AFORL: Offline RL under Episodic Affine Sensor Drift via Transform-Invariant Belief Updates

Liz Lemma Future Detective

January 20, 2026

Abstract

Offline reinforcement learning (RL) is appealing when online interaction is costly, but deployment often fails under non-stationary sensing pipelines. Prior work (e.g., FORL) treats episodic non-stationarity as additive offsets, leveraging zero-shot time-series forecasting and diffusion-based multimodal state beliefs conditioned on offset-invariant delta observations. In real systems, however, observation drift frequently includes unknown per-channel gains (unit conversions, normalization changes, calibration drift), not just biases. We generalize the FORL setting to episodically constant affine observation maps $o_t = A_j s_t + b_j$ with diagonal (or low-rank) A_j and develop AFORL, a transform-invariant belief-update framework that retains FORL's retrospective constraint advantage. The key idea is to replace raw delta observations with per-dimension log-centered delta features that cancel diagonal scaling and additive bias, enabling a conditional generative belief model trained solely on stationary offline data to remain valid at test time. AFORL then fits affine parameters per candidate state trajectory in closed form and fuses them with a forecast prior over (A_j, b_j) to select a consistent state estimate for control. We provide identifiability conditions and upper bounds on state-estimation and value loss, along with matching lower bounds showing excitation is necessary. We outline benchmark protocols extending D4RL/OGBench with real-world time-series gains and biases; experiments would validate robustness under uniform scaling, per-channel scaling, and combined bias+scaling without policy retraining.

Table of Contents

3. 3. Transform-Invariant Features: derive invariants for diagonal scaling + bias; stability issues (near-zero deltas) and smoothing; invariants for low-rank/structured A as optional extension.

4. 4. AFORL Algorithm: conditional belief generation on invariant features; per-candidate affine fitting (OLS) and Bayesian selection with forecast priors; how to produce control state estimates for a frozen policy.

5. 5. Theory I — Invariance and Identifiability: invariance lemma; conditions under which episode-wise affine parameters are identifiable given a candidate state window; symmetry/pathology discussion.

6. 6. Theory II — Error and Value Bounds: propagate generative belief approximation error and forecast prior error into state-estimation bounds; derive value-loss bounds under Lipschitz assumptions; match with minimax lower bounds (necessity of excitation).

7. 7. Complexity and Practical Considerations: runtime per step, memory, and ways to reduce sampling cost; few-step belief models as optional add-on.

8. 8. Experimental Design (Recommended): new benchmark suite for affine drifts; comparison baselines; ablations (invariants vs raw deltas, diagonal vs uniform scaling, forecast quality, window length).

9. 9. Related Work: FORL, robust offline RL under observation corruption, non-stationary RL with forecasting, classical system ID and filtering under unknown sensor gain.

10. 10. Limitations and Extensions: beyond diagonal A (block/low-rank), clipping/nonlinearities, action-dependent drift, partial anchors; open problems.

1 Introduction and Motivation

Offline reinforcement learning is frequently adopted in deployment settings where online exploration is constrained by safety, cost, or latency. In such pipelines the learned policy is often trained on a fixed dataset collected under a specific sensing stack and a specific preprocessing chain. The resulting policy π is then frozen and evaluated (or deployed) under the assumption that the state variables presented to the policy at test time are "the same" as those seen during data collection. Our point of departure is that this assumption fails in a systematic and, by 2026, routine manner: sensing and preprocessing drift across deployments, across hardware revisions, and even across episodes within the same deployment day, while the underlying dynamics and reward remain essentially unchanged.

A widely discussed special case is additive observation bias, for example due to an offset in a sensor, a baseline shift in a learned perception embedding, or a changed origin in coordinate conventions. Additive offsets are problematic but conceptually limited: they can sometimes be removed by recentering, by maintaining running means, or by relying on policies that are robust to global translations. In practice, however, sensor drift is more accurately modeled as affine. Per-dimension gains change under recalibration, temperature, optical exposure, compression, quantization, or normalization steps that are inserted or modified by downstream engineering teams. In addition, these gains and offsets are often constant over short time horizons (an episode, a rollout, or a task attempt) but can change abruptly between horizons, e.g., due to resets, reinitializations, or adaptive filtering that restarts at episode boundaries. This leads us to a test-time observation map of the form

$$o_t \;=\; A_j s_t + b_j,$$

where the pair (A_j, b_j) is fixed within episode j but may vary across j. Even when A_j is diagonal, strictly positive, and bounded away from 0, the induced distribution shift can be severe: a policy trained on clean states may interpret scaled state components as increased velocity, larger distances, or more urgent errors, and respond with inappropriate actions.

These affine shifts are not merely a modeling convenience; they capture concrete failure modes in modern deployment pipelines. First, in robotics and embodied control, the same physical state can be reported under different scaling due to unit conversions (meters vs. centimeters), camera intrinsics updates, or changes in state-estimation filters. Second, in industrial control and forecasting-driven decision systems, a standardized feature pipeline may apply per-batch normalization whose parameters are computed from recent data; after a reset, these parameters can differ, producing an episode-wise affine transform of the features. Third, in multi-tenant inference stacks, the policy may receive state proxies that are the output of another learned model

(a perception encoder or a latent state estimator) whose calibration changes after routine retraining; the resulting mapping is often well-approximated locally by a diagonal gain and bias when inspected in the coordinates relevant to the downstream policy. In all these cases, the environment transition kernel and reward function may remain effectively unchanged, yet the control performance deteriorates because the policy is evaluated on a distorted representation of the state.

A natural response is to seek robustness by augmenting training with random affine transformations or by learning policies that are invariant to such transformations. Such approaches, however, can be overly conservative: the policy may discard information in dimensions where scale matters for optimal control, or it may require substantial additional data coverage to avoid pathological invariances. Another response is online system identification: estimate (A_j, b_j) from the observation stream and invert the transform. This is also nontrivial in the offline setting, because the policy is frozen and cannot actively excite the system to reveal the transform, and because the observation model is confounded with latent state evolution. In particular, without sufficient state variation within an episode, the affine parameters are not identifiable, and any method that claims uniform recovery must fail on such instances.

We propose AFORL, an inference-time method designed to sit between a frozen offline policy and a drifting affine observation stream. The method is organized around a simple principle: rather than attempting to directly infer (A_j, b_j) from raw observations, we construct *transform-invariant* features from short windows of experience and use these features to recover a belief over the latent state in the coordinate system on which π was trained. Concretely, we consider windowed differences Δo_t to eliminate the episode-wise bias b_j, and we apply per-dimension logarithms and within-window centering to eliminate the additive effect of $\log a_{j,d}$ under positive diagonal gains. This produces a feature map $\phi(\tau(t, w))$ whose distribution is (up to controlled numerical smoothing) the same under the clean training observations and the affine-distorted test observations. Thus, a conditional model trained offline to approximate $p(s_t \mid \phi)$ remains applicable at test time without retraining and without access to ground-truth affine parameters.

AFORL then combines this invariance with a candidate-selection mechanism that restores global consistency. Since $\phi(\tau)$ is invariant, it cannot on its own determine the absolute scale and location of the latent state. We therefore treat state inference as a multimodal belief problem: from $\phi(\tau)$ we sample k candidate latent states (or short latent trajectories) using a conditional generative model trained on the offline dataset. For each candidate, we fit the best episode-wise affine parameters via closed-form least squares over the same window and score the candidate by a Bayesian criterion that incorporates an episode-start forecast prior over (A_j, b_j). The prior term is intended to exploit auxiliary information that is often available in deploy-

ment: calibration metadata, historical telemetry, or a time-series model that predicts drift patterns across episodes. The residual term enforces within-window agreement between the candidate latent trajectory and the observed trajectory under a single affine map, thereby preferring candidates that admit a coherent explanation of the data.

Our technical contributions are correspondingly threefold. First, we formalize and prove the invariance property of the log-centered delta features under diagonal positive gains and additive bias, with explicit control of the smoothing parameter used to avoid singularities at small deltas. This invariance is the key to transferring an offline-trained conditional belief model to the test-time observation regime. Second, we establish an identifiability and stability statement for the episode-wise affine parameter fit *given* a candidate latent trajectory: under a mild excitation condition (nondegenerate empirical variance in each dimension over the window), the least-squares solution is unique and varies continuously with candidate error. This yields a principled way to score candidate trajectories and to quantify how state estimation error propagates into the inferred affine parameters. Third, we combine these elements into an end-to-end selection guarantee: if the learned conditional generator is within total variation ε of the true conditional and if the Bayesian selector enjoys a score margin separating the true mode from spurious explanations, then the expected state-estimation error is bounded by a term scaling with ε, a term capturing forecast-prior miscalibration, and a term determined by the scoring noise parameter. Finally, to avoid overclaiming, we complement the upper bounds with a lower bound showing the necessity of excitation: when the state exhibits negligible variation along a dimension within an episode, neither the affine parameters nor the latent state can be uniformly recovered from observations, and any estimator must incur non-vanishing error on an appropriate family of instances.

The practical implication is that offline policies need not be retrained, nor must we assume access to clean states at deployment. Instead, we treat the deployment problem as one of fast, bounded-memory inference: from a short observation-action window we compute invariant statistics, generate a small set of plausible latent states, and select the one whose implied affine map is both internally consistent and externally plausible under a forecast prior. This approach is explicitly designed to match the operational constraints of offline RL deployment: fixed policies, limited online compute, episodically changing sensing conditions, and evaluation criteria that include not only mean performance but also tail risk induced by transient miscalibration.

2 Problem Setup

We formalize the deployment mismatch of interest as a discrepancy in the observation map, while keeping the underlying controlled Markov process

fixed between training and test. Throughout, the continuous latent state space is $\mathcal{S} \subset \mathbb{R}^n$ and the continuous action space is $\mathcal{A} \subset \mathbb{R}^m$. The (train and test) dynamics are given by a transition kernel $T(\cdot \mid s, a)$ on $\mathcal{S}$, and rewards are given by a measurable function $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. Episodes start from an initial distribution ρ_0 over $\mathcal{S}$ (typically uniform over $\mathcal{S}$ when specified by a benchmark, but not assumed known in closed form).

Offline training environment and dataset. The training environment is an episodic MDP

$$\mathcal{M}_{\text{train}} \;=\; (\mathcal{S}, \mathcal{A}, T, R, \rho_0),$$

equipped with the identity observation map, i.e., the agent observes s_t itself at training time. We assume access to an offline dataset

$$\mathcal{D} \;=\; \{(s_t, a_t, s_{t+1}, r_t)\},$$

collected by some behavior policy (not necessarily known) interacting with $\mathcal{M}_{\text{train}}$. In particular, $\mathcal{D}$ contains clean state coordinates in the representation in which the downstream controller is intended to operate. From $\mathcal{D}$ we train (by any offline RL procedure) a policy $\pi(\cdot \mid s)$, which is then *frozen* and will not be updated online. We emphasize that our setting is not online adaptation in the RL sense: we do not change π at test time, and we do not assume access to rewards or dynamics beyond what is implied by interaction.

