mathcal{R}_{\mathcal{F}}^\star - \mathcal{R}_{\mathcal{H}}^\star = \|P_{\mathcal{S}_R} m\|_{L_2}^2. \tag{50}$$

Finally, because $P_{\mathcal{S}_R} m \in \mathcal{S}_R$, there exists $\gamma \in \mathbb{R}^{d_z}$ such that

$$P_{\mathcal{S}_R} m = \gamma^\top R. \tag{51}$$

It follows that

$$\begin{aligned}\|P_{\mathcal{S}_R} m\|_{L_2}^2 &= \mathbb{E}\left[(\gamma^\top R)^2\right] \\ &= \gamma^\top \mathbb{E}\left[R R^\top\right] \gamma \\ &= \gamma^\top \Sigma_R \gamma. \end{aligned} \tag{52}$$

Since $\mathbb{E}[R] = 0$, $\Sigma_R = \mathrm{Cov}(R)$. The gain is strictly positive if and only if $P_{\mathcal{S}_R} m \neq 0$ in L_2. □

B.2 Proof of Theorem 1

Proof. Throughout the proof, we condition on the realized inputs and the induced design matrices. Thus, $\mathbf{m}$, F, H, P_F, and P_H are fixed within this conditional analysis, while the expectation is taken over the conditional label noise.

For any $G \in \{F, H\}$, using $\widehat{\mathbf{m}}_G = P_G \mathbf{y}$ and $\mathbf{y} = \mathbf{m} + \boldsymbol{\varepsilon}$ gives

$$\begin{aligned}\widehat{\mathbf{m}}_G - \mathbf{m} &= P_G(\mathbf{m} + \boldsymbol{\varepsilon}) - \mathbf{m} \\ &= -(I_n - P_G)\mathbf{m} + P_G \boldsymbol{\varepsilon}. \end{aligned} \tag{53}$$

Because P_G is an orthogonal projector,

$$P_G^\top = P_G, \qquad P_G^2 = P_G, \qquad (I_n - P_G) P_G = 0. \tag{54}$$

Therefore, the two terms on the right-hand side of Eq. 53 are orthogonal, and

$$\|\widehat{\mathbf{m}}_G - \mathbf{m}\|_2^2 = \|(I_n - P_G)\mathbf{m}\|_2^2 + \|P_G \boldsymbol{\varepsilon}\|_2^2. \tag{55}$$

Taking the conditional expectation yields

$$\begin{aligned}\mathbb{E}\left[\|P_G\boldsymbol{\varepsilon}\|_2^2 \;\middle|\; X_{1:n}\right] &= \mathbb{E}\left[\boldsymbol{\varepsilon}^\top P_G^\top P_G \boldsymbol{\varepsilon} \mid X_{1:n}\right] \\ &= \mathbb{E}\left[\boldsymbol{\varepsilon}^\top P_G \boldsymbol{\varepsilon} \mid X_{1:n}\right] \\ &= \operatorname{tr}\left(P_G \mathbb{E}\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top \mid X_{1:n}\right]\right) \\ &= \operatorname{tr}\left(P_G \Sigma_\varepsilon\right), \end{aligned} \tag{56}$$

where the final equality follows from $\mathbb{E}[\boldsymbol{\varepsilon} \mid X_{1:n}] = \mathbf{0}$ and the definition of Σ_ε.

Substituting Eq. 56 into Eq. 55 and dividing by n gives

$$\mathcal{E}_G^{(n)} = \frac{\|(I_n - P_G)\mathbf{m}\|_2^2}{n} + \frac{\operatorname{tr}\left(P_G \Sigma_\varepsilon\right)}{n}, \tag{57}$$

which proves Eq. 13.

We next compare the baseline and augmented designs. Since

$$\operatorname{col}(F) \subseteq \operatorname{col}(H), \tag{58}$$

their orthogonal projectors satisfy

$$P_H P_F = P_F P_H = P_F. \tag{59}$$

Define

$$\tilde{D} := P_H - P_F. \tag{60}$$

Using Eq. 59,

$$\begin{aligned}\tilde{D}^2 &= (P_H - P_F)^2 \\ &= P_H^2 - P_H P_F - P_F P_H + P_F^2 \\ &= P_H - P_F = \tilde{D}. \end{aligned} \tag{61}$$

Moreover, $\tilde{D}^\top = \tilde{D}$. Hence, $\tilde{D} = P_H - P_F$ is the orthogonal projector onto the additional directions in the augmented design space.

The residual projector for the baseline space decomposes as

$$I_n - P_F = (I_n - P_H) + (P_H - P_F). \tag{62}$$

The two projectors on the right-hand side are orthogonal because

$$\begin{aligned}(I_n - P_H)(P_H - P_F) &= P_H - P_F - P_H^2 + P_H P_F \\ &= 0. \end{aligned} \tag{63}$$

Therefore, the Pythagorean identity gives

$$\|(I_n - P_F)\mathbf{m}\|_2^2 = \|(I_n - P_H)\mathbf{m}\|_2^2 + \|(P_H - P_F)\mathbf{m}\|_2^2. \tag{64}$$

Subtracting the risk decomposition for H from that for F and applying Eq. 64, we obtain

$$\begin{aligned}\mathcal{E}_F^{(n)} - \mathcal{E}_H^{(n)} = &\frac{\|(P_H - P_F)\mathbf{m}\|_2^2}{n} \\ &- \frac{\operatorname{tr}\left[(P_H - P_F)\Sigma_\varepsilon\right]}{n}. \end{aligned} \tag{65}$$

This proves Eq. 14. Consequently, the augmented estimator has lower conditional in-design function-estimation risk if and only if the margin in Eq. 14 is positive.

Finally, suppose that the conditional noise is isotropic:

$$\Sigma_\varepsilon = \sigma_\varepsilon^2 I_n. \tag{66}$$

Then

$$\begin{aligned}\frac{\operatorname{tr}\left[(P_H - P_F)\Sigma_\varepsilon\right]}{n} &= \frac{\sigma_\varepsilon^2 \operatorname{tr}\left(P_H - P_F\right)}{n} \\ &= \frac{\sigma_\varepsilon^2 \left[\operatorname{tr}(P_H) - \operatorname{tr}(P_F)\right]}{n} \\ &= \frac{\sigma_\varepsilon^2 \left[\operatorname{rank}(H) - \operatorname{rank}(F)\right]}{n}, \end{aligned} \tag{67}$$

where we used the fact that the trace of an orthogonal projector equals its rank and that $\operatorname{rank}(P_G) = \operatorname{rank}(G)$. □

B.3 Proof of Corollary 1

Proof. Under the centered-feature convention, both ordinary least-squares estimators are fitted without an intercept. Since $\operatorname{rank}(\Sigma_R) = r$, throughout this proof we work in the r-dimensional effective residual subspace of R, and continue to use R, γ, and Σ_R for the corresponding reduced coordinates.

Let

$$\mathbf{\Psi}_n := \begin{bmatrix} \Psi_1^\top \\ \vdots \\ \Psi_n^\top \end{bmatrix} \in \mathbb{R}^{n\times p}, \qquad \mathbf{R}_n := \begin{bmatrix} R_1^\top \\ \vdots \\ R_n^\top \end{bmatrix} \in \mathbb{R}^{n\times r}, \tag{68}$$

and define the augmented design matrix

$$\mathbf{W}_n := [\mathbf{\Psi}_n \quad \mathbf{R}_n] \in \mathbb{R}^{n\times(p+r)}. \tag{69}$$

Since (Ψ, R) is jointly Gaussian and $\mathbb{E}[R\Psi^\top] = 0$, the random vectors Ψ and R are independent. The covariance of the augmented feature vector is therefore

$$\Sigma_W := \begin{bmatrix} \Sigma_{\Psi\Psi} & 0 \\ 0 & \Sigma_R \end{bmatrix} \succ 0. \tag{70}$$

We first calculate the risk of the baseline estimator. The response model can be written as

$$\mathbf{y} = \mathbf{\Psi}_n\theta + \mathbf{u}, \qquad \mathbf{u} := \mathbf{R}_n\gamma + \boldsymbol{\varepsilon}, \tag{71}$$

where $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_n)^\top$. Because R is independent of Ψ and ε is independent of (Ψ, R), the effective baseline noise $\mathbf{u}$ is independent of $\mathbf{\Psi}_n$ and has variance

$$\tau_F^2 := \operatorname{Var}\left(\gamma^\top R + \varepsilon\right) = \gamma^\top\Sigma_R\gamma + \sigma^2 = \Delta_{\text{struct}} + \sigma^2. \tag{72}$$

The baseline OLS coefficient estimator is

$$\widehat{\theta} = \left(\mathbf{\Psi}_n^\top\mathbf{\Psi}_n\right)^{-1}\mathbf{\Psi}_n^\top\mathbf{y}, \tag{73}$$

and hence

$$\widehat{\theta} - \theta = \left(\mathbf{\Psi}_n^\top\mathbf{\Psi}_n\right)^{-1}\mathbf{\Psi}_n^\top\mathbf{u}. \tag{74}$$

Conditional on $\mathbf{\Psi}_n$,

$$\operatorname{Cov}\left(\widehat{\theta} - \theta \mid \mathbf{\Psi}_n\right) = \tau_F^2\left(\mathbf{\Psi}_n^\top\mathbf{\Psi}_n\right)^{-1}. \tag{75}$$

For an independent test context X', let Ψ' and R' denote the corresponding baseline and residual coordinates. The baseline prediction is $\widehat{m}_{\mathcal{F}}(X') = \widehat{\theta}^\top\Psi'$. Its error relative to the regression function is

$$\widehat{m}_{\mathcal{F}}(X') - m(X') = (\widehat{\theta} - \theta)^\top\Psi' - \gamma^\top R'. \tag{76}$$

The two terms on the right-hand side are uncorrelated because Ψ' and R' are independent and centered. Therefore,

$$\begin{aligned}\mathfrak{R}_{\mathcal{F}}^{\text{out}} &= \mathbb{E}\left[\left((\widehat{\theta} - \theta)^\top\Psi'\right)^2\right] + \mathbb{E}\left[(\gamma^\top R')^2\right] \\ &= \mathbb{E}\left[\left((\widehat{\theta} - \theta)^\top\Psi'\right)^2\right] + \Delta_{\text{struct}}.\end{aligned} \tag{77}$$

Conditioning on the training design and using Eq. 75,

$$\begin{aligned}&\mathbb{E}\left[\left((\widehat{\theta} - \theta)^\top\Psi'\right)^2 \,\middle|\, \mathbf{\Psi}_n\right] \\ &\qquad = \tau_F^2\operatorname{tr}\left[\Sigma_{\Psi\Psi}\left(\mathbf{\Psi}_n^\top\mathbf{\Psi}_n\right)^{-1}\right].\end{aligned} \tag{78}$$

Since the rows of $\mathbf{\Psi}_n$ are independent Gaussian vectors,

$$\mathbf{\Psi}_n^\top\mathbf{\Psi}_n \sim \operatorname{Wishart}_p\left(\Sigma_{\Psi\Psi}, n\right). \tag{79}$$

For $n > p + 1$,

$$\mathbb{E}\left[\left(\mathbf{\Psi}_n^\top \mathbf{\Psi}_n\right)^{-1}\right] = \frac{\Sigma_{\Psi\Psi}^{-1}}{n-p-1}. \tag{80}$$

Consequently,

$$\begin{aligned}\mathfrak{R}_{\mathcal{F}}^{\text{out}} &= \Delta_{\text{struct}} + \tau_F^2 \frac{\operatorname{tr}\left(\Sigma_{\Psi\Psi}\Sigma_{\Psi\Psi}^{-1}\right)}{n-p-1} \\ &= \Delta_{\text{struct}} + \left(\sigma^2 + \Delta_{\text{struct}}\right)\frac{p}{n-p-1}.\end{aligned} \tag{81}$$

This proves Eq. 17.

We next consider the augmented estimator. Define

$$\vartheta := \begin{bmatrix}\theta \\ \gamma\end{bmatrix} \in \mathbb{R}^{p+r}. \tag{82}$$

The realizable augmented model is

$$\mathbf{y} = \mathbf{W}_n \vartheta + \boldsymbol{\varepsilon}. \tag{83}$$

The augmented OLS estimator satisfies

$$\widehat{\vartheta} - \vartheta = \left(\mathbf{W}_n^\top \mathbf{W}_n\right)^{-1} \mathbf{W}_n^\top \boldsymbol{\varepsilon}, \tag{84}$$

and, conditional on $\mathbf{W}_n$,

$$\operatorname{Cov}\left(\widehat{\vartheta} - \vartheta \mid \mathbf{W}_n\right) = \sigma^2 \left(\mathbf{W}_n^\top \mathbf{W}_n\right)^{-1}. \tag{85}$$

Let

$$W' := \begin{bmatrix}\Psi' \\ R'\end{bmatrix}. \tag{86}$$

The augmented prediction error is

$$\widehat{m}_{\mathcal{H}}(X') - m(X') = (\widehat{\vartheta} - \vartheta)^\top W'. \tag{87}$$

Therefore,

$$\mathfrak{R}_{\mathcal{H}}^{\text{out}} = \sigma^2 \mathbb{E}\left[\operatorname{tr}\left(\Sigma_W \left(\mathbf{W}_n^\top \mathbf{W}_n\right)^{-1}\right)\right]. \tag{88}$$

Because the rows of $\mathbf{W}_n$ are independent Gaussian vectors with covariance Σ_W,

$$\mathbf{W}_n^\top \mathbf{W}_n \sim \operatorname{Wishart}_{p+r}\left(\Sigma_W, n\right). \tag{89}$$

When $n > p + r + 1$,

$$\mathbb{E}\left[\left(\mathbf{W}_n^\top \mathbf{W}_n\right)^{-1}\right] = \frac{\Sigma_W^{-1}}{n-p-r-1}. \tag{90}$$

It follows that

$$\begin{aligned}\mathfrak{R}_{\mathcal{H}}^{\text{out}} &= \sigma^2 \frac{\operatorname{tr}\left(\Sigma_W \Sigma_W^{-1}\right)}{n-p-r-1} \\ &= \sigma^2 \frac{p+r}{n-p-r-1},\end{aligned} \tag{91}$$

which proves Eq. 18.

