COMPOSITIONAL GENERALIZATION VIA GEOMETRIC ABSTRACTIONS IN SEQUENTIAL DECISION-MAKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Compositional generalization, the ability to recombine known components in new settings, in sequential decision-making requires identifying which parts of earlier rollouts remain useful for new tasks. Existing methods often consider reusing temporally extended skills and predictive models, but not rich local geometry and dynamics. We propose a geometric abstraction to quantify local transition geometry and dynamics and enable efficient compositional generalization in sequential decision-making. The proposed geometric descriptor aggregates both first- and second-order statistics of lifted one-step transitions within a trajectory segment in a positive semidefinite cone. We show that these descriptors make shared structure explicit, distinguish algebraic composition in an abstract representation space, and reveal opportunities for generalization. In particular, we prove that they are well defined up to coordinate gauge, complete for the induced low-order additive signal class, additive under valid segment composition, and minimally sufficient among admissible additive descriptors. Under explicit matrix-sufficiency, Bellman-compatibility, and local-smoothness assumptions, we further derive a local first-order approximation of action-value variation in matrix space, motivating reuse of matrix increments learned from existing geometry and dynamics. The proposed matrix-to-value mapping is plug-in compatible with standard model-free and model-based baselines to bootstrap value-function learning in new tasks, while obstruction filtering rejects implausible compositions. Empirically, MSRL achieves the best average finite-budget target AUC (0.73), improving over MSRL from scratch (0.65), the strongest source-initialized transfer baseline TD-MPC-PT+FT (0.63), and the strongest from-scratch baseline TD-MPC (0.57).

1 INTRODUCTION

Sequential decision making is formalized through states, actions, rewards, transition laws, and Bellman recursions (Bellman, 1966; Puterman, 2014; Sutton et al., 1998). Modern reinforcement learning has made this formalism effective through high-capacity value and policy models (Watkins & Dayan, 1992; Mnih et al., 2015; Schulman et al., 2017; Haarnoja et al., 2018). Reusing knowledge across related tasks, however, remains difficult: even when tasks share local geometry or dynamics, it is not obvious how to identify transferable structure, leverage trajectories or demonstrations with different appearances, or improve value and policy learning without inducing infeasible behavior (Taylor & Stone, 2009; Lazaric, 2012; Rusu et al., 2016). Existing methods reuse temporally extended skills, predictive occupancies, goal-conditioned values, or behaviorally compressed states (Sutton et al., 1999; Ravindran, 2003; Barreto et al., 2019; Dayan, 1993; Barreto et al., 2017; 2018; Machado et al., 2017; Schaul et al., 2015; Andrychowicz et al., 2017; Ferns et al., 2004; Givan et al., 2003; Li et al., 2006; Abel et al., 2016; Castro, 2020; Zhang et al., 2020), but they generally do not expose local transition geometry and dynamics through a fixed descriptor with an explicit segment-composition rule. Forward–Backward representations instead learn reward-free long-horizon state relations for zero-shot task specification; our goal is different: a descriptor computed directly from observed transition segments with exact segment addition and a separate realizability test (Touati et al., 2022).

This paper proposes a novel geometric abstraction to quantify local transition dynamics/geometry and to enable efficient compositional generalization in sequential decision-making. We view a trajectory segment as a finite consecutive block of transitions from a rollout, where segments

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.

may share common prefixes, suffixes, full rollouts, or partial overlaps. This allows us to describe trajectories/demonstrations of different appearances (e.g., horizon length, task realizations, and action/space representations) through a common set of geometric descriptors within one unified space. These descriptors make shared hidden structures explicit, distinguish algebraic composition in an abstract representation space, and reveal opportunities for compositional generalization. We detail our **four contributions** below.

(1) We introduce the trajectory-segment matrix $M(\tau)$ ***as a geometric descriptor*** (Section 2) to capture low-order local geometry and dynamics in a compositional form. Given a trajectory segment $\tau = ((o_t, a_t, r_t, o_{t+1}))_{t=0}^{T-1}$, where o_t and o_{t+1} are consecutive observations, a_t is the action, r_t is the reward, and T is the segment length, we define a fixed step lift $\psi_t(\tau)$ that records the embedded transition midpoint, directed displacement, action-conditioned midpoints and displacements, reward, and a constant channel. We then define $M(\tau) = \sum_{t=0}^{T-1} \psi_t(\tau)\psi_t(\tau)^\top$, a covariance-like matrix aggregating second-order statistics of lifted one-step transitions in a positive semidefinite cone (Tuzel et al., 2006; Pennec et al., 2006; Muandet et al., 2017; Johnson, 1970).

The goal of this abstraction is to capture critical local geometry and dynamics that induce compositional generalization structures into value learning, while capturing only minimal information needed for sequential decision-making. More precisely, the abstraction is chosen to retain the low-order transition information used by the value learning: where transitions occur, which directions and actions induce movement, how rewards align with them, and how first-order totals accumulate (Kondor & Lafferty, 2002; Gärtner et al., 2003; Mahadevan & Maggioni, 2007). Because $M(\tau)$ is a commutative sum, it does not encode the full chronological sequence (Graves, 2012; Vaswani et al., 2017) which often has different appearances across tasks or environments. The benefit of our proposed abstraction is a transparent algebra: harnessing compatible segments in compositional generalization becomes matrix addition (while preserving structure of the abstraction), and overlapping local restrictions can be assembled by inclusion–exclusion operators.

(2) We prove a set of key foundational properties of the proposed geometric descriptor (Section 3). We prove that $M(\tau)$ is well defined up to coordinate gauge, complete for the induced low-order additive signal class, additive under valid segment composition, recoverable from overlapping restrictions, and separable from environment-level realizability through an obstruction score. We also prove that $M(\tau)$ is minimally sufficient among admissible additive descriptors. These results establish an algebra on the $M(\tau)$ construction and justify the use of the proposed abstraction for transferring and learning value functions in sequential decision-making.

(3) We develop a matrix-space reinforcement learning (MSRL) theory (Section 5) by connecting the descriptor to value in Section 4. Under explicit matrix-sufficiency, Bellman-compatibility, and local-smoothness assumptions, small admissible matrix increments have first-order additive effects on the centered action value. This gives a local rationale for reusing matrix increments rather than a global linearity claim. MSRL pretrains an $M \to V$ branch on source environments, uses it to initialize the target critic, and learns target-specific residual structure from target interaction. The same interface can be added to standard model-free or model-based backbones (Fujimoto et al., 2018; Janner et al., 2019; Hafner et al., 2025; Hansen et al., 2022; Taylor & Stone, 2009; Lazaric, 2012; Rusu et al., 2016).

(4) We validate our MSRL method using source-side structural diagnostics and source-initialized transfer from simple environments to complex target tasks. Under matched target interaction budgets, source-pretrained MSRL improves finite-budget AUC over its from-scratch counterpart and the strongest source-initialized baseline, while convergence curves report full-budget behavior separately.

We finally conclude with a discussion of limitations in Section 7.

2 Matrix Proxy for Sampled Compositional Structure

This section defines the descriptor used throughout the paper. The goal is to turn observed rollout data into a fixed-dimensional object that can be stored, compared, composed with other segments, and supplied to a matrix-conditioned value model. A trajectory segment is any finite consecutive block of transitions from a rollout; different segments may have different lengths and may overlap when extracted from the same trajectory. Once a common step lift is fixed, every such segment is mapped to one matrix in a shared positive semidefinite cone.

Fixed step lift. For each environment e, a fixed deterministic adapter maps benchmark variables into common spaces $\mathcal{O}$ and $\mathcal{A}$. We call a family of environments *aligned* when corresponding geometric quantities and action channels occupy the same dimensions and block order wherever matrices are compared or transferred; no cross-environment semantic encoder is learned. This is an explicit interface condition, not a claim that arbitrary environments are automatically aligned. Fix $\phi : \mathcal{O} \to \mathbb{R}^d, \qquad \xi : \mathcal{A} \to \mathbb{R}^q$, for the aligned family under study.

For a finite observable trajectory segment $\tau = ((o_t, a_t, r_t, o_{t+1}))_{t=0}^{T-1}$, the local index t ranges over the T transitions inside this segment; o_t and o_{t+1} are consecutive observations, a_t is the action applied at the transition, and r_t is the observed reward. Define $x_t := \phi(o_t), \quad x_t^+ := \phi(o_{t+1}), \quad c_t := \frac{x_t + x_t^+}{2}, \quad \delta_t := x_t^+ - x_t$. Here c_t is the midpoint of the embedded one-step transition, and δ_t is its directed displacement. To place local position, directed displacement, action dependence, reward, and additive first-order statistics in one common space, define the segment-dependent lifted vector $\psi_t(\tau) := \begin{bmatrix} c_t^\top & \delta_t^\top & \big(\xi(a_t) \otimes c_t\big)^\top & \big(\xi(a_t) \otimes \delta_t\big)^\top & r_t & 1 \end{bmatrix}^\top \in \mathbb{R}^m$, where $m := 2d + 2qd + 2$. All lifted quantities are assumed finite throughout the definition.

Definition 2.1 (Matrix proxy induced by a trajectory segment)**.** The matrix proxy of τ is $M(\tau) := \sum_{t=0}^{T-1} \psi_t(\tau)\psi_t(\tau)^\top \in \mathbb{S}_+^m$.

The outer-product aggregation can be viewed as a finite-dimensional PSD descriptor of lifted local transitions; PSD matrix geometry, covariance descriptors, kernel embeddings, and graph kernels provide useful background for this representation choice (Johnson, 1970; Tuzel et al., 2006; Muandet et al., 2017; Vishwanathan et al., 2010; Shervashidze et al., 2011). The role here is operational: segments, prefixes, and full rollouts all have the same matrix dimension, so a replay library can store them as reusable increments and a single value model can evaluate accumulated matrix states.

Because $M(\tau)$ is a commutative sum of one-step outer products, it records low-order additive statistics of the lifted transitions rather than the full chronological sequence. It may distinguish a segment from its time reversal when the lifted transition vectors change, but it does not distinguish two segments that contain the same multiset of lifted one-step vectors in a different order. Symbolic transition-pattern examples are deferred to Appendix C; blockwise interpretation is deferred to Appendix A.

3 Structural Guarantees of the Matrix Proxy

The six properties below are the requirements needed later for matrix-space control. Gauge covariance makes comparisons meaningful under aligned coordinate changes. Completeness specifies exactly which low-order additive signals are retained. Composition and gluing justify reusing stored local segments in a common matrix algebra. Minimal sufficiency states that, within the admissible additive class, no additional history representation is needed to recover those signals. Obstruction separates formal matrix compositions from candidates that are realizable in the environment.

3.1 Gauge covariance and well-definedness

Theorem 3.1 (Well-definedness up to gauge)**.** *For every finite trajectory segment satisfying Definition 2.1, $M(\tau)$ is well defined, symmetric, and positive semidefinite. Moreover, if $\widetilde{\phi}(o) = S\phi(o), \qquad \widetilde{\xi}(a) = U\xi(a)$, with $S \in GL(d)$ and $U \in GL(q)$, then $\widetilde{M}(\tau) = HM(\tau)H^\top, \qquad H := \mathrm{BlkDiag}(S, S, U \otimes S, U \otimes S, 1, 1)$. Hence the intrinsic descriptor is the congruence class $[M(\tau)]$, rather than any particular coordinate realization.*

This gauge-covariant reading is consistent with geometric representation-learning principles in which the coordinate realization is less important than the transformation law of the descriptor (Bronstein et al., 2021).

The next definition fixes the information claimed to be preserved. The class is intentionally low-order: it contains the additive first-order and quadratic signals exposed by the lift, which are the signals used by the matrix-conditioned value interface below.

3.2 COMPLETENESS FOR THE CHOSEN LOW-ORDER CLASS

Definition 3.2 (Low-order additive compositional class)**.** Write $\bar{\psi}_t := \begin{bmatrix} c_t^\top & \delta_t^\top & \big(\xi(a_t)\otimes c_t\big)^\top & \big(\xi(a_t)\otimes \delta_t\big)^\top & r_t \end{bmatrix}^\top \in \mathbb{R}^{m-1}$, $\psi_t = \begin{bmatrix} \bar{\psi}_t^\top & 1 \end{bmatrix}^\top$. Let $\mathfrak{F}_{\mathrm{aq}}$ be the class of trajectory-segment functionals F for which there exist $a \in \mathbb{R}$, $b \in \mathbb{R}^{m-1}$, and $C \in \mathbb{S}^{m-1}$ such that $F(\tau) = \sum_{t=0}^{T-1} \Big(a + b^\top \bar{\psi}_t + \bar{\psi}_t^\top C \bar{\psi}_t \Big)$. Thus $\mathfrak{F}_{\mathrm{aq}}$ contains exactly the additive first-order and quadratic statistics induced by the fixed lift. These are the low-order signals of sampled compositional structure preserved by the descriptor.

Theorem 3.3 (Completeness)**.** *For every $F \in \mathfrak{F}_{\mathrm{aq}}$, there exists $A \in \mathbb{S}^m$ such that $F(\tau) = \langle A, M(\tau)\rangle$ for every finite trajectory segment τ. Conversely, every $A \in \mathbb{S}^m$ defines a functional in $\mathfrak{F}_{\mathrm{aq}}$ through the same formula. Consequently, $M(\tau) = M(\tau') \iff F(\tau) = F(\tau')$ for all $F \in \mathfrak{F}_{\mathrm{aq}}$. Equivalently, $M(\tau)$ captures exactly the chosen low-order additive information induced by the fixed lift.*

The next two results give the representation-level algebra. A valid concatenation $\tau = \tau^{(1)} \star \tau^{(2)}$ means that $\tau^{(1)}$ and $\tau^{(2)}$ are consecutive segment blocks from the same rollout, or from compatible rollouts whose terminal and initial observations match under the aligned representation. Such concatenation becomes matrix addition, while overlapping local restrictions combine through inclusion–exclusion in the same ambient space.

3.3 COMPOSITION AND GLUING

Theorem 3.4 (Composition)**.** *If $\tau = \tau^{(1)} \star \tau^{(2)}$ is a valid concatenation, then $M(\tau) = M(\tau^{(1)}) + M(\tau^{(2)})$. Therefore concatenated segments induce composed descriptors in the same matrix space.*

Each local segment contributes one matrix increment interpreted as sampled local structure. The theorem says that compositional assembly is linear in matrix space even though the raw observations arrive in sequential form.

Theorem 3.5 (Local-to-global gluing)**.** *Let $\{I_k\}_{k=1}^K$ be an interval cover of $\{0, \dots, T-1\}$. For each nonempty $S \subseteq \{1, \dots, K\}$, write $I_S := \bigcap_{k\in S} I_k$, $\tau_S := \tau|_{I_S}$ when $I_S \neq \varnothing$. Here $\tau|_{I_S}$ denotes the subsegment containing exactly the transitions whose local indices lie in I_S. Let $\mathcal{N} := \{S \subseteq \{1, \dots, K\} : S \neq \varnothing,\ I_S \neq \varnothing\}$. Then $M(\tau) = \sum_{S\in\mathcal{N}} (-1)^{|S|+1} M(\tau_S)$. Hence the global matrix is recovered exactly from overlapping local restrictions by inclusion–exclusion: matrices from individual intervals include all local transitions, and matrices from nonempty intersections subtract the duplicate transition contributions introduced by overlaps.*

More generally, any family of matrices $\{M_S\}_{S\in\mathcal{N}}$ indexed by the same nonempty overlaps defines a unique candidate glued matrix $\widehat{M} := \sum_{S\in\mathcal{N}} (-1)^{|S|+1} M_S \in \mathbb{S}^m$. This is a representation-level local-to-global construction in the same ambient dimension. Whether $\widehat{M}$ is realizable by an actual trajectory in a given environment is addressed separately by Theorem 3.10.

The gluing theorem formalizes the local-to-global interpretation used by replay libraries: the learner may store windows, prefixes, or partially overlapping fragments rather than a single complete task-level trajectory, and the matrix algebra specifies how to assemble their nonduplicated contributions.

The next result concerns minimal sufficiency within the admissible additive class. This is the relevant comparison class for the method because the matrix-conditioned value interface and segment library use additive first-order and quadratic statistics of the fixed lift, not unrestricted history representations.