Test-time environment as a sequence of episodic POMDPs. At test time the agent faces a sequence of episodes indexed by $j \in \{1, 2, \dots\}$. The latent controlled process within each episode is governed by the *same* $(\mathcal{S}, \mathcal{A}, T, R, \rho_0)$ as in training; thus, if the agent could observe s_t, the optimality properties of π would transfer in the usual stationary sense. The distribution shift arises because the agent no longer observes s_t directly. Instead, within episode j the agent receives observations $o_t \in \mathbb{R}^n$ generated deterministically by an episode-wise affine map

$$o_t \;=\; A_j s_t + b_j, \qquad t = 0, 1, \dots, \tag{1}$$

where $A_j \in \mathbb{R}^{n \times n}$ is diagonal and $b_j \in \mathbb{R}^n$. We write $A_j = \operatorname{diag}(a_{j,1}, \dots, a_{j,n})$ and assume strict positivity and boundedness

$$0 < a_{\min} \le a_{j,d} \le a_{\max} \qquad \text{for all } d \in \{1, \dots, n\},$$

where $a_{\min}, a_{\max}$ are fixed constants known to the agent. The pair (A_j, b_j) is constant over timesteps within episode j but can vary arbitrarily across episodes; in particular, the sequence $\{(A_j, b_j)\}_j$ is not assumed Markovian nor stationary. We allow the change to be abrupt at episode boundaries, modeling resets, recalibrations, or per-episode preprocessing choices.

This defines a POMDP for each j (latent state s_t, observation o_t), coupled across episodes by the nonstationary parameter sequence (A_j, b_j). We stress that the environment does not provide (A_j, b_j) during the episode, and we do not assume delayed reveals. Consequently, the agent must act under partial observability induced purely by the sensing map (1).

Forecast priors over affine parameters. In addition to the online observation stream, we assume that at the start of each episode j the agent is given a probabilistic *forecast prior* Π_j over (A_j, b_j). This prior is intended to capture auxiliary information external to the MDP interaction—for example, calibration metadata, a time-series model predicting drift patterns across episodes, or historical statistics from similar deployments. Formally, Π_j is a distribution supported on diagonal A with entries in $[a_{\min}, a_{\max}]$ and on some bounded region for b (or more generally, a distribution for which evaluation of $\log p_{\Pi_j}(A, b)$ is available). We do *not* assume that Π_j is calibrated; indeed, part of the analysis later will quantify how mis-specification in Π_j impacts inference and control.

Interface to the frozen policy. The frozen policy π expects an input in the clean state coordinate system. At test time, however, the agent only sees o_t and must construct an estimate $\hat{s}_t \in \mathcal{S}$ to feed into π. We therefore introduce an estimator (or belief-update rule) $\mathcal{E}$ which, at each time t in episode j, maps the available history and the episode prior to an estimated latent state:

$$\hat{s}_t \;=\; \mathcal{E}(\Pi_j,\; o_{0:t},\; a_{0:t-1})\,,$$

where $a_{0:t-1}$ are the actions actually executed (possibly randomized under π). The deployed controller is the composition $\pi \circ \mathcal{E}$: at time t it selects $a_t \sim \pi(\cdot \mid \hat{s}_t)$ and the environment transitions according to $T(\cdot \mid s_t, a_t)$ and yields reward $R(s_t, a_t)$.

While the estimator may be history-dependent, we will ultimately be interested in estimators implementable under bounded online resources, e.g. using a sliding window of length w and bounded per-step compute. These constraints matter operationally: the estimator must run in real time, and it must not require storing the entire episode history.

Evaluation criteria: estimation and control. We evaluate performance along two axes: the quality of state reconstruction and the induced control performance of the frozen policy when driven by reconstructed states.

First, for state estimation, we consider per-timestep losses such as ℓ_2 error $\|\hat{s}_t - s_t\|_2$ and its expectation under the test-time interaction distribution. Because deployment failures are often dominated by rare but severe miscalibrations, we also consider *tail* metrics, such as the expected maximum error

within an episode,

$$\mathbb{E}\Big[\max_{0\le t\le T}\|\hat{s}_t - s_t\|_2\Big],$$

or high quantiles of $\|\hat{s}_t - s_t\|_2$ over time. These metrics are sensitive to transient inference failures (e.g. during the first few steps of an episode, when little information is available).

Second, for control performance, let $V^{\pi,\text{oracle}}$ denote the value (expected discounted return, or finite-horizon return as specified by the benchmark) achieved by the frozen policy when it is provided the true latent state s_t at test time, and let $V^{\pi\circ\hat{s}}$ denote the value when the policy is instead driven by $\hat{s}_t$ produced by $\mathcal{E}$. Our primary control metric is the *value loss*

$$V^{\pi,\text{oracle}} - V^{\pi\circ\hat{s}},$$

with expectation taken over the randomness of ρ_0, transitions T, policy sampling, and any randomness in the estimator. This comparison isolates the effect of observation drift and inference error from any intrinsic suboptimality of π.

Finally, although our goal is to supply $\hat{s}_t$ to π, it is often operationally useful to also output an episode-wise estimate of the affine parameters (A_j, b_j), together with uncertainty. Such estimates can be used for monitoring, debugging, or triggering safe fallback modes. We treat this as an optional byproduct; the primary requirement is accurate state reconstruction for control.

What makes the problem nontrivial. Two structural aspects drive the difficulty. First, the affine parameters are confounded with the latent state: observing o_t alone does not identify s_t without additional assumptions or information. Second, the policy is frozen and cannot be modified to actively excite the system for identification; the action sequence is endogenous to the estimator via $\pi(\cdot \mid \hat{s}_t)$, so inference errors can compound into future trajectories. Any successful method must therefore (i) leverage offline information from $\mathcal{D}$ about plausible state evolution, (ii) exploit the within-episode constancy of (A_j, b_j), and (iii) incorporate the episode-start forecast prior Π_j whenever it is informative, while remaining robust when it is not. The next section develops the transform-invariant features that allow us to connect offline training distributions to the test-time observation stream.

3 Transform-Invariant Features

Our estimator must connect two distributions: the offline distribution in which we observed clean states s_t, and the test-time distribution in which we observe $o_t = A_j s_t + b_j$ with (A_j, b_j) unknown. Since we will not update the policy online, the only viable route is to build a representation of the

test-time history whose law is (approximately) the same as a corresponding representation computed offline. We therefore construct features $\phi(\tau)$ that are invariant to the episode-wise bias b_j and to the per-dimension gains $a_{j,d}$ in the diagonal matrix A_j.

Windowed histories. Fix a window length $w \geq 1$. For $t \geq 1$ define the observation differences

$$\Delta o_t \;:=\; o_t - o_{t-1}, \qquad \Delta s_t \;:=\; s_t - s_{t-1}.$$

We will form features from a sliding window of recent deltas and actions. Concretely, for $t \geq w$ we define the windowed history tuple

$$\tau(t,w) \;:=\; \big((\Delta o_{t-w+1}, a_{t-w}),\, (\Delta o_{t-w+2}, a_{t-w+1}),\, \ldots,\, (\Delta o_t, a_{t-1})\big),$$

and write $\tau(t,w) = \{(\Delta o_i, a_{i-1})\}_{i=t-w+1}^{t}$ when convenient. (Any consistent alignment of actions within the window suffices; we use a_{i-1} to emphasize that Δo_i is induced by the transition generated under a_{i-1}.)

Bias invariance by differencing. Within a fixed episode j, the observation map is constant, so for all $t \geq 1$,

$$\Delta o_t \;=\; o_t - o_{t-1} \;=\; A_j s_t + b_j - (A_j s_{t-1} + b_j) \;=\; A_j \Delta s_t.$$

Thus differencing eliminates b_j exactly. This is the first invariance: any statistic of $(\Delta o_{t-w+1:t}, a_{t-w:t-1})$ depends on (A_j, b_j) only through A_j.

Diagonal scale invariance by log-centering. Because A_j is diagonal with strictly positive entries, the dth coordinate satisfies

$$\Delta o_t[d] \;=\; a_{j,d}\, \Delta s_t[d], \qquad a_{j,d} > 0.$$

A direct normalization $\Delta o_t[d]/\|\Delta o_t[d]\|$ is unstable in one dimension, and ratio-based normalizations are sensitive to small denominators. We instead use a log-amplitude representation, which turns multiplicative gains into additive offsets that can be removed by centering.

Fix a small smoothing constant $\eta > 0$ (discussed below) and define, for each t and coordinate d,

$$u_{t,d} \;:=\; \log\big(|\Delta o_t[d]| + \eta\big).$$

Over the window $i \in \{t-w+1, \ldots, t\}$, let

$$\bar{u}_{t,d} \;:=\; \frac{1}{w} \sum_{i=t-w+1}^{t} u_{i,d}, \qquad z_{t,d} \;:=\; u_{t,d} - \bar{u}_{t,d}.$$

We then define the feature map as the collection of centered log-deltas paired with actions:

$$\phi(\tau(t,w)) \;:=\; \big\{(z_i, a_{i-1})\big\}_{i=t-w+1}^{t}, \qquad z_i = (z_{i,1}, \ldots, z_{i,n}) \in \mathbb{R}^n. \tag{2}$$

The role of the centering is that, ignoring η for the moment,

$$\log|\Delta o_t[d]| \;=\; \log a_{j,d} + \log|\Delta s_t[d]|.$$

Averaging over the window adds the same $\log a_{j,d}$ term, so subtraction cancels it. Hence $z_{t,d}$ depends on the latent deltas $\Delta s_{t-w+1:t}[d]$ but not on the unknown gain $a_{j,d}$, and it already does not depend on b_j because we started from Δo. In particular, when $\eta = 0$ and $\Delta s_i[d] \neq 0$ over the window,

$$z_{t,d} \;=\; \log|\Delta s_t[d]| - \frac{1}{w}\sum_{i=t-w+1}^{t} \log|\Delta s_i[d]|.$$

Therefore the random variable $\phi(\tau(t,w))$ computed from test-time observations has the same distribution as the same construction computed offline from clean deltas, provided that the latent process (s_t, a_t) follows the same controlled dynamics.