Subtracting Eq. 91 from Eq. 81 gives

$$\begin{aligned}\mathfrak{R}_{\mathcal{F}}^{\text{out}} - \mathfrak{R}_{\mathcal{H}}^{\text{out}} &= \Delta_{\text{struct}} + \left(\sigma^2 + \Delta_{\text{struct}}\right)\frac{p}{n-p-1} - \sigma^2 \frac{p+r}{n-p-r-1} \\ &= \frac{n-1}{n-p-1}\left[\Delta_{\text{struct}} - \frac{\sigma^2 r}{n-p-r-1}\right].\end{aligned} \tag{92}$$

This proves Eq. 19.

Since $n > p + r + 1$, the multiplicative factor $(n-1)/(n-p-1)$ is strictly positive. Hence,

$$\mathfrak{R}_{\mathcal{F}}^{\text{out}} > \mathfrak{R}_{\mathcal{H}}^{\text{out}} \tag{93}$$

if and only if

$$\Delta_{\text{struct}} > \frac{\sigma^2 r}{n-p-r-1}. \tag{94}$$

This establishes Eq. 20. □

B.4 Transfer from Fixed-Design to Population Risk

Beyond the Gaussian reference model, a uniform comparison between empirical and population norms provides a sufficient condition for transferring the conditional in-design risk ordering to conditional population risk. Let μ denote the population distribution of X, and define

$$\|h\|_n^2 := \frac{1}{n}\sum_{i=1}^{n} h(X_i)^2, \qquad \|h\|_\mu^2 := \mathbb{E}_{X\sim\mu}[h(X)^2]. \tag{95}$$

For $\mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}$, let $\widehat{m}_\mathcal{G} \in \mathcal{G}$ denote the fitted function whose evaluations on $X_{1:n}$ give the fitted vector $\widehat{\mathbf{m}}_G$, and define

$$e_\mathcal{G} := \widehat{m}_\mathcal{G} - m. \tag{96}$$

$$\mathcal{E}_{\mathcal{G},\mu}^{(n)} := \mathbb{E}\big[\|e_\mathcal{G}\|_\mu^2 \mid X_{1:n}\big], \qquad \mathcal{E}_\mathcal{G}^{(n)} := \mathbb{E}\big[\|e_\mathcal{G}\|_n^2 \mid X_{1:n}\big]. \tag{97}$$

$$\mathfrak{M}_n := \mathcal{E}_\mathcal{F}^{(n)} - \mathcal{E}_\mathcal{H}^{(n)}, \qquad \mathfrak{S}_n := \mathcal{E}_\mathcal{F}^{(n)} + \mathcal{E}_\mathcal{H}^{(n)}. \tag{98}$$

Proposition 2 (Realizable transfer certificate)**.** *Suppose that $m \in \mathcal{H}$ and that, for some $0 < \eta < 1$,*

$$(1-\eta)\|h\|_\mu^2 \le \|h\|_n^2 \le (1+\eta)\|h\|_\mu^2, \qquad \forall h \in \mathcal{H}. \tag{99}$$

Then

$$\frac{\mathfrak{M}_n - \eta\mathfrak{S}_n}{1-\eta^2} \le \mathcal{E}_{\mathcal{F},\mu}^{(n)} - \mathcal{E}_{\mathcal{H},\mu}^{(n)} \le \frac{\mathfrak{M}_n + \eta\mathfrak{S}_n}{1-\eta^2}. \tag{100}$$

In particular, $\mathfrak{M}_n > \eta\mathfrak{S}_n$ certifies lower conditional population risk for the augmented estimator, whereas $\mathfrak{M}_n < -\eta\mathfrak{S}_n$ certifies the reverse ordering.

Proof. Since $\mathcal{H}$ is a linear space, $\widehat{m}_\mathcal{F} \in \mathcal{F} \subseteq \mathcal{H}$, $\widehat{m}_\mathcal{H} \in \mathcal{H}$, and $m \in \mathcal{H}$ imply that both $e_\mathcal{F}$ and $e_\mathcal{H}$ belong to $\mathcal{H}$. Hence Eq. 99 gives

$$\frac{\|e_\mathcal{G}\|_n^2}{1+\eta} \le \|e_\mathcal{G}\|_\mu^2 \le \frac{\|e_\mathcal{G}\|_n^2}{1-\eta}, \qquad \mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}. \tag{101}$$

Taking conditional expectations yields

$$\frac{\mathcal{E}_\mathcal{G}^{(n)}}{1+\eta} \le \mathcal{E}_{\mathcal{G},\mu}^{(n)} \le \frac{\mathcal{E}_\mathcal{G}^{(n)}}{1-\eta}, \qquad \mathcal{G} \in \{\mathcal{F}, \mathcal{H}\}. \tag{102}$$

Combining the lower bound for $\mathcal{F}$ with the upper bound for $\mathcal{H}$ gives

$$\begin{aligned}\mathcal{E}_{\mathcal{F},\mu}^{(n)} - \mathcal{E}_{\mathcal{H},\mu}^{(n)} &\ge \frac{\mathcal{E}_\mathcal{F}^{(n)}}{1+\eta} - \frac{\mathcal{E}_\mathcal{H}^{(n)}}{1-\eta} \\ &= \frac{\mathfrak{M}_n - \eta\mathfrak{S}_n}{1-\eta^2}.\end{aligned} \tag{103}$$

Similarly,

$$\begin{aligned}\mathcal{E}_{\mathcal{F},\mu}^{(n)} - \mathcal{E}_{\mathcal{H},\mu}^{(n)} &\le \frac{\mathcal{E}_\mathcal{F}^{(n)}}{1-\eta} - \frac{\mathcal{E}_\mathcal{H}^{(n)}}{1+\eta} \\ &= \frac{\mathfrak{M}_n + \eta\mathfrak{S}_n}{1-\eta^2}.\end{aligned} \tag{104}$$

These inequalities establish Eq. 100.

Since $1-\eta^2 > 0$, the lower bound is positive whenever $\mathfrak{M}_n > \eta\mathfrak{S}_n$, implying

$$\mathcal{E}_{\mathcal{H},\mu}^{(n)} < \mathcal{E}_{\mathcal{F},\mu}^{(n)}.$$

Likewise, if $\mathfrak{M}_n < -\eta\mathfrak{S}_n$, the upper bound is negative and hence

$$\mathcal{E}_{\mathcal{F},\mu}^{(n)} < \mathcal{E}_{\mathcal{H},\mu}^{(n)}.$$

□

B.5 PROOF OF THE FIXED-POLICY VALUE-LEARNING BRIDGE

Proof. We first establish the contraction and projected fixed-point properties. For any $f \in L_2(\mu^\pi)$, Jensen's inequality gives

$$\begin{aligned}\|P^\pi f\|_{2,\mu^\pi}^2 &= \mathbb{E}_{(S,A)\sim\mu^\pi}\left[\left(\mathbb{E}\left[f(S',A') \mid S,A\right]\right)^2\right] \\ &\le \mathbb{E}_{(S,A)\sim\mu^\pi}\left[\mathbb{E}\left[f(S',A')^2 \mid S,A\right]\right] \\ &= \mathbb{E}_{(S',A')\sim\mu^\pi}\left[f(S',A')^2\right] \\ &= \|f\|_{2,\mu^\pi}^2 , \end{aligned} \tag{105}$$

where the third line follows from the stationarity of μ^π. Thus, P^π is nonexpansive in $L_2(\mu^\pi)$.

Since the orthogonal projector $\Pi_\mathcal{G}$ is also nonexpansive, for any $Q, Q' \in L_2(\mu^\pi)$,

$$\begin{aligned}\|\Pi_\mathcal{G}\mathcal{T}^\pi Q - \Pi_\mathcal{G}\mathcal{T}^\pi Q'\|_{2,\mu^\pi} &= \gamma\,\|\Pi_\mathcal{G}P^\pi(Q-Q')\|_{2,\mu^\pi} \\ &\le \gamma\,\|Q-Q'\|_{2,\mu^\pi} . \end{aligned} \tag{106}$$

Hence, $\Pi_\mathcal{G}\mathcal{T}^\pi$ is a γ-contraction and admits a unique fixed point $Q_\mathcal{G} \in \mathcal{G}$.

We next prove Eq. 22. Because $\Pi_\mathcal{G}Q^\pi$ is the best approximation of Q^π in $\mathcal{G}$, and $Q_\mathcal{G} \in \mathcal{G}$,

$$\begin{aligned} a_\mathcal{G} &= \|(I-\Pi_\mathcal{G})Q^\pi\|_{2,\mu^\pi} \\ &= \inf_{g\in\mathcal{G}} \|g - Q^\pi\|_{2,\mu^\pi} \\ &\le \|Q_\mathcal{G} - Q^\pi\|_{2,\mu^\pi} . \end{aligned} \tag{107}$$

Let

$$d_\mathcal{G} := Q_\mathcal{G} - Q^\pi. \tag{108}$$

Using $Q_\mathcal{G} = \Pi_\mathcal{G}\mathcal{T}^\pi Q_\mathcal{G}$ and $Q^\pi = \mathcal{T}^\pi Q^\pi$, we obtain

$$\begin{aligned} d_\mathcal{G} &= \Pi_\mathcal{G}\mathcal{T}^\pi Q_\mathcal{G} - Q^\pi \\ &= \Pi_\mathcal{G}\left(\mathcal{T}^\pi Q_\mathcal{G} - \mathcal{T}^\pi Q^\pi\right) + \Pi_\mathcal{G}Q^\pi - Q^\pi \\ &= \gamma\Pi_\mathcal{G}P^\pi d_\mathcal{G} - (I-\Pi_\mathcal{G})Q^\pi. \end{aligned} \tag{109}$$

The first term on the right-hand side belongs to $\mathcal{G}$, whereas the second belongs to $\mathcal{G}^\perp$. They are therefore orthogonal. Consequently,

$$\begin{aligned}\|d_\mathcal{G}\|_{2,\mu^\pi}^2 &= \gamma^2\,\|\Pi_\mathcal{G}P^\pi d_\mathcal{G}\|_{2,\mu^\pi}^2 + \|(I-\Pi_\mathcal{G})Q^\pi\|_{2,\mu^\pi}^2 \\ &\le \gamma^2\,\|d_\mathcal{G}\|_{2,\mu^\pi}^2 + a_\mathcal{G}^2, \end{aligned} \tag{110}$$

where nonexpansiveness of both $\Pi_\mathcal{G}$ and P^π was used in the inequality. Rearranging gives

$$\|Q_\mathcal{G} - Q^\pi\|_{2,\mu^\pi} \le \frac{a_\mathcal{G}}{\sqrt{1-\gamma^2}}. \tag{111}$$

Together with Eq. 107, this establishes Eq. 22.

We now prove the fitted-iteration bound. For fitted Q evaluation, write

$$\widehat{Q}_{k+1} = \Pi_\mathcal{G}\mathcal{T}^\pi\widehat{Q}_k + e_k, \tag{112}$$

where e_k denotes the regression error relative to the exact projected Bellman update. Subtracting the projected fixed-point equation $Q_\mathcal{G} = \Pi_\mathcal{G}\mathcal{T}^\pi Q_\mathcal{G}$ from Eq. 112 gives

$$\begin{aligned}\widehat{Q}_{k+1} - Q_\mathcal{G} &= \Pi_\mathcal{G}\mathcal{T}^\pi\widehat{Q}_k - \Pi_\mathcal{G}\mathcal{T}^\pi Q_\mathcal{G} + e_k \\ &= \gamma\Pi_\mathcal{G}P^\pi\left(\widehat{Q}_k - Q_\mathcal{G}\right) + e_k. \end{aligned} \tag{113}$$

Taking norms and using nonexpansiveness yields

$$\left\|\widehat{Q}_{k+1} - Q_\mathcal{G}\right\|_{2,\mu^\pi} \le \gamma\left\|\widehat{Q}_k - Q_\mathcal{G}\right\|_{2,\mu^\pi} + \|e_k\|_{2,\mu^\pi} . \tag{114}$$

Iterating this recursion gives

$$\left\|\widehat{Q}_K - Q_{\mathcal{G}}\right\|_{2,\mu^\pi} \le \gamma^K \left\|\widehat{Q}_0 - Q_{\mathcal{G}}\right\|_{2,\mu^\pi} + \sum_{j=0}^{K-1} \gamma^{K-1-j} \left\|e_j\right\|_{2,\mu^\pi}. \tag{115}$$

Finally, the triangle inequality and Eq. 111 imply

$$\begin{aligned}\left\|\widehat{Q}_K - Q^\pi\right\|_{2,\mu^\pi} &\le \left\|\widehat{Q}_K - Q_{\mathcal{G}}\right\|_{2,\mu^\pi} + \|Q_{\mathcal{G}} - Q^\pi\|_{2,\mu^\pi} \\ &\le \gamma^K \left\|\widehat{Q}_0 - Q_{\mathcal{G}}\right\|_{2,\mu^\pi} + \sum_{j=0}^{K-1} \gamma^{K-1-j} \left\|e_j\right\|_{2,\mu^\pi} + \frac{a_{\mathcal{G}}}{\sqrt{1-\gamma^2}}.\end{aligned} \tag{116}$$

This proves Eq. 23.