3.4 MINIMAL SUFFICIENCY

Definition 3.6 (Admissible additive descriptor)**.** A descriptor D is called admissible if, for some fixed $p \in \mathbb{N}$, it maps each finite trajectory segment τ to a point $D(\tau) \in \mathbb{R}^p$, is additive under valid concatenation, and is sufficient for all functionals in $\mathfrak{F}_{\mathrm{aq}}$ in the following factorization sense: for every $F \in \mathfrak{F}_{\mathrm{aq}}$, there exists a Borel measurable map $h_F : \mathbb{R}^p \to \mathbb{R}$ such that $F(\tau) = h_F(D(\tau))$ for every finite trajectory segment τ.

Theorem 3.7 (Minimal sufficiency)**.** *The matrix proxy M is minimally sufficient within the class of admissible additive descriptors: if D is any admissible descriptor, then there exists a measurable map $\rho : \mathbb{R}^p \to \mathbb{S}^m$ such that $M(\tau) = \rho(D(\tau))$ for every τ. Hence M is the coarsest admissible additive descriptor for the class $\mathfrak{F}_{\mathrm{aq}}$, up to measurable post-processing.*

The final result adds the environment-specific layer. Composition and gluing are closed at the level of descriptor algebra, but realizability depends on the actual environment.

3.5 Realizability and obstruction

Definition 3.8 (Reachable matrix set)**.** For an environment e and horizon T, define the reachable matrix set $\mathcal{R}_e^{(T)} := \{M(\tau) : \tau \text{ is realizable in } e \text{ with horizon } T\}$.

Definition 3.9 (Obstruction score)**.** For a candidate matrix $\widehat{M} \in \mathbb{S}^m$, define $\mathrm{Obs}_e^{(T)}(\widehat{M}) := \inf_{M \in \mathcal{R}_e^{(T)}} \|\widehat{M} - M\|_F$.

Theorem 3.10 (Obstruction)**.** *Let $\widehat{M}$ be a matrix obtained by composition or gluing of local matrices, and let T be the candidate horizon encoded by its constant-channel block, $\widehat{M}_{\mathbf{11}} = T$. If $\mathrm{Obs}_e^{(T)}(\widehat{M}) > 0$, then $\widehat{M}$ does not correspond to any realizable trajectory in environment e at horizon T. Therefore positive obstruction is a certificate of inconsistency with the environment.*

No converse is claimed without additional closure assumptions on $\mathcal{R}_e^{(T)}$. The theorem separates two levels of validity: algebraic validity in matrix space and realizability in the actual sequential decision environment.

4 Value as a Low-Order Signal on Matrix Space

The structural theorems above explain why $M(\tau)$ is compositional and sufficient for a clear low-order information class. The control interface additionally requires a relationship between the matrix descriptor and value, connecting the representation to the Bellman view of sequential decision making (Bellman, 1966; Puterman, 2014; Sutton et al., 1998). Theorem 3.3 and Appendix C show that M exactly retains the lift-induced low-order structural quantities, including reversal parity, path endpoint displacement, cycle closure, and oriented circulation. We do not claim that M preserves the full topology of the grounded transition graph; the preserved content is the matrix-visible structural information induced by the common lift. The remaining question for control is whether those preserved directions are sufficient for Bellman recursion and local value variation.

Assumption 4.1 (Matrix sufficiency for optimal value)**.** For each environment e in the aligned family, let $Q_e^\star(o, h, a)$ denote the optimal action-value function when the current observation is o, the past rollout prefix is h, and the candidate action is a. There exists a measurable function $\bar{Q}_e^\star : \mathcal{O} \times \mathbb{S}_+^m \times \mathcal{A} \to \mathbb{R}$ such that, for every reachable rollout prefix $h_t = \tau_{0:t-1}$ and every action a, $Q_e^\star(o_t, h_t, a) = \bar{Q}_e^\star(o_t, M(h_t), a)$. Equivalently, any two reachable prefixes with the same current observation and the same accumulated matrix state induce the same optimal action values.

Assumption 4.1 is not an existence claim for an arbitrary auxiliary function. It is a sufficiency hypothesis for the optimal value: after conditioning on the current observation, the history dependence that is relevant to optimal control is assumed to enter through the accumulated matrix state $M(h_t)$. This need not hold for arbitrary partially observed or nonstationary environments. It specifies the regime in which the matrix descriptor is intended to act as a control-relevant state abstraction, analogous in role to goal-conditioned value parameterizations, successor-feature decompositions, and bisimulation-based state abstractions (Schaul et al., 2015; Barreto et al., 2017; 2018; Ferns et al., 2004; Zhang et al., 2020; Gelada et al., 2019).

Assumption 4.2 (Lifted Bellman compatibility)**.** For each environment e, the lifted pair (o, M) admits a well-defined one-step transition kernel. That is, for every reachable (o, M) and action a, there exists a kernel $P_e(dr, do' \mid o, M, a)$ such that the conditional law of the next reward and observation, given any rollout prefix h_t satisfying $(o_t, M(h_t)) = (o, M)$, is $P_e(\cdot \mid o, M, a)$. With the one-step lift denoted by $\psi(o, a, r, o')$ and $M^+ = M + \psi(o, a, r, o')\psi(o, a, r, o')^\top$, the matrix-conditioned optimal value satisfies $\bar{Q}_e^\star(o, M, a) = \mathbb{E}_{(r,o') \sim P_e(\cdot \mid o, M, a)} \left[r + \gamma \sup_{a' \in \mathcal{A}} \bar{Q}_e^\star(o', M^+, a') \right]$, for a discount factor $\gamma \in [0, 1)$, whenever the right-hand side is well defined.

Assumption 4.2 is the lifted-state Markov condition needed to write Bellman recursion on (o, M). It is a strong modeling assumption, but it is explicit: the claim is not that M is universally sufficient, but that when the matrix state is a sufficient statistic for the value-relevant history, dynamic programming can be carried out in the lifted matrix state space. For the remainder of this section, fix an environment e and suppress the subscript e on P_e, $\bar{Q}^\star_e$, and related lifted quantities.

Proposition 4.3 (Exact Bellman reduction to the lifted matrix state)**.** *Define the lifted Bellman optimality operator on measurable $f : \mathcal{O} \times \mathbb{S}^m_+ \times \mathcal{A} \to \mathbb{R}$ by $(\mathcal{T}_M f)(o, M, a) := \mathbb{E}\big[r + \gamma \max_{a' \in \mathcal{A}} f\big(o', M + \psi(o, a, r, o')\psi(o, a, r, o')^\top, a'\big) \mid o, M, a\big]$. Under Assumptions 4.1 and 4.2, $\bar{Q}^\star = \mathcal{T}_M \bar{Q}^\star$, and for every reachable rollout prefix $\tau_{0:t-1}$, $Q^\star_e(o_t, \tau_{0:t-1}, a) = \bar{Q}^\star(o_t, M(\tau_{0:t-1}), a)$. Therefore the Bellman recursion closes exactly on the lifted state (o, M).*

This proposition is included to fix the lifted Bellman notation used below. It follows directly from Assumptions 4.1 and 4.2: under these assumptions, Bellman recursion can be written on (o, M) instead of the full history.

Assumption 4.4 (Ambient C^1 matrix-value representative)**.** For every fixed (o, a), there exist an open set $\mathcal{U}_{o,a} \subseteq \mathbb{S}^m$ containing the reachable matrix states used for evaluation and a function $\widehat{Q}^\star_{o,a} : \mathcal{U}_{o,a} \to \mathbb{R}$ such that $\widehat{Q}^\star_{o,a}(M) = \bar{Q}^\star(o, M, a)$ for every such reachable M, and $\widehat{Q}^\star_{o,a}$ is Fréchet differentiable on $\mathcal{U}_{o,a}$ with derivative locally bounded in the operator norm induced by the Frobenius norm.

Remark 4.5 (Interpretation of the smoothness assumption)*.* The realizable matrix-state set need not be open in $\mathbb{S}^m_+$, so Fréchet differentiability of $M \mapsto \bar{Q}^\star(o, M, a)$ is not naturally stated on the constrained domain itself. Assumption 4.4 therefore postulates a differentiable ambient representative in matrix space. It is not a claim that elementary edits of the grounded transition graph are infinitesimal: a single added edge may induce the finite update $M \mapsto M + \psi\psi^\top$. The first-order statements below apply only to local comparisons whose induced matrix perturbation is small in Frobenius norm.

Theorem 4.6 (First-order Bellman-value linearization in matrix space)**.** *Fix (o, a) and a reference matrix M. Let $H_0, H_1 \in \mathbb{S}^m$ be admissible compositional perturbations such that $M + H_0$, $M + H_1$, and $M + H_0 + H_1$ lie in the relevant domain, and let $\|H_0\|_F + \|H_1\|_F \to 0$. Under Assumptions 4.1, 4.2, and 4.4, there exists $W_{o,M,a} \in \mathbb{S}^m$, the Riesz representative of $D\widehat{Q}^\star_{o,a}(M)$ under the Frobenius inner product, such that $\bar{Q}^\star(o, M + H, a) = \bar{Q}^\star(o, M, a) + \langle W_{o,M,a}, H\rangle + r_M(H)$, $r_M(H) = o(\|H\|_F)$. Consequently, $\bar{Q}^\star(o, M + H_0 + H_1, a) - \bar{Q}^\star(o, M, a) = \big[\bar{Q}^\star(o, M + H_0, a) - \bar{Q}^\star(o, M, a)\big] + \big[\bar{Q}^\star(o, M + H_1, a) - \bar{Q}^\star(o, M, a)\big] + o(\|H_0\|_F + \|H_1\|_F)$. In particular, when 0 lies in the ambient differentiability neighborhood and the expression is centered at $M = 0$, $\bar{Q}^\star(o, H_0 + H_1, a) \approx \bar{Q}^\star(o, H_0, a) + \bar{Q}^\star(o, H_1, a) - \bar{Q}^\star(o, 0, a)$. Thus, conditional on the lifted-state sufficiency assumptions, sufficiently small admissible matrix increments have first-order additive effects on the centered value; this is a local statement, not a global linearity claim for finite stored segments.*

Theorem 4.6 gives the local value rule used by MSRL: small compositional matrix increments have approximately additive effects on the matrix-conditioned value. The statement is local. It does not claim exact additivity of optimal value, and it does not imply that adding an arbitrary discrete segment produces a small value change. The approximation applies when the induced perturbation is small in Frobenius norm.

The exactly affine case is a useful sanity check, but the method does not rely on exact linearity; the operative claim is the local first-order approximation in Theorem 4.6.

Corollary 4.7 (Centered compositional value law)**.** *Define the centered matrix-conditioned value $\widetilde{Q}^\star(o, M, a) := \bar{Q}^\star(o, M, a) - \bar{Q}^\star(o, 0, a)$. Under the assumptions of Theorem 4.6, $\widetilde{Q}^\star(o, H_0 + H_1, a) = \widetilde{Q}^\star(o, H_0, a) + \widetilde{Q}^\star(o, H_1, a) + o(\|H_0\|_F + \|H_1\|_F)$. Hence the centered value law is approximately additive on compositional matrix increments.*

A further consequence is local value stability: matrix-preserving transformations leave the matrix-conditioned value unchanged, while perturbations orthogonal to the local value gradient are invisible to first order. We state this auxiliary result in Appendix F.10, because the main algorithm only uses the first-order compositional rule above.

At the population level, the matrix-conditioned value law also implies that the full transition history cannot improve the optimal squared-loss value target once (o, M) is known; the formal sufficiency statement is deferred to Appendix F.10.

Taken together, the lifted Bellman closure and Theorem 4.6 justify using M as the planning state for MSRL: the Bellman equation is written on (o, M), and local value variation is governed by compositional matrix increments. Auxiliary stability and regression-sufficiency consequences are given in Appendix F.10.

5 Matrix-Space Reinforcement Learning for Compositional Generalization

MSRL has three interfaces: a fixed lift constructs Z_t, a target critic and policy learn control on (o_t, Z_t), and a bounded segment library supplies candidate increments that are filtered by a target-trained obstruction proxy. Across environments, only the source-trained matrix-to-value branch initializes the target critic; the lift coordinates are fixed by the declared alignment, while the target observation pathway, actor or action head, critic remainder, policy optimization, and obstruction proxy remain target-specific. Formal training details are in Appendix D, with pseudocode in Algorithm 1.

Lifted planning state. For a rollout prefix $\tau_{0:t-1}$, define $Z_t := M(\tau_{0:t-1}) = \sum_{s=0}^{t-1} \psi_s \psi_s^\top$. Then $Z_{t+1} = Z_t + \psi_t \psi_t^\top$. We use the lifted state $\widetilde{s}_t := (o_t, Z_t)$, where o_t supplies the current local decision context and Z_t stores the accumulated sampled structure available to the agent through past interaction.

Composable segment increments. A local trajectory segment $\sigma \in \mathcal{L}$ contributes the matrix increment $M(\sigma)$. By Theorem 3.4, a compatible segment can be appended to the current lifted state by $Z_t \mapsto Z_t + M(\sigma)$. Thus the segment library is not a separate symbolic planner: it is a collection of matrix increments that can be evaluated by the same value model used for ordinary lifted states.

Segment library. The replay memory stores a bounded library $\mathcal{L}$ of local trajectory segments. Each entry stores $M(\sigma)$, its head action, entry context, exit observation, and optional compatibility tags. In the reported target-control experiments, $\mathcal{L}$ contains at most $5{,}000$ one-transition entries, so candidate construction is bounded retrieval rather than enumeration of replay subtrajectories; full details are in Appendix E.1.

Admissible composed futures. Given the current lifted state (o_t, Z_t) and a candidate action a, define the action-conditioned candidate set $\mathcal{C}(o_t, Z_t, a) \subseteq \Big\{ \big(o^{\mathrm{tail}}(\sigma),\, Z_t + M(\sigma)\big) : \sigma \in \mathcal{L},\ a^{\mathrm{head}}(\sigma) \sim_e a \Big\}$. Here $\sim_e$ denotes compatibility in the fixed action encoding: exact equality for discrete actions and the implementation's encoded compatibility rule for continuous actions. A candidate $(\widehat{o}, \widehat{Z}) \in \mathcal{C}(o_t, Z_t, a)$ is retained only if the segment entry context is compatible with the current local context and its realizability score satisfies $g_\omega(o_t, Z_t, \widehat{o}, \widehat{Z}) \le \varepsilon_{\mathrm{obs}}$, where g_ω is a learned proxy for obstruction. Thus composed futures remain tied both to the current action and to a well-defined future local context.

Action-conditioned matrix-space planning. MSRL can be implemented on top of either a discrete-action critic or an actor–critic method for continuous control. The parameter vector θ denotes the target critic parameters, including the matrix-value branch initialized from the source-trained parameters η_{src}. The lookahead score below is not a separate learned network; it is computed from the same critic evaluated on obstruction-filtered composed futures.

For discrete actions, let $V_\theta(o, Z) := \max_{a \in \mathcal{A}} Q_\theta(o, Z, a)$. For continuous-control tasks, let π_ν denote the target actor and use the actor-induced value $V_\theta(o, Z) := Q_\theta(o, Z, \pi_\nu(o, Z))$, with the dependence on ν suppressed in V_θ for notational economy. For each candidate action a, define $U_\theta(o_t, Z_t, a) := \begin{cases} \max\limits_{(\widehat{o}, \widehat{Z}) \in \mathcal{C}(o_t, Z_t, a)} V_\theta(\widehat{o}, \widehat{Z}), & \mathcal{C}(o_t, Z_t, a) \neq \varnothing, \\ 0, & \mathcal{C}(o_t, Z_t, a) = \varnothing. \end{cases}$ Action selection uses

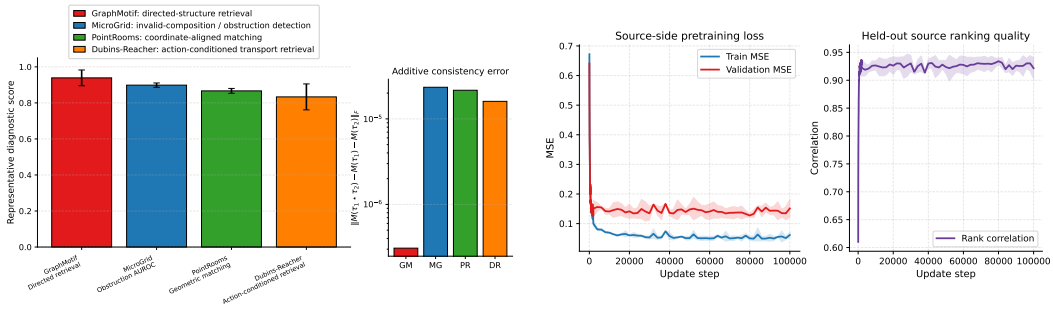


Figure 1: Part I source diagnostics and value pretraining. Left: the diagnostic score is a normalized task-specific metric: GraphMotif directed-structure retrieval accuracy, MicroGrid obstruction AUROC, PointRooms coordinate-aligned matching accuracy, or Dubins-Reacher action-conditioned transport retrieval accuracy; higher is better, and scores are not averaged across diagnostics. Additive consistency reports $\|M(\tau_1 \star \tau_2) - M(\tau_1) - M(\tau_2)\|_F$. Right: MSE is mean-squared error between $F_{\eta_{\mathrm{src}}}(Z)$ and $V_{\mathrm{src}}(Z)$; rank correlation is between predicted and target source values on held-out prefixes/segments. Takeaway: $M(\tau)$ is interpretable and supports source-side $M \to V$ pretraining.