Stability near zero and the choice of η. The logarithm requires care when $|\Delta o_t[d]|$ is small. The additive smoothing η serves two purposes: it makes $u_{t,d}$ well-defined, and it controls the sensitivity of the feature to small perturbations in $\Delta o_t[d]$. Indeed, the map $x \mapsto \log(|x| + \eta)$ is globally Lipschitz with constant $1/\eta$:

$$\big|\log(|x| + \eta) - \log(|y| + \eta)\big| \;\leq\; \frac{1}{\eta}|x - y|.$$

Thus, for deterministic observations, η also controls numerical stability; for stochastic observations (or modeling noise), it regularizes high-variance regions where $\Delta o_t[d] \approx 0$. The trade-off is that large η reduces invariance fidelity: when $|\Delta o_t[d]| \ll \eta$ across the entire window, all $u_{t,d}$ concentrate near $\log \eta$ and the centered features $z_{t,d}$ become nearly 0, carrying little information about $\Delta s_t[d]$. This loss of information is not merely an artifact of the feature design: it corresponds to a genuine lack of excitation along coordinate d over the window, which we later show implies a statistical non-identifiability barrier.

Operationally, one may choose η as a small quantile of empirical $|\Delta s_t[d]|$ magnitudes in the offline dataset (possibly per dimension), or adopt a conservative global η and augment (2) with simple flags such as $\mathbf{1}\{|\Delta o_t[d]| < \eta\}$ to indicate low-signal regimes. A robust variant replaces the window mean $\bar{u}_{t,d}$ by a median or Huberized mean, which is less sensitive to outliers in $|\Delta o_t[d]|$ (e.g., spikes caused by saturations or resets) while preserving the cancellation of the additive $\log a_{j,d}$ shift.

Handling sign changes and negative gains (optional extension). Our primary development assumes $a_{j,d} > 0$ so that a scale change does not flip signs. If negative gains are possible, then $\Delta o_t[d] = a_{j,d}\Delta s_t[d]$ implies

$$\operatorname{sign}(\Delta o_t[d]) \;=\; \operatorname{sign}(a_{j,d})\operatorname{sign}(\Delta s_t[d]),$$

so the sign carries an additional discrete ambiguity. One can extend ϕ by (i) keeping the centered log-amplitude features defined above (which depend only on $|a_{j,d}|$) and (ii) adding sign features such as $\operatorname{sign}(\Delta o_t[d])$ or short sign patterns over the window. At inference time, $\operatorname{sign}(a_{j,d})$ may be treated as an episode-wise latent variable with a prior (potentially part of Π_j), or marginalized by enumerating sign configurations in low dimension.

Why invariance matters for offline-to-test transfer. The key point is that $\phi(\tau(t, w))$ is computable from test-time data but does not depend on the unknown episode-wise affine nuisance parameters. Consequently, we can train a conditional model on offline data that predicts (or samples) the latent state given ϕ without ever seeing the affine shift at training time. Concretely, from the offline dataset we can compute the same feature map using clean deltas:

$$u^{\text{off}}_{t,d} := \log(|\Delta s_t[d]| + \eta), \quad z^{\text{off}}_{t,d} := u^{\text{off}}_{t,d} - \frac{1}{w}\sum_{i=t-w+1}^{t} u^{\text{off}}_{i,d},$$

and set $\phi^{\text{off}}(\tau) := \{(z^{\text{off}}_i, a_{i-1})\}$. Under matched dynamics, ϕ^{off} and ϕ are identically distributed (up to the controlled effect of η), which justifies learning $p(s_t \mid \phi)$ offline and applying it at deployment.

Optional extension: structured non-diagonal A_j. The diagonal assumption yields per-coordinate invariants. If A_j is not diagonal, exact invariants of the same simplicity typically do not exist without further structure. Nevertheless, two structured extensions are often tractable.

First, for block-diagonal A_j with small blocks, the above construction applies blockwise after an appropriate choice of coordinates (or after grouping coordinates into blocks and replacing per-coordinate log-amplitudes by log-norms within each block).

Second, for low-rank deviations from diagonal, one may combine the diagonal-invariant features with a small set of projection-based statistics. For example, let $P \in \mathbb{R}^{r\times n}$ be a fixed random projection with $r \ll n$, and define projected deltas $\Delta\tilde{o}_t := P\Delta o_t$. If $A_j \approx D_j$ is approximately diagonal, then the per-coordinate invariants still stabilize the dominant nuisance, while the projected features provide weak residual information about off-diagonal coupling. In such cases we treat ϕ as a *partially* invariant representation: it removes the leading confounders exactly (bias and per-coordinate scaling)

and leaves a controlled residual mismatch that can be absorbed as modeling error in the subsequent conditional belief model. Our analysis and algorithmic development will focus on the diagonal case, where invariance is exact and the remaining inference problem can be cleanly separated into (i) learning a conditional belief over s_t given ϕ and (ii) selecting among candidates by fitting episode-wise affine parameters.

4 AFORL: Belief Generation, Affine Fitting, and Bayesian Selection

We now describe the full deployment-time estimator. The construction in Section 3 gives us a window feature map $\phi(\tau(t, w))$ whose law matches between offline training (clean states) and test-time deployment (affine observations), up to the controlled smoothing effects of η. AFORL uses this invariant representation to (i) sample a *conditional belief* over latent states from an offline-trained generative model, and then (ii) select among sampled candidates by fitting episode-wise affine parameters and scoring them against a forecast prior.

Objects learned offline and frozen at deployment. Offline, we assume we have already trained a policy π on the clean-state dataset $\mathcal{D}$; crucially, π is fixed at deployment and expects a state input in $\mathcal{S}$ (not an observation input in the transformed space). In addition, AFORL trains a conditional generative model (a "belief generator") on the same offline data:

$$p_\theta(s_t \mid \phi(\tau(t, w))) \approx p(s_t \mid \phi(\tau(t, w))),$$

where ϕ is computed from clean deltas Δs (not from Δo) during training. Any expressive conditional density model is admissible (normalizing flow, diffusion model, autoregressive model, or a mixture model), and we emphasize two requirements: (1) the model must support efficient sampling, since test-time inference is sampling-based; and (2) the model should represent multimodality, since the invariant features necessarily discard information about (A_j, b_j) and thus cannot always determine s_t uniquely.

In practice we may choose to condition on the full window $\phi(\tau(t, w)) = \{(z_i, a_{i-1})\}_{i=t-w+1}^{t}$ and output either (a) a distribution over s_t only, or (b) a joint distribution over the state window $s_{t-w+1:t}$. The latter is slightly more expensive but simplifies scoring, since affine fitting is naturally a windowed regression.

Test-time inputs and the per-episode forecast prior. At the start of episode j we receive a forecast prior Π_j over the nuisance parameters (A_j, b_j). We treat Π_j as an *external* source of information, potentially miscalibrated,

which is used only in a Bayesian selection step. We do not assume that Π_j is correct, only that it can sometimes break symmetries that remain after applying ϕ.

Within episode j, at time t we have the history of observations and actions, but only through a bounded window: we maintain a FIFO buffer of the most recent pairs $(\Delta o_i, a_{i-1})$. From this we compute $\phi(\tau(t, w))$ using (2). The output at each step is a state estimate $\hat{s}_t$ which is then fed into the frozen policy $\pi(\cdot \mid \hat{s}_t)$ to select a_t.

Warm start before the window is full. For $t < w$ the invariant window is not yet available. We therefore use the forecast prior as a bootstrap mechanism: for example, letting $(\bar{A}_j, \bar{b}_j)$ denote a representative point estimate from Π_j (mean, MAP, or median), we set

$$\hat{s}_t \;:=\; \bar{A}_j^{-1}(o_t - \bar{b}_j),$$

where the inverse is taken coordinate-wise since $\bar{A}_j$ is diagonal. This warm start is not meant to be statistically optimal; it simply ensures that the policy receives a state-like input immediately, while AFORL accumulates enough evidence to switch to invariant-feature inference.

Candidate generation from invariant features. Once $t \geq w$, we form $\phi(\tau(t, w))$ and sample k candidate latent states:

$$s_t^{(1)}, \ldots, s_t^{(k)} \;\sim\; p_\theta(\,\cdot \mid \phi(\tau(t, w))).$$

When affine fitting is performed on a whole window (as it will be below), we require candidate latent values across the same indices as the observation window. There are two compatible implementations.

1. *Window-trajectory sampling.* Train the belief generator to sample

$$(s_{t-w+1:t}^{(r)})_{r=1}^{k} \;\sim\; p_\theta(\,\cdot \mid \phi(\tau(t, w))),$$

   i.e., directly output a joint sample of the state window given the feature window. This makes the scoring step immediate.

2. *Particle-style propagation.* Maintain k particles across time, storing their recent windows, and at each step resample or rejuvenate them using $p_\theta(\cdot \mid \phi)$. This amortizes the cost of producing window trajectories but requires additional bookkeeping.

For clarity of exposition we proceed with the window-trajectory view; the analysis in the next section depends only on having candidate state windows with non-degenerate variation.

Per-candidate affine fitting by closed-form OLS. Fix a candidate window $s^{(r)}_{t-w+1:t}$ and the corresponding observation window $o_{t-w+1:t}$. Since A_j is diagonal, we fit each coordinate independently by ordinary least squares. For each dimension $d \in \{1, \dots, n\}$ we solve

$$(\hat{a}_{r,d}, \hat{b}_{r,d}) \;\in\; \arg\min_{a\in\mathbb{R},\, b\in\mathbb{R}} \sum_{i=t-w+1}^{t} \big(o_i[d] - a\, s_i^{(r)}[d] - b\big)^2.$$

Writing $\bar{s}_{r,d} := \frac{1}{w}\sum_{i=t-w+1}^{t} s_i^{(r)}[d]$ and $\bar{o}_d := \frac{1}{w}\sum_{i=t-w+1}^{t} o_i[d]$, the unconstrained OLS solution has the familiar closed form

$$\hat{a}_{r,d} \;=\; \frac{\sum_{i=t-w+1}^{t}(s_i^{(r)}[d] - \bar{s}_{r,d})(o_i[d] - \bar{o}_d)}{\sum_{i=t-w+1}^{t}(s_i^{(r)}[d] - \bar{s}_{r,d})^2}, \qquad \hat{b}_{r,d} \;=\; \bar{o}_d - \hat{a}_{r,d}\bar{s}_{r,d},$$

provided the denominator is nonzero. This is precisely where an excitation condition enters: if the candidate state is (nearly) constant along a coordinate over the window, the regression becomes ill-conditioned, and the affine parameters are effectively unidentifiable from that coordinate alone. In implementation we may stabilize by adding a small ridge term to the denominator, but the theory will later show that this corresponds to an irreducible statistical barrier in the truly unexcited regime.