Finally, for nested value spaces $\mathcal{F} \subseteq \mathcal{H}$, the best-approximation residuals satisfy

$$\begin{aligned}a_{\mathcal{H}} &= \inf_{h \in \mathcal{H}} \|h - Q^\pi\|_{2,\mu^\pi} \\ &\le \inf_{f \in \mathcal{F}} \|f - Q^\pi\|_{2,\mu^\pi} = a_{\mathcal{F}}.\end{aligned} \tag{117}$$

Thus, enlarging the value-function space weakly reduces the best-approximation residual,

$$a_{\mathcal{H}} \le a_{\mathcal{F}},$$

while the additional representation may increase the finite-sample regression errors accumulated across fitted iterations. Together with Eq. 23, this shows how the critic-side approximation–estimation trade-off propagates to fixed-policy value learning: solver augmentation improves the approximation component whenever it enlarges the useful value space, but its overall benefit depends on whether this gain offsets the additional fitted-regression error. □

Conditional one-step regression interpretation. Fix an iterate $\widehat{Q}_k$. Let $Y_i^{(k)}$ denote the Bellman regression target at input X_i, and define

$$m_k(X) := \mathcal{T}^\pi \widehat{Q}_k(X), \qquad \varepsilon_{k,i} := Y_i^{(k)} - m_k(X_i). \tag{118}$$

Conditional on $\widehat{Q}_k$ and the realized inputs $X_{1:n}$, suppose that

$$\mathbb{E}\left[\boldsymbol{\varepsilon}_k \mid X_{1:n}, \widehat{Q}_k\right] = \mathbf{0}_n. \tag{119}$$

Then m_k, the design matrices, and the associated projection operators are fixed under this conditioning, so Theorem 1 applies to the comparison of the baseline and augmented representations for this common one-step Bellman regression target. Independent unbiased target samples, or appropriate sample splitting or cross-fitting, can ensure the required conditional mean-zero property; repeated reuse of the same data may violate it.

B.6 Proof of Theorem 2

Proof. The argument is local. In particular, Eq. 27 and the matched total-derivative identity require differentiability of T at the reference action, whereas the reverse inclusion in Eq. 28 applies on any neighborhood where T is C^1. Hence, for a piecewise-C^1 solver map, the same conclusions hold within each smooth region; no claim is made at regime-switching boundaries. Let

$$h(a, z) := \widetilde{q}(a, z) - q(a, z). \tag{120}$$

Since $q \sim_T \widetilde{q}$, the difference vanishes identically on the solver graph:

$$h(a, T(a)) = 0, \qquad \forall a \in \mathcal{U}. \tag{121}$$

Differentiating Eq. 121 with respect to a and applying the chain rule gives

$$\nabla_a h(a, z)|_{z=T(a)} + J_T(a)^\top \nabla_z h(a, T(a)) = 0. \tag{122}$$

Therefore,

$$\begin{aligned}&(\nabla_a \widetilde{q}(a,z) - \nabla_a q(a,z))|_{z=T(a)} \\ &\quad = -J_T(a)^\top \left[\nabla_z \widetilde{q}(a,T(a)) - \nabla_z q(a,T(a))\right]. \end{aligned} \tag{123}$$

The right-hand side belongs to $\mathrm{Range}(J_T(a)^\top)$, which establishes Eq. 27. Since this holds for every $\widetilde{q} \sim_T q$, it also implies

$$\mathcal{A}_{\mathrm{semi}}(a;q) \subseteq \mathrm{Range}\left(J_T(a)^\top\right). \tag{124}$$

We next establish the reverse inclusion for the unrestricted C^1 extension class. For any $v \in \mathbb{R}^{d_z}$, consider the graph-vanishing perturbation

$$h_v(a',z) = v^\top \left(z - T(a')\right). \tag{125}$$

Since T is continuously differentiable, h_v is C^1, and hence $\widetilde{q}_v := q + h_v$ is an admissible C^1 critic extension. For every $a' \in \mathcal{U}$,

$$h_v(a', T(a')) = v^\top \left(T(a') - T(a')\right) = 0. \tag{126}$$

Hence, the critic

$$\widetilde{q}_v := q + h_v \tag{127}$$

is solver-graph equivalent to q.

Holding z fixed, the partial action gradient of the perturbation is

$$\nabla_{a'} h_v(a',z) = -J_T(a')^\top v. \tag{128}$$

Thus, at the reference action a,

$$\begin{aligned}&(\nabla_a \widetilde{q}_v(a,z) - \nabla_a q(a,z))|_{z=T(a)} \\ &\quad = -J_T(a)^\top v. \end{aligned} \tag{129}$$

As v ranges over $\mathbb{R}^{d_z}$,

$$\left\{-J_T(a)^\top v : v \in \mathbb{R}^{d_z}\right\} = \mathrm{Range}\left(J_T(a)^\top\right). \tag{130}$$

Therefore,

$$\mathrm{Range}\left(J_T(a)^\top\right) \subseteq \mathcal{A}_{\mathrm{semi}}(a;q). \tag{131}$$

Combining Eqs. 124 and 131 yields

$$\mathcal{A}_{\mathrm{semi}}(a;q) = \mathrm{Range}\left(J_T(a)^\top\right), \tag{132}$$

which proves Eq. 28.

Finally, because the two composed critics agree on the solver graph,

$$q(a,T(a)) = \widetilde{q}(a,T(a)), \qquad \forall a \in \mathcal{U}, \tag{133}$$

their derivatives with respect to a must also agree. Applying the chain rule to both sides gives

$$\begin{aligned}&\nabla_a q(a,z)|_{z=T(a)} + J_T(a)^\top \nabla_z q(a,T(a)) \\ &\quad = \nabla_a \widetilde{q}(a,z)|_{z=T(a)} + J_T(a)^\top \nabla_z \widetilde{q}(a,T(a)). \end{aligned} \tag{134}$$

This establishes Eq. 30. Thus, exact values on the solver graph uniquely determine the derivative along the graph, whereas the detached partial action derivative remains ambiguous within $\mathrm{Range}(J_T(a)^\top)$. □

C Experimental Protocols for Main-Text Results

Common experimental controls. We collect the representation variants and actor-update interfaces used throughout the main text and appendices. Each experiment uses the subset specified in its protocol. In the same-information representation comparisons, every method retains the complete baseline context X; variants change the auxiliary features supplied to the predictor or critic, or remove the auxiliary branch.

- **Context:** uses X with a zero-filled auxiliary block, retaining the auxiliary branch for architectural matching.
- **Exact dual:** uses (X, Z), where $Z = T(X)$ is the matched deterministic solver representation. Dual and Context + Dual denote the corresponding dual-augmented variants in experiment-specific figures and tables; they retain the baseline context.
- **Predicted:** replaces Z with $\widehat{Z}(X)$ from a predictor trained only on the corresponding training split, testing surrogate recovery of the solver representation.
- **Shuffled:** permutes exact solver representations across samples, preserving their empirical marginal distribution while breaking their samplewise alignment with X.
- **Random:** uses an independent random vector with the same dimensionality as Z, controlling for arbitrary auxiliary inputs.
- **Parameter-matched (Matched):** uses deterministic nuisance features of X with the same auxiliary dimensionality and architecture as Exact dual but without solver-specific structure, controlling for generic deterministic feature expansion.
- **X-only:** uses X without the auxiliary branch, isolating the effect of the additional representation pathway.

For the resource-sensitivity experiment (Appendix D.2), the following variants also retain the complete nominal context:

- **Primal summary:** adds aggregate allocation, cancellation, cost, delay, and slack statistics from the nominal solver solution.
- **Full primal:** adds the complete nominal primal solution together with the Primal summary features.
- **Primal + dual:** augments Full primal with the matched dual features.
- **Primal + shuffled dual:** augments Full primal with shuffled dual features, testing whether gains depend on their alignment with the nominal context.
- **Active-set summary:** adds constraint-binding indicators and family-level binding summaries at the nominal solution.

For the actor-gradient experiments, $X = (s, a)$ and $Z = T(s, a)$. The following names distinguish actor-update interfaces:

- **Context:** uses a zero-filled auxiliary block and a reparameterized actor update.
- **Detached:** uses the dual-augmented critic but holds Z fixed during actor differentiation, retaining only the partial derivative with respect to the action.
- **Total:** differentiates through $Z = T(s, a)$, including the solver-response contribution to the actor gradient.
- **Score:** uses a likelihood-ratio actor update without differentiating the critic or solver representation with respect to the action.

The extension coefficient κ indexes graph-equivalent critic extensions within each interface; it is not an additional representation variant.

For MASAC (Appendix D.6), zero-valued, random, and mismatched auxiliary representations provide the corresponding controls. Mismatched representations break the correspondence between the current critic input and its solver representation, as specified in that protocol. The observation-based (o, a) setting is not a strict same-information comparison when o omits optimizer-side inputs. The complete-context comparison separately tests shared, invertible observation scaling.

Within each fixed-data comparison, methods use the same contexts, targets, train–evaluation splits, normalization rules, and optimization budgets unless stated otherwise. Online control comparisons use matched training and evaluation protocols and paired seeds; collected trajectories and training targets may differ as policies evolve. Experiment-specific architecture choices and matching rules are reported with each protocol.

Controlled characterization. Figures 2a– 2b isolate the critic-side structural-gain mechanism through a synthetic same-information construction. We sample $X \sim \mathcal{N}(0, I_8)$ and define the deterministic representation $Z = AX + Lh(X)$, with regression target $m(X) = \theta^\top X + \gamma^\top Z + c\, w(X)$. The six components of $h(X)$ and the additional function $w(X)$ are mutually orthonormal quadratic Hermite features, orthogonal to affine functions of X. For the affine baseline class, AX is recoverable, while $R = Lh(X)$ is the residual representation with $\Sigma_R = LL^\top$. Across 230 controlled configurations, we vary residual structure and target alignment, thereby varying $\Delta_{\mathrm{struct}} = \gamma^\top \Sigma_R \gamma$. For each configuration, the analytic structural gain is compared with the oracle projection-risk gap evaluated on $50{,}000$ independently sampled contexts. Fig. 2a compares these quantities across configurations, while Fig. 2b varies residual magnitude and target relevance separately to isolate their respective effects. Fig. 2c separately tests the Gaussian reference crossover. We fix $p = 8$, $r = 1$, $\sigma^2 = 0.25$, and $\Delta_{\mathrm{struct}} = 0.04$, and evaluate $n \in \{12, 14, 16, 18, 24, 32, 50, 100\}$. For each sample size, 1000 independent training sets are drawn from the same population, and the baseline and augmented estimators are fitted to matched samples and evaluated out of sample under the common population. The empirical risk ordering is then compared with the theoretical crossover $n^\star = 16.25$.

Neural and value learning. The prediction experiments use a synthetic eight-demand allocation task. The context contains all demands and the shared budget, and the solver projects the demand vector onto nonnegative allocations satisfying that budget. The solver representation is the shared-budget multiplier. The regression target is negative total unmet demand, with additive Gaussian label noise of standard deviation 0.25; held-out function MSE is evaluated against the noise-free target. For Fig. 3a, each replicate draws training, validation, and test sets independently from this population. Context, Exact dual, and Random inputs are constructed from the same underlying samples. The neural predictors are trained under paired random seeds and evaluated on the common test set. We use $n \in \{100, 200, 500, 1000, 3000, 10000, 30000\}$ and 10 paired seeds for each sample size. Fig. 3b compares regression classes on the same controlled population. Linear regression is fitted directly, the frozen-feature model fits a prediction layer on a fixed nonlinear representation, and the end-to-end MLP jointly optimizes all trainable parameters. Within each function class, Context and Exact dual use matched samples and are compared through held-out function MSE. For Fig. 3c, a sequential version of the allocation task uses autoregressive demands and budget gaps, a fixed stochastic budget-adjustment policy, and noisy rewards penalizing unmet demand and squared adjustment. We generate a transition dataset under this policy and compute the solver representation for each decision context. Context, Exact dual, Predicted, Shuffled, Random, and Matched inputs are constructed from the same transition data according to the control definitions above. Each critic is trained by neural FQE on the same fixed-policy dataset, and the resulting value functions are evaluated against discounted Monte Carlo returns at held-out episode-initial states under the same policy. The FQE comparison uses 10 paired seeds.

Actor-gradient identifiability. For Fig. 4a, we construct graph-equivalent extensions with $\kappa \in \{-3, -1, 0, 1, 3\}$ from a common reference critic. For each of ten seeds, the same batch of 256 samples is used for all extensions and interfaces. We verify on-graph value agreement and measure relative actor gradient change $\|G_\kappa - G_0\|_2 / \|G_0\|_2$; the total derivative is also checked against finite differences. A separate frozen-critic experiment (Fig. 6) uses 20 paired actor-training replicates per interface and $\kappa \in \{-2, -1, 0, 1, 2\}$, with matched initializations and update budgets, holding the critic and update data fixed. It measures RMS deterministic-action differences from the same interface at $\kappa = 0$ on common probes. Figs. 4b and 4c report the joint online actor–critic experiment with ten paired seeds: return improvement and policy sensitivity, respectively. Its protocol is given in Appendix D.4.1.

Full-RL validation. Fig. 5a evaluates the representation effect in a controlled sequential decision task. The controlled and ACN experiments use dueling Double-DQN with discrete capacity-allocation actions and state-dependent nominal dual features. The Q-network receives the complete state and the auxiliary representation and outputs a value for each action. At each decision step, an inner convex optimization problem determines the resource allocation and returns its KKT dual variables, so that the controlled task explicitly exposes which resource constraint is most binding under the current decision context. Exact dual supplies these solver duals to the critic, Context retains the same auxiliary branch with a zero-valued input, Random replaces the duals with a dimension-matched random input, and X-only removes the auxiliary branch. The environment, solver calls,

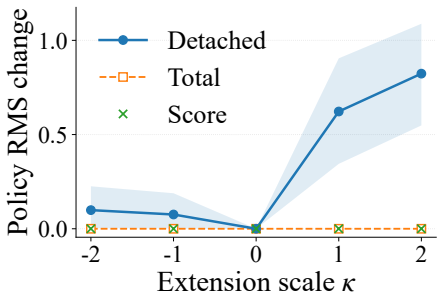


Figure 6: Frozen-critic policy-update experiment. Actors are updated using a fixed critic and update data. Policy sensitivity is the RMS deterministic-action difference from the corresponding $\kappa = 0$ policy on common probes.