$S_\theta(o_t, Z_t, a) := Q_\theta(o_t, Z_t, a) + \beta\, U_\theta(o_t, Z_t, a), \qquad \beta \geq 0$. In source-augmented target training, the source-trained matrix-value branch initializes the target critic, while the target observation adapter, action head, policy, and obstruction proxy are trained with target-environment data. The planning term biases the target policy toward admissible segment continuations whose matrix increments are both value-relevant and plausible under the learned obstruction filter.

6 Experiments

We evaluate three questions: *Q1* whether $M(\tau)$ exposes interpretable structure in simple source environments; *Q2* whether MSRL is competitive on complex target control; and *Q3* whether source-trained matrix-value knowledge improves target learning under the same target interaction budget. Part I validates $M(\tau)$ and pretrains $F_{\eta_{\mathrm{src}}}$ on inspectable source environments; Part II transfers only this matrix-value branch to complex targets, while target policies, action heads, and observation adapters are trained target-side.

6.1 Part I: Interpretable Heterogeneous Source Environments

Part I computes trajectory matrices in GraphMotif, MicroGrid, PointRooms, and Dubins-Reacher, whose known structures make the descriptor inspectable. These environments test directed motifs, invalid composition/obstruction, coordinate-aligned geometry, and action-conditioned transport; they also provide source matrix states for pretraining $F_{\eta_{\mathrm{src}}}$. Figure 1 answers Q1 by showing that matrix-space diagnostics recover these structures and that the resulting source value predictor is usable before Part II transfer.

6.2 Part II: Complex Control with Source-Augmented Target Training

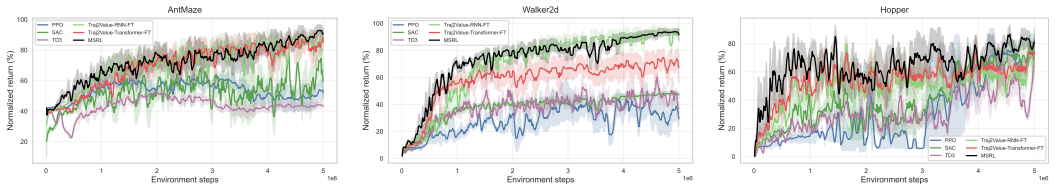
Part II tests transfer to AntMaze, Reacher/Pusher, Walker2d, Hopper, and Humanoid variants. We pretrain $F_{\eta_{\mathrm{src}}}(Z) \approx V_{\mathrm{src}}(Z)$ on source matrix states, initialize $\eta_{\mathrm{tgt}}^{(0)} = \eta_{\mathrm{src}}$, and then train the target critic and policy with target interactions. Thus MSRL transfers a matrix-value initialization, not a source policy, source action model, or zero-shot controller.

We compare with from-scratch RL baselines, source-initialized transfer baselines, and MSRL ablations. Normalized few-shot AUC over the first 25% of target training answers Q2–Q3 under matched target budgets, while full learning curves report terminal behavior. Table entries are mean$\pm$standard deviation over five seeds. Details and negative controls are in Appendix E.

Figure 1 answers Q1 by validating interpretable source-side structure and additivity. Table 1 answers Q2: MSRL from scratch improves over the strongest target-only baseline in average AUC (0.65

Table 1: Normalized few-shot AUC on the complex-control target suite. Entries are mean±std over seeds, normalized within each target family. Baselines include from-scratch control, source-initialized transfer, and MSRL ablations; the key comparison is MSRL from scratch versus MSRL with source-pretrained $F_{\eta_{\mathrm{src}}}$ initialization.

Method	Initialized component	AntMaze	Reacher/Pusher	Walker2d	Hopper	Humanoid	Avg. AUC
PPO	none	0.36±0.07	0.41±0.06	0.45±0.05	0.53±0.06	0.34±0.07	0.40
SAC	none	0.43±0.06	0.50±0.05	0.53±0.05	0.51±0.05	0.42±0.06	0.48
TD3	none	0.41±0.06	0.48±0.06	0.51±0.05	0.50±0.05	0.40±0.06	0.46
MBPO	none	0.47±0.05	0.48±0.05	0.57±0.04	0.55±0.05	0.46±0.06	0.52
DreamerV3	none	0.50±0.05	0.57±0.04	0.60±0.04	0.58±0.04	0.49±0.05	0.55
TD-MPC	none	0.49±0.05	0.58±0.04	0.62±0.04	0.60±0.04	0.51±0.05	0.57
SF-GPI-FT	successor features	0.52±0.05	0.55±0.04	0.58±0.04	0.56±0.04	0.47±0.05	0.53
UVFA-FT	universal value encoder/head	0.51±0.05	0.57±0.04	0.60±0.04	0.58±0.04	0.49±0.05	0.55
DBC/Bisimulation-FT	bisimulation encoder/value head	0.53±0.05	0.58±0.04	0.61±0.04	0.59±0.04	0.50±0.05	0.56
Traj2Value-RNN-FT	recurrent trajectory-value encoder	0.55±0.05	0.60±0.04	0.63±0.04	0.61±0.04	0.52±0.05	0.58
Traj2Value-Transformer-FT	transformer trajectory-value encoder	0.57±0.05	0.62±0.04	0.65±0.04	0.63±0.04	0.54±0.05	0.60
Subgraph-GNN-to-Value-FT	graph value encoder	0.56±0.05	0.61±0.04	0.64±0.04	0.62±0.04	0.53±0.05	0.59
DreamerV3-PT+FT	latent world model/value	0.58±0.04	0.63±0.04	0.60±0.04	0.64±0.04	0.55±0.05	0.61
TD-MPC-PT+FT	latent dynamics/value	0.60±0.04	0.65±0.04	0.68±0.04	0.66±0.04	0.53±0.05	0.63
MSRL w/o library	target-trained matrix value	0.54±0.05	0.60±0.05	0.63±0.04	0.61±0.04	0.52±0.05	0.58
MSRL w/o obstruction	target-trained matrix value	0.57±0.05	0.62±0.04	0.65±0.04	0.63±0.04	0.54±0.05	0.60
MSRL from scratch	target-trained matrix value	0.61±0.04	0.67±0.04	0.70±0.04	0.68±0.04	0.59±0.05	0.65
MSRL + pretrained $F_{\eta_{\mathrm{src}}}$ init	matrix-value branch	**0.70±0.04**	**0.75±0.03**	**0.78±0.03**	**0.76±0.03**	**0.67±0.04**	**0.73**



(a) AntMaze (b) Walker2d (c) Hopper

Figure 2: Part II convergence curves. Each panel reports normalized return versus target-environment steps; solid curves are means over five seeds and shaded bands show across-seed variability. MSRL transfers only the source-pretrained $F_{\eta_{\mathrm{src}}}$ initialization and then fine-tunes on target data. Together with Table 1, the curves report both finite-budget adaptation and full-budget behavior under the same interaction budget.

vs. 0.57). Table 1 and Figure 2 answer Q3: source-pretrained matrix-value initialization further raises AUC to 0.73 and accelerates target adaptation, so the gain is matrix-value transfer rather than zero-shot source-policy transfer.

7 Limitations and Conclusion

Limitations. The method is not universal. It requires an aligned transition lift across source and target settings, and $M(\tau)$ preserves only the low-order additive affine-quadratic information induced by that lift. As a commutative sum of one-step outer products, $M(\tau)$ does not encode arbitrary chronological order beyond what is visible in the lifted transition vectors. The matrix-conditioned Bellman interface also depends on sufficiency and smoothness assumptions, so the approximate additive value rule is a local modeling rationale, not an exact dynamic-programming identity for arbitrary matrix sums. Finally, the obstruction certificate is exact only for the true reachable matrix set; in experiments, it is implemented by a learned surrogate filter.

Within these limitations, we proposed a matrix-centered view of compositional generalization in sequential decision making. Trajectory segments become reusable positive semidefinite matrix increments; concatenation and overlap assembly become matrix operations; and source-trained matrix-value knowledge initializes target-side learning under an aligned lift. Experiments support this interface through descriptor diagnostics and finite-budget source-augmented target training.

8 STATEMENTS

AI USE STATEMENT

In this work, we used generative AI tools primarily for language editing, including grammar correction, improving clarity and fluency, and refining the presentation of the manuscript. We also used generative AI tools to assist with running, debugging, and checking experimental code and computational workflows, as well as to help check mathematical derivations and improve the rigor, consistency, and clarity of proof presentation. Generative AI was not used to determine the core research ideas, theoretical results, experimental design, or scientific conclusions of this work. All AI-assisted mathematical arguments, code execution, outputs, and manuscript edits were independently inspected and verified by the authors. The authors take full responsibility for the accuracy, originality, reproducibility, and final content of the manuscript.

REFERENCES

David Abel, David Hershkowitz, and Michael Littman. Near optimal behavior via approximate state abstraction. In *International Conference on Machine Learning*, pp. 2915–2923. PMLR, 2016.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P Van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Andre Barreto, Diana Borsa, John Quan, Tom Schaul, David Silver, Matteo Hessel, Daniel Mankowitz, Augustin Zidek, and Remi Munos. Transfer in deep reinforcement learning using successor features and generalised policy improvement. In *International Conference on Machine Learning*, pp. 501–510. PMLR, 2018.

André Barreto, Diana Borsa, Shaobo Hou, Gheorghe Comanici, Eser Aygün, Philippe Hamel, Daniel Toyama, Shibl Mourad, David Silver, Doina Precup, et al. The option keyboard: Combining skills in reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Richard Bellman. Dynamic programming. *science*, 153(3731):34–37, 1966.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Pablo Samuel Castro. Scalable methods for computing state similarity in deterministic markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 10069–10076, 2020.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In *Uai*, volume 4, pp. 162–169, 2004.

Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pp. 1587–1596. PMLR, 2018.

Thomas Gärtner, Peter Flach, and Stefan Wrobel. On graph kernels: Hardness results and efficient alternatives. In *Learning theory and kernel machines: 16th annual conference on learning theory and 7th kernel workshop, COLT/kernel 2003, washington, DC, USA, August 24-27, 2003. proceedings*, pp. 129–143. Springer, 2003.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *International conference on machine learning*, pp. 2170–2179. PMLR, 2019.

Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in markov decision processes. *Artificial intelligence*, 147(1-2):163–223, 2003.

Alex Graves. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

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.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

Charles Royal Johnson. Positive definite matrices. *The American Mathematical Monthly*, 77(3): 259–264, 1970.

Risi Imre Kondor and John Lafferty. Diffusion kernels on graphs and other discrete structures. In *Proceedings of the 19th international conference on machine learning*, volume 2002, pp. 315–322, 2002.

Alessandro Lazaric. Transfer in reinforcement learning: a framework and a survey. In *Reinforcement Learning: State-of-the-Art*, pp. 143–173. Springer, 2012.

Lihong Li, Thomas J Walsh, and Michael L Littman. Towards a unified theory of state abstraction for mdps. *AI&M*, 1(2):3, 2006.

Marlos C Machado, Clemens Rosenbaum, Xiaoxiao Guo, Miao Liu, Gerald Tesauro, and Murray Campbell. Eigenoption discovery through the deep successor representation. *arXiv preprint arXiv:1710.11089*, 2017.

Sridhar Mahadevan and Mauro Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *Journal of Machine Learning Research*, 8(10), 2007.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. Kernel mean embedding of distributions: A review and beyond. *Foundations and TrendsÂ® in Machine Learning*, 10(1-2):1–141, 2017.

Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A riemannian framework for tensor computing. *International Journal of computer vision*, 66(1):41–66, 2006.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Balaraman Ravindran. Smdp homomorphisms: An algebraic approach to abstraction in semi markov decision processes. 2003.

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

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

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9), 2011.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(7), 2009.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? *arXiv preprint arXiv:2209.14935*, 2022.

Oncel Tuzel, Fatih Porikli, and Peter Meer. Region covariance: A fast descriptor for detection and classification. In *European conference on computer vision*, pp. 589–600. Springer, 2006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

S Vichy N Vishwanathan, Nicol N Schraudolph, Risi Kondor, and Karsten M Borgwardt. Graph kernels. *The Journal of Machine Learning Research*, 11:1201–1242, 2010.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3):279–292, 1992.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

A Matrix Block Structure

This appendix records the block form of the matrix proxy. A trajectory segment is treated as an observable sampler of local compositional structure, and $M(\tau)$ stores a low-order matrix proxy for that sampled structure.

A.1 Step Lift and Block Notation

Write

$$\psi_t = \begin{bmatrix} c_t \\ \delta_t \\ \chi_t^c \\ \chi_t^\delta \\ r_t \\ 1 \end{bmatrix}, \qquad \chi_t^c := \xi(a_t) \otimes c_t, \qquad \chi_t^\delta := \xi(a_t) \otimes \delta_t.$$

Then

$$M(\tau) = \sum_{t=0}^{T-1} \psi_t \psi_t^\top = \begin{bmatrix} M_{cc} & M_{c\delta} & M_{c,\chi^c} & M_{c,\chi^\delta} & M_{cr} & M_{c1} \\ M_{\delta c} & M_{\delta\delta} & M_{\delta,\chi^c} & M_{\delta,\chi^\delta} & M_{\delta r} & M_{\delta 1} \\ M_{\chi^c,c} & M_{\chi^c,\delta} & M_{\chi^c,\chi^c} & M_{\chi^c,\chi^\delta} & M_{\chi^c r} & M_{\chi^c 1} \\ M_{\chi^\delta,c} & M_{\chi^\delta,\delta} & M_{\chi^\delta,\chi^c} & M_{\chi^\delta,\chi^\delta} & M_{\chi^\delta r} & M_{\chi^\delta 1} \\ M_{rc} & M_{r\delta} & M_{r,\chi^c} & M_{r,\chi^\delta} & M_{rr} & M_{r1} \\ M_{1c} & M_{1\delta} & M_{1,\chi^c} & M_{1,\chi^\delta} & M_{1r} & M_{11} \end{bmatrix}.$$

The block labels correspond to local position, directed transport, action-conditioned position, action-conditioned transport, reward, and the constant channel.

A.2 Local Position Moment

The block

$$M_{cc} = \sum_t c_t c_t^\top$$

records the distribution of local transition contexts sampled by the segment. In the graph-matrix interpretation, this block is the node-like second moment of the sampled local structure.

A.3 Directed Transport Energy

The block

$$M_{\delta\delta} = \sum_t \delta_t \delta_t^\top$$

records the anisotropy and magnitude of directed displacement. It distinguishes horizontal, vertical, isotropic, and curved transport patterns at low order.

A.4 Incidence and Direction-Sensitive Coupling

The mixed block

$$M_{c\delta} = \sum_t c_t \delta_t^\top$$

records the dependence between local context and directed transport. This block is direction-sensitive: it changes under edge reversal and detects oriented incidence patterns such as directed paths and cycles. It is not, by itself, a full chronological-order encoding, because the descriptor is a commutative sum over one-step lifted outer products.

A.5 Action-Conditioned Blocks

If ξ is one-hot, then

$$M_{\chi^c,\chi^c} = \sum_t (\xi(a_t) \otimes c_t)(\xi(a_t) \otimes c_t)^\top$$

records the local contexts associated with each action, while

$$M_{\chi^\delta,\chi^\delta} = \sum_t (\xi(a_t) \otimes \delta_t)(\xi(a_t) \otimes \delta_t)^\top$$

records action-conditioned directed transport. The mixed block

$$M_{\chi^c,\chi^\delta} = \sum_t (\xi(a_t) \otimes c_t)(\xi(a_t) \otimes \delta_t)^\top$$

captures action-conditioned incidence.