Because we assume $a_{j,d} \in [a_{\min}, a_{\max}]$ with $a_{\min} > 0$, we also project the fitted slope into this interval:

$$\hat{a}_{r,d} \;\leftarrow\; \min\{a_{\max}, \max\{a_{\min}, \hat{a}_{r,d}\}\},$$

and recompute $\hat{b}_{r,d} = \bar{o}_d - \hat{a}_{r,d}\bar{s}_{r,d}$. Collecting coordinates yields $\hat{A}_r = \mathrm{diag}(\hat{a}_{r,1}, \dots, \hat{a}_{r,n})$ and $\hat{b}_r = (\hat{b}_{r,1}, \dots, \hat{b}_{r,n})$.

Bayesian scoring with a forecast prior. We next score each candidate by combining (i) its empirical consistency with the observation window under the best-fitting affine map, and (ii) its plausibility under the episode prior Π_j. We introduce a small noise level $\sigma^2 > 0$ purely as a scoring temperature (even when $o_t = A_j s_t + b_j$ is deterministic), and define the residual

$$\mathrm{resid}_r \;:=\; \sum_{i=t-w+1}^{t} \big\|o_i - \hat{A}_r s_i^{(r)} - \hat{b}_r\big\|_2^2.$$

The Bayesian score is then

$$\mathrm{score}_r \;:=\; \log p_{\Pi_j}(\hat{A}_r, \hat{b}_r) \;-\; \frac{1}{2\sigma^2}\,\mathrm{resid}_r,$$

where p_{Π_j} denotes a density (or unnormalized score) for the prior. The term resid_r favors candidates that admit a *single* episode-wise affine map matching

the window, while the prior term favors candidates whose implied $(\hat{A}_r, \hat{b}_r)$ are consistent with the forecast. This combination is essential in regimes where ϕ is invariant but not fully informative: several distinct latent windows can yield similar ϕ, yet only one of them induces affine parameters that are both consistent over time and plausible under Π_j.

We then select

$$r^\star \in \arg\max_{r\in\{1,\dots,k\}} \text{score}_r, \qquad \hat{s}_t := s_t^{(r^\star)}.$$

Optionally, we also output $(\hat{A}_j, \hat{b}_j) := (\hat{A}_{r^\star}, \hat{b}_{r^\star})$ as an episode-wise estimate, and we may smooth these estimates across overlapping windows to reduce variance.

Control using the frozen policy. Having produced $\hat{s}_t$, we act by composing π with the estimator:

$$a_t \sim \pi(\cdot \mid \hat{s}_t),$$

and execute a_t in the environment. No policy gradients, value updates, or online system identification loops are performed. All adaptation is confined to state estimation. This separation is deliberate: it allows us to express the control impact of estimation error through Lipschitz properties of the frozen value function (cf. Theorem 4), and it avoids the instability of online RL under misspecified observations.

Computational remarks. The OLS fitting step admits an $O(n)$-per-candidate implementation if we maintain sufficient statistics over the window (sums of $s_i^{(r)}[d]$, $s_i^{(r)}[d]^2$, $o_i[d]$, and $s_i^{(r)}[d]o_i[d]$). The dominant cost is typically sampling from p_θ. For this reason AFORL is naturally compatible with few-step flows or distilled diffusion models, and with moderate sample counts k.

The next section formalizes the two core facts that make AFORL analyzable: (i) the invariance property of ϕ, and (ii) identifiability (and stability) of the affine fit given a candidate state window with sufficient excitation.

5 Theory I: Invariance and Identifiability

This section isolates the two structural facts on which AFORL rests. First, the window features $\phi(\tau(t, w))$ computed from test-time observations are (approximately) distributionally identical to the same construction computed from clean offline states. Second, conditional on a candidate latent window, the episode-wise affine nuisance parameters are identifiable (and stably estimable) provided the candidate exhibits sufficient variation over

the window. We also record the main symmetries and pathologies that explain why the belief generator must represent multimodality and why an excitation condition is unavoidable.

5.1 Invariance of the window feature map

We recall that within a fixed episode j the observation map is

$$o_t \;=\; A_j s_t + b_j, \qquad A_j = \mathrm{diag}(a_{j,1}, \ldots, a_{j,n}), \quad a_{j,d} \in [a_{\min}, a_{\max}],\ a_{\min} > 0,$$

and we compute deltas $\Delta o_t = o_t - o_{t-1}$. Bias cancels immediately:

$$\Delta o_t \;=\; A_j \Delta s_t.$$

Thus, for each coordinate d, $\Delta o_t[d] = a_{j,d} \Delta s_t[d]$. The feature map used for belief generation is based on log-magnitudes, stabilized by $\eta > 0$,

$$u_{t,d} \;:=\; \log(|\Delta o_t[d]| + \eta), \qquad z_{t,d} \;:=\; u_{t,d} - \frac{1}{w} \sum_{i=t-w+1}^{t} u_{i,d},$$

and $\phi(\tau(t,w))$ collects the window $\{(z_{i,\cdot}, a_{i-1})\}_{i=t-w+1}^{t}$ (we suppress minor indexing choices).

Lemma (feature invariance up to smoothing). Fix an episode j and a coordinate d. Define $u^s_{t,d} := \log(|\Delta s_t[d]| + \eta)$ and the corresponding centered quantity $z^s_{t,d} := u^s_{t,d} - \frac{1}{w} \sum_{i=t-w+1}^{t} u^s_{i,d}$. Then

$$z_{t,d} \;=\; z^s_{t,d} + \xi_{t,d},$$

where $\xi_{t,d}$ is a deterministic correction induced by η satisfying the uniform bound

$$|\xi_{t,d}| \;\le\; \frac{\eta}{\min\{|\Delta s_t[d]|, |\Delta s_{t-w+1}[d]|, \ldots, |\Delta s_t[d]|\} + \eta} + \frac{1}{w} \sum_{i=t-w+1}^{t} \frac{\eta}{|\Delta s_i[d]| + \eta}.$$

In particular, whenever $|\Delta s_i[d]| \gg \eta$ throughout the window, we have $|\xi_{t,d}| \ll 1$ and $z_{t,d} \approx z^s_{t,d}$.

Proof sketch. For $a > 0$, $\log(|ax| + \eta) = \log a + \log(|x| + \eta/a)$. Centering over the window cancels the constant $\log a$, leaving only the difference between using η and η/a. The displayed bound follows from the Lipschitz estimate $|\log(y+\alpha) - \log(y+\beta)| \le |\alpha - \beta|/(y + \min\{\alpha, \beta\})$ with $y = |x|$. $\square$

Consequence for offline-to-online transfer. The lemma implies that, up to the controlled perturbation ξ, the random variable $\phi(\tau(t, w))$ computed from $(\Delta o, a)$ at test time has the same law as the corresponding feature computed from $(\Delta s, a)$ in the offline environment (under the same policy-induced trajectory distribution). This is the sense in which an offline-trained conditional model $p_\theta(s_t \mid \phi)$ remains applicable at deployment: the conditioning signal has been "factored" to remove the episode-wise nuisance (A_j, b_j).

Two qualifications are essential. First, invariance is per-coordinate and uses diagonality of A_j; for non-diagonal A_j, $\Delta o_t[d]$ mixes coordinates and the log-centering trick no longer isolates a single $\log a_{j,d}$. Second, the sign of $a_{j,d}$ matters: the above uses $a_{j,d} > 0$. If gains may be negative, one must either (i) model $\mathrm{sign}(a_{j,d})$ as an additional latent and incorporate sign features $\mathrm{sign}(\Delta o_t[d])$, or (ii) restrict to magnitude-only objectives and accept an inherent sign ambiguity.

5.2 Identifiability of episode-wise affine parameters given a candidate

AFORL does not attempt to infer (A_j, b_j) directly from ϕ; rather, it uses ϕ to generate candidate latent windows and then checks which candidates admit a *single* affine map consistent with the observation window. This motivates the following elementary identifiability statement.

Proposition (per-coordinate OLS is unique under excitation). Fix a coordinate d and a window $\{t - w + 1, \ldots, t\}$. Given any candidate latent values $\{\tilde{s}_i[d]\}_{i=t-w+1}^{t}$ and observed values $\{o_i[d]\}_{i=t-w+1}^{t}$, consider the least-squares fit

$$(\hat{a}_d, \hat{b}_d) \;\in\; \arg\min_{a,b\in\mathbb{R}} \sum_{i=t-w+1}^{t} \big(o_i[d] - a\,\tilde{s}_i[d] - b\big)^2.$$

If the empirical variance of the candidate regressor is nonzero,

$$\sum_{i=t-w+1}^{t} \big(\tilde{s}_i[d] - \bar{\tilde{s}}_d\big)^2 \;>\; 0, \qquad \bar{\tilde{s}}_d := \frac{1}{w} \sum_{i=t-w+1}^{t} \tilde{s}_i[d],$$

then the minimizer is unique and equals the usual closed form. Moreover, if the true relation is $o_i[d] = a_{j,d} s_i[d] + b_{j,d}$ and the candidate satisfies $\max_i |\tilde{s}_i[d] - s_i[d]| \le \delta_s$, then

$$|\hat{a}_d - a_{j,d}| \;\le\; \frac{C}{\sum_i (\tilde{s}_i[d] - \bar{\tilde{s}}_d)^2}\,\delta_s, \qquad |\hat{b}_d - b_{j,d}| \;\le\; C'\,\delta_s,$$

for explicit constants C, C' depending on $\max_i |o_i[d] - \bar{o}_d|$ and the candidate variance (and, after slope projection, on $a_{\min}, a_{\max}$ as well).

Interpretation. The condition $\sum_i (\tilde{s}_i[d] - \bar{\tilde{s}}_d)^2 > 0$ is precisely a windowed excitation condition. Without it, (a, b) are not identifiable even if the candidate were correct: if $s_i[d] \equiv c$ in the window, then $o_i[d] = a\,c + b$ is constant, and infinitely many pairs (a, b) yield the same constant. The stability estimate exhibits the same phenomenon quantitatively: as the candidate variance shrinks, the slope estimate becomes arbitrarily sensitive to small candidate errors.

This is not merely an artifact of least squares. The pathology is information-theoretic: if the latent state does not vary along coordinate d, then the observation stream contains no leverage to separate scaling from bias in that coordinate. This is the origin of the lower bound we later state as a necessity of excitation.

5.3 Symmetries, multimodality, and failure modes

The invariance of ϕ is deliberately achieved by discarding information, and this inevitably introduces symmetries. We record the ones most relevant for understanding why AFORL combines a multimodal belief generator with a forecast prior.

Residual symmetries after applying ϕ. Even in the diagonal, positive-gain setting, ϕ is not injective in general: distinct latent windows can induce identical centered log-delta patterns, especially when the underlying dynamics admit sign flips, periodicity, or near-linear regimes where Δs takes values from a small set. Consequently, the conditional distribution $p(s_t \mid \phi)$ can be genuinely multimodal, and any unimodal predictor (e.g. a conditional mean) can be arbitrarily misleading for control.