Q-network architectures, interaction budgets, and optimization budgets are otherwise matched, with paired seeds and matched initialization used for comparison. This setting provides a controlled end-to-end test of whether the critic-side representation benefit survives repeated policy and value updates rather than only one-step regression. Fig. 5b evaluates the same question under realistic workload variation using ACN charging traces. The ACN traces provide the time-varying charging workload and are replayed on a fixed coupled-resource charging network constructed for the experiment; the network topology itself is therefore held fixed while the workload varies according to the recorded traces. The same Double-DQN and inner-solver pipeline is then trained and evaluated under these replayed workloads. May and June traces are used for training, July is reserved for development, and September is used only for the formal evaluation. The model configuration, formal evaluation protocol, and ten RL seeds are fixed before access to the September evaluation data, preventing the formal split from influencing model selection or tuning. Fig. 5c provides a complementary view of learning dynamics using the July development split. The already specified training runs are evaluated at 12 fixed checkpoints, without retraining on the development data or using the September formal evaluation split. Exact dual, Context, and Random are evaluated with common checkpoint and seed protocols, and learning efficiency is summarized by the area under the evaluation-return curve as a function of training interactions. The checkpoint curves are reported directly without smoothing or manually enforcing monotonicity, so this panel serves as a development-side diagnostic of learning dynamics rather than an additional formal test set.

Key implementation settings for the prediction, neural FQE, and Double-DQN experiments are summarized in Table 2. Settings for the controlled characterization and local actor-gradient audits are given above; the online actor–critic and MASAC protocols are detailed in Appendices D.4.1 and D.6, respectively.

D Additional Experimental Details and Results

D.1 Neural Target-Relevance and Representation Specificity

The main-text experiments establish that deterministic solver representations can improve neural prediction and fixed-policy value estimation. Here, we remove Bellman recursion and use end-to-end neural regression to isolate two representation-level questions: how the benefit changes with target relevance, and whether it is specific to the matched solver representation rather than to an arbitrary auxiliary feature.

For Fig. 7a, we use the residual solver coordinate $R = Z - B\Psi$ from the critic-side analysis and construct a family of regression targets of the form $m_\alpha(X) = m_0(X) + \alpha\,\gamma^\top R(X)$. We vary $\alpha \in \{0, 0.25, 0.5, 1\}$ while keeping the sampled contexts, network architecture, optimization budget, and paired randomness fixed. Thus, α directly controls the contribution of the residual solver structure to the target and is not a correlation coefficient. For each setting, Context and Exact dual are trained on matched data, and the paired reduction in held-out function MSE is reported. As shown in Fig. 7a, the benefit increases as the residual–target coupling becomes stronger, consistent with the relevance mechanism identified by the critic-side analysis. A nonzero gain remains at $\alpha = 0$. This

Table 2: Key implementation settings for the prediction, neural FQE, and Double-DQN comparisons. Each listed comparison uses ten paired seeds. Data counts are per seed; network sizes denote hidden-layer widths. "lr" denotes learning rate.

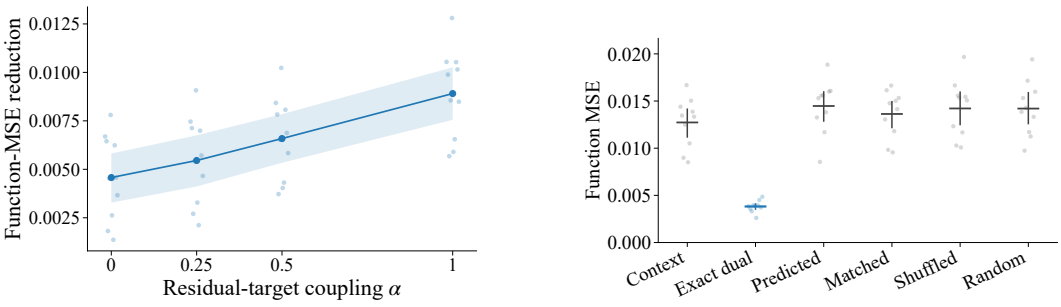
Experiment	Learner	Data and evaluation	Optimization
Sample-size scaling (Fig. 3a)	ReLU MLP, 128×128	$n = 100$–$30{,}000$ training; $5{,}000$ validation; $30{,}000$ test	AdamW, lr 10^{-3}; batch $\min(256, n)$; update cap $\max(400, \lceil 100n/256 \rceil)$
Function classes: linear (Fig. 3b)	Linear regression with intercept	500 training; $20{,}000$ test	Direct ridge solve; coefficient 10^{-10}
Function classes: frozen features	Context plus 8 fixed tanh features; linear readout	500 training; $20{,}000$ test	Direct ridge solve; coefficient 10^{-4}
Function classes: neural	ReLU MLP, 32×32	500 training; $1{,}000$ validation; $20{,}000$ test	AdamW, lr 10^{-3}; batch 64; at most 600 updates
Fixed-policy FQE (Fig. 3c)	ReLU MLP, 32×32	$125 \times 40 = 5{,}000$ training transitions; 300 evaluation episodes of 40 steps	AdamW, lr 5×10^{-4}; batch 256; 10 Bellman iterations, 100 updates each; $\gamma = 0.5$
Controlled full RL (Fig. 5a)	Dueling Double-DQN; 64 actions	$60{,}000$ environment steps; 24 monitoring and 100 final evaluation episodes, each 80 steps	AdamW, lr 3×10^{-4}; batch 256; replay $50{,}000$; $\gamma = 0.99$
ACN full RL (Fig. 5b)	Dueling Double-DQN; 16 actions	$60{,}000$ environment steps; 8 July development days; September test traces	Adam, lr 3×10^{-4}; batch 256; replay $100{,}000$; $\gamma = 0.97$

Neural regressors select the checkpoint with the lowest validation label MSE; FQE uses the final Bellman iterate, with primary MSE evaluated against discounted Monte Carlo returns at the 300 episode-initial states (one rollout per state). The Double-DQN networks use a 512–384–256 state encoder, a 128–128 auxiliary branch, and width-128 dueling heads, all with ReLU; X-only omits the auxiliary branch. Both RL experiments use $4{,}000$ warm-up steps, one gradient update per environment step thereafter, a target-network copy every 500 steps, and evaluation every $5{,}000$ steps. The ϵ-greedy exploration probability decreases linearly from 1 to 0.05 over $40{,}000$ steps. For the neural models listed here, gradient-norm clipping is 10; weight decay is 10^{-4} for regression/FQE and 10^{-5} for full RL.

does not imply a positive structural gain when the residual is target-irrelevant; rather, it shows that finite-budget end-to-end neural training can exhibit additional optimization or conditioning effects beyond the exact linear and frozen-feature decomposition. For Fig. 7b, we fix $\alpha = 1$ and compare the original held-out function MSE across Exact dual, Context, Predicted, Matched, Shuffled, and Random. All controls use the same underlying samples and matched training budgets, with their auxiliary inputs constructed according to the definitions in Appendix C. The matched solver representation achieves substantially lower error than the alternative controls. This comparison therefore isolates representation specificity before Bellman recursion, complementing the fixed-policy FQE result in the main text.

D.2 Initial-state resource sensitivity

We evaluate nominal solver representations for a specified resource-sensitivity target in a continuous linear-programming (LP) relaxation of airline recovery. The model uses 2025 flight schedules for carrier DL from the U.S. Bureau of Transportation Statistics (BTS) Reporting Carrier On-Time Performance dataset, together with weather observations from NOAA's ISD-Lite archive at ATL, DTW, MSP, SLC, JFK, LGA, BOS, SEA, LAX, and MCO. Baseline capacities and initial-delay distributions are estimated from January–May data. Passenger-weight proxies use monthly route frequencies and flight distances, with the priority threshold calibrated on January–May data. Costs and slack penalties are modeling assumptions. The target is an optimal-cost response in this model, not an observed airline loss or an RL return.



(a) (b)

Figure 7: Neural target relevance and representation specificity.

Each example is a freshly initialized recovery window. Start times are sampled uniformly with replacement from the 21 half-hour grid points between 06:00 and 16:00, inclusive, with scenario-specific random draws. Earlier scheduled departures are treated as cleared, no aircraft state is carried over from prior execution, and remaining flights receive initial delays sampled from the development-period distribution. The LP includes up to 64 pending flights, an eight-slot horizon with 30-minute slots, five delay options per flight (0–4 slots), cancellation variables, and slack variables for priority and global-budget constraints. The model uses airport-local clock conventions. Examples are sampled only at initialization, not along sequential policy trajectories.

Let X denote the complete nominal LP context, b the vector of right-hand-side values for the departure-capacity, arrival-capacity, gate-capacity, and cancellation-budget constraints, and $J_X^*(b)$ the minimum cost with all other fields held fixed. The prediction target is

$$Y_\epsilon(X) = \frac{J_X^*((1-\epsilon)b) - J_X^*(b)}{\epsilon}, \qquad \epsilon \in \{0.02, 0.05, 0.10\}. \tag{135}$$

The delay budget is unchanged. Nominal features are computed once and held fixed while each tightened LP is solved separately. Optimal objective values from the tightened solves are used only to construct labels; tightened solver outputs are not used as input features. Labels are neither clipped nor filtered by magnitude. The primary tightening level is 0.02; 0.05 and 0.10 are prespecified robustness conditions. A separate predictor is trained for each tightening level.

The 946-dimensional context contains all variable fields determining the LP: flight masks, airport indices, schedules, durations, weights, priorities, predecessor and readiness information, capacities, and budgets. The remaining coefficients are fixed by the configuration. Under the fixed solver convention, nominal solver outputs are deterministic functions of X, so adding them as features preserves the information set.

Dual has 31 coordinates. For each airport and each of the three capacity families, it sums capacity–price products over the horizon, using nonnegative inequality prices; the final coordinate is the cancellation budget multiplied by its price. This representation is aligned with the tested tightening direction. Primal summary has 35 coordinates: per-airport assigned departures, arrivals, and cancellations, together with nominal cost, total delay in slots, summed priority slack, and the two global-budget slacks. Full primal supplies all 450 padded primal variables together with this summary, including nominal cost. Active-set summary is a 508-dimensional representation containing binding indicators and family-level binding summaries. Inequality binding is determined using a slack tolerance of 10^{-7}. This representation is not a complete optimal-basis description.

Every predictor receives the complete context X. The eight learned variants are Context (all auxiliary blocks zero-filled), Dual, Primal summary, Full primal, Primal + dual, Active-set summary, Shuffled dual, and Primal + shuffled dual. Primal + dual and Primal + shuffled dual augment Full primal with matched and shuffled dual features, respectively. All variants use 1,970 input coordinates and fill disabled blocks with zeros, giving identical total parameter counts while enabling different feature blocks. For each seed and data split, shuffling uses a fixed permutation with no fixed points. Training shuffles are restricted to the selected training subset; validation and test shuffles use only features from their respective splits, without labels, as diagnostic controls.

Formal training dates are July 1–24, validation dates July 25–31, and test dates August 1–20. With 48 scenarios per date, these splits contain 1,152, 336, and 960 examples, respectively. Each of the 20 seeds (100–119) selects 1,000 training examples without replacement; the selected subset is shared across variants and tightening levels. June pilot examples are excluded. Each feature is centered using its selected training-subset mean and divided by the larger of its training standard deviation and one. Targets use the same training-only normalization rule. No feature clipping is applied. Reported training, validation, and test MSEs use the original target units.

All neural predictors have two hidden layers of width 128 with SiLU activations and 268,929 parameters. They are trained with AdamW for 1,500 updates, using a learning rate of 3×10^{-4}, weight decay of 10^{-4}, batch size 128, and a gradient-norm clipping threshold of five. Initialization, training subsets, minibatch sequences, and update budgets are paired across variants, yielding $8 \times 3 \times 20 = 480$ fits. Training and validation MSE are recorded at initialization and every 100 updates. Primary results use the checkpoint with the lowest validation MSE among recorded steps 100–1,500, breaking ties in favor of the earliest checkpoint. Step zero is used only for diagnostics. Models at step 1,500 are retained as a secondary robustness check. Test data are not used to select either the estimator or the checkpoint.

Test MSE is computed in the original target units by averaging squared errors within each date and then weighting the 20 test dates equally. Table 3 reports means and sample standard deviations across seeds. Paired reductions subtract candidate MSE from reference MSE within each seed. Tables 4 and 6 report seven contrasts at each of the three tightening levels. Individual two-sided 95% Student-t confidence intervals are computed from 20 seed replicates. Holm-adjusted p-values cover all 21 comparisons separately for each estimator; the step-1,500 estimator is secondary. Percentage reductions reported in the text are computed as $100(1 - \overline{M}_{\mathrm{cand}}/\overline{M}_{\mathrm{ref}})$, where the bars denote means across seeds, rather than averaging per-seed percentages.