A.6 Reward and Constant Blocks

The reward-coupling blocks

$$M_{cr}, \quad M_{\delta r}, \quad M_{\chi^c r}, \quad M_{\chi^\delta r}$$

record the alignment of reward with local position, directed transport, and action-conditioned transport. The block

$$M_{rr} = \sum_t r_t^2$$

records second-order reward magnitude. The constant channel gives

$$M_{11} = T, \qquad M_{c1} = \sum_t c_t, \qquad M_{\delta 1} = \sum_t \delta_t,$$

$$M_{\chi^c 1} = \sum_t \chi_t^c, \qquad M_{\chi^\delta 1} = \sum_t \chi_t^\delta, \qquad M_{r1} = \sum_t r_t.$$

Thus segment length, average local bias, net transport, action-conditioned first-order totals, and cumulative reward remain linearly recoverable from the same descriptor.

B Structural Interpretation

This appendix records the intended reading of the structural theorems in Section 3. All statements are relative to the common step lift fixed in Section 2. The descriptor preserves the low-order additive information induced by that lift; it does not preserve every history-level property.

B.1 Gauge Covariance

Theorem 3.1 is a descriptor-level statement. It establishes that $M(\tau)$ is a fixed-dimensional positive semidefinite matrix whenever the lifted quantities are finite. The gauge statement is covariance under a common global change of observable and action coordinates. The intrinsic descriptor is therefore the congruence class of $M(\tau)$, not a particular coordinate realization.

B.2 Low-Order Completeness

Theorem 3.3 is intentionally stated for $\mathfrak{F}_{\mathrm{aq}}$. The class contains exactly the additive first-order and quadratic statistics induced by the fixed lift. Because the lift includes a constant channel, the induced class includes segment length, cumulative reward, and first-order local-structure summaries in addition to second-order moments.

B.3 Segment Composition

Theorem 3.4 states that valid concatenation of observed segments becomes addition in matrix space. A stored segment contributes the increment $M(\sigma)$, and the accumulated state is updated by adding this increment.

B.4 Local-to-Global Gluing

Theorem 3.5 gives an inclusion–exclusion formula for recovering a global descriptor from overlapping local restrictions. The same formula can also form a candidate glued matrix from local descriptors. That candidate is a representation-level object; realizability in a particular environment is a separate question.

B.5 Minimal Sufficiency

Theorem 3.7 is a statement inside the class of fixed-dimensional additive descriptors that preserve all functionals in $\mathfrak{F}_{\mathrm{aq}}$. It does not assert global minimality over arbitrary history representations. It says that, within the stated admissible class, the matrix proxy is already the coarsest sufficient object up to measurable post-processing.

B.6 Realizability Obstruction

Theorem 3.10 separates algebraic validity in matrix space from realizability in the environment. A candidate matrix may be well defined as a sum or an inclusion–exclusion combination while still falling outside the reachable set $\mathcal{R}_e^{(T)}$. Positive obstruction certifies that no actual rollout of the matching horizon in the environment produces that matrix.

B.7 Matrix-Value Approximation

Theorem 4.6 connects the representation to control. Under a matrix-conditioned value law and local smoothness in matrix coordinates, value differences induced by small compositional increments are first-order additive. Corollary 4.7 gives the centered form of the same approximation.

C Symbolic Subgraph Patterns

This appendix gives symbolic examples of what the matrix proxy records for different local subgraph patterns. No numerical coordinates are fixed. Instead, we describe the algebraic identities that the blocks of M must satisfy under reversal symmetry, directed cycles, sequential paths, and action-labeled subgraphs. These examples should be read as structural interpretations of the block formulas in Appendix A, not as additional assumptions used in the main theorems.

C.1 Edge-Indexed Subgraph Descriptor

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a finite directed multigraph sampled by a trajectory segment or by a local subgraph extraction procedure. Each directed edge $e \in \mathcal{E}$ has a tail vertex u_e, a head vertex v_e, an action label a_e, and a reward r_e. Write

$$x_e^- := \phi(u_e), \qquad x_e^+ := \phi(v_e), \qquad \eta_e := \xi(a_e),$$

and define the edge midpoint and directed displacement by

$$c_e := \frac{x_e^- + x_e^+}{2}, \qquad \delta_e := x_e^+ - x_e^-.$$

The edge lift is

$$\psi_e = \begin{bmatrix} c_e \\ \delta_e \\ \eta_e \otimes c_e \\ \eta_e \otimes \delta_e \\ r_e \\ 1 \end{bmatrix}.$$

The symbolic subgraph matrix is

$$M(\mathcal{G}) := \sum_{e \in \mathcal{E}} \psi_e \psi_e^\top.$$

When $\mathcal{G}$ is exactly the directed multiset of transitions in a trajectory segment τ, this definition coincides with $M(\tau)$.

For the rest of this appendix, write

$$\chi_e^c := \eta_e \otimes c_e, \qquad \chi_e^\delta := \eta_e \otimes \delta_e.$$

The blocks of $M(\mathcal{G})$ are therefore

$$M_{cc} = \sum_{e \in \mathcal{E}} c_e c_e^\top, \qquad M_{c\delta} = \sum_{e \in \mathcal{E}} c_e \delta_e^\top, \qquad M_{\delta\delta} = \sum_{e \in \mathcal{E}} \delta_e \delta_e^\top,$$

$$M_{c1} = \sum_{e\in\mathcal{E}} c_e, \qquad M_{\delta 1} = \sum_{e\in\mathcal{E}} \delta_e, \qquad M_{11} = |\mathcal{E}|,$$

$$M_{cr} = \sum_{e\in\mathcal{E}} r_e c_e, \qquad M_{\delta r} = \sum_{e\in\mathcal{E}} r_e \delta_e, \qquad M_{rr} = \sum_{e\in\mathcal{E}} r_e^2, \qquad M_{r1} = \sum_{e\in\mathcal{E}} r_e,$$

and the action-conditioned blocks are

$$M_{\chi^c,\chi^c} = \sum_{e\in\mathcal{E}} (\eta_e \otimes c_e)(\eta_e \otimes c_e)^\top,$$

$$M_{\chi^\delta,\chi^\delta} = \sum_{e\in\mathcal{E}} (\eta_e \otimes \delta_e)(\eta_e \otimes \delta_e)^\top,$$

$$M_{\chi^c,\chi^\delta} = \sum_{e\in\mathcal{E}} (\eta_e \otimes c_e)(\eta_e \otimes \delta_e)^\top.$$

Using the Kronecker identity

$$(p \otimes u)(q \otimes v)^\top = (pq^\top) \otimes (uv^\top),$$

these may also be written as

$$M_{\chi^c,\chi^c} = \sum_{e\in\mathcal{E}} (\eta_e \eta_e^\top) \otimes (c_e c_e^\top),$$

$$M_{\chi^\delta,\chi^\delta} = \sum_{e\in\mathcal{E}} (\eta_e \eta_e^\top) \otimes (\delta_e \delta_e^\top),$$

and

$$M_{\chi^c,\chi^\delta} = \sum_{e\in\mathcal{E}} (\eta_e \eta_e^\top) \otimes (c_e \delta_e^\top).$$

Thus each symbol in the matrix has a direct interpretation as an additive edge statistic.

C.2 Reversal Parity and Symmetric Subgraphs

A directed edge reversal changes the sign of displacement but preserves the midpoint. More precisely, if $\bar{e}$ is the reverse of e, so that

$$u_{\bar{e}} = v_e, \qquad v_{\bar{e}} = u_e,$$

then

$$c_{\bar{e}} = c_e, \qquad \delta_{\bar{e}} = -\delta_e.$$

If, in addition, the reversed edge carries the same action feature and the same reward,

$$\eta_{\bar{e}} = \eta_e, \qquad r_{\bar{e}} = r_e,$$

then

$$\chi^c_{\bar{e}} = \chi^c_e, \qquad \chi^\delta_{\bar{e}} = -\chi^\delta_e.$$

Thus the channels

$$c, \quad \chi^c, \quad r, \quad 1$$

are even under reversal, while the channels

$$\delta, \quad \chi^\delta$$

are odd under reversal.

Lemma C.1 (Block parity under reversal symmetry)**.** *Suppose $\mathcal{G}$ is reversal-symmetric in the following sense: there is a fixed-point-free involution $e \mapsto \bar{e}$ on $\mathcal{E}$ such that*

$$u_{\bar{e}} = v_e, \qquad v_{\bar{e}} = u_e, \qquad \eta_{\bar{e}} = \eta_e, \qquad r_{\bar{e}} = r_e.$$

Then every block coupling an even channel with an odd channel is zero. In particular,

$$M_{c\delta} = 0, \qquad M_{c,\chi^\delta} = 0, \qquad M_{\chi^c,\delta} = 0, \qquad M_{\chi^c,\chi^\delta} = 0,$$

$$M_{r\delta} = 0, \qquad M_{r,\chi^\delta} = 0, \qquad M_{1\delta} = 0, \qquad M_{1,\chi^\delta} = 0,$$

and the transposed blocks are zero as well.

Proof. Choose one representative from each reversal pair and call the representative set $\mathcal{R}$, so that

$$\mathcal{E} = \mathcal{R} \cup \{\bar{e} : e \in \mathcal{R}\}$$

as a disjoint union. Let Y_e be any even channel, meaning one of

$$c_e, \quad \chi^c_e, \quad r_e, \quad 1,$$

and let Z_e be any odd channel, meaning one of

$$\delta_e, \quad \chi^\delta_e.$$

By the reversal assumptions,

$$Y_{\bar{e}} = Y_e, \qquad Z_{\bar{e}} = -Z_e.$$

Therefore the contribution of a reversal pair to the mixed block M_{YZ} is

$$Y_e Z_e^\top + Y_{\bar{e}} Z_{\bar{e}}^\top = Y_e Z_e^\top + Y_e(-Z_e)^\top = Y_e Z_e^\top - Y_e Z_e^\top = 0.$$

Summing over all representative pairs gives $M_{YZ} = 0$. The same argument gives $M_{ZY} = 0$, or equivalently $M_{ZY} = M_{YZ}^\top = 0$. Applying this parity argument to all even–odd channel pairs gives the displayed identities. □

The surviving blocks in a reversal-symmetric subgraph are exactly the even–even and odd–odd blocks. For example,

$$M_{cc} = 2\sum_{e\in\mathcal{R}} c_e c_e^\top, \qquad M_{\delta\delta} = 2\sum_{e\in\mathcal{R}} \delta_e \delta_e^\top,$$
$$M_{\chi^c,\chi^c} = 2\sum_{e\in\mathcal{R}} (\eta_e \otimes c_e)(\eta_e \otimes c_e)^\top,$$
$$M_{\chi^\delta,\chi^\delta} = 2\sum_{e\in\mathcal{R}} (\eta_e \otimes \delta_e)(\eta_e \otimes \delta_e)^\top,$$
$$M_{cr} = 2\sum_{e\in\mathcal{R}} r_e c_e, \qquad M_{rr} = 2\sum_{e\in\mathcal{R}} r_e^2, \qquad M_{11} = 2|\mathcal{R}|.$$

Thus a reversal-symmetric subgraph has no first-order oriented incidence in the even–odd blocks, but it can still have nonzero local position mass, transport energy, action-conditioned transport energy, reward mass, and horizon.

If the reversed edge has a different action feature or a different reward, the full parity cancellation above need not hold for all action- or reward-coupled blocks. The purely geometric cancellation

$$M_{c\delta} = 0, \qquad M_{\delta 1} = 0$$

still follows from $c_{\bar{e}} = c_e$ and $\delta_{\bar{e}} = -\delta_e$, but blocks such as M_{χ^c,χ^δ}, $M_{\chi^\delta 1}$, or $M_{\delta r}$ may remain nonzero unless the corresponding labels also match across reversal pairs.

C.3 Directed Edge Reversal

The previous subsection considered a subgraph containing both directions of every edge. We now compare a directed subgraph with its fully reversed version.

Let $\mathcal{G}^{\mathrm{rev}}$ be obtained from $\mathcal{G}$ by reversing every directed edge while keeping the same action feature and reward attached to the reversed edge. For each edge,

$$c_e^{\mathrm{rev}} = c_e, \qquad \delta_e^{\mathrm{rev}} = -\delta_e, \qquad \eta_e^{\mathrm{rev}} = \eta_e, \qquad r_e^{\mathrm{rev}} = r_e.$$

Therefore the blocks containing an even number of displacement-type channels are unchanged:

$$M_{cc}(\mathcal{G}^{\mathrm{rev}}) = M_{cc}(\mathcal{G}), \qquad M_{\delta\delta}(\mathcal{G}^{\mathrm{rev}}) = M_{\delta\delta}(\mathcal{G}),$$
$$M_{\chi^c,\chi^c}(\mathcal{G}^{\mathrm{rev}}) = M_{\chi^c,\chi^c}(\mathcal{G}), \qquad M_{\chi^\delta,\chi^\delta}(\mathcal{G}^{\mathrm{rev}}) = M_{\chi^\delta,\chi^\delta}(\mathcal{G}),$$
$$M_{cr}(\mathcal{G}^{\mathrm{rev}}) = M_{cr}(\mathcal{G}), \qquad M_{rr}(\mathcal{G}^{\mathrm{rev}}) = M_{rr}(\mathcal{G}), \qquad M_{11}(\mathcal{G}^{\mathrm{rev}}) = M_{11}(\mathcal{G}).$$

The blocks containing exactly one displacement-type channel change sign:

$$M_{c\delta}(\mathcal{G}^{\mathrm{rev}}) = -M_{c\delta}(\mathcal{G}), \qquad M_{\delta 1}(\mathcal{G}^{\mathrm{rev}}) = -M_{\delta 1}(\mathcal{G}),$$
$$M_{\delta r}(\mathcal{G}^{\mathrm{rev}}) = -M_{\delta r}(\mathcal{G}), \qquad M_{\chi^c,\chi^\delta}(\mathcal{G}^{\mathrm{rev}}) = -M_{\chi^c,\chi^\delta}(\mathcal{G}),$$
$$M_{\chi^\delta 1}(\mathcal{G}^{\mathrm{rev}}) = -M_{\chi^\delta 1}(\mathcal{G}), \qquad M_{\chi^\delta r}(\mathcal{G}^{\mathrm{rev}}) = -M_{\chi^\delta r}(\mathcal{G}).$$

Thus M distinguishes a directed subgraph from its edge-reversed version precisely through the odd-parity blocks. This is a direction-sensitive property, not a claim that M stores arbitrary chronological order.

C.4 Sequential Paths

Let P be a directed sequential path

$$v_0 \to v_1 \to \cdots \to v_L.$$

Write

$$x_i := \phi(v_i), \qquad \delta_i := x_i - x_{i-1}, \qquad c_i := \frac{x_{i-1} + x_i}{2}, \qquad 1 \le i \le L.$$

For this path, the geometric blocks satisfy

$$M_{11}(P) = L,$$

$$M_{\delta 1}(P) = \sum_{i=1}^{L} \delta_i = \sum_{i=1}^{L} (x_i - x_{i-1}) = x_L - x_0,$$

and

$$M_{c1}(P) = \sum_{i=1}^{L} c_i = \frac{1}{2} x_0 + \sum_{i=1}^{L-1} x_i + \frac{1}{2} x_L.$$

The local mass and transport energy are

$$M_{cc}(P) = \sum_{i=1}^{L} \frac{(x_{i-1} + x_i)(x_{i-1} + x_i)^\top}{4},$$

$$M_{\delta\delta}(P) = \sum_{i=1}^{L} (x_i - x_{i-1})(x_i - x_{i-1})^\top.$$

The incidence block is

$$M_{c\delta}(P) = \sum_{i=1}^{L} \frac{x_{i-1} + x_i}{2} (x_i - x_{i-1})^\top.$$

Its symmetric part telescopes. Indeed, for each edge,

$$c_i \delta_i^\top + \delta_i c_i^\top = x_i x_i^\top - x_{i-1} x_{i-1}^\top.$$

Therefore

$$M_{c\delta}(P) + M_{\delta c}(P) = \sum_{i=1}^{L} \left(c_i \delta_i^\top + \delta_i c_i^\top \right) = x_L x_L^\top - x_0 x_0^\top.$$

Equivalently,

$$\mathrm{Sym}(M_{c\delta}(P)) := \frac{1}{2} \left(M_{c\delta}(P) + M_{c\delta}(P)^\top \right) = \frac{1}{2} \left(x_L x_L^\top - x_0 x_0^\top \right).$$

The skew-symmetric part records oriented turning or circulation along the path:

$$M_{c\delta}(P) - M_{\delta c}(P) = \sum_{i=1}^{L} \left(c_i \delta_i^\top - \delta_i c_i^\top \right) = \sum_{i=1}^{L} \left(x_{i-1} x_i^\top - x_i x_{i-1}^\top \right).$$

Equivalently,

$$\mathrm{Skew}(M_{c\delta}(P)) := \frac{1}{2} \left(M_{c\delta}(P) - M_{c\delta}(P)^\top \right) = \frac{1}{2} \sum_{i=1}^{L} \left(x_{i-1} x_i^\top - x_i x_{i-1}^\top \right).$$

The reversed path

$$P^{\mathrm{rev}} : v_L \to v_{L-1} \to \cdots \to v_0$$

has the same midpoint sequence as an unordered edge set but all displacements are negated. Therefore

$$M_{cc}(P^{\mathrm{rev}}) = M_{cc}(P), \qquad M_{\delta\delta}(P^{\mathrm{rev}}) = M_{\delta\delta}(P), \qquad M_{11}(P^{\mathrm{rev}}) = M_{11}(P),$$

while

$$M_{c\delta}(P^{\mathrm{rev}}) = -M_{c\delta}(P), \qquad M_{\delta 1}(P^{\mathrm{rev}}) = -M_{\delta 1}(P).$$

Thus M distinguishes a path from its edge-reversed path through the sign of directed incidence and net transport. However, if two trajectories traverse exactly the same directed edges with exactly the same lifted one-step vectors but in a different temporal order, then their matrices are identical, because M is a sum over one-step outer products.