Sign ambiguity and near-zero deltas. Because ϕ uses $\log(|\Delta o| + \eta)$, it is insensitive to the sign of Δo unless sign is separately included. When Δs frequently changes sign, magnitude-only features can identify a set of plausible states but not a unique one. Moreover, when $|\Delta s|$ is frequently below η, the smoothing correction in Section 5.1 dominates and the features become less informative; empirically this manifests as degraded candidate quality from $p_\theta(\cdot \mid \phi)$ unless η is tuned to the noise/scale of deltas.

Affine fitting can be vacuous without excitation. The scoring step penalizes candidates that cannot be explained by a *single* affine map over the window. However, if a candidate window is nearly constant in a coordinate, then *any* affine parameters can fit that coordinate well, so the residual offers no discrimination. In such regimes the selector must rely on other coordinates (if excited) and on the prior Π_j. This clarifies why we do not view Π_j as optional ornamentation: in ambiguous regimes it supplies the only principled tie-breaker among symmetry-related candidates.

Why diagonality matters for identifiability. In the diagonal case, each coordinate admits an independent two-parameter regression, and excitation can be checked coordinate-wise. For a general full matrix A_j, identifiability would couple coordinates and require significantly stronger conditions (e.g. persistent excitation in multiple directions plus structural constraints such as sparsity or low rank). AFORL is therefore designed to exploit the diagonal structure in both the invariance map and the affine fitting step.

The next section uses the invariance and identifiability facts above as inputs: we propagate approximation error of the learned conditional belief and mis-specification of the forecast prior into explicit state-estimation and value-loss bounds, and we relate the excitation requirement to a matching minimax lower bound.

6 Theory II: Error and Value Bounds

We now propagate the two imperfect ingredients of AFORL—(i) approximation error of the learned conditional belief generator and (ii) mis-specification of the episode-start forecast prior—into explicit state-estimation and control-performance bounds. The resulting statements are "plug-in" in the sense that they treat the invariance and per-candidate identifiability facts from Section 5 as black-box inputs.

6.1 From conditional belief error to candidate coverage

Fix a test episode j and a time $t \geq w$. Let $\phi_t := \phi(\tau(t, w))$ denote the invariant feature computed from the observation window and executed actions. Let $p(\cdot \mid \phi_t)$ denote the *true* conditional distribution of the latent state s_t induced by the MDP dynamics and the policy-induced trajectory distribution, and let $\hat{p}_\theta(\cdot \mid \phi_t)$ be the learned conditional belief generator used by AFORL. We quantify statistical mismatch by the total variation bound

$$\mathrm{TV}(\hat{p}_\theta(\cdot \mid \phi_t),\, p(\cdot \mid \phi_t)) \;\leq\; \varepsilon.$$

We emphasize that ϕ_t is computed from test observations o but is distributionally aligned with the offline construction (up to the controlled smoothing perturbation), so ε should be interpreted as an offline generalization error rather than a domain-shift error.

The first step is to convert ε into a statement that among k i.i.d. samples from $\hat{p}_\theta(\cdot \mid \phi_t)$, at least one is near the true s_t with high probability. We formalize "near" by an arbitrary measurable target set $G_t \subset \mathcal{S}$ (e.g. an ℓ_2 ball around s_t) and use only the elementary inequality

$$\hat{p}_\theta(G_t \mid \phi_t) \;\geq\; p(G_t \mid \phi_t) \;-\; \varepsilon.$$

If we draw $\{s_t^{(r)}\}_{r=1}^k \overset{\text{i.i.d.}}{\sim} \hat{p}_\theta(\cdot \mid \phi_t)$, then conditional on ϕ_t,

$$\mathbb{P}\Big(\exists r \le k:\ s_t^{(r)} \in G_t \ \Big|\ \phi_t\Big) \ = \ 1-(1-\hat{p}_\theta(G_t \mid \phi_t))^k \ \ge \ 1-(1-p(G_t \mid \phi_t)+\varepsilon)^k\,. \tag{3}$$

Thus, for any level $\alpha \in (0,1)$, choosing $k \gtrsim \log(1/\alpha)/(p(G_t \mid \phi_t)-\varepsilon)$ ensures α-failure probability as soon as $p(G_t \mid \phi_t) > \varepsilon$. This calculation is the only place where k enters the analysis: increasing k increases the chance that AFORL "covers" at least one candidate from each relevant mode of the true conditional.

6.2 Bayesian selection with a miscalibrated forecast prior

Given each candidate sample $s_t^{(r)}$, AFORL fits nuisance parameters $(\hat{A}_r, \hat{b}_r)$ by per-coordinate OLS over the same observation window and scores the candidate by

$$\text{score}_r \ := \ \log p_{\Pi_j}(\hat{A}_r, \hat{b}_r) - \frac{1}{2\sigma^2}\,\text{resid}_r, \qquad \text{resid}_r := \sum_{i=t-w}^{t} \|o_i - \hat{A}_r s_i^{(r)} - \hat{b}_r\|_2^2,$$

where $\sigma^2 > 0$ is a modeling noise used only for scoring. In the idealized case where (a) the candidate window equals the true latent window and (b) the OLS step is well-posed by excitation in each coordinate, we have $\text{resid}_r = 0$ and $(\hat{A}_r, \hat{b}_r) = (A_j, b_j)$. More generally, the stability part of the identifiability result from Section 5.2 implies that if a candidate trajectory window is uniformly close to the true one, then the implied affine parameters and the residual are correspondingly small. We therefore reduce selection correctness to a *score margin* condition: there exists $\gamma > 0$ such that, in expectation conditional on ϕ_t, the (near-)true candidate has score at least γ larger than any alternative mode that can explain ϕ_t but induces inconsistent affine parameters across time.

To expose the effect of forecast error, we introduce an abstract misspecification measure δ_Π controlling log-density error in a neighborhood of the true parameters:

$$\Big|\log p_{\Pi_j}(A,b) - \log p_{\Pi_j^\star}(A,b)\Big| \ \le \ \delta_\Pi \qquad \text{for all } (A,b) \text{ in a set containing } (A_j,b_j) \text{ and the competing f} \tag{4}$$

Here $\Pi_j^\star$ denotes an "oracle" prior that would correctly reflect the episode-wise distribution of (A_j, b_j). Condition (4) is deliberately weak: it does not require calibration, only that the log prior not arbitrarily distort the relative ordering of plausible parameter pairs.

Under a margin condition and (4), the selector inherits robustness: an additive perturbation of the log prior by at most δ_Π can reduce the effective margin by at most $O(\delta_\Pi)$, while the residual term contributes an additional perturbation on the order of σ through the factor $(2\sigma^2)^{-1}$. Combining these

with the coverage bound (3) yields the qualitative form recorded in Theorem 3: the state error is controlled by a term proportional to ε (candidate coverage), a term proportional to δ_Π (tie-breaking distortion), and a term proportional to σ (selection noise floor).

6.3 A bound on state-estimation error

We state the consequence in the form we will use downstream. Fix $t \geq w$ and let $\hat{s}_t$ be AFORL's chosen sample. Assume: (i) $\mathrm{TV}(\hat{p}_\theta, p) \leq \varepsilon$, (ii) windowed excitation holds so that OLS is unique and stable per coordinate for any candidate within a neighborhood of the truth, and (iii) a separability/margin condition as described above. Then there exist constants C_1, C_2, C_3 (depending on the margin γ, window length w, the candidate-OLS stability constants, and mild tail properties of the score) such that

$$\mathbb{E}\|\hat{s}_t - s_t\|_2 \;\leq\; C_1\,\varepsilon \;+\; C_2\,\delta_\Pi \;+\; C_3\,\sigma, \tag{5}$$

where the expectation is taken over the trajectory randomness and the sampling randomness in AFORL. The dependence on k is implicit in C_1 through the coverage probability (3): for fixed G_t one may write $C_1 = C_1(k)$ with $C_1(k) \to 0$ as $k \to \infty$ when $p(G_t \mid \phi_t)$ is bounded away from 0.

Equation (5) should be read as a *decomposition* rather than a sharp constant-level inequality: ε reflects how well the offline-trained generator captures the true conditional under invariant features; δ_Π reflects forecast quality; and σ reflects the "temperature" of the Bayesian selection rule.

6.4 From state error to value loss for frozen policies

We now translate state-estimation error into performance degradation when deploying the frozen policy π on $\hat{s}_t$ rather than s_t. Let $V^{\pi,\mathrm{oracle}}$ denote the return obtained by executing π with access to the true latent state, and let $V^{\pi\circ\hat{s}}$ denote the return when π is fed the estimated state $\hat{s}_t$ at each step.

A convenient sufficient condition is Lipschitz regularity of the action-value function. Suppose that for the true latent MDP,

$$\sup_{a\in\mathcal{A}}\big|Q^\pi(s,a) - Q^\pi(s',a)\big| \;\leq\; L_Q\,\|s - s'\|_2 \qquad \text{for all } s, s' \in \mathcal{S}.$$

Then, writing $a_t \sim \pi(\cdot \mid s_t)$ for the oracle action and $\hat{a}_t \sim \pi(\cdot \mid \hat{s}_t)$ for the deployed action, the one-step suboptimality induced by estimation can be bounded (in expectation) by the mismatch in the argument of Q^π, yielding a per-step loss $O(L_Q\mathbb{E}\|\hat{s}_t - s_t\|_2)$. Standard discounted telescoping then implies the episodic/discounted return gap bound recorded abstractly as Theorem 4:

$$V^{\pi,\mathrm{oracle}} - V^{\pi\circ\hat{s}} \;\lesssim\; \frac{L_Q}{1-\gamma}\,\sup_t \mathbb{E}\|\hat{s}_t - s_t\|_2, \tag{6}$$

up to the usual compounding terms if one insists on uniform-in-time rather than average-in-time bounds. Substituting (5) into (6) yields an explicit control-level decomposition into generative, forecast, and scoring contributions.

6.5 Matching lower bounds: necessity of excitation

Finally, we clarify why the excitation condition is not merely technical. If, along some coordinate d, the latent state is (nearly) constant over the window, then the observation stream contains no information to separate gain from bias: many pairs $(a_{j,d}, b_{j,d})$ explain the same $o_i[d]$. This creates an irreducible ambiguity in $s_t[d]$ itself because $s_t[d] = (o_t[d] - b_{j,d})/a_{j,d}$ depends on both unknowns.