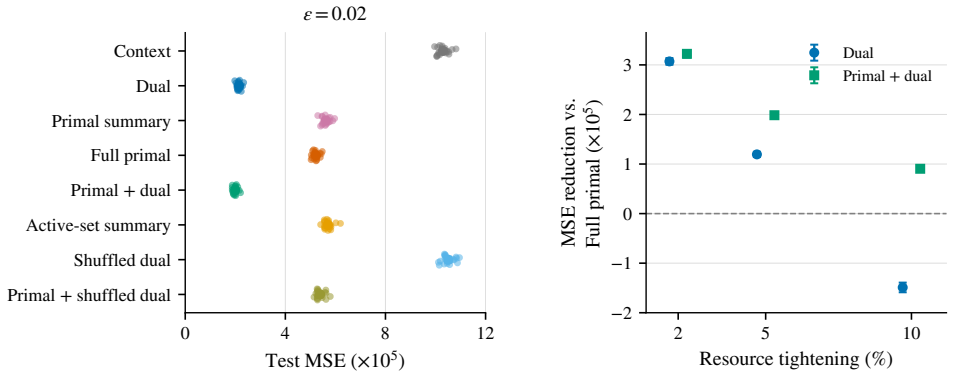
Seed intervals condition on the fixed date dataset. Separately, we average paired effects across the fitted seed ensemble and bootstrap the 20 test dates using 10,000 resamples. These date intervals condition on the fitted models and are not joint seed-and-date intervals. Table 5 reports the three central comparisons and positive-date counts.

At 2%, Dual lowers mean test MSE relative to Context, Primal summary, and Full primal by 79.1%, 61.8%, and 58.8% (Fig. 8). Primal + dual lowers Full-primal MSE by 61.8% and Primal + shuffled dual MSE by 63.0%. These comparisons improve in all 20 seed pairs. Dual versus Full primal and both incremental comparisons also improve on all 20 dates after averaging across seeds. At 5%, Dual retains a 22.5% mean advantage over Full primal, positive on 17 of 20 dates. At 10%, Dual alone has 33.0% higher mean MSE than Full primal. Nevertheless, adding Dual to Full primal lowers MSE by 37.5% and 20.0% at 5% and 10%, improving in every seed pair and on every seed-averaged test date.

Figure 9 shows the test-MSE sweep and checkpoint robustness. At step 1,500, Dual lowers Full-primal MSE by 57.0% and 6.0% at 2% and 5%; Primal + dual lowers it by 61.8%, 34.1%, and 21.8% across the three amplitudes. These comparisons improve in all seed pairs. Figure 10 shows training and validation errors at 2%: all methods substantially reduce training error, but their validation errors differ. Of 480 fits, 293 select step 100, motivating the fixed-budget diagnostics. Under primary model selection, Context improves on a training-mean constant predictor at every amplitude (Table 3).

Two non-neural baselines are also evaluated. The pressure-sum baseline predicts $\widehat{Y}_{\mathrm{press}}(X) = \mathbf{1}^{\top} Z(X)$ using the raw, unstandardized dual-pressure features. This is a dual-based first-order estimate for the specified tightening direction and requires no predictor training. The training-mean constant predicts the mean training label for the corresponding seed and tightening level for every test example. Both baselines use only nominal features or training labels, respectively. They are reported separately from the learned representation variants.

At 2%, the pressure-sum baseline has MSE $171{,}192$, below all learned variants in Table 3, including the learned Dual predictor's $214{,}775$. Its error increases with tightening amplitude, and Primal + dual has lower MSE at 5% and 10% (Fig. 9). This diagnostic makes the target's connection to classical LP sensitivity explicit. The experiment supports finite-sample learning benefits and complementarity with primal features for this target, without implying universal Dual superiority or higher control returns.



(a) Representation controls.

(b) Gains over full primal.

Figure 8: Solver representations for initial-state cost-sensitivity prediction. (a) Test MSE at 2% resource tightening; dots show 20 training seeds and larger markers show means with individual 95% seed t intervals. (b) Paired MSE reductions relative to Full primal across tightening levels; positive values favor the candidate. Bars are individual 95% seed t intervals. Horizontal offsets distinguish the candidates. All learned methods also receive the complete nominal context and use validation-selected models.

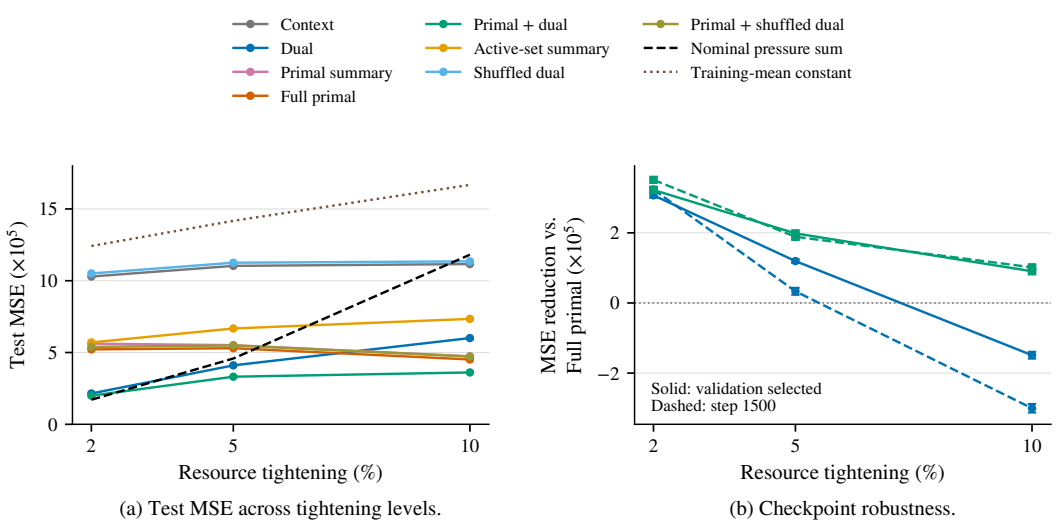
Table 3: Test sensitivity MSE ($\times 10^5$; lower is better). Learned variants use validation-selected models and report mean $\pm$ sample SD across 20 training seeds. Bold marks the lowest mean MSE among learned variants in each column. Non-neural baselines are shown separately: the pressure sum is deterministic on the fixed test data, and the constant row reports the mean MSE across seed-specific training-mean predictors.

Representation	$\epsilon = 0.02$	0.05	0.10
Learned representation variants			
Context	10.293 ± 0.229	11.035 ± 0.166	11.160 ± 0.200
Dual	2.148 ± 0.086	4.105 ± 0.107	6.006 ± 0.156
Primal summary	5.617 ± 0.169	5.513 ± 0.256	4.731 ± 0.150
Full primal	5.218 ± 0.129	5.299 ± 0.125	4.516 ± 0.170
Primal + dual	$\mathbf{1.995} \pm 0.091$	$\mathbf{3.315} \pm 0.077$	$\mathbf{3.612} \pm 0.116$
Active-set summary	5.703 ± 0.179	6.675 ± 0.248	7.343 ± 0.283
Shuffled dual	10.507 ± 0.248	11.254 ± 0.206	11.350 ± 0.241
Primal + shuffled dual	5.392 ± 0.174	5.492 ± 0.122	4.723 ± 0.214
Non-neural baselines			
Nominal pressure sum	1.712	4.583	11.812
Training-mean constant	12.417	14.172	16.677

Recorded maxima over nominal and tightened solves are 1.11×10^{-16} for equality residuals, 3.55×10^{-15} for inequality violations, 1.78×10^{-15} for stationarity residuals, and 1.36×10^{-12} for primal–dual gaps. All 480 sets of saved predictions and validation selections were checked against their records.

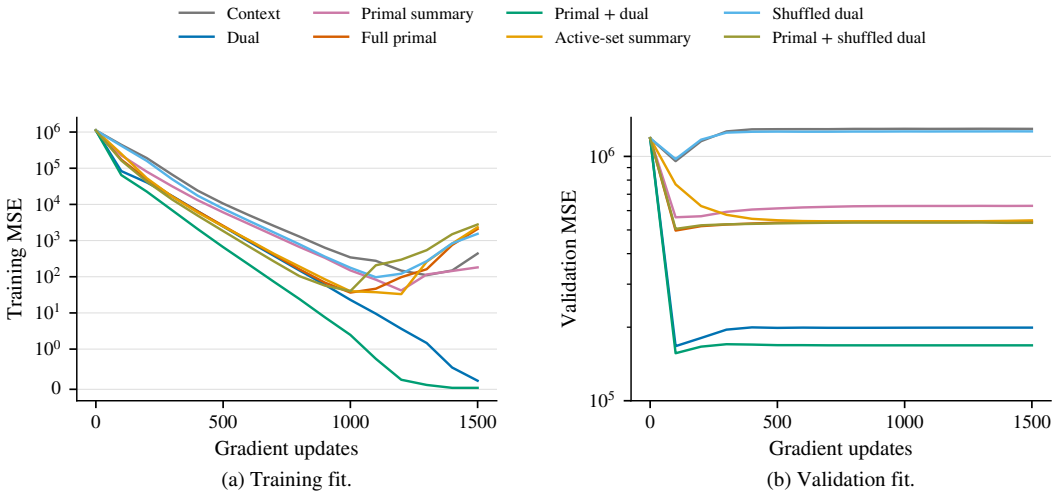
D.3 Application-Backed PowerGrid Evaluation

We further evaluate the critic-side representation effect on dynamic power system benchmarks constructed from modified PYPOWER/MATPOWER Case30 and Case57 systems Zimmerman et al. (2011); Yetkin et al. (2024). At each decision step, a convex quadratic DC optimal power flow (DC-OPF) problem determines the resource allocation subject to load, renewable, network, and intertemporal constraints. The evaluated policy is fixed throughout the FQE experiments. Context uses the complete state–action context $X_t = (s_t, a_t)$, whereas Exact dual additionally receives the deterministic KKT pressure representation $Z_t = T(X_t)$ extracted from the corresponding DC-OPF solution.



(a) Test MSE across tightening levels. (b) Checkpoint robustness.

Figure 9: Sensitivity-target diagnostics. (a) Mean test MSE at each of the three tightening levels, with nominal pressure-sum and training-mean constant baselines; learned curves use validation-selected models. (b) Paired gains over Full primal for Dual (blue circles) and Primal + dual (green squares): solid lines use validation selection and dashed lines use step 1,500. Bars are individual 95% seed t intervals; lines connect tested amplitudes only.



(a) Training fit. (b) Validation fit.

Figure 10: Mean training and validation MSE across 20 seeds at 2% tightening. All scheduled checkpoints are shown without smoothing. The training axis is linear up to MSE one and logarithmic beyond it; the validation axis is logarithmic. These are fitting diagnostics, not RL learning curves.

For both systems, trajectories are generated under the same fixed policy and split into training, validation, and test sets. The compared FQE critics are trained under paired data, initialization, and evaluation conditions, and value-estimation error is measured against Monte Carlo returns on held-out trajectories. Case30 serves as the primary benchmark, while Case57 provides a larger cross-case evaluation with a different network scale and constraint structure. The benchmark dimensions, data split, optimization budget, and compared representation modes are summarized in Table 7.

Fig. 11a evaluates representation specificity on Case30 using initial-state value MSE. Exact dual achieves a 28.6% lower error than Context and also outperforms the Shuffled and Random controls, indicating that the gain is associated with the matched solver representation rather than an arbitrary auxiliary input. Fig. 11b repeats the evaluation on the larger Case57 system. Exact dual again

Table 4: Validation-selected paired MSE reductions ($\times 10^5$), reference minus candidate. Intervals are individual 95% seed t intervals. Holm-adjusted p-values cover the 21 comparisons in this table; the step-1500 family is secondary. C: Context; D: Dual; PS: Primal summary; P: Full primal; PD: Primal + dual; A: Active-set summary; SD: Shuffled dual; PSD: Primal + shuffled dual.

ϵ (%)	Cand./ref.	Reduction	Seed 95% CI	p_{Holm}	Positive seeds
2	D / C	8.145	$[8.035, 8.255]$	1.3×10^{-29}	20/20
2	D / PS	3.470	$[3.398, 3.542]$	3.2×10^{-26}	20/20
2	D / P	3.070	$[3.003, 3.138]$	9.1×10^{-26}	20/20
2	D / A	3.555	$[3.474, 3.636]$	1.7×10^{-25}	20/20
2	D / SD	8.359	$[8.232, 8.486]$	9.2×10^{-29}	20/20
2	PD / P	3.223	$[3.149, 3.297]$	1.9×10^{-25}	20/20
2	PD / PSD	3.396	$[3.299, 3.493]$	9.4×10^{-24}	20/20
5	D / C	6.930	$[6.847, 7.013]$	1.3×10^{-30}	20/20
5	D / PS	1.408	$[1.290, 1.526]$	1.2×10^{-15}	20/20
5	D / P	1.195	$[1.142, 1.247]$	2.2×10^{-20}	20/20
5	D / A	2.570	$[2.458, 2.682]$	2.2×10^{-20}	20/20
5	D / SD	7.150	$[7.042, 7.258]$	9.2×10^{-29}	20/20
5	PD / P	1.985	$[1.930, 2.039]$	4.4×10^{-24}	20/20
5	PD / PSD	2.177	$[2.110, 2.244]$	3.2×10^{-23}	20/20
10	D / C	5.154	$[5.080, 5.227]$	3.3×10^{-29}	20/20
10	D / PS	-1.275	$[-1.352, -1.198]$	6.8×10^{-18}	0/20
10	D / P	-1.490	$[-1.587, -1.392]$	2.2×10^{-17}	0/20
10	D / A	1.336	$[1.218, 1.454]$	1.4×10^{-15}	20/20
10	D / SD	5.344	$[5.239, 5.448]$	1.0×10^{-26}	20/20
10	PD / P	0.904	$[0.850, 0.959]$	6.8×10^{-18}	20/20
10	PD / PSD	1.111	$[1.022, 1.200]$	7.3×10^{-16}	20/20

Table 5: Date-level robustness for the three central contrasts. Date bootstrap intervals ($\times 10^5$) resample 20 test dates after averaging paired effects across the fitted seed ensemble. Positive dates count positive seed-averaged effects. These intervals condition on the fitted models and are not joint seed-and-date intervals. Abbreviations follow Table 4.