C.5 Directed Cycles

A directed cycle is a sequential path with

$$v_L = v_0, \qquad x_L = x_0.$$

The path identities above immediately give

$$M_{\delta 1}(C) = x_L - x_0 = 0.$$

Moreover,

$$M_{c\delta}(C) + M_{\delta c}(C) = x_L x_L^\top - x_0 x_0^\top = 0.$$

Since $M_{\delta c}(C) = M_{c\delta}(C)^\top$, this is equivalent to

$$M_{c\delta}(C)^\top = -M_{c\delta}(C).$$

Thus, for a closed directed cycle, the geometric incidence block $M_{c\delta}$ is skew-symmetric. Its skew-symmetric component records oriented circulation. If the same cycle is traversed in the reverse direction, then

$$M_{c\delta}(C^{\mathrm{rev}}) = -M_{c\delta}(C),$$

while

$$M_{cc}(C^{\mathrm{rev}}) = M_{cc}(C), \qquad M_{\delta\delta}(C^{\mathrm{rev}}) = M_{\delta\delta}(C), \qquad M_{11}(C^{\mathrm{rev}}) = M_{11}(C).$$

If both orientations of the same cycle are included with matched actions and rewards, then the odd-parity blocks cancel as in Lemma C.1, and in particular

$$M_{c\delta} = 0.$$

The cycle identity $M_{\delta 1} = 0$ is purely geometric. The action-conditioned first-order transport

$$M_{\chi^\delta 1}(C) = \sum_{i=1}^{L} \xi(a_i) \otimes \delta_i$$

need not vanish unless the action features factor out of the sum. For example, if all edges in the cycle have the same action feature η, then

$$M_{\chi^\delta 1}(C) = \eta \otimes \sum_{i=1}^{L} \delta_i = 0.$$

If the action labels vary around the cycle, then $M_{\chi^\delta 1}(C)$ records action-conditioned circulation and may be nonzero even though $M_{\delta 1}(C) = 0$.

C.6 Action-Labeled Subgraphs

Assume in this subsection that ξ is one-hot. For each action a, write e_a for its one-hot vector and define the action-specific edge set

$$\mathcal{E}_a := \{e \in \mathcal{E} : a_e = a\}.$$

Partition the tensorized blocks according to action labels. Then the (a, b)-block of M_{χ^c,χ^c} is

$$[M_{\chi^c,\chi^c}]_{ab} = \sum_{e\in\mathcal{E}} (e_a^\top \xi(a_e))(\xi(a_e)^\top e_b) c_e c_e^\top.$$

Since $\xi(a_e)$ is one-hot, this equals

$$[M_{\chi^c,\chi^c}]_{ab} = \begin{cases} \sum_{e\in\mathcal{E}_a} c_e c_e^\top, & a = b, \\ 0, & a \neq b. \end{cases}$$

Similarly,

$$\left[M_{\chi^\delta,\chi^\delta}\right]_{ab} = \begin{cases} \sum_{e\in\mathcal{E}_a} \delta_e \delta_e^\top, & a = b, \\ 0, & a \neq b, \end{cases}$$

and

$$\big[M_{\chi^c,\chi^\delta}\big]_{ab} = \begin{cases} \sum_{e\in\mathcal{E}_a} c_e\delta_e^\top, & a=b, \\ 0, & a\neq b. \end{cases}$$

The mixed non-action/action blocks decompose as

$$[M_{c,\chi^c}]_a = \sum_{e\in\mathcal{E}_a} c_e c_e^\top, \qquad \big[M_{c,\chi^\delta}\big]_a = \sum_{e\in\mathcal{E}_a} c_e\delta_e^\top,$$

and

$$[M_{\delta,\chi^c}]_a = \sum_{e\in\mathcal{E}_a} \delta_e c_e^\top, \qquad \big[M_{\delta,\chi^\delta}\big]_a = \sum_{e\in\mathcal{E}_a} \delta_e\delta_e^\top.$$

Thus two subgraphs can have the same unconditioned geometric blocks

$$M_{cc}, \qquad M_{c\delta}, \qquad M_{\delta\delta},$$

while having different action-conditioned blocks if the same geometry is assigned to different action labels. This is the sense in which M separates pure directed transport from action-conditioned directed transport.

C.7 What the Symbolic Patterns Show

The preceding identities imply the following structural reading. A reversal-symmetric subgraph has vanishing even–odd blocks, so the matrix records local mass and transport energy but no net oriented incidence. A directed path has

$$M_{\delta 1} = x_L - x_0$$

and

$$M_{c\delta} + M_{\delta c} = x_L x_L^\top - x_0 x_0^\top,$$

so its endpoint displacement and endpoint second-moment difference are linearly visible from the geometric blocks. A directed cycle has

$$M_{\delta 1} = 0, \qquad M_{c\delta} + M_{\delta c} = 0,$$

so its incidence block is skew-symmetric and records oriented circulation. An action-labeled subgraph decomposes the same geometric statistics by action label through the tensorized channels.

These facts explain why the descriptor is naturally interpreted as a low-order additive graph signal of sampled directed substructure. The descriptor is not a complete encoding of chronological order. If two trajectories produce the same multiset of lifted one-step vectors, then

$$\sum_t \psi_t\psi_t^\top$$

is identical for both. The information preserved by M is exactly the additive affine-quadratic information associated with the chosen lift, as formalized by Theorem 3.3.

C.8 One-Dimensional Motion Segment

Consider a one-dimensional segment with $d = 1$, no action channel, and rewards suppressed. A step has midpoint c_t and displacement δ_t, so the descriptor reduces to blocks involving

$$\sum_t c_t^2, \qquad \sum_t c_t\delta_t, \qquad \sum_t \delta_t^2, \qquad \sum_t c_t, \qquad \sum_t \delta_t, \qquad T.$$

The block $\sum_t c_t\delta_t$ records the orientation of transport relative to location.

C.9 Order-Reversal Pair

Two segments may visit the same local contexts with the same displacement magnitudes but traverse them in opposite order. In that case, the local-position block and transport-energy block can be similar, while the incidence block $M_{c\delta}$ changes. This is the basic mechanism by which the descriptor separates order-sensitive sampled structure from unordered occupancy.

C.10 Action-Labeled Segment Pair

If two segments share the same geometric displacement but use different action labels, the tensorized blocks M_{χ^c,χ^c}, $M_{\chi^\delta,\chi^\delta}$, and M_{χ^c,χ^δ} differ. The descriptor therefore separates pure transport from action-conditioned transport.

C.11 Overlapping Segment Assembly

For an interval cover $\{I_k\}_{k=1}^K$, local descriptors over the restrictions $\tau|_{I_k}$ double count overlap regions. Inclusion–exclusion removes the double counting and reconstructs the global matrix exactly for a genuine rollout.

C.12 Unrealizable Candidate Composition

A candidate matrix can be assembled from locally valid pieces while violating a global environment constraint, such as a locked door, one-way passage, or resource requirement. In that case the candidate may be algebraically valid in $\mathbb{S}^m$, while its distance to $\mathcal{R}_e^{(T)}$ remains positive.

D Matrix-Space Reinforcement Learning Details

This appendix gives the implementation-level form of the matrix-space planning interface.

D.1 Lifted Replay State

At time t, the algorithm stores

$$\widetilde{s}_t = (o_t, Z_t), \qquad Z_t = M(\tau_{0:t-1}).$$

After observing (o_t, a_t, r_t, o_{t+1}), the lifted state is updated by

$$Z_{t+1} = Z_t + \psi_t \psi_t^\top.$$

The observation o_t supplies the current local decision context, and Z_t supplies accumulated sampled structure.

D.2 Segment Library

The segment library $\mathcal{L}$ stores local trajectory segments σ, their matrix increments $\Delta(\sigma) = M(\sigma)$, a head action, entry and exit context summaries, and optional environment tags. Library entries can be drawn from replay, from short successful rollouts, or from mined subsequences with high reuse frequency.

D.3 Candidate Future Set

For a lifted state (o, Z) and action a, the candidate set contains segment continuations whose head action matches a and whose entry context is compatible with o:

$$\mathcal{C}(o, Z, a) \subseteq \{(o^{\mathrm{tail}}(\sigma), Z + \Delta(\sigma)) : \sigma \in \mathcal{L},\ a^{\mathrm{head}}(\sigma) = a\}.$$

The set is filtered by the learned obstruction proxy g_ω.

D.4 Obstruction Proxy

The exact obstruction score requires access to the reachable matrix set $\mathcal{R}_e^{(T)}$, which is generally unavailable. The planner therefore uses a learned proxy

$$g_\omega(o, Z, \widehat{o}, \widehat{Z})$$

trained to distinguish realizable rollout continuations from composed candidates that fail environment rollout checks.

Algorithm 1 Matrix-Space Reinforcement Learning

Require: replay buffer $\mathcal{B}$, segment library $\mathcal{L}$, obstruction threshold $\varepsilon_{\mathrm{obs}}$, lookahead weight β

Initialize critic Q_θ, target critic $Q_{\bar{\theta}}$, obstruction proxy g_ω
for each episode **do**
  Observe o_0, set $Z_0 \leftarrow 0$
  for each time step t **do**
    Form a finite action-candidate set $\mathcal{A}_t$: all actions in the discrete case, or candidates proposed by the continuous-control backbone
    for each $a \in \mathcal{A}_t$ **do**
      Build $\mathcal{C}(o_t, Z_t, a)$ from compatible segments in $\mathcal{L}$
      Remove candidates with $g_\omega(o_t, Z_t, \widehat{o}, \widehat{Z}) > \varepsilon_{\mathrm{obs}}$
      Compute $U_\theta(o_t, Z_t, a)$, using 0 if the filtered set is empty
      Compute $S_\theta(o_t, Z_t, a) = Q_\theta(o_t, Z_t, a) + \beta U_\theta(o_t, Z_t, a)$
    end for
    Select and execute a_t using the scores S_θ
    Observe r_t, o_{t+1}, compute ψ_t, and update $Z_{t+1} = Z_t + \psi_t \psi_t^\top$
    Add transition and optional local segments to $\mathcal{B}$ and $\mathcal{L}$
    Update Q_θ and g_ω from replay
  end for
end for

D.5 Training Objective

The lifted critic can be trained with the native temporal-difference target of the chosen backbone on $(o_t, Z_t, a_t, r_t, o_{t+1}, Z_{t+1})$. For discrete actions, a standard target is

$$y_t = r_t + \gamma \max_{a'} Q_{\bar{\theta}}(o_{t+1}, Z_{t+1}, a').$$

For continuous control with target actor $\pi_{\bar{\nu}}$, the corresponding target is

$$y_t = r_t + \gamma Q_{\bar{\theta}}(o_{t+1}, Z_{t+1}, \pi_{\bar{\nu}}(o_{t+1}, Z_{t+1})),$$

with any entropy or other backbone-specific terms retained when applicable. The critic loss is

$$\mathcal{L}_Q(\theta) = \mathbb{E}[(Q_\theta(o_t, Z_t, a_t) - y_t)^2].$$

The continuous-control actor is updated with the native backbone objective using the same matrix-conditioned critic; MSRL introduces no separate policy network. The obstruction proxy can be trained with binary cross entropy or margin losses, depending on the construction of positive and negative composed futures.

D.6 Action Selection

The action score combines a learned one-step critic with an admissible lookahead term:

$$S_\theta(o, Z, a) = Q_\theta(o, Z, a) + \beta U_\theta(o, Z, a).$$

The fallback value $U_\theta = 0$ for empty candidate sets keeps action selection well defined even when the segment library has no admissible continuation for a given action.

D.7 Cross-Environment Transfer

Cross-environment reuse is defined only for environments with a shared aligned lift (ϕ, ξ). The common lift gives matrices of the same size and comparable block semantics. It does not imply transfer across arbitrary unrelated environments; transfer claims require matching assumptions or empirical validation.

D.8 PSEUDOCODE

E EXPERIMENTAL PROTOCOLS AND ADDITIONAL RESULTS

This appendix provides the source-suite construction, target-suite construction, source-augmented target-training protocol, baseline definitions, fairness constraints, and the additional numerical summaries used to support the main experiments.

E.1 COMPACT EXPERIMENTAL LIFT AND TRANSFERRED MODULE

The experiments use a compact instantiation of the trajectory-segment matrix. For each transition, we construct

$$\psi_t^{\mathrm{exp}} = \begin{bmatrix} \bar{x}_t \\ \Delta\bar{x}_t \\ \bar{a}_t \\ \bar{r}_t \\ 1 \end{bmatrix}, \qquad \Delta\bar{x}_t = \bar{x}_{t+1} - \bar{x}_t, \qquad M_{\mathrm{exp}}(\tau) = \sum_t \psi_t^{\mathrm{exp}} (\psi_t^{\mathrm{exp}})^\top.$$

Here $\bar{x}_t$ is an aligned geometric state channel, $\Delta\bar{x}_t$ is a directed displacement channel, $\bar{a}_t$ is a padded or encoded action channel, $\bar{r}_t$ is a normalized reward channel, and the constant channel records segment length and first-order statistics through cross terms. The deterministic experimental adapters are

$$\bar{x}_t = \mathrm{Pad}_8(\mathrm{Norm}_e(\mathrm{Geom}_e(o_t))), \qquad \bar{a}_t = \mathrm{Pad}_{10}(\mathrm{Code}_e(a_t)).$$

Here Geom_e selects named benchmark geometry, Norm_e applies fixed environment-specific normalization, and Code_e is a fixed discrete action code for graph and grid sources or a normalized continuous action vector for continuous-control tasks; unused slots are zero-padded. No cross-environment semantic encoder is learned. This compact lift omits the Kronecker action-position blocks of the full theoretical lift in Section 2; all experimental descriptor claims are relative to ψ_t^{exp}.

For graph and grid environments, Geom_e selects embedded node or cell coordinates. For continuous-control environments, it selects normalized geometric quantities such as position, velocity summaries, goal-relative coordinates, end-effector coordinates, object coordinates, or center-of-mass coordinates. In the reported configuration,

For graph and grid environments, $\bar{x}_t$ is an embedded node or cell coordinate and $\bar{a}_t$ is a discrete action code. For continuous-control environments, $\bar{x}_t$ contains normalized geometric coordinates such as position, velocity summaries, goal-relative coordinates, end-effector coordinates, object coordinates, or center-of-mass coordinates. In the reported configuration,

$$d_x = 8, \qquad d_a = 10, \qquad \dim(\psi_t^{\mathrm{exp}}) = 2d_x + d_a + 2 = 28,$$

so $M_{\mathrm{exp}}(\tau) \in \mathbb{S}_+^{28}$.