Theorem 5 formalizes this as a minimax lower bound in one dimension: for any estimator based only on $(o_{0:t}, a_{0:t-1})$, if the latent trajectory has vanishing empirical variance, then there exist distinct parameter pairs within the admissible class that produce identical observations, forcing a non-vanishing worst-case state error. Two corollaries are immediate for our setting. First, no algorithm—including AFORL—can guarantee uniform recovery without excitation; at best one can rely on other excited coordinates or on prior information. Second, in regimes where excitation intermittently fails, the forecast prior Π_j is not optional: it is the only principled mechanism available to break symmetries that are information-theoretically unbreakable from in-episode data alone.

In summary, the theory presents a coherent triad: invariance makes offline-to-online conditioning feasible; excitation makes affine nuisance estimation well-posed; and conditional belief quality together with forecast quality determines the quantitative state and value degradation through bounds of the form (5)–(6).

7 Complexity and Practical Considerations

We now record the computational profile of AFORL and several implementation choices that materially affect wall-clock performance at deployment. Throughout, we consider a fixed episode j, dimension $n = \dim(\mathcal{S})$, window length w, and k candidate samples per step for $t \geq w$.

Per-step runtime decomposition. At each step, AFORL performs three conceptually separate computations: (i) update invariant features, (ii) generate candidate latent states from the conditional belief model, and (iii) score candidates by fitting (A_j, b_j) over the window (or by maintaining equivalent sufficient statistics) and taking an arg max. Writing C_{gen} for the cost of one forward evaluation of the conditional generator and N for the number of

denoising steps (if the generator is a diffusion model), the per-step cost for $t \geq w$ admits the coarse bound

$$\text{time}(t) \;=\; O(nw) + O(k\,N\,C_{\text{gen}}) + O(knw) \qquad \text{(naive OLS on a length-}w\text{ window).} \tag{7}$$

The first term accounts for feature computation: deltas, per-dimension log-transform, and window-centering. This term is typically memory-bandwidth limited and scales linearly in both n and w. The second term is typically dominant when the belief model is a high-capacity diffusion sampler. The third term is the candidate-wise regression and residual computation.

A simple but important observation is that the OLS fitting can be reduced from $O(knw)$ to $O(kn)$ by maintaining per-candidate sufficient statistics over the sliding window. For each candidate r and dimension d, define

$$S_1^{(r)}[d] = \sum_{i=t-w}^{t} s_i^{(r)}[d], \quad S_2^{(r)}[d] = \sum_{i=t-w}^{t} s_i^{(r)}[d]^2, \quad O_1[d] = \sum_{i=t-w}^{t} o_i[d], \quad SO^{(r)}[d] = \sum_{i=t-w}^{t} s_i^{(r)}[d]\,o_i[d],$$

together with the window length $\ell = w+1$. Then the closed-form regression coefficients in each coordinate are

$$\hat{a}_{r,d} = \frac{SO^{(r)}[d] - \frac{1}{\ell} S_1^{(r)}[d]\,O_1[d]}{S_2^{(r)}[d] - \frac{1}{\ell}(S_1^{(r)}[d])^2}, \qquad \hat{b}_{r,d} = \frac{1}{\ell} O_1[d] - \hat{a}_{r,d}\,\frac{1}{\ell} S_1^{(r)}[d], \tag{8}$$

whenever the denominator is nonzero (the excitation condition). Because a sliding window update changes each sum by "add newest, remove oldest," the statistics can be updated in $O(n)$ per candidate, yielding

$$\text{time}(t) \;=\; O(nw) \;+\; O(k\,N\,C_{\text{gen}}) \;+\; O(kn), \tag{9}$$

where the residual resid_r can also be computed from maintained sums (or computed approximately, cf. below) rather than re-iterating over the full window.

Memory footprint and buffering. The strictly necessary episode-local memory consists of (a) the observation/action history of length w (or equivalently the delta history), and (b) the information needed to score candidates. If one stores raw buffers, the history cost is $O((n+m)w)$. Candidate storage depends on whether we retain a candidate trajectory window $\{s_{t-w:t}^{(r)}\}$ for each r. The most direct implementation stores the full window per candidate, costing $O(knw)$, which may be prohibitive when k and w are both large.

In practice, we can often avoid storing the full candidate trajectory by choosing a belief model that outputs a *window* of latent states in one shot, i.e. sampling $s_{t-w:t}^{(r)} \sim \hat{p}_\theta(\cdot \mid \phi_t)$ jointly. This shifts complexity from memory

to sampling but enables one-pass computation of sufficient statistics. Alternatively, if the belief model produces only $s_t^{(r)}$ at each step, then maintaining candidate-wise sufficient statistics requires a consistent notion of candidate identity over time; this can be implemented by a particle-like mechanism (propagate candidates forward under a learned dynamics model, or resample with ancestor tracking). When such temporal bookkeeping is undesirable, the simplest reliable strategy is to sample short windows (rather than single states) and accept the associated increase in generator output dimension.

Reducing sampling cost: adaptive k, early pruning, and amortized scoring. The dependence on k enters only through candidate coverage and selection. Consequently, we can treat k as a *runtime dial* and adjust it online. A straightforward rule is to begin each episode with a small k and increase it only when the selector is uncertain, e.g. when the top scores are close:

$$k(t+1) = \begin{cases} k(t) & \text{if } \mathrm{score}_{(1)} - \mathrm{score}_{(2)} \geq \Delta_{\mathrm{conf}}, \\ \min\{k_{\max},\ \lceil \alpha k(t) \rceil\} & \text{otherwise,} \end{cases}$$

where $\mathrm{score}_{(1)} \geq \mathrm{score}_{(2)}$ are the best two candidate scores and $\Delta_{\mathrm{conf}} > 0$ is a tunable confidence gap. A complementary heuristic is *early pruning*: compute a cheap proxy score first, discard a large fraction of candidates, and run full OLS and residual evaluation only on the surviving set. For instance, one may first fit $(\hat{A}_r, \hat{b}_r)$ on a subsampled set of indices in the window (say every other timestep), or compute only the prior term $\log p_{\Pi_j}(\hat{A}_r, \hat{b}_r)$ after a quick fit, then evaluate resid_r precisely only for the best few.

When n is large, the OLS stage itself can become nontrivial. Because A_j is diagonal, the fitting and residual decomposes over dimensions; we can therefore prune dimensions as well. If the forecast prior strongly concentrates some coordinates of A_j (or if some observation coordinates are known to be stable), we may fix those dimensions to prior means and fit only the uncertain subset, reducing both compute and variance.

Few-step belief models as an optional add-on. The generator cost $O(k\,N\,C_{\mathrm{gen}})$ is the primary deployment bottleneck when N is large. We therefore view *few-step* conditional samplers as a practically important specialization. Two standard choices are: (i) a conditional normalizing flow or autoregressive model, for which $N = 1$ (or a small constant), and (ii) a diffusion model distilled into a small-number-of-steps sampler (e.g. by progressive distillation), reducing N substantially at the expense of an additional offline training stage.

The salient requirement for our theory is not the sampling mechanism but the conditional approximation property $\mathrm{TV}(\hat{p}_\theta(\cdot \mid \phi), p(\cdot \mid \phi)) \leq \varepsilon$. Empirically, we expect few-step models to trade increased ε for reduced runtime. AFORL exposes this trade-off transparently: one may increase

k to compensate for a larger ε when runtime permits, or decrease k when the sampler is accurate enough that candidate coverage is not the limiting factor. This suggests a natural operating envelope: on hardware-constrained platforms, we prefer a low-N generator and moderate k; on high-throughput accelerators, we can afford larger k (and possibly larger w) even with a heavier sampler.

Numerical and systems considerations. We note four low-level choices that tend to dominate stability in real deployments. First, the log feature uses $\log(|\Delta o| + \eta)$; the smoothing η should be chosen to avoid numerical explosion near $\Delta o = 0$ while not washing out informative small deltas. Second, the OLS denominator in (8) should be regularized (or clamped) when empirical variance is small, both for numerical stability and to reflect the information-theoretic lower bound under weak excitation. Third, it is typically beneficial to constrain fitted gains by projection onto $[a_{\min}, a_{\max}]$ after OLS:

$$\hat{a}_{r,d} \leftarrow \Pi_{[a_{\min}, a_{\max}]}(\hat{a}_{r,d}),$$

which reduces the influence of pathological candidates whose implied scaling is implausible. Fourth, the entire scoring loop over $r = 1, \ldots, k$ is embarrassingly parallel; a vectorized implementation that batches candidate evaluation is often the difference between real-time feasibility and failure.

In summary, AFORL is computationally dominated by conditional sampling and can be made efficient by (a) maintaining sliding-window sufficient statistics for diagonal OLS, (b) using adaptive sampling and early pruning, and (c) optionally adopting few-step conditional belief models. These modifications do not change the logical structure of the method; they alter only the constants implicit in (7)–(9) and the effective approximation error ε induced by the chosen generator family.

8 Experimental Design (Recommended)

We outline an experimental protocol intended to isolate the deployment-time difficulty induced by episodically varying affine observation maps, while keeping the underlying MDP and the offline policy π fixed. The guiding principle is that *training is always performed under identity observations* (as in the offline dataset $\mathcal{D}$), and *only test-time observations are affinely transformed*. This separation prevents conflating robustness to sensor drift with standard generalization or policy-learning effects.

A benchmark suite for episodic affine drifts. Given any continuous-control benchmark with state $s_t \in \mathbb{R}^n$ and action $a_t \in \mathbb{R}^m$, we propose a derived test suite, denoted informally by AffineDrift, constructed as follows. We take the canonical offline dataset $\mathcal{D} = \{(s_t, a_t, s_{t+1}, r_t)\}$ collected

under identity observation, train a policy π on s_t (any offline RL algorithm may be used), and then freeze π for all subsequent evaluation. At test time we replace the identity observation by

$$o_t \;=\; A_j s_t + b_j, \qquad A_j = \mathrm{diag}(a_{j,1}, \ldots, a_{j,n}), \qquad a_{j,d} \in [a_{\min}, a_{\max}],$$

with (A_j, b_j) held constant within episode j and resampled (or evolved) across episodes.

To make the suite diagnostically useful, we recommend reporting results under at least three drift families:

1. **I.i.d. episode maps:** sample $a_{j,d} \sim \mathrm{Unif}[a_{\min}, a_{\max}]$ and $b_{j,d} \sim \mathrm{Unif}[-B, B]$ independently across episodes. This stresses per-episode adaptation without temporal forecasting structure.

2. **Smooth drifts (forecastable):** evolve latent parameters by a stable AR(1) process in log-space, e.g. $\log a_{j+1,d} = \lambda \log a_{j,d} + (1-\lambda)\xi_{j,d}$ with $\xi_{j,d}$ i.i.d., and similarly $b_{j+1,d} = \lambda b_{j,d} + (1-\lambda)\zeta_{j,d}$. This is the setting in which a forecast prior Π_j is meaningful.