ϵ (%)	Cand./ref.	Date bootstrap 95% CI	Positive dates
2	D / P	$[2.568, 3.558]$	20/20
2	PD / P	$[2.808, 3.622]$	20/20
2	PD / PSD	$[2.971, 3.802]$	20/20
5	D / P	$[0.669, 1.727]$	17/20
5	PD / P	$[1.617, 2.355]$	20/20
5	PD / PSD	$[1.797, 2.552]$	20/20
10	D / P	$[-2.086, -0.885]$	2/20
10	PD / P	$[0.724, 1.089]$	20/20
10	PD / PSD	$[0.919, 1.308]$	20/20

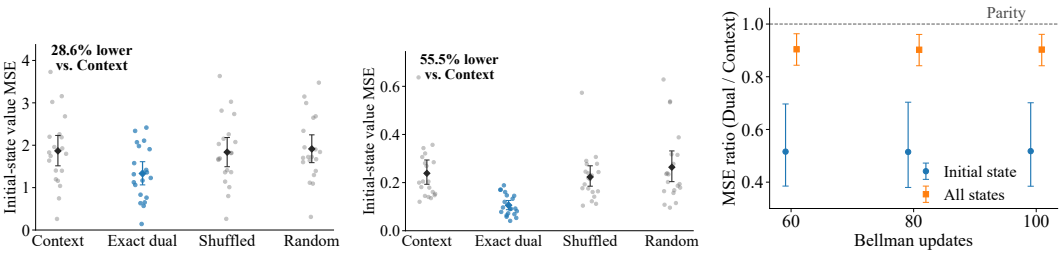
achieves the lowest value-estimation error, with a 55.5% reduction relative to Context. This result shows that the observed representation benefit persists under a different grid scale and constraint structure. Fig. 11c evaluates sensitivity to the FQE optimization budget on Case57. The plotted metric is the MSE ratio between Exact dual and Context, so values below one favor Exact dual. For both initial-state and all-state evaluation, the ratio remains below one across all tested budgets, with a consistently larger reduction for the initial-state metric.

D.4 Additional Diagnostics for Actor-Gradient Identifiability

Within each diagnostic, compared configurations use matched settings and paired randomness. When gradient interfaces or extension fields are varied for a fixed critic, the critic parameters and evaluation samples are held fixed. For the policy-update experiment, actors start from matched initializations and use the same frozen critic and update data.

Table 6: Fixed step-1500 paired MSE reductions ($\times 10^5$), reference minus candidate. Intervals are individual 95% seed t intervals. Holm-adjusted p-values cover the 21 comparisons in this table; the step-1500 family is secondary. C: Context; D: Dual; PS: Primal summary; P: Full primal; PD: Primal + dual; A: Active-set summary; SD: Shuffled dual; PSD: Primal + shuffled dual.

ϵ (%)	Cand./ref.	Reduction	Seed 95% CI	p_{Holm}	Positive seeds
2	D / C	10.089	$[9.940, 10.238]$	7.2×10^{-29}	20/20
2	D / PS	3.559	$[3.487, 3.630]$	1.8×10^{-26}	20/20
2	D / P	3.235	$[3.159, 3.311]$	2.8×10^{-25}	20/20
2	D / A	3.278	$[3.210, 3.345]$	2.8×10^{-26}	20/20
2	D / SD	10.080	$[9.913, 10.246]$	5.6×10^{-28}	20/20
2	PD / P	3.511	$[3.446, 3.575]$	4.3×10^{-27}	20/20
2	PD / PSD	3.594	$[3.503, 3.686]$	1.1×10^{-24}	20/20
5	D / C	8.475	$[8.315, 8.636]$	6.7×10^{-27}	20/20
5	D / PS	0.555	$[0.480, 0.631]$	9.9×10^{-12}	20/20
5	D / P	0.333	$[0.234, 0.432]$	2.1×10^{-6}	20/20
5	D / A	1.475	$[1.370, 1.580]$	1.1×10^{-16}	20/20
5	D / SD	8.461	$[8.271, 8.650]$	1.2×10^{-25}	20/20
5	PD / P	1.888	$[1.819, 1.956]$	8.1×10^{-22}	20/20
5	PD / PSD	2.063	$[1.976, 2.150]$	1.1×10^{-20}	20/20
10	D / C	6.821	$[6.691, 6.952]$	7.6×10^{-27}	20/20
10	D / PS	-2.819	$[-2.927, -2.712]$	2.1×10^{-21}	0/20
10	D / P	-3.003	$[-3.126, -2.881]$	6.0×10^{-21}	0/20
10	D / A	-0.360	$[-0.488, -0.232]$	1.2×10^{-5}	3/20
10	D / SD	6.729	$[6.565, 6.892]$	5.1×10^{-25}	20/20
10	PD / P	1.016	$[0.958, 1.075]$	3.2×10^{-18}	20/20
10	PD / PSD	1.176	$[1.108, 1.245]$	3.4×10^{-18}	20/20



(a) Case30 specificity (b) Case57 cross-case evaluation (c) Budget robustness

Figure 11: Application-backed PowerGrid value learning.

Fig. 12a compares the magnitude of the detached action component $\|\partial_a q\|_2$ with that of the solver-response component $\|J_T^\top \partial_z q\|_2$ on trained critics. Points near the equal-magnitude reference indicate that the two contributions are of comparable scale. The solver-response component is frequently nonnegligible, showing that detaching the solver pathway can remove a substantial part of the matched actor gradient.

Fig. 12b tests whether the detached-gradient sensitivity persists beyond a single constructed extension. We evaluate 16 independently generated perturbation fields that vanish on the solver graph, while holding the reference critic and evaluation samples fixed. For each field, the vertical axis reports the relative change in the detached semi-gradient per unit extension coefficient κ. Nonzero responses across the independently generated fields show that the detached gradient depends systematically on the off-graph extension rather than on one particular construction. In contrast, the on-graph critic values and the matched total-derivative and score-function interfaces remain invariant.

Fig. 12c examines how policy updates change when the solver-response component $J_T^\top \partial_z q$ is progressively attenuated while the forward critic values are kept unchanged. For $Z = T(s, a)$, define $Z_\eta = \mathrm{sg}(Z) + \eta\big(Z - \mathrm{sg}(Z)\big), \eta \in [0, 1]$, where $\mathrm{sg}(\cdot)$ denotes stop-gradient. For every η, the forward

Table 7: **PowerGrid benchmark structure and fixed-policy FQE protocol.**

Setting	Case30	Case57
Benchmark-specific configuration		
Grid topology	MATPOWER/PYPOWER Case30	MATPOWER/PYPOWER Case57
State dimension	118	212
Action dimension	4	4
Compact KKT dimension	31	31
Matched auxiliary block width	90	158
Matched critic input width	212	374
Primary FQE budget	60×20	60×20
Additional budget audit	–	100×40
Compared modes	C/E/S/R/Z	C/E/S/R/Z; C/E/Z for audit
Dynamic optimization structure		
Inner problem	Convex quadratic DC-OPF	
Exogenous dynamics	Stochastic load, renewables, and line derating	
Sequential coupling	Generator output and storage SOC	
Network constraints	Power balance, branch flow, and thermal limits	
Resource constraints	Generation, ramping, storage, and reserve constraints	
Feasibility safeguards	Renewable curtailment and load shedding	
Pressure families	Nodal-price, congestion, generation, ramp, storage, and reserve	
Fixed-policy FQE protocol		
Policy	Deterministic, dual-free, and fixed during FQE	
Critic context	Complete context $X_t = (s_t, a_t)$	
Dual representation	Compact $Z_t = T(X_t)$; current-step solver information only	
Bellman targets	Frozen preceding critic; detached Bellman targets	
Training pairing	Matched initialization and minibatch ordering	
Data split	200 episodes $\times$ 48 steps; $120/40/40$ train/validation/test	
Evaluation	FQE target $\gamma = 0.5$; 20 paired seeds; Monte Carlo value MSE	

C: Context; E: Exact dual; S: Shuffled; R: Random; Z: Zero. FQE budget denotes Bellman updates $\times$ maximum regression epochs per update. The 31-dimensional compact KKT representation is zero-padded to the case-specific auxiliary block width so that all compared critics have matched input dimensions and architectures.

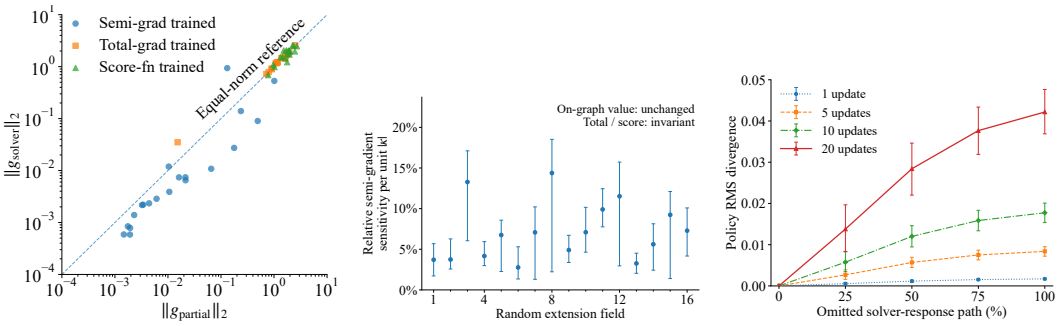
value satisfies $Z_\eta = Z$, while its backward derivative with respect to the action is scaled by η. Thus, this intervention leaves the critic evaluation and direct action path unchanged while continuously attenuating the $J_T^\top \partial_z q$ contribution. The horizontal axis reports the omitted fraction $100(1-\eta)\%$: 0% corresponds to the matched total derivative and 100% to the fully detached semi-gradient. Policy root-mean-square (RMS) divergence measures the action difference from the corresponding total-gradient policy on common probe states. The four curves report this divergence after 1, 5, 10, and 20 actor updates. Policy divergence increases as a larger fraction of the solver-response gradient is removed and becomes more pronounced over repeated updates. These results show that modifying only the backward solver-response pathway, while preserving the forward critic values, can induce a systematic policy-level effect.

D.4.1 Online actor–critic extension test

We use a four-queue allocation task with 24-step episodes. The 15-dimensional state contains normalized queue lengths, service efficiencies, a capacity state, normalized previous allocations, and two phase coordinates. The actor selects four priority coordinates and one shared-budget coordinate, which parameterize

$$x^\star(s,a) = \arg \min_{x \geq 0,\, \mathbf{1}^\top x \leq B(s,a)} \tfrac{1}{2}\|x - d(s,a)\|_2^2. \tag{136}$$

The target allocation $d(s,a)$ is a smooth positive function of queue lengths, efficiencies, previous allocations, and the priority actions, constructed using softplus. The shared budget $B(s,a)$ uses a sigmoid mapping of the capacity state, budget action, and episode phase, with lower and upper limits of 1.5 and 7.5.



(a) Gradient magnitude (b) Extension robustness (c) Gradient-path dose response

Figure 12: Additional diagnostics for actor-gradient identifiability.

Service equals allocation multiplied by efficiency. Queues evolve by adding arrivals and subtracting service, with lengths bounded between zero and 30 and an overflow penalty above 20 before clipping. Arrivals have periodic means and Gaussian noise; efficiencies and capacity follow bounded autoregressive dynamics with periodic targets and Gaussian disturbances. Initial queues, efficiencies, and capacity are sampled randomly, with previous allocations initialized to zero. Rewards penalize backlog, energy, allocation changes, budget use, overflow, and deviation from a reference allocation based on current queue urgency and capacity. The environment construction and numerical parameters are fixed across all interfaces and extensions.

The representation $Z = T(s, a)$ is the elementwise $\log(1 + \cdot)$ transform of one shared-budget and four nonnegativity multipliers. It is deterministic, so $\sigma(s, a, Z) = \sigma(s, a)$. All actors observe the state, and all critics receive (s, a) plus a block of width five. Context fills this block with zeros and uses a reparameterized actor update. Detached holds Z fixed during the actor backward pass; Total differentiates through the piecewise differentiable solver response; Score uses a likelihood-ratio update with a leave-one-out minibatch baseline.

In normalized critic-output units, we use

$$q_{\phi,\kappa}(s, a, z) = q_\phi(s, a, z) + \kappa g(s, a)^\top [z - T(s, a)], \qquad \kappa \in \{-1, 0, 1\}. \tag{137}$$

The field g is a fixed affine map followed by elementwise $\tanh$ and division by $\sqrt{5}$, with $\|g(s, a)\|_2 \leq 1$ and initialization seed 409173. For fixed critic parameters, all extensions agree on the solver graph. We execute a separate training run for each interface and extension, using matched initialization and random streams within each seed. Context is run only at $\kappa = 0$. Ten seeds (0–9) give $10 \times (1 + 3 \times 3) = 100$ formal runs. Once policies diverge, their visited states and fitted critics may also differ.

Each of 80 iterations collects 24 fresh episodes under the current policy and fits the critic to their discounted Monte Carlo returns using 16 minibatch updates. During the first ten iterations, only the critic is updated. Each subsequent iteration additionally performs eight actor updates. Historical return batches are not replayed; network parameters and Adam states persist across iterations. The critic parameters and collected state dataset remain fixed during these eight updates; state minibatches and policy actions are resampled at each update. Each run uses 46,080 training environment steps, 1,280 critic updates, and 560 actor updates. Both networks have two hidden layers of width 128 with SiLU activations, with 19,205 actor and 19,969 critic parameters. The policy is a tanh-transformed Gaussian with fixed pre-transform standard deviation 0.25. Adam learning rates are 3×10^{-4} and 8×10^{-4} for actor and critic, respectively. The minibatch size is 256, the discount factor is 0.95, and the gradient-norm clipping threshold is 10. Discounted Monte Carlo targets are divided by 20 before critic fitting; reported returns remain in the original reward units.