The online accumulated matrix state is

$$Z_t = M_{\mathrm{exp}}(\tau_{0:t-1}), \qquad Z_{t+1} = Z_t + \psi_t^{\mathrm{exp}} (\psi_t^{\mathrm{exp}})^\top.$$

The transferable module in our method is the matrix-to-value branch

$$F_\eta(Z) \approx V(Z).$$

In source environments, this branch is trained to predict Monte Carlo returns, fitted value targets, or critic targets from Z. In target environments, the source-pretrained parameter vector η_{src} initializes the target value branch:

$$\eta_{\mathrm{tgt}}^{(0)} = \eta_{\mathrm{src}}.$$

The branch is then further optimized using target-environment data. Target-specific observation adapters, action heads, and policies are not transferred from the source environments.

This protocol is a source-augmented target-training protocol, not a zero-shot control protocol. The source-trained branch is expected to provide a useful inductive bias and initialization, not a complete target-environment solution before target training.

Table 2: Hyperparameters used in the reported MSRL experiments with the compact experimental lift. The matrix state is constructed from the aligned geometric channel, displacement channel, padded action channel, normalized reward, and bias term.

Quantity	Value
Geometric channel dimension d_x	8
Action channel dimension d_a	10
Matrix dimension	28×28
Discount factor γ	0.99
Learning rate	3×10^{-4}
Batch size	128 target updates; 256 source pretraining
Replay size	100,000
Segment-library size	5,000 segments
Segment length used for target-control library	1 transition
Obstruction threshold $\varepsilon_{\mathrm{obs}}$	0.55
Lookahead weight β	0.35
Source pretraining updates	100,000
Target adaptation steps	5,000,000 environment steps
Few-shot target budget	first 25% of target training, i.e., 1,250,000 steps
Number of seeds	5

E.2 Simple source environments

The source environments are deliberately simple and heterogeneous. Their role is twofold: they expose interpretable matrix structure, and they provide source data for pretraining F_η. Only one source family is grid-based.

Table 3: Simple source suite. The suite is heterogeneous: graph, grid, continuous point-mass, and low-dimensional control environments are all included.

Environment	Type	What is varied	Primary diagnostic
GraphMotif	Directed graph MDP	Paths, cycles, branches, reversal pairs, lollipop motifs	Order sensitivity, directed incidence, motif retrieval
MicroGrid	Small grid/maze	Occupancy-equivalent paths, bottlenecks, blocked passages, key-door variants	Gluing, obstruction, local-to-global consistency
PointRooms	Continuous point mass	Rooms, walls, coordinate transforms, continuous corridors	Coordinate alignment and geometric matching
Dubins-Reacher	Low-dimensional control	Heading-dependent motion, planar reaching, action-dependent transport	Action-conditioned dynamics and local transport

Descriptor diagnostics. For each source family, we sample trajectory segments and compute $M(\tau)$. We evaluate whether the descriptor preserves useful structural information before it is used for control. The diagnostics are: order-sensitive retrieval, reversal detection, motif classification, coordinate-aligned matching, composition error, gluing error, and invalid-composition AUROC.

Source-side value targets. For source pretraining, each sampled segment or prefix receives a value target. Depending on the source environment, this target is a Monte Carlo return-to-go, a fitted value target, or a critic target from a source-domain RL agent. We train the matrix-to-value model

$$F_{\eta_{\mathrm{src}}}(Z) \approx V_{\mathrm{src}}(Z)$$

on mixed batches from GraphMotif, MicroGrid, PointRooms, and Dubins-Reacher. Environment identity is not provided to the value branch unless explicitly used in a diagnostic ablation. This encourages the value branch to learn reusable matrix-value regularities rather than memorizing one source environment.

Additional source-side numerical diagnostics are reported in Tables 4 and 5. We do not include extra visualizations in the submitted appendix.

Table 4: Representative descriptor-level diagnostics for the source suite. Each environment reports the diagnostic matched to its design, so scores should be interpreted within each row rather than averaged across heterogeneous tasks. Values are mean±std over seeds.

Environment	Representative diagnostic	Diagnostic score ↑	Additive consistency error ↓
GraphMotif	Directed motif / reversal retrieval	0.939 ± 0.054	$3.08\times10^{-7} \pm 1.37\times10^{-8}$
MicroGrid	Invalid-composition / obstruction detection	0.889 ± 0.035	$2.34\times10^{-5} \pm 4.81\times10^{-6}$
PointRooms	Coordinate-aligned trajectory matching	0.867 ± 0.017	$2.16\times10^{-5} \pm 3.96\times10^{-6}$
Dubins-Reacher	Action-channel local-transport retrieval	0.833 ± 0.088	$1.60\times10^{-5} \pm 2.52\times10^{-6}$

Table 5: Source pretraining results for $F_{\eta_{\mathrm{src}}}$. Values are reported as mean $\pm$ standard deviation over seeds.

Source data	Training value error ↓	Validation value error ↓	Rank correlation ↑
GraphMotif only	0.012 ± 0.004	0.028 ± 0.009	0.987 ± 0.006
MicroGrid only	0.078 ± 0.021	0.174 ± 0.052	0.902 ± 0.031
PointRooms only	0.132 ± 0.034	0.286 ± 0.071	0.854 ± 0.038
Dubins-Reacher only	0.041 ± 0.013	0.091 ± 0.028	0.948 ± 0.019
Mixed source suite	0.062 ± 0.006	0.130 ± 0.043	0.933 ± 0.024

E.3 Complex target environments

The complex target suite evaluates whether source-pretrained matrix-value initialization improves learning in substantially different domains. These targets are not all instances of the same environment class. They differ in state dimension, action dimension, dynamics, and reward structure. The shared component is the aligned matrix lift.

Table 6: Complex target environments. Each family may contain several variants; the main text reports a representative target per family and the appendix reports full per-variant results.

Family	Representative targets	Geometric channel used for $\bar{x}_t$	Metrics
AntMaze	U-maze, medium maze, large maze	Ant center of mass, goal-relative position, velocity summary	Success, return, AUC
Reacher/Pusher	reaching, pushing, obstacle pushing	End-effector coordinates, object coordinates, goal-relative vector	Success, return, AUC
Walker2d	velocity target, goal target, terrain variant	Torso coordinates, velocity, target-relative coordinate	Return, AUC
Hopper	velocity target, terrain variant	Body position, velocity, height/contact summary	Return, AUC
Humanoid	standing, walking, direction target	Center of mass, torso-orientation summary, velocity	Return, AUC

Target training variants. For each target environment, we evaluate MSRL from scratch and MSRL initialized with the source-pretrained $F_{\eta_{\mathrm{src}}}$ branch. The target policy, action head, and observation adapter are always target-specific. The transferred source model initializes only the matrix-value branch. We also evaluate ablations without the segment library, without obstruction filtering, and without source pretraining.

Target metrics. The primary metric is few-shot AUC under a fixed target interaction budget. We also report final normalized return, task success rate when applicable, and steps-to-threshold. Steps-to-threshold measures the number of target interactions required to reach a fixed percentage of the best final return observed in the comparison set. A successful pretraining method should improve early learning and reduce steps-to-threshold without harming final performance.

The available target learning curves are reported in Figure 2. The appendix reports protocol details and tabular summaries rather than additional curves.

E.4 BASELINES

We use two baseline groups. The first group consists of standard from-scratch control baselines. The second group consists of source-initialized transfer baselines that are matched to our transfer setting.

Table 7: From-scratch control baselines. These methods do not use source pretraining.

Method	Role in comparison
PPO	On-policy policy-gradient baseline.
SAC	Entropy-regularized off-policy actor-critic baseline.
TD3	Deterministic off-policy actor-critic baseline for continuous actions.
MBPO	Model-based policy optimization baseline using learned short-horizon dynamics.
DreamerV3	Latent model-based RL baseline.
TD-MPC	Latent dynamics and model-predictive control baseline.

Table 8: Source-initialized transfer baselines. These baselines are the most important comparison group for the transfer claim. Each method uses the same simple source suite and then performs source-augmented target training on the same complex target environments.

Method	Source-side trained object	Target initialization	Target training
SF-GPI-FT	Successor-feature encoder and reward adapter	Initialize successor features or feature encoder	Fine-tune reward adapter, critic, and policy on target data
UVFA-FT	Universal value-function encoder/head	Initialize value encoder and value head	Fine-tune target value function and policy
DBC/Bisimulation-FT	Bisimulation-style encoder and value head	Initialize latent encoder and value head	Fine-tune encoder, critic, and policy with target transitions
Traj2Value-RNN-FT	Recurrent trajectory-to-value encoder	Initialize sequence encoder and value head	Fine-tune sequence value predictor and target policy
Traj2Value-Transformer-FT	Transformer trajectory-to-value encoder	Initialize transformer encoder and value head	Fine-tune sequence value predictor and target policy
Subgraph-GNN-to-Value-FT	Trajectory-induced graph encoder and value head	Initialize graph encoder and value head	Fine-tune graph value predictor and target policy
DreamerV3-PT+FT	Latent world model, reward, and value modules	Initialize latent model and value modules	Fine-tune world model and policy on target data
TD-MPC-PT+FT	Latent dynamics and terminal value model	Initialize latent dynamics/value modules	Fine-tune dynamics, value, and MPC policy on target data

For fairness, every pretrain-to-finetune baseline uses the same source environments, source trajectories, source value targets, source pretraining budget, target environments, target seeds, and target interaction budget. When parameter counts are not identical, we report them and keep the transferred value or representation modules in the same scale range.

E.5 PRETRAINING AND FINE-TUNING PROTOCOL

This subsection defines the core transfer experiment.

Step 1: collect source data. Collect trajectories from the simple source suite: GraphMotif, MicroGrid, PointRooms, and Dubins-Reacher. For each trajectory prefix or segment, compute the accumulated matrix state Z and a value target. The same source trajectories and value targets are used for MSRL and all pretrain-to-finetune baselines.

Step 2: pretrain source modules. For MSRL, train

$$F_{\eta_{\mathrm{src}}}(Z) \approx V_{\mathrm{src}}(Z)$$

on mixed source batches. For baseline methods, train their corresponding transferable modules on the same source data: successor features for SF-GPI-FT, a universal value encoder for UVFA-FT, a bisimulation encoder for DBC/Bisimulation-FT, trajectory encoders for Traj2Value methods, a graph encoder for Subgraph-GNN-to-Value-FT, and latent dynamics/value modules for DreamerV3-PT+FT and TD-MPC-PT+FT.

Step 3: initialize target models. For each complex target environment, initialize the relevant target-side module using the source-pretrained parameters. For MSRL,

$$\eta_{\mathrm{tgt}}^{(0)} = \eta_{\mathrm{src}}.$$

The target-specific observation adapter, action head, and policy are initialized normally unless otherwise stated. This avoids transferring source actions into incompatible target action spaces.

Step 4: fine-tune on target data. Train each method on the target environment under the same interaction budget. MSRL fine-tunes the matrix-value branch together with the target critic and policy. A conservative variant may use a source-anchored regularizer,

$$\mathcal{L}_{\mathrm{target}} = \mathcal{L}_{\mathrm{RL}} + \lambda_{\mathrm{src}} \|\eta - \eta_{\mathrm{src}}\|_2^2,$$

but the default setting simply initializes from η_{src} and then trains with the target RL objective. We report both if the regularized version is used.

Step 5: evaluate transfer gain. The main transfer metric is the improvement from source initialization:

$$\Delta_{\mathrm{AUC}} = \mathrm{AUC}(\text{pretrained initialization}) - \mathrm{AUC}(\text{matched from scratch}).$$

For MSRL:

$$\Delta_{\mathrm{AUC}}^{\mathrm{MSRL}} = \mathrm{AUC}(\mathrm{MSRL} + F_{\eta_{\mathrm{src}}}\ \mathrm{init}) - \mathrm{AUC}(\mathrm{MSRL\ from\ scratch}).$$

We also report final performance, success rate, and steps-to-threshold. The key hypothesis is that source-pretrained matrix-value initialization improves low-data target learning more reliably than raw trajectory, graph, successor-feature, bisimulation, or latent-model pretraining.

The aggregate AUC comparison is reported in Table 1, and the matched MSRL source-initialization gain is summarized in Table 9. Negative controls are reported in Table 10.

Table 9: Matched MSRL source-initialization gain. Gains are computed as the normalized few-shot AUC of MSRL initialized with the source-pretrained matrix-value branch minus MSRL trained from scratch under the same target interaction budget.

Comparison	AntMaze	Reacher/Pusher	Walker2d	Hopper	Humanoid	Avg. Δ_{AUC}
MSRL pretrained init – MSRL from scratch	0.09	0.08	0.08	0.08	0.08	0.08

We report a matched transfer gain only for MSRL because MSRL from scratch and MSRL with $F_{\eta_{\mathrm{src}}}$ initialization share the same architecture and target-training protocol. For source-initialized external baselines, Table 1 reports absolute normalized few-shot AUC rather than cross-method gain values, since a uniquely matched from-scratch counterpart is not defined for every pretraining mechanism.

E.6 Negative controls and ablations

The following controls test whether the observed gain comes from meaningful matrix-value pretraining rather than from extra training compute.

Random initialization control. Replace $F_{\eta_{\mathrm{src}}}$ with a randomly initialized branch of the same architecture and then fine-tune on the target environment.

Shuffled source-value control. Pretrain F_η on source matrices paired with shuffled source value targets. This keeps the same source compute and architecture but destroys the matrix-value relation.

Unaligned lift control. Pretrain F_η using a source lift whose coordinate alignment, padding, or channel order is intentionally mismatched with the target lift. This tests whether consistent matrix geometry is necessary.

Observation-only pretraining control. Pretrain an observation-value head on the source suite and use it to initialize a target value head without the matrix branch. This tests whether the improvement comes specifically from matrix-value pretraining.

Frozen-branch diagnostic. As a diagnostic only, we may freeze $F_{\eta_{\mathrm{src}}}$ during target training and train only the target policy and adapters. This is not the main transfer setting. It tests whether full fine-tuning is necessary and should not be interpreted as a zero-shot control result.

MSRL architectural ablations. We evaluate MSRL without the segment library, without obstruction filtering, without action-conditioned matrix blocks, without reward coupling, and without source-pretrained initialization.

The available negative-control results are summarized in Table 10.

Table 10: Negative controls for source initialization. Entries report normalized few-shot AUC under the same target interaction budget.

Variant	AntMaze	Reacher/Pusher	Walker2d	Hopper	Humanoid	Avg.
MSRL from scratch	0.61	0.67	0.70	0.68	0.59	0.65
MSRL + random F_η init	0.60	0.66	0.69	0.67	0.58	0.64
MSRL + shuffled-source-value F_η init	0.56	0.62	0.64	0.62	0.53	0.59
MSRL + unaligned-lift F_η init	0.55	0.60	0.63	0.61	0.52	0.58
MSRL + observation-only pretraining	0.58	0.64	0.67	0.65	0.56	0.62
MSRL + pretrained $F_{\eta_{\mathrm{src}}}$ init	**0.70**	**0.75**	**0.78**	**0.76**	**0.67**	**0.73**

E.7 Fairness checklist

Table 11: Transfer fairness checklist used for the reported source-to-target comparisons.

Controlled item	Reported control
Source environments	Same GraphMotif, MicroGrid, PointRooms, and Dubins-Reacher source suite for all pretraining methods.
Source trajectories	Same source trajectories and trajectory prefixes for all pretraining methods.
Source targets	Same return-to-go or fitted value targets for all methods that learn value predictors.
Source compute	Same number of pretraining updates or a clearly reported compute-normalized budget.
Target environments	Same complex target variants and same random seeds.
Target interaction budget	Same target-training interaction budget for all methods.
Target evaluation	Same evaluation frequency, number of evaluation episodes, and normalization scheme.
Architecture scale	Comparable parameter counts for transferable value or representation modules when possible.
Transferred component	Clearly state which component is initialized from source pretraining and which components remain target-specific.