3. **Adversarial regime shifts:** alternate between a small set of modes (e.g. "nominal sensors" and "miscalibrated sensors"), which induces multimodality in (A_j, b_j) and tests whether the inference stage can resolve aliasing using in-episode evidence.

We further recommend two structural variants: *uniform scaling* ($A_j = a_j I$) as an easier special case, and *diagonal scaling* as the default. When reporting diagonal results, it is useful to include a setting in which only a subset of coordinates drift (e.g. $a_{j,d} = 1$ for $d \notin \mathcal{I}$), since in many practical systems only certain sensors are miscalibrated.

Forecast priors and controlled miscalibration. Because AFORL explicitly consumes an episode-start prior Π_j over (A_j, b_j), the benchmark should vary prior quality in a controlled way. We recommend three prior sources: (i) an *oracle* prior centered at the true parameters with known covariance (upper bound on performance), (ii) a *learned forecaster* trained on past revealed parameters or proxy signals (realistic), and (iii) a *uninformative* prior (e.g. product uniform over admissible ranges). To parameterize miscalibration continuously, one may define a prior family $\Pi_j^{(\beta)}$ that interpolates between informative and uninformative priors, e.g. by tempering the oracle log-density or inflating covariance by a factor $\beta \geq 1$. We then report performance as a function of β (or any other scalar proxy for δ_Π in Theorem 3), since this directly probes the robustness of Bayesian selection to forecast error.

Baselines (comparison set). To evaluate the contribution of each component, we suggest the following baselines, all acting through the same frozen π:

1. **Oracle state (upper bound):** feed s_t to π. This isolates control difficulty from state estimation.

2. **No correction:** feed o_t directly to π (equivalently, pretend $A_j = I, b_j = 0$). This quantifies the raw brittleness of offline policies under affine sensor drift.

3. **Prior-only inversion:** use the prior mean (or MAP) $(\bar{A}_j, \bar{b}_j)$ and set $\hat{s}_t = \bar{A}_j^{-1}(o_t - \bar{b}_j)$. This measures what forecasting alone can do without in-episode adaptation.

4. **Online diagonal regression with a dynamics prior:** estimate (A_j, b_j) online by regressing o_t on a one-step predicted $\tilde{s}_t$ obtained from a learned dynamics model trained on $\mathcal{D}$. This baseline captures the classical "predict-then-calibrate" approach, but is sensitive to model bias in T.

5. **Recurrent estimator on raw observations:** train an RNN/transformer to output $\hat{s}_t$ from $(o_{0:t}, a_{0:t-1})$, supervised on identity data (where $o_t = s_t$) and evaluated under affine drift. This tests whether generic sequence models implicitly learn invariances, and typically fails without explicit augmentation.

6. **AFORL (full):** invariant features ϕ, conditional generator, OLS fitting, and Bayesian selection with Π_j.

Where feasible, we also recommend a *domain-randomized* training baseline: retrain the policy (or the estimator) with synthetic affine augmentations during offline training. This is not comparable as a deployment-only method, but it clarifies the extent to which robustness can be "baked in" versus "inferred at test time."

Core ablations. We suggest ablations aligned with the logical structure of AFORL:

1. **Invariant features vs. raw deltas:** replace $\phi(\tau)$ by the uncentered raw delta history $\{(\Delta o_i, a_{i-1})\}$, or by centered deltas without the log transform. This isolates the effect of Theorem 1's scale cancellation.

2. **Diagonal vs. uniform scaling:** evaluate the same method family under $A_j = a_j I$ and under diagonal A_j. The gap quantifies how much difficulty arises from per-coordinate ambiguity rather than a single global gain.

3. **Forecast quality:** sweep the prior miscalibration parameter (e.g. covariance inflation) and report sensitivity, since AFORL's selector explicitly combines likelihood (residual) and prior.

4. **Window length w:** sweep w on a logarithmic grid. Small w tests fast adaptation but weak identifiability; large w improves stability but may introduce lag under nonstationary latent dynamics. This sweep is also the cleanest empirical probe of the excitation requirement in Theorem 5.

5. **Candidate budget (k, σ):** sweep k and the scoring noise σ^2 to quantify the practical trade-off between compute and candidate coverage. We recommend reporting both mean performance and tail risk (e.g. CVaR over episodes) as k varies.

Metrics and reporting. If the simulator provides access to s_t (common in benchmarks), we recommend reporting (i) $\mathbb{E}\|\hat{s}_t - s_t\|_2$ aggregated over timesteps and episodes, (ii) a tail metric such as $\max_t \|\hat{s}_t - s_t\|_2$ per episode (or its percentile), and (iii) induced control loss, measured by episodic return under $\pi \circ \hat{s}$ compared to the oracle return under π with true s_t. Because the failure modes under affine drift are often episodic (catastrophic miscalibration in a minority of episodes), we recommend reporting not only the mean return but also quantiles (median, 10th percentile) and CVaR.

Finally, for reproducibility and interpretability, we recommend logging the per-episode fitted parameters $(\hat{A}_j, \hat{b}_j)$ and comparing them to ground truth when available, as well as reporting the selector's score gap $\text{score}_{(1)} - \text{score}_{(2)}$ as a proxy for epistemic uncertainty. These diagnostics help distinguish errors due to insufficient excitation (small OLS denominators), generator mismatch (no candidate explains the observations), and prior miscalibration (the correct candidate is penalized by Π_j), which correspond to distinct theoretical terms in our bounds.

9 Related Work

Offline RL under deployment shift. A recurring theme in offline reinforcement learning is that a policy learned from a fixed dataset can be brittle when deployed under distribution shift, even when the latent dynamics (T, R) remain unchanged; see, e.g., surveys of offline RL and OOD generalization **??**. Our setting isolates a particular and practically common shift: a change in the *observation map* rather than in the MDP itself. This differs from standard covariate shift analyses that assume the same semantics for the state coordinates at train and test time. Here, the semantics are preserved at the latent level (the state s_t), but the policy only receives an affine proxy $o_t = A_j s_t + b_j$ whose parameters vary by episode. The technical

consequence is that the offline policy π is not merely queried off-distribution; it is queried on inputs that are *systematically miscalibrated* by a structured, episodically constant transform.

Robustness to observation corruption and sensor drift. There is extensive work on robustness to corrupted observations in RL, including additive noise, partial observability, missingness, and adversarial perturbations **??**. A typical remedy is to train policies to be invariant via data augmentation or adversarial training, or to train recurrent policies that can denoise through temporal aggregation. In the offline setting, robustness is often pursued via conservative objectives, uncertainty penalties, or model-based regularization **??**. These approaches are not tailored to the episodically constant affine structure: in our problem the corruption is not i.i.d. per time step, and its parameters are shared across all timesteps within an episode. We exploit precisely this shared structure by (i) constructing per-window features $\phi(\tau)$ invariant to the nuisance parameters (A_j, b_j) and (ii) performing episode-wise parameter fitting given candidate latent trajectories. This is closer in spirit to classical *calibration* than to generic robustification.

FORL and forecasting-augmented RL. A separate line of work (which we refer to broadly as *forecasting-augmented RL*, or FORL) studies non-stationary environments in which exogenous, slowly varying parameters affect either dynamics or rewards, and one seeks to forecast these parameters and condition control on the forecast **??**. In such methods, the forecast is typically used to adapt the policy itself (e.g., by conditioning on a context variable, updating a belief state, or performing online RL). Our use of a forecast prior Π_j is different in two respects. First, Π_j concerns *observation parameters* rather than the latent MDP, so the optimal control law in latent space is unchanged; what changes is the map from observations to the inputs expected by the frozen policy. Second, we treat the forecaster as providing a *prior* in a Bayesian selection step rather than as an oracle context: the in-episode evidence, summarized by residuals under candidate explanations, is allowed to override a miscalibrated forecast. This interplay between a possibly wrong prior and in-episode likelihood is central to our guarantees (cf. the explicit δ_Π term).

Non-stationary RL, meta-RL, and online adaptation. Non-stationary RL and meta-RL address settings where the environment changes across episodes and agents must adapt, often by learning a latent context and updating it online **???**. Many such methods train policies end-to-end to incorporate adaptation mechanisms (e.g., RNN policies or gradient-based adaptation). By contrast, we impose the deployment constraint that the policy π is *frozen* (as is typical when π is a safety-certified controller or a

costly-to-retrain model). Consequently, adaptation must occur in an *estimator* that feeds $\hat{s}_t$ to π, rather than in the policy parameters. This shifts the technical focus from regret-style adaptation bounds to state-estimation error and induced value loss bounds for a fixed policy.

POMDP filtering, learned observers, and belief compression. Our test-time problem is a POMDP with deterministic observation function $o_t = A_j s_t + b_j$ and an episode-level latent variable (A_j, b_j). Classical filtering would maintain a belief over (s_t, A_j, b_j) given $(o_{0:t}, a_{0:t-1})$, which is generally intractable except for linear-Gaussian models (Kalman filters) or specific conjugate families **?**. Contemporary approaches learn neural observers or latent-state models and infer beliefs via amortized inference (e.g., variational RNNs, sequential VAEs) **??**. Our method can be viewed as an amortized filter with two design choices motivated by structure: (i) we compress the history into transform-invariant features $\phi(\tau)$ so that a generator trained under identity observations remains valid at test time, and (ii) we separate *candidate generation* (a learned conditional prior over s_t) from *episode-wise calibration* (OLS fitting of (A_j, b_j) and Bayesian scoring). This modularity is largely absent in monolithic learned filters, which tend to entangle dynamics, observation, and calibration in a single network and thus require training-time exposure to the corruption family.

Classical system identification with unknown sensor gain/bias. The subproblem of estimating (A_j, b_j) from paired signals resembles sensor calibration and errors-in-variables regression **?**. In control and signal processing, one often estimates sensor gains using known excitation signals, reference measurements, or redundant sensors (anchors). In our setting, we have neither reference sensors nor direct access to s_t; instead, we obtain *hypothesized* latent trajectories from a learned conditional model and then fit gains/biases by regression. This is reminiscent of blind calibration and self-calibration problems, where identifiability hinges on excitation and non-degeneracy conditions. Our Theorem 5 aligns with this classical necessity: if the latent signal has vanishing empirical variance in a coordinate, then gain/bias are not identifiable. Theorem 2 may be seen as a stability statement for the regression step under perturbations in the regressor (here, the candidate $\tilde{s}$).