Returns are undiscounted episode reward sums for the stochastic policy. Monitoring uses 40 episodes at initialization, iteration one, and every five iterations, with matched random streams across methods, extensions, and checkpoints within each seed. Initial and final policies are additionally evaluated on 120 matched episodes from streams separate from training and monitoring. These held-out evaluations define the final-minus-initial and final-policy contrasts. Return AUC is the trapezoidal integral of checkpoint returns with respect to training environment steps, divided

by 46,080. Fig. 13 shows checkpoint learning curves, and Fig. 4b shows held-out improvement over initialization. The final monitoring point and the held-out final return therefore use different evaluation episodes. Tables 8 and 9 report all extension settings and paired return contrasts. Confidence intervals are computed from the ten seed replicates using two-sided 95% Student-t intervals. Learning-curve bands are pointwise, and contrast intervals are individual and unadjusted for multiple comparisons.

At each monitoring checkpoint, we record deterministic actions on 256 fixed reachable probe states. These are generated under the neutral policy $a = \tanh(0.25\xi)$, where $\xi \sim \mathcal{N}(0, I_5)$, covering episode steps 0, 6, 12, and 18 equally. The probes are common to all runs and are not used for training. For seed j, let

$$d_{j,t}(\kappa)^2 = \frac{1}{256 \cdot 5} \sum_{i=1}^{256} \|\bar{a}_{j,t,\kappa}(s_i) - \bar{a}_{j,t,0}(s_i)\|_2^2,$$
$$D_{j,t} = \sqrt{\frac{d_{j,t}(-1)^2 + d_{j,t}(1)^2}{2}}. \tag{138}$$

Each comparison uses the same actor interface, and t indexes training checkpoints. The deterministic action is $\bar{a}(s) = \tanh(\mu(s))$, where μ is the actor's pre-transform mean. Fig. 4c shows the mean and pointwise 95% interval of $D_{j,t}$ over ten seeds, with individual final seed values. This display pools the squared differences of both nonzero extensions within each seed; it does not treat extensions as independent replicates. A larger value indicates greater extension sensitivity, not higher return. Fig. 14 retains the separate final policy and return differences for every coefficient.

At each checkpoint, we additionally perform a local extension audit using the first 64 common probes and matched action noise. Within each audit, actor and critic parameters are fixed while κ is varied, allowing on-graph values and actor gradients to be compared without changes in the trained model or evaluation inputs.

Local audits report zero on-graph value differences across extensions at all recorded checkpoints. For Total and Score, the audited gradient differences and recorded policy differences are also zero. Within each of these interfaces and each seed, final actor hashes and final critic hashes match across κ, and evaluation returns are identical. Detached changes both gradients and trained policies; its mean final action RMS differences are 0.680 and 1.000 for $\kappa = -1$ and $+1$, respectively. The confidence intervals for its mean paired return shifts both include zero (Fig. 14(b)); the observed policy sensitivity therefore does not establish a nonzero mean return shift at either tested nonzero coefficient.

At $\kappa = 0$, Total and Score improve over initialization by 9.23 (95% CI: $[4.82, 13.65]$) and 11.62 (95% CI: $[7.67, 15.58]$), respectively, with nine positive seed pairs each. Context and Detached have negative mean gains. Total-minus-Context and Score-minus-Context intervals both include zero (Table 9). These results support the predicted extension invariance under the tested online training protocol. The return measurements complement this mechanism test by documenting learning gains for Total and Score, without establishing a universal performance ranking.

D.5 Supplementary Full-RL Robustness

The main text evaluates full-RL performance through final return, formal real-trace evaluation, and average learning dynamics. The experiments below use the same underlying controlled and real-trace settings but examine complementary robustness aspects that are not shown in the main figures, including training-budget persistence, temporal stability, and seed-level consistency.

Fig. 15a provides a complementary training-budget analysis on the controlled full-RL task. Unlike the main-text learning-dynamics plot, which reports seed-averaged return trajectories, this analysis examines the paired return advantage of Exact dual over Context, Random, and X-only at each checkpoint. The advantage remains positive across all 12 checkpoints, and all three comparisons are positive for all 10 paired seeds at every checkpoint. Thus, the result verifies that the controlled-task advantage persists across training budgets and is not driven by either the final checkpoint or a small subset of favorable seeds.

Fig. 15b decomposes the frozen September ACN evaluation into four consecutive periods. The paired return gain of Exact dual over Context remains positive in all four periods, with the corre-

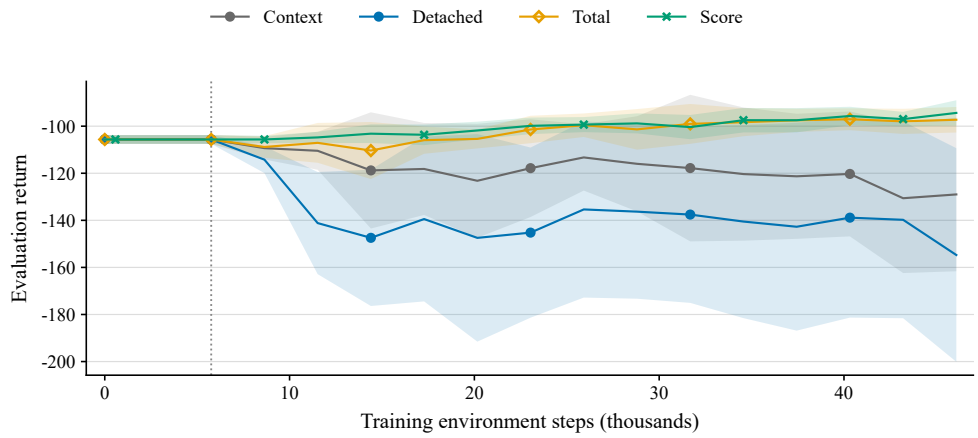
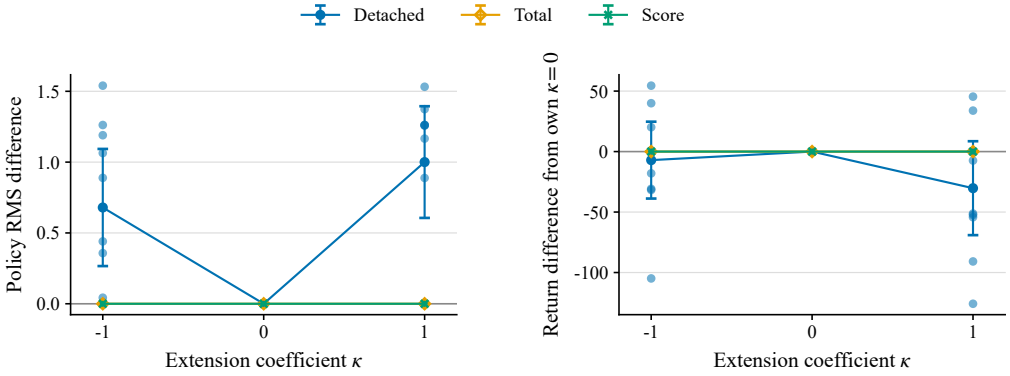


Figure 13: Online learning at $\kappa = 0$. Lines show mean stochastic-policy checkpoint returns and bands show pointwise 95% Student-t intervals over ten paired seeds. Each checkpoint evaluates 40 episodes per seed. The dotted line marks the end of the critic-only warm-up at 5,760 training environment steps. These monitoring episodes are separate from the initial/final evaluations used in Fig. 4b.



(a) Final policy sensitivity.

(b) Final return sensitivity.

Figure 14: Final online extension sensitivity relative to the same interface and seed at $\kappa = 0$: (a) deterministic-action RMS difference; (b) stochastic-policy return difference. Small colored dots show individual seed values, which may overlap; larger markers and bars show means and individual, unadjusted 95% Student-t confidence intervals over ten seeds. Return differences use the held-out final evaluations. Lines connect only the tested coefficients. Total and Score overlap at zero.

sponding 95% confidence intervals above zero. Thus, the formal evaluation advantage is not driven by a single portion of the September workload trace. This decomposition is a temporal analysis of the same frozen evaluation rather than an additional independent test set.

Fig. 15c examines early-training consistency across paired seeds. For each seed, the evaluation-return curve from 5k to 30k interactions is summarized by its area under the learning curve (AUC). Exact dual achieves a positive AUC gain over both Context and Random for all 10 paired seeds. This complements the seed-averaged learning curves in the main text by showing that the early-training advantage is not driven by only a small subset of runs.

Table 8: Online actor–critic results over ten paired seeds. Final return uses the 120-episode held-out evaluation of the final stochastic policy; normalized return AUC uses the 40-episode monitoring evaluations. Entries are mean ± sample SD across seeds. All tested extensions are included; higher is better.

Interface	κ	Final return	Return AUC
Context	0	-129.89 ± 46.02	-116.63 ± 16.95
Detached	-1	-161.77 ± 44.55	-148.10 ± 30.30
Detached	0	-154.71 ± 62.60	-134.22 ± 38.68
Detached	1	-184.98 ± 55.76	-164.18 ± 40.55
Total	-1	-97.09 ± 6.42	-102.66 ± 4.10
Total	0	-97.09 ± 6.42	-102.66 ± 4.10
Total	1	-97.09 ± 6.42	-102.66 ± 4.10
Score	-1	-94.70 ± 6.23	-101.02 ± 3.46
Score	0	-94.70 ± 6.23	-101.02 ± 3.46
Score	1	-94.70 ± 6.23	-101.02 ± 3.46

Table 9: Paired held-out return contrasts at $\kappa = 0$. The first four rows compare each final policy with its own initialization; the remaining rows compare final policies with Context. Returns use 120 matched evaluation episodes per seed. Intervals are individual, unadjusted two-sided 95% Student-t intervals over ten seeds. Positive pairs count seeds with a positive difference.

Contrast	Mean difference	95% CI	Positive pairs
Context final − own initial	-23.57	$[-57.12, 9.98]$	$3/10$
Detached final − own initial	-48.39	$[-93.03, -3.75]$	$3/10$
Total final − own initial	9.23	$[4.82, 13.65]$	$9/10$
Score final − own initial	11.62	$[7.67, 15.58]$	$9/10$
Detached − Context	-24.82	$[-83.50, 33.87]$	$5/10$
Total − Context	32.81	$[-1.35, 66.96]$	$8/10$
Score − Context	35.19	$[-0.88, 71.27]$	$8/10$

D.6 Additional MASAC full-RL validation

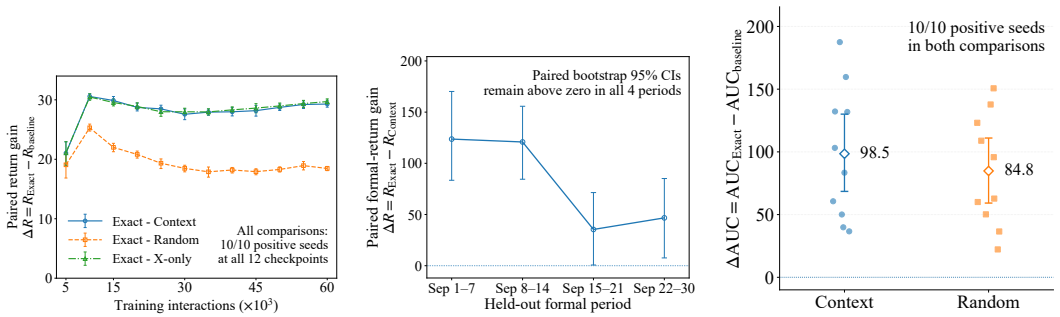
We further evaluate solver-derived representations in a constrained multi-agent resource-allocation task using multi-agent soft actor–critic (MASAC). The environment features stochastic request arrivals and coupled resource constraints, while an inner convex program maps the agents' high-level actions to continuous resource allocations and returns the associated KKT pressure representation. This setting differs from both the controlled full-RL task and the ACN workload experiment in the main text, providing an additional evaluation under a distinct multi-agent optimization structure.

D.6.1 Original observation and complete-context comparisons

We consider two complementary critic-input regimes without the additional fixed observation scaling introduced below, with results reported in Table 10. Let o_i denote the 18-dimensional local observation of agent i, and let s_i denote its complete 42-dimensional non-action state used in the complete-context critic. Specifically, s_i contains o_i together with a 24-dimensional optimizer-side request block. For the centralized critic, o, s, and a denote the corresponding joint concatenations across the four agents.

The first regime uses the observation–action input (o, a). The observation o omits some inputs to the inner problem, so the matched dual may expose optimization information not explicitly available in (o, a). We therefore compare the exact dual with zero-valued, random, and mismatched auxiliary representations in this regime. This comparison evaluates the practical benefit of solver augmentation under partial observation, but is not used as a strict same-information test.

The second regime uses the complete optimization context $X = (s, a)$. The Context and Context + Exact dual critics use identical architectures and input widths and differ only in whether the auxiliary block is zero-valued or contains the matched deterministic representation $Z = T(X)$. Since Z is fully determined by X, this pair constitutes the primary same-information full-RL comparison. It



(a) Training-budget persistence (b) Formal temporal robustness (c) Early-budget consistency

Figure 15: Supplementary full-RL robustness of solver representations.

tests whether solver augmentation remains beneficial when the dual adds no information beyond the complete optimization context. Comparisons are made within each input regime; the observation-based and complete-context variants use different state-side representations and are not intended as a cross-regime ablation.

In the same-information comparison, Context + Exact dual improves the late evaluation return from -4097.8 to 21.1 and the normalized learning AUC from -3256.1 to -698.8. Across the five paired seeds, the mean improvements are 4118.9 in late return and 2557.3 in normalized AUC, with 95% paired t-confidence intervals of $[3473.4, 4764.4]$ and $[2211.5, 2903.1]$, respectively. Context + Exact dual reaches sustained nonnegative return in all $5/5$ paired seeds, whereas the Context baseline does not do so in any seed. The observation-based regime exhibits the same qualitative pattern, with the matched dual substantially outperforming the zero-valued, random, and mismatched auxiliary controls.