F PROOFS

F.1 PROOF OF THEOREM 3.1

Proof. Fix a finite trajectory segment

$$\tau = ((o_t, a_t, r_t, o_{t+1}))_{t=0}^{T-1}.$$

The segment is finite by definition, and the theorem assumes that all lifted quantities appearing in Definition 2.1 are finite. Hence, for every $t \in \{0, \dots, T-1\}$, the vectors

$$x_t = \phi(o_t), \qquad x_t^+ = \phi(o_{t+1}), \qquad c_t = \frac{x_t + x_t^+}{2}, \qquad \delta_t = x_t^+ - x_t$$

are finite vectors in $\mathbb{R}^d$, the action feature $\xi(a_t)$ is a finite vector in $\mathbb{R}^q$, and the reward r_t is a finite real number. The Kronecker products $\xi(a_t) \otimes c_t$ and $\xi(a_t) \otimes \delta_t$ therefore have finite entries in $\mathbb{R}^{qd}$. Consequently

$$\psi_t = \begin{bmatrix} c_t \\ \delta_t \\ \xi(a_t) \otimes c_t \\ \xi(a_t) \otimes \delta_t \\ r_t \\ 1 \end{bmatrix} \in \mathbb{R}^{2d+2qd+2} = \mathbb{R}^m$$

is a well-defined finite-dimensional real vector for each t.

For each t, the outer product $\psi_t \psi_t^\top$ is an $m \times m$ real matrix whose (i,j)-entry is $(\psi_t)_i (\psi_t)_j$. Since the entries of ψ_t are finite, every entry of $\psi_t \psi_t^\top$ is finite. Because T is finite, the matrix

$$M(\tau) = \sum_{t=0}^{T-1} \psi_t \psi_t^\top$$

is a finite sum of finite real matrices. Thus $M(\tau)$ is well defined as an element of $\mathbb{R}^{m \times m}$. Moreover, for every t,

$$(\psi_t \psi_t^\top)^\top = (\psi_t^\top)^\top \psi_t^\top = \psi_t \psi_t^\top,$$

so each summand is symmetric. A finite sum of symmetric matrices is symmetric, and hence

$$M(\tau)^\top = \left(\sum_{t=0}^{T-1} \psi_t \psi_t^\top \right)^\top = \sum_{t=0}^{T-1} (\psi_t \psi_t^\top)^\top = \sum_{t=0}^{T-1} \psi_t \psi_t^\top = M(\tau).$$

Therefore $M(\tau) \in \mathbb{S}^m$.

It remains to prove positive semidefiniteness. Let $v \in \mathbb{R}^m$ be arbitrary. Then

$$v^\top M(\tau) v = v^\top \left(\sum_{t=0}^{T-1} \psi_t \psi_t^\top \right) v = \sum_{t=0}^{T-1} v^\top \psi_t \psi_t^\top v.$$

Since $v^\top \psi_t$ is a scalar and $\psi_t^\top v = v^\top \psi_t$, each summand satisfies

$$v^\top \psi_t \psi_t^\top v = (v^\top \psi_t)(\psi_t^\top v) = (v^\top \psi_t)^2 \geq 0.$$

Thus

$$v^\top M(\tau) v = \sum_{t=0}^{T-1} (v^\top \psi_t)^2 \geq 0.$$

Because v was arbitrary, $M(\tau)$ is positive semidefinite. If $T = 0$, the same argument is understood with the empty sum

$$M(\tau) = 0,$$

and the zero matrix is symmetric and positive semidefinite, so the conclusion also holds for the empty segment.

We now prove the covariance statement. Suppose

$$\widetilde{\phi}(o) = S\phi(o), \qquad \widetilde{\xi}(a) = U\xi(a),$$

where $S \in GL(d)$ and $U \in GL(q)$. For the transformed observation features one has

$$\widetilde{x}_t = \widetilde{\phi}(o_t) = S\phi(o_t) = Sx_t, \qquad \widetilde{x}_t^+ = \widetilde{\phi}(o_{t+1}) = S\phi(o_{t+1}) = Sx_t^+.$$

Therefore the transformed midpoint is

$$\widetilde{c}_t = \frac{\widetilde{x}_t + \widetilde{x}_t^+}{2} = \frac{Sx_t + Sx_t^+}{2} = S\frac{x_t + x_t^+}{2} = Sc_t,$$

and the transformed displacement is

$$\widetilde{\delta}_t = \widetilde{x}_t^+ - \widetilde{x}_t = Sx_t^+ - Sx_t = S(x_t^+ - x_t) = S\delta_t.$$

For the action-conditioned blocks, the elementary Kronecker identity

$$(Au) \otimes (Bv) = (A \otimes B)(u \otimes v)$$

gives

$$\widetilde{\xi}(a_t) \otimes \widetilde{c}_t = (U\xi(a_t)) \otimes (Sc_t) = (U \otimes S)(\xi(a_t) \otimes c_t),$$

and similarly

$$\widetilde{\xi}(a_t) \otimes \widetilde{\delta}_t = (U\xi(a_t)) \otimes (S\delta_t) = (U \otimes S)(\xi(a_t) \otimes \delta_t).$$

The reward coordinate and the constant coordinate are not changed by this gauge transformation. Hence, with

$$H := \mathrm{BlkDiag}(S, S, U \otimes S, U \otimes S, 1, 1),$$

the transformed lift satisfies

$$\widetilde{\psi}_t = H\psi_t$$

for every t. Consequently,

$$\widetilde{M}(\tau) = \sum_{t=0}^{T-1} \widetilde{\psi}_t \widetilde{\psi}_t^\top = \sum_{t=0}^{T-1} (H\psi_t)(H\psi_t)^\top.$$

For each summand,

$$(H\psi_t)(H\psi_t)^\top = H\psi_t\psi_t^\top H^\top,$$

and therefore

$$\widetilde{M}(\tau) = \sum_{t=0}^{T-1} H\psi_t\psi_t^\top H^\top = H\left(\sum_{t=0}^{T-1} \psi_t\psi_t^\top\right) H^\top = HM(\tau)H^\top.$$

Since S and U are invertible, the Kronecker product $U \otimes S$ is invertible with inverse $U^{-1} \otimes S^{-1}$. Thus H is invertible, with inverse

$$H^{-1} = \mathrm{BlkDiag}(S^{-1}, S^{-1}, U^{-1} \otimes S^{-1}, U^{-1} \otimes S^{-1}, 1, 1).$$

Hence coordinate changes in the observable and action lifts act on $M(\tau)$ by invertible congruence. The coordinate-free content of the construction is therefore the congruence class generated by these allowed gauge transformations, rather than a particular matrix representative. □

F.2 Proof of Theorem 3.3

Proof. Throughout the proof, the Frobenius inner product on matrices is

$$\langle A, B \rangle := \mathrm{tr}(A^\top B).$$

For vectors $u, v \in \mathbb{R}^m$ and a matrix $A \in \mathbb{R}^{m \times m}$, we shall repeatedly use the identity

$$\langle A, uv^\top \rangle = \mathrm{tr}(A^\top uv^\top) = \mathrm{tr}(v^\top A^\top u) = v^\top A^\top u.$$

In particular, if $A = A^\top$ and $u = v = \psi_t$, then

$$\langle A, \psi_t\psi_t^\top \rangle = \psi_t^\top A\psi_t.$$

Let $F \in \mathfrak{F}_{\mathrm{aq}}$. By Definition 3.2, there exist coefficients

$$a \in \mathbb{R}, \qquad b \in \mathbb{R}^{m-1}, \qquad C \in \mathbb{S}^{m-1}$$

such that, for every finite segment τ,

$$F(\tau) = \sum_{t=0}^{T-1} \left(a + b^\top \bar{\psi}_t + \bar{\psi}_t^\top C \bar{\psi}_t\right), \qquad \psi_t = \begin{bmatrix} \bar{\psi}_t \\ 1 \end{bmatrix}.$$

Define the symmetric block matrix

$$A := \begin{bmatrix} C & \frac{1}{2}b \\ \frac{1}{2}b^\top & a \end{bmatrix} \in \mathbb{S}^m.$$

For each time index t, direct block multiplication gives

$$A\psi_t = \begin{bmatrix} C & \frac{1}{2}b \\ \frac{1}{2}b^\top & a \end{bmatrix} \begin{bmatrix} \bar{\psi}_t \\ 1 \end{bmatrix} = \begin{bmatrix} C\bar{\psi}_t + \frac{1}{2}b \\ \frac{1}{2}b^\top \bar{\psi}_t + a \end{bmatrix}.$$

Therefore

$$\psi_t^\top A \psi_t = \begin{bmatrix} \bar{\psi}_t^\top & 1 \end{bmatrix} \begin{bmatrix} C\bar{\psi}_t + \frac{1}{2}b \\ \frac{1}{2}b^\top \bar{\psi}_t + a \end{bmatrix} = \bar{\psi}_t^\top C \bar{\psi}_t + \frac{1}{2}\bar{\psi}_t^\top b + \frac{1}{2} b^\top \bar{\psi}_t + a.$$

Since $\bar{\psi}_t^\top b = b^\top \bar{\psi}_t$, the two linear half-terms combine:

$$\psi_t^\top A \psi_t = \bar{\psi}_t^\top C \bar{\psi}_t + b^\top \bar{\psi}_t + a.$$

Using the identity $\langle A, \psi_t \psi_t^\top \rangle = \psi_t^\top A \psi_t$, we obtain

$$\langle A, \psi_t \psi_t^\top \rangle = a + b^\top \bar{\psi}_t + \bar{\psi}_t^\top C \bar{\psi}_t.$$

Summing this equality over t yields

$$\begin{aligned} \langle A, M(\tau) \rangle &= \left\langle A, \sum_{t=0}^{T-1} \psi_t \psi_t^\top \right\rangle \\ &= \sum_{t=0}^{T-1} \langle A, \psi_t \psi_t^\top \rangle \\ &= \sum_{t=0}^{T-1} \left(a + b^\top \bar{\psi}_t + \bar{\psi}_t^\top C \bar{\psi}_t\right) \\ &= F(\tau). \end{aligned}$$

Thus every functional in $\mathfrak{F}_{\mathrm{aq}}$ is represented by a linear functional of $M(\tau)$.

Conversely, let $A \in \mathbb{S}^m$ be arbitrary. Partition A according to the decomposition $\mathbb{R}^m = \mathbb{R}^{m-1} \oplus \mathbb{R}$:

$$A = \begin{bmatrix} C & u \\ u^\top & \alpha \end{bmatrix}, \qquad C \in \mathbb{R}^{(m-1)\times(m-1)}, \quad u \in \mathbb{R}^{m-1}, \quad \alpha \in \mathbb{R}.$$

Since A is symmetric, its upper-left block satisfies $C = C^\top$, so $C \in \mathbb{S}^{m-1}$. For each t,

$$A\psi_t = \begin{bmatrix} C & u \\ u^\top & \alpha \end{bmatrix} \begin{bmatrix} \bar{\psi}_t \\ 1 \end{bmatrix} = \begin{bmatrix} C\bar{\psi}_t + u \\ u^\top \bar{\psi}_t + \alpha \end{bmatrix},$$

and hence

$$\psi_t^\top A \psi_t = \begin{bmatrix} \bar{\psi}_t^\top & 1 \end{bmatrix} \begin{bmatrix} C\bar{\psi}_t + u \\ u^\top \bar{\psi}_t + \alpha \end{bmatrix} = \bar{\psi}_t^\top C \bar{\psi}_t + \bar{\psi}_t^\top u + u^\top \bar{\psi}_t + \alpha.$$

Because $\bar{\psi}_t^\top u = u^\top \bar{\psi}_t$, this becomes

$$\psi_t^\top A \psi_t = \bar{\psi}_t^\top C \bar{\psi}_t + 2u^\top \bar{\psi}_t + \alpha.$$

Therefore

$$\begin{aligned}\langle A, M(\tau)\rangle &= \sum_{t=0}^{T-1} \langle A, \psi_t \psi_t^\top \rangle \\ &= \sum_{t=0}^{T-1} \psi_t^\top A \psi_t \\ &= \sum_{t=0}^{T-1} \left(\alpha + 2u^\top \bar{\psi}_t + \bar{\psi}_t^\top C \bar{\psi}_t\right).\end{aligned}$$

This is a member of $\mathfrak{F}_{\mathrm{aq}}$, with coefficients

$$a = \alpha, \qquad b = 2u, \qquad C = C.$$

Thus every symmetric matrix A defines a functional in the low-order additive class through the formula $F_A(\tau) = \langle A, M(\tau)\rangle$.

It remains to prove the equivalence statement. Suppose first that $M(\tau) = M(\tau')$. Let $F \in \mathfrak{F}_{\mathrm{aq}}$ be arbitrary. By the first part of the proof, there exists $A \in \mathbb{S}^m$ such that

$$F(\eta) = \langle A, M(\eta)\rangle$$

for every finite segment η. Applying this to τ and τ', and using $M(\tau) = M(\tau')$, gives

$$F(\tau) = \langle A, M(\tau)\rangle = \langle A, M(\tau')\rangle = F(\tau').$$

Hence equality of matrix proxies implies equality of all functionals in $\mathfrak{F}_{\mathrm{aq}}$.

Conversely, suppose

$$F(\tau) = F(\tau') \qquad \text{for every } F \in \mathfrak{F}_{\mathrm{aq}}.$$

The converse representation already proved implies that, for every $A \in \mathbb{S}^m$, the functional

$$F_A(\eta) := \langle A, M(\eta)\rangle$$

belongs to $\mathfrak{F}_{\mathrm{aq}}$. Applying the assumed equality to F_A gives

$$\langle A, M(\tau)\rangle = \langle A, M(\tau')\rangle \qquad \text{for every } A \in \mathbb{S}^m.$$

Equivalently,

$$\langle A, M(\tau) - M(\tau')\rangle = 0 \qquad \text{for every } A \in \mathbb{S}^m.$$

By Theorem 3.1, both $M(\tau)$ and $M(\tau')$ are symmetric, so the difference

$$\Delta := M(\tau) - M(\tau')$$

also belongs to $\mathbb{S}^m$. Choosing $A = \Delta$ in the preceding identity gives

$$0 = \langle \Delta, \Delta\rangle = \mathrm{tr}(\Delta^\top \Delta) = \|\Delta\|_F^2.$$

A Frobenius norm is zero if and only if the matrix is zero. Hence $\Delta = 0$, and therefore

$$M(\tau) = M(\tau').$$

This proves both directions of the equivalence and shows that $M(\tau)$ captures exactly the additive affine-quadratic information induced by the fixed lift. □

F.3 Proof of Theorem 3.4

Proof. Let

$$\tau^{(1)} = ((o_t^{(1)}, a_t^{(1)}, r_t^{(1)}, o_{t+1}^{(1)}))_{t=0}^{T_1-1}$$

and

$$\tau^{(2)} = ((o_s^{(2)}, a_s^{(2)}, r_s^{(2)}, o_{s+1}^{(2)}))_{s=0}^{T_2-1}$$

be finite trajectory segments. Suppose

$$\tau = \tau^{(1)} \star \tau^{(2)}$$

is a valid concatenation. Validity means that the terminal observation of the first segment matches the initial observation of the second segment, so the raw transitions can be placed one after another without inserting any additional transition at the boundary. The concatenated segment has length $T_1 + T_2$.

Let $\psi_t^{(1)}$ be the lifted vector associated with the t-th transition of $\tau^{(1)}$, and let $\psi_s^{(2)}$ be the lifted vector associated with the s-th transition of $\tau^{(2)}$. Since the fixed lift (ϕ, ξ) is applied transition by transition, the lifted vectors of the concatenated trajectory are precisely

$$\psi_t = \psi_t^{(1)} \qquad \text{for } 0 \le t \le T_1 - 1,$$

and

$$\psi_{T_1+s} = \psi_s^{(2)} \qquad \text{for } 0 \le s \le T_2 - 1.$$

No extra outer product appears at the joining point, because the descriptor sums over actual transitions and the concatenation operation does not create a new transition between the two pieces.

By Definition 2.1,

$$M(\tau) = \sum_{t=0}^{T_1+T_2-1} \psi_t \psi_t^\top.$$

Splitting the finite sum at T_1 gives

$$M(\tau) = \sum_{t=0}^{T_1-1} \psi_t \psi_t^\top + \sum_{t=T_1}^{T_1+T_2-1} \psi_t \psi_t^\top.$$

Using the identities for the lifted vectors on the first and second parts, the first sum becomes

$$\sum_{t=0}^{T_1-1} \psi_t \psi_t^\top = \sum_{t=0}^{T_1-1} \psi_t^{(1)} (\psi_t^{(1)})^\top = M(\tau^{(1)}),$$

while the second sum, after the change of index $t = T_1 + s$, becomes

$$\sum_{t=T_1}^{T_1+T_2-1} \psi_t \psi_t^\top = \sum_{s=0}^{T_2-1} \psi_{T_1+s} \psi_{T_1+s}^\top = \sum_{s=0}^{T_2-1} \psi_s^{(2)} (\psi_s^{(2)})^\top = M(\tau^{(2)}).$$

Combining these two identities yields

$$M(\tau) = M(\tau^{(1)}) + M(\tau^{(2)}).$$

Thus valid concatenation of trajectory segments is represented exactly as addition in the common matrix space. □

F.4 Proof of Theorem 3.5

Proof. Let $\tau = ((o_t, a_t, r_t, o_{t+1}))_{t=0}^{T-1}$ be fixed, and let ψ_t denote the lifted vector associated with its t-th transition. For an interval $I \subseteq \{0, \dots, T-1\}$, the restricted trajectory $\tau|_I$ consists exactly of those transitions whose indices lie in I. Hence its matrix descriptor is

$$M(\tau|_I) = \sum_{t \in I} \psi_t \psi_t^\top.$$

This identity is simply the definition of M applied to the restricted segment; the same ambient lift is used, so no change of dimension or coordinates occurs when passing from τ to $\tau|_I$.