Blind inverse problems and structural restrictions. If A_j were an unrestricted full matrix, the decomposition $o_t = A_j s_t + b_j$ would become a blind linear inverse problem closely related to blind deconvolution, ICA, and matrix factorization; without further structure one expects non-identifiability and computational obstacles **??**. The diagonal (or block/low-rank) restriction on A_j plays the same role as incoherence or sparsity assumptions in

those literatures: it reduces the hypothesis class to one where both identifiability and efficient estimation are plausible. Our use of per-dimension log-centering is specifically tuned to diagonal positive gains, providing an invariance that would not hold for generic mixing matrices.

Invariant representation learning and test-time adaptation. A closely related conceptual thread is invariant representation learning, where one aims to remove nuisance variation while preserving task-relevant information **?**. In vision and robotics, explicit invariances (e.g., to illumination, contrast, or affine transforms) are often engineered, while recent approaches seek to learn invariances by augmentation. Our feature map $\phi(\tau)$ is an explicit invariance derived from the affine structure and the episode-wise constancy of (A_j, b_j): differencing removes b_j, and window-centering in log-space removes $\log a_{j,d}$. This differs from generic test-time adaptation methods that update normalization statistics or model parameters at deployment **?**; we adapt *beliefs* over latent states while keeping both π and the generator fixed.

Summary of the distinction. Across these literatures, the closest analogues combine (i) an estimator that maps corrupted observations to latent states and (ii) a control policy operating in latent space. The distinctive feature of our approach is that the estimator is designed to be *trainable only on identity-observation offline data* while remaining applicable under episodic affine drift via analytic invariances and a structured Bayesian selection step incorporating a forecast prior. This design choice is what allows us to cleanly separate the learning problem (approximating $p(s_t \mid \phi)$ under the stationary MDP) from the deployment problem (resolving episode-wise calibration ambiguity under (A_j, b_j)).

10 Limitations and Extensions

We conclude by delineating the boundary of the present analysis and several directions in which the AFORL template plausibly extends. Our aim is not to assert that the same proofs go through verbatim, but rather to make explicit which parts of the construction are structural (and thus robust) and which are tuned to the diagonal, strictly-positive affine family.

Beyond diagonal gains: block structure and low rank. The pivotal simplification in our feature invariance is that, for each coordinate d, the observation difference obeys $\Delta o_t[d] = a_{j,d}\,\Delta s_t[d]$, so that log-centering cancels $\log a_{j,d}$ coordinate-wise. If A_j is block-diagonal with blocks of size $p > 1$, then $\Delta o_t^{(b)} = A_j^{(b)} \Delta s_t^{(b)}$ mixes coordinates within a block. In this case a coordinate-wise scalar invariance is unavailable, but one may still hope to

construct *block-wise* invariants. One candidate is to work with norms or singular values: for a block b,

$$\log\big(\|\Delta o_t^{(b)}\|_2 + \eta\big) = \log\big(\|A_j^{(b)} \Delta s_t^{(b)}\|_2 + \eta\big),$$

whose centered version removes only a scalar component (an "average gain") rather than the full matrix. More informative invariants can be built from ratios $\|\Delta o_t^{(b)}\|/\|\Delta o_{t'}^{(b)}\|$ or from Gram matrices $\Delta o_{t_1:t_w}^{(b)} (\Delta o_{t_1:t_w}^{(b)})^\top$, which transform as $A_j^{(b)} G_s (A_j^{(b)})^\top$. Such constructions suggest a natural extension of AFORL in which $\phi(\tau)$ comprises block-wise statistics and the episode-wise fitting step replaces per-dimension OLS with a structured regression over each block (possibly with a low-rank or orthogonality prior). However, identifiability becomes more delicate: if $A_j^{(b)}$ is an arbitrary $p \times p$ matrix, then without additional restrictions (e.g., symmetry, positive definiteness, sparsity, or known eigenvectors) one generally cannot expect uniqueness from a short window.

A related extension is to low-rank perturbations around diagonal structure, e.g. $A_j = D_j + U_j V_j^\top$ with $\mathrm{rank}(U_j V_j^\top) = r \ll n$. Here one may preserve the diagonal invariance approximately by treating $U_j V_j^\top \Delta s_t$ as an additional "noise" term whose magnitude depends on excitation along the low-rank subspace. This motivates a two-stage selector: (i) diagonal AFORL to obtain an initial $\hat{s}_t$ and implied $(\hat{D}_j, \hat{b}_j)$, and (ii) a refinement step that fits (U_j, V_j) via regularized least squares on the residuals. A rigorous analysis would require controlling the interaction between generator error and the low-rank fit, and quantifying when the low-rank component is distinguishable from sampling noise and model mismatch.

Nonlinear observation effects: clipping, saturation, and monotone distortions. Many sensors exhibit saturation or clipping, leading to observations of the form

$$o_t = \mathrm{clip}(A_j s_t + b_j; \ell, u),$$

or, more generally, $o_t = g_j(A_j s_t + b_j)$ for a monotone nonlinearity g_j that is constant within an episode. Our invariance arguments rely on affine structure and on the algebra of differences and logarithms; clipping breaks both, since differencing does not remove b_j once saturation occurs and the mapping is no longer injective. That said, several partial remedies appear feasible. First, one can detect saturation by checking whether o_t lies near ℓ or u for many consecutive steps, and then downweight those timesteps in the fitting objective or excise them from the window. Second, if g_j is monotone and smooth (e.g. a sigmoid-like sensor response), then order statistics and rank-based features are invariant to monotone transforms, suggesting a different choice of $\phi(\tau)$ built from within-window ranks of $\Delta o_t[d]$ rather than centered

log-magnitudes. This would trade off quantitative information (magnitudes) for robustness (ordering), and would likely increase the sample complexity in w needed to localize s_t.

An open theoretical issue here is to characterize the minimal conditions under which one can still recover useful control-relevant state information when the observation map is only approximately affine on the encountered range. In practice, one expects local linearization to be adequate provided s_t remains in a regime where g_j is nearly linear and the forecaster prior Π_j keeps the inferred affine parameters away from degenerate values.

Action-dependent or time-varying drift within an episode. We assumed that (A_j, b_j) is constant within episode j. If the sensor drift depends on action or evolves slowly in time, e.g. $o_t = A_{j,t}s_t + b_{j,t}$ with $A_{j,t}$ following a smooth process, then the window-centering step no longer cancels a constant $\log a_{j,d}$, and the OLS fit produces parameters that average across the window. A modest extension is to posit a parametric drift model, such as $a_{j,t,d} = a_{j,0,d} + \alpha_{j,d}t$ (or an AR(1) process), and to replace OLS by a regression that fits $(a_{j,0,d}, \alpha_{j,d}, b_{j,0,d}, \beta_{j,d})$ on the window. The Bayesian score can then incorporate a forecast prior over drift coefficients. The price is that identifiability requires stronger excitation: one must now distinguish genuine state changes from changes in the observation map, which is impossible if both are allowed to vary arbitrarily.

The action-dependent case, $A_{j,t} = A(a_t)$, is qualitatively harder because the corruption is then coupled to the control channel. In that regime, the estimator risks "explaining away" inconsistencies in o_t by attributing them to the action-dependent sensor, which can induce a feedback loop. A principled treatment would likely require either (i) known functional form for $A(a)$, (ii) occasional calibration actions, or (iii) additional instrumentation (anchors). We view this as a natural frontier where purely observational invariances are insufficient.

Partial anchors and occasional ground truth. Our formulation explicitly avoids reliance on reference measurements. Nonetheless, many deployments provide *partial anchors*: some state coordinates may be trusted (unaffected by drift), or one may occasionally observe a calibrated snapshot (e.g. at episode start), or receive delayed reveals of (A_j, b_j) for a subset of episodes. Each of these anchors can be integrated into AFORL by adding terms to the score or by constraining the regression fit. For example, if a subset of coordinates $\mathcal{I} \subset [n]$ is known to have $a_{j,d} = 1$ and $b_{j,d} = 0$, then candidates inconsistent with $o_t[d] = s_t[d]$ for $d \in \mathcal{I}$ can be rejected immediately, reducing multimodality and improving identifiability for the remaining coordinates through dynamic coupling. Similarly, if we observe a single calibrated pair $(s_{\text{ref}}, o_{\text{ref}})$ at some time, then $b_j = o_{\text{ref}} - A_j s_{\text{ref}}$ collapses part

of the ambiguity, leaving only A_j to infer; the regression then becomes more stable, especially under short windows.

From a theoretical standpoint, anchors effectively supply excitation externally, circumventing the lower bound in Theorem 5. Quantifying how many anchors (in time or in coordinates) suffice to guarantee a desired error level remains an appealing question, particularly when anchors are noisy and intermittent.

Modeling and algorithmic limitations. Our bounds depend on a generator accuracy parameter ε and on a separability margin γ in the Bayesian selector. In complex environments, the true conditional $p(s_t \mid \phi)$ may be highly multimodal, and the margin between modes may be small; then the required sample count k can be prohibitive. Moreover, our scoring step treats each candidate window independently except through the fitted $(\hat{A}_j, \hat{b}_j)$, whereas the true posterior over $(s_{0:t}, A_j, b_j)$ is temporally coupled. One extension is to maintain a particle filter over (A_j, b_j) across time, with AFORL-style conditional generation providing proposals for s_t; this would replace the per-step argmax by a resampling-and-weighting scheme and could reduce variance at the cost of additional computation.

A second limitation is that we implicitly assume the offline state distribution is sufficiently rich that the generator can learn $p(s_t \mid \phi)$ from identity-observation data alone. If the offline dataset $\mathcal{D}$ lacks coverage of the trajectories induced at test time by $\pi \circ \hat{s}$, then even perfect invariance cannot prevent extrapolation error. This is not unique to AFORL; it is the classical offline RL coverage problem manifesting in the estimator rather than in the policy.

Open problems. We highlight several concrete questions. (i) Can one characterize minimal excitation conditions for structured (block/low-rank) A_j that parallel Theorem 5, and derive sharp stability constants for the corresponding regression step? (ii) Can one replace the heuristic score resid/$(2\sigma^2)$ by a likelihood consistent with the generator and obtain end-to-end calibrated posteriors over (s_t, A_j, b_j)? (iii) How should one choose the window length w adaptively, trading bias (non-stationarity within the window) against variance (insufficient excitation)? (iv) Finally, can one design policies π (still trained offline) that are *implicitly probing* in the sense that, while optimizing reward, they also induce the excitation needed for calibration, thereby mitigating the necessity result when anchors are absent?

These extensions, while technically nontrivial, preserve the organizing principle of our approach: separate (a) an invariance-informed compression of recent history, (b) a learned conditional generator trained under the identity observation map, and (c) an explicit episode-wise calibration step that adjudicates among candidate explanations using a forecast prior and

in-episode evidence.