The complete-context comparison also exhibits a clear separation in late-stage operating behavior. Over the final three evaluation checkpoints, the mean queue length decreases from 41.74 under Context to 3.41 under Context + Exact dual, while the timeout rate decreases from 81.2% to 3.1% and the drop rate from 42.2% to 0.0%. The constraint-activity statistics in Table 10 further indicate a corresponding change in the resulting resource-allocation regime. These operational statistics are descriptive and are not used to establish a causal mechanism for the return improvement.

Implementation and training details are summarized in Table 11.

D.6.2 Shared observation scaling

We supplement the original MASAC experiments with a complete-context comparison under shared observation scaling. Both methods use $\widetilde{o}_i = D^{-1} o_i$, where D is a fixed positive diagonal matrix specified by the environment configuration. The divisors for the 18 observation coordinates, in their original order, are

$$\begin{aligned}\operatorname{diag}(D) = \big(&50, 14, 2.176, 2.176, 2.176, 2.368, 2.368, 2.368,\\ &3.84, 3.84, 3.84, 15, 15, 15, 1, 192, 108.8, 1\big).\end{aligned}$$

No coordinates are removed or clipped, and the divisors are not fitted to trajectories. The 24-dimensional optimizer-side request block retains its existing physical scaling. Since the observation transformation is invertible and the complete optimizer context is retained, the added solver representation remains deterministic given the critic's context and action.

Both centralized critics receive the same scaled observations, joint actions, optimizer-side request blocks, and an auxiliary block of width five per agent. Context fills this block with zeros; Context + Dual (the Exact dual variant) uses the matched budget-pressure representation with its existing $\log(1 + \cdot)$ transform. The actors receive the same scaled local observations. The resulting critic input width is 200 in both methods. The original nonlinear normalization is disabled; both methods instead use the fixed observation scaling specified above.

Table 10: **Original MASAC results without additional fixed observation scaling.**

Representation design		Control performance			Constraint activity		
Base input	Auxiliary input	Late return ↑	Normalized AUC ↑	Sustained $R \geq 0$	GPU active	BW active	KV binding
(o, a)	**0**	-3887.5 ± 971.8	-2887.3 ± 780.9	$0/5$	99.5	88.8	0.54
(o, a)	Random	-3965.2 ± 788.7	-2919.8 ± 693.3	$0/5$	99.6	88.4	0.37
(o, a)	Mismatched dual	-3936.1 ± 872.2	-2868.9 ± 730.2	$0/5$	99.6	88.6	0.40
(o, a)	Exact dual	4.6 ± 30.6	-988.1 ± 500.9	$4/5$	94.1	92.1	5.92
$X = (s, a)$	**0**	-4097.8 ± 464.6	-3256.1 ± 215.1	$0/5$	99.6	88.7	0.35
$X = (s, a)$	**Exact dual**	$\mathbf{21.1 \pm 1.5}$	$\mathbf{-698.8 \pm 169.3}$	$\mathbf{5/5}$	93.4	91.2	6.56

Results are reported as mean ± standard deviation over five paired training seeds. For the complete-context rows, the reported SD uses divisor five; the shared-scaling results below instead use sample SD (divisor four). (o, a) denotes the observation–action input based on the agents' local observations. $X = (s, a)$ denotes the complete optimization context, where each s_i contains the local observation together with the optimizer-side request information required by the inner problem. Within each input regime, **0** is a zero-valued auxiliary block with the same width as the dual representation. Random uses a dimension-matched random auxiliary input, whereas the mismatched control uses a solver representation from a different decision context. The $X = (s, a)$ rows constitute the primary same-information comparison, since the exact dual satisfies $Z = T(X)$. Late return is averaged over the final three evaluation checkpoints. Normalized AUC is the trapezoidal area under the evaluation-return curve divided by the evaluation span. Sustained $R \geq 0$ denotes the number of seeds achieving nonnegative evaluation return for at least three consecutive evaluation checkpoints. GPU and BW report active-constraint frequencies, while KV reports the key–value cache binding frequency; all three are reported as percentages, with percent signs omitted from the table for compactness.

We retain the task, network architectures, MASAC hyperparameters, and evaluation budget in Table 11, applying the shared scaling above. Each run collects 40,000 environment steps and performs 3,951 actor and 3,951 critic updates. Both methods have 288,792 actor parameters and 732,162 critic parameters. Five paired training seeds use matched initial models and training settings, giving ten completed runs; their trajectories can differ as policies evolve.

Solver features are matched to the behavior, target-policy, and actor actions; the latter two are recomputed for the corresponding sampled actions. The actor update holds the dual representation fixed during differentiation. This experiment evaluates online MASAC with a detached semi-gradient interface; extension invariance is tested separately in Appendix D.4.1.

Each evaluation uses deterministic policies for ten 200-step episodes. Episode seeds are shared across methods, training seeds, and checkpoints. Return is the undiscounted sum of the four agents' rewards over an episode. Evaluations occur at step zero and every 1,500 steps through step 39,000. The primary late-return metric averages the final three scheduled checkpoints (36,000, 37,500, and 39,000); it is not a separate evaluation of the 40,000-step model. Normalized return AUC is the trapezoidal integral over the recorded checkpoints divided by 39,000. No checkpoints are selected by return, and no curve smoothing or extrapolation is used.

At each checkpoint, mean queue length averages all agent–time observations over the ten evaluation episodes. Timeout rate is the fraction of processed requests whose modeled completion times exceed their deadlines; its denominator includes both on-time and timed-out processed requests. Drop rate is dropped arrivals divided by all arrivals, including admitted and dropped requests. Each rate is computed from counts pooled across all agents, steps, and ten episodes, with the denominator floored at one. Late queue length and rates are arithmetic means of the three checkpoint metrics within each training seed. Figure 16 shows the queue trajectories, while Fig. 17 displays the paired late return, timeout, and drop measurements.

Tables 12 and 13 report the method summaries and paired contrasts. Means and sample SDs use five training-seed replicates. Curve bands and paired-contrast intervals are pointwise or individual two-sided 95% Student-t intervals, respectively; they are not simultaneous intervals. Repeated checkpoints and evaluation episodes are not treated as independent training replicates. The intervals characterize training-seed variation on the fixed evaluation scenarios. Archived unscaled runs and smoke runs are not pooled with this comparison.

Table 11: **Implementation details for the original MASAC experiments.**

Parameter	Value	Parameter	Value
Environment, constraints, and representation			
Number of agents	4	Episode horizon	200 env. steps
Request arrivals	Poisson, mean 2.5 requests per agent per step	Queue capacity	50 requests per agent
Maximum batch size	$B_{\max} = 6$ requests per agent	Per-agent action	$a_i = (B_i, F_i, C_i)$; 3 coordinates
Shared coupling	Shared computation-resource constraint	Local constraints	Communication and KV-cache occupancy constraints
Inner optimization	Convex continuous resource allocation	Solver representation	$\dim(Z_i) = 5$
Agent observation	$\dim(o_i) = 18$	Complete state	$\dim(s_i) = 42$
Optimizer-side block	24 dimensions: $[a_{1:B}, b_{1:B}, m_{1:B}, \text{mask}_{1:B}]$	Context ordering	Earliest-deadline-first (EDF)
Pressure features	GPU/BW/KV budget pressures, request-pressure maximum, and concentration	Same-information relation	$Z = T(s, a)$
Observation-based critic input	$4(18 + 3 + 5) = 104$	Complete-context critic input	$4(42 + 3 + 5) = 200$
Exact-dual alignment	Recomputed for matched target and actor actions	Auxiliary controls	(o, a): zero/random/mismatched/exact; X: zero/exact
MASAC training and evaluation			
Actor network	2×256 ReLU	Critic network	Twin centralized 2×512 ReLU
Actor learning rate	3×10^{-4}	Critic learning rate	1×10^{-4}
Entropy temperature	Fixed $\alpha = 0.1$	Discount factor	$\gamma = 0.9$
Target-update coefficient	$\tau = 0.001$	Replay batch size	64 transitions
Replay capacity	10^5 transitions	Warm-up period	500 env. steps
Update cadence	1 update per 10 env. steps	Training horizon	40,000 env. steps
Training seeds	5 paired seeds	Evaluation interval	Every 1,500 env. steps
Evaluation episodes	10 episodes per checkpoint	Observation normalization	Disabled (original runs)

Each complete state s_i comprises the 18-dimensional local observation o_i and the 24-dimensional optimizer-side request block. The actors always use the local observation o_i; the two regimes differ in the information supplied to the centralized critic. The final five-dimensional auxiliary block is zero-valued, random, mismatched, or the matched solver representation according to the experimental mode.

Mean late return rises from -4483.50 for Context to 26.00 for Context + Dual. The paired improvement is 4509.50, with a 95% interval of $[4418.76, 4600.24]$, and all five seed differences are positive. Mean late queue length falls from 45.00 to 2.68, timeout rate from 88.78% to 2.34%, and drop rate from 49.23% to zero in these evaluations. The queue trajectories show early congestion followed by recovery in the dual variant, whereas Context remains congested. Monitoring returns initially deteriorate for both methods; the dual variant subsequently recovers. A post-hoc comparison with the same runs' initial evaluations gives a mean initial-to-late gain of 19.39 for Context + Dual, with interval $[16.25, 22.53]$ and five positive seed pairs. This diagnostic indicates improvement beyond initialization, in addition to separation from the deteriorated Context baseline; it is not an additional prespecified primary endpoint.

These results support a dual-augmentation benefit under the tested shared scaling and training protocol. The comparison does not isolate the mechanism of this benefit or assess primal alternatives or actor-gradient correctness.

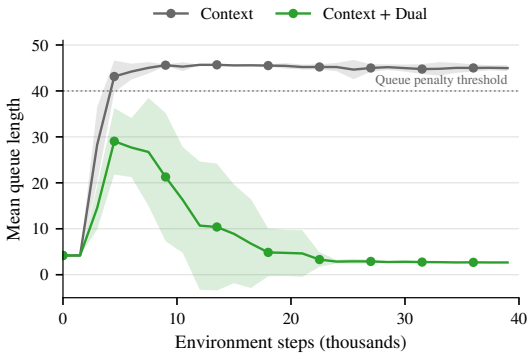
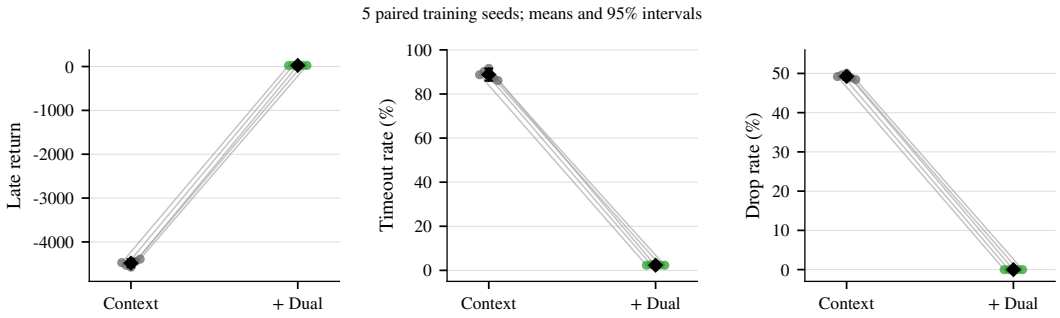


Figure 16: Queue dynamics under shared fixed observation scaling. Lines show mean queue length; bands are pointwise 95% Student-t intervals across five paired training seeds, each evaluated on ten common episodes per checkpoint. The dotted line marks the queue-penalty threshold. All scheduled checkpoints are included; training ends at 40,000 steps and monitoring at 39,000. Symmetric confidence bands are not constrained to be nonnegative. Return summaries and paired contrasts are reported in Tables 12 and 13.



(a) Late return. (b) Timeouts. (c) Drops.

Figure 17: Paired late-stage outcomes under shared observation scaling: (a) return, (b) timeout rate, and (c) drop rate. Each colored point averages one training seed's final three scheduled evaluation checkpoints; gray lines connect matched seeds. Black diamonds and bars show means and individual 95% Student-t intervals across five seeds. "+ Dual" denotes Context + Dual. All five pairs are shown, with small horizontal offsets to distinguish overlap.

E THE USE OF LARGE LANGUAGE MODELS (LLMS)

We used LLMs as writing assistants to polish the manuscript and improve its readability. They also helped identify relevant theoretical literature, provided feedback on experimental design, and assisted with result analysis.

Table 12: MASAC with shared observation scaling over five paired training seeds. Entries are mean $\pm$ sample SD. Late metrics average the checkpoints at 36,000, 37,500, and 39,000 environment steps. AUC is normalized by the 39,000-step evaluation span. Bold marks the higher mean primary endpoint (late return).

Critic input	Late return ↑	Return AUC ↑	Late mean queue ↓	Late timeout (%) ↓	Late drop (%) ↓
Context	-4483.50 ± 72.53	-4174.32 ± 54.14	45.00 ± 0.49	88.78 ± 2.21	49.23 ± 0.62
Context + Dual	$\mathbf{26.00 \pm 1.79}$	-546.44 ± 327.71	2.68 ± 0.09	2.34 ± 0.18	0.00 ± 0.00

Table 13: Paired contrasts for the shared-scaling experiment. Intervals are individual, unadjusted two-sided 95% Student-t intervals across five training seeds. The return difference is the primary contrast; AUC is secondary. The initial-to-late comparison is a post-hoc descriptive diagnostic computed from the existing monitoring evaluations. Positive pairs count seed differences above zero. For compactness, C denotes Context and $C+D$ denotes Context + Dual.

Contrast	Role	Mean difference	95% CI	Positive pairs
Late return: $(C+D) - C$	Primary	4509.50	$[4418.76, 4600.24]$	5/5
Return AUC: $(C+D) - C$	Secondary	3627.88	$[3221.80, 4033.95]$	5/5
$C+D$: late $-$ initial return	Post hoc	19.39	$[16.25, 22.53]$	5/5