Let $\{I_k\}_{k=1}^K$ be an interval cover of $\{0, \dots, T-1\}$. For each nonempty $S \subseteq \{1, \dots, K\}$, define

$$I_S = \bigcap_{k \in S} I_k.$$

The theorem restricts attention to the index family

$$\mathcal{N} = \{S \subseteq \{1, \dots, K\} : S \neq \varnothing,\ I_S \neq \varnothing\}.$$

For $S \in \mathcal{N}$, the descriptor of the overlap restriction is

$$M(\tau_S) = M(\tau|_{I_S}) = \sum_{t \in I_S} \psi_t \psi_t^\top.$$

Therefore the inclusion–exclusion expression equals

$$\sum_{S \in \mathcal{N}} (-1)^{|S|+1} M(\tau_S) = \sum_{S \in \mathcal{N}} (-1)^{|S|+1} \sum_{t \in I_S} \psi_t \psi_t^\top.$$

All sums are finite, so the order of summation may be interchanged. This gives

$$\sum_{S \in \mathcal{N}} (-1)^{|S|+1} M(\tau_S) = \sum_{t=0}^{T-1} \left(\sum_{\substack{S \in \mathcal{N} \\ t \in I_S}} (-1)^{|S|+1} \right) \psi_t \psi_t^\top.$$

It remains to compute the scalar coefficient multiplying each fixed transition contribution $\psi_t \psi_t^\top$.

Fix $t \in \{0, \dots, T-1\}$. Since $\{I_k\}_{k=1}^K$ covers $\{0, \dots, T-1\}$, the set

$$K(t) := \{k \in \{1, \dots, K\} : t \in I_k\}$$

is nonempty. For any nonempty $S \subseteq \{1, \dots, K\}$, we have

$$t \in I_S \quad \iff \quad t \in \bigcap_{k \in S} I_k \quad \iff \quad t \in I_k \text{ for every } k \in S \quad \iff \quad S \subseteq K(t).$$

Moreover, whenever $S \subseteq K(t)$ and $S \neq \varnothing$, the intersection I_S is nonempty because it contains t. Hence such an S automatically belongs to $\mathcal{N}$. Thus

$$\{S \in \mathcal{N} : t \in I_S\} = \{S : S \neq \varnothing,\ S \subseteq K(t)\}.$$

The coefficient of $\psi_t \psi_t^\top$ is therefore

$$\sum_{\substack{S \in \mathcal{N} \\ t \in I_S}} (-1)^{|S|+1} = \sum_{\varnothing \neq S \subseteq K(t)} (-1)^{|S|+1}.$$

Because $K(t)$ is nonempty, the binomial identity gives

$$\sum_{S \subseteq K(t)} (-1)^{|S|} = (1-1)^{|K(t)|} = 0.$$

Separating the empty subset term from this sum yields

$$1 + \sum_{\varnothing \neq S \subseteq K(t)} (-1)^{|S|} = 0.$$

Multiplying by -1, we obtain

$$\sum_{\varnothing \neq S \subseteq K(t)} (-1)^{|S|+1} = 1.$$

Thus every transition contribution $\psi_t \psi_t^\top$ appears with coefficient exactly one in the inclusion–exclusion reconstruction. Substituting this coefficient back into the reordered sum gives

$$\sum_{S \in \mathcal{N}} (-1)^{|S|+1} M(\tau_S) = \sum_{t=0}^{T-1} \psi_t \psi_t^\top = M(\tau).$$

This proves exact local-to-global recovery for an actual global trajectory.

For the more general representation-level statement, suppose an arbitrary family

$$\{M_S\}_{S \in \mathcal{N}} \subseteq \mathbb{S}^m$$

is given. Since $\mathbb{S}^m$ is a real vector space, every finite linear combination of elements of $\mathbb{S}^m$ is again an element of $\mathbb{S}^m$. Therefore

$$\widehat{M} := \sum_{S \in \mathcal{N}} (-1)^{|S|+1} M_S$$

is a well-defined element of $\mathbb{S}^m$. Its uniqueness follows from the fact that finite sums in a vector space are unique: once the family $\{M_S\}_{S \in \mathcal{N}}$ and the coefficients $(-1)^{|S|+1}$ are fixed, there is exactly one matrix equal to that linear combination. This establishes the candidate glued matrix at the representation level. The argument does not imply that $\widehat{M}$ is positive semidefinite, that its constant channel is a valid horizon, or that it is reachable by an environment trajectory; those are separate realizability properties handled by the obstruction theorem. □

F.5 Proof of Theorem 3.7

Proof. We use the factorization meaning of admissible sufficiency in Definition 3.6. Thus, if D is admissible, then for every $F \in \mathfrak{F}_{\mathrm{aq}}$ there exists a Borel measurable map

$$h_F : \mathbb{R}^p \to \mathbb{R}$$

such that

$$F(\tau) = h_F(D(\tau)) \qquad \text{for every finite trajectory segment } \tau.$$

This measurable factorization assumption is the point that allows one to construct the measurable post-processing map ρ.

Let

$$s := \dim(\mathbb{S}^m) = \frac{m(m+1)}{2}.$$

Equip $\mathbb{S}^m$ with the Frobenius inner product

$$\langle A, B\rangle = \mathrm{tr}(A^\top B).$$

Since $\mathbb{S}^m$ is a finite-dimensional real Hilbert space under this inner product, choose an orthonormal basis

$$B_1, \dots, B_s$$

of $\mathbb{S}^m$. Thus every $N \in \mathbb{S}^m$ has the unique expansion

$$N = \sum_{j=1}^{s} \langle B_j, N\rangle B_j,$$

and the coordinates $\langle B_j, N\rangle$ determine N uniquely.

For each $j \in \{1, \dots, s\}$, define a trajectory functional

$$F_j(\tau) := \langle B_j, M(\tau)\rangle.$$

By Theorem 3.3, every functional of the form $\tau \mapsto \langle A, M(\tau)\rangle$, with $A \in \mathbb{S}^m$, belongs to $\mathfrak{F}_{\mathrm{aq}}$. Taking $A = B_j$, we obtain

$$F_j \in \mathfrak{F}_{\mathrm{aq}} \qquad \text{for every } j.$$

Since D is admissible and therefore sufficient for all functionals in $\mathfrak{F}_{\mathrm{aq}}$, there exists a Borel measurable map

$$h_j : \mathbb{R}^p \to \mathbb{R}$$

such that

$$F_j(\tau) = h_j(D(\tau)) \qquad \text{for every } \tau.$$

Equivalently,

$$\langle B_j, M(\tau)\rangle = h_j(D(\tau)) \qquad \text{for every } \tau$$

and for every $j \in \{1, \dots, s\}$.

Define

$$\rho : \mathbb{R}^p \to \mathbb{S}^m$$

by

$$\rho(y) := \sum_{j=1}^{s} h_j(y) B_j.$$

This map is Borel measurable. Indeed, identify $\mathbb{S}^m$ with $\mathbb{R}^s$ through its coordinates in the basis $B_1, \dots, B_s$. Under this identification, ρ has coordinate functions

$$y \mapsto h_1(y), \dots, y \mapsto h_s(y),$$

all of which are Borel measurable. Since a map into a finite-dimensional Euclidean space is Borel measurable if and only if its coordinate functions are Borel measurable, ρ is Borel measurable.

We now show that ρ recovers M from D. Fix any trajectory segment τ. For each $i \in \{1, \dots, s\}$,

$$\langle B_i, \rho(D(\tau))\rangle = \left\langle B_i, \sum_{j=1}^{s} h_j(D(\tau))B_j \right\rangle$$
$$= \sum_{j=1}^{s} h_j(D(\tau))\langle B_i, B_j\rangle.$$

Because $B_1, \dots, B_s$ is orthonormal,

$$\langle B_i, B_j\rangle = \delta_{ij},$$

and hence

$$\langle B_i, \rho(D(\tau))\rangle = h_i(D(\tau)).$$

By construction of h_i,

$$h_i(D(\tau)) = F_i(\tau) = \langle B_i, M(\tau)\rangle.$$

Therefore

$$\langle B_i, \rho(D(\tau))\rangle = \langle B_i, M(\tau)\rangle \qquad \text{for every } i.$$

The two symmetric matrices $\rho(D(\tau))$ and $M(\tau)$ have identical coordinates in the orthonormal basis $B_1, \dots, B_s$. Hence they are equal:

$$\rho(D(\tau)) = M(\tau).$$

Since τ was arbitrary, this proves

$$M(\tau) = \rho(D(\tau)) \qquad \text{for every finite trajectory segment } \tau.$$

It remains to justify the minimal-sufficiency interpretation. Strictly speaking, the admissible descriptors in Definition 3.6 take values in Euclidean spaces, whereas M takes values in $\mathbb{S}^m$. This is only a notational difference, because the vector space $\mathbb{S}^m$ is canonically identifiable with $\mathbb{R}^s$, for example by the half-vectorization map vech. Under this identification, M is a fixed-dimensional descriptor.

The descriptor M is additive under valid concatenation by Theorem 3.4. It is sufficient for $\mathfrak{F}_{\mathrm{aq}}$ by Theorem 3.3: if $F \in \mathfrak{F}_{\mathrm{aq}}$, then there exists $A \in \mathbb{S}^m$ such that

$$F(\tau) = \langle A, M(\tau)\rangle.$$

The map

$$N \mapsto \langle A, N\rangle$$

is linear, hence continuous, hence Borel measurable. Therefore F factors measurably through M.

Finally, the factorization proved above says that for every admissible descriptor D, the matrix proxy M is obtained by measurable post-processing of D. In the usual sufficiency preorder, a descriptor D_1 is no finer than a descriptor D_2 when D_1 is a measurable function of D_2. The identity

$$M = \rho \circ D$$

therefore says that every admissible descriptor D is at least as fine as M. Since M itself is admissible, M is the coarsest admissible additive descriptor preserving all functionals in $\mathfrak{F}_{\mathrm{aq}}$, up to measurable post-processing. □

F.6 Proof of Theorem 3.10

Proof. Let $e_{\mathrm{c}} \in \mathbb{R}^m$ denote the standard basis vector corresponding to the constant channel of the lift ψ_t, namely the last coordinate in

$$\psi_t = \begin{bmatrix} c_t \\ \delta_t \\ \xi(a_t) \otimes c_t \\ \xi(a_t) \otimes \delta_t \\ r_t \\ 1 \end{bmatrix}.$$

For any trajectory segment τ of horizon T, the constant-channel entry of $M(\tau)$ is

$$e_{\mathrm{c}}^{\top} M(\tau) e_{\mathrm{c}} = e_{\mathrm{c}}^{\top} \left(\sum_{t=0}^{T-1} \psi_t \psi_t^{\top} \right) e_{\mathrm{c}} = \sum_{t=0}^{T-1} (e_{\mathrm{c}}^{\top} \psi_t)^2.$$

Since $e_{\mathrm{c}}^{\top} \psi_t = 1$ for every t, this becomes

$$e_{\mathrm{c}}^{\top} M(\tau) e_{\mathrm{c}} = \sum_{t=0}^{T-1} 1 = T.$$

Thus the constant-channel block records the horizon for every realizable matrix. The theorem is applied to a candidate $\widehat{M}$ whose constant-channel block encodes the same horizon T. If the encoded value is not a nonnegative integer, then it cannot equal the horizon of any finite trajectory segment, and non-realizability is immediate. Hence the only nontrivial case is the stated case in which the encoded value is a valid horizon $T \in \mathbb{N}_0$.

Recall the reachable set

$$\mathcal{R}_e^{(T)} = \{M(\tau) : \tau \text{ is realizable in } e \text{ with horizon } T\}.$$

The obstruction score is

$$\mathrm{Obs}_e^{(T)}(\widehat{M}) = \inf_{M \in \mathcal{R}_e^{(T)}} \|\widehat{M} - M\|_F,$$

with the standard convention that the infimum over an empty set is $+\infty$. If $\mathcal{R}_e^{(T)} = \varnothing$, then no trajectory of horizon T is realizable in environment e. In that case $\widehat{M}$ cannot correspond to any realizable trajectory at horizon T, so the conclusion holds.

Assume now that $\mathcal{R}_e^{(T)} \neq \varnothing$. Suppose, for contradiction, that $\widehat{M}$ corresponds to a realizable trajectory in environment e at horizon T. Then there exists a trajectory segment τ, realizable in e with horizon T, such that

$$\widehat{M} = M(\tau).$$

By the definition of $\mathcal{R}_e^{(T)}$, this implies

$$M(\tau) \in \mathcal{R}_e^{(T)}.$$

The distance from $\widehat{M}$ to the reachable set is bounded above by its distance to any particular reachable element. Therefore

$$\mathrm{Obs}_e^{(T)}(\widehat{M}) = \inf_{M \in \mathcal{R}_e^{(T)}} \|\widehat{M} - M\|_F \leq \|\widehat{M} - M(\tau)\|_F.$$

Since $\widehat{M} = M(\tau)$, the right-hand side is

$$\|\widehat{M} - M(\tau)\|_F = \|0\|_F = 0.$$

The Frobenius norm is nonnegative, so the infimum of such nonnegative distances is also nonnegative:

$$\mathrm{Obs}_e^{(T)}(\widehat{M}) \geq 0.$$

Combining the two inequalities gives

$$\mathrm{Obs}_e^{(T)}(\widehat{M}) = 0.$$

This contradicts the assumption

$$\mathrm{Obs}_e^{(T)}(\widehat{M}) > 0.$$

Therefore no realizable trajectory in environment e at horizon T can have matrix proxy equal to $\widehat{M}$. Positive obstruction is consequently a certificate of inconsistency with the environment. The argument proves only this one-way implication; it does not assert that zero obstruction implies exact realizability unless additional closedness or attainment assumptions are imposed on $\mathcal{R}_e^{(T)}$. □

F.7 Proof of Proposition 4.3

Proof. By definition of $\mathcal{T}_M$,

$$(\mathcal{T}_M \bar{Q}^\star)(o, M, a) = \mathbb{E}\left[r + \gamma \max_{a' \in \mathcal{A}} \bar{Q}^\star\big(o', M + \psi(o, a, r, o')\psi(o, a, r, o')^\top, a'\big) \mid o, M, a\right].$$

Assumption 4.2 states exactly that the right-hand side equals $\bar{Q}^\star(o, M, a)$ whenever the lifted transition law is well defined. Hence

$$\bar{Q}^\star(o, M, a) = (\mathcal{T}_M \bar{Q}^\star)(o, M, a)$$

pointwise, which is the fixed-point identity $\bar{Q}^\star = \mathcal{T}_M \bar{Q}^\star$.

Next, Assumption 4.1 states that for every environment e in the aligned family and every rollout prefix $\tau_{0:t-1}$,

$$Q_e^\star(o_t, \tau_{0:t-1}, a) = \bar{Q}^\star(o_t, M(\tau_{0:t-1}), a).$$

Thus the optimal action-value of the original history-dependent problem is obtained by evaluating the lifted Bellman fixed point at the matrix state generated by the prefix. Therefore the Bellman recursion closes exactly on the lifted state (o, M). □

F.8 Proof of Theorem 4.6

Proof. Fix $o \in \mathcal{O}$, $a \in \mathcal{A}$, and a reference matrix M in the relevant matrix domain. Assumptions 4.1 and 4.2 identify $\bar{Q}^\star(o, M, a)$ as the optimal Bellman value associated with the lifted state (o, M). The analytic expansion itself uses Assumption 4.4.

By Assumption 4.4, there exist an open set

$$\mathcal{U}_{o,a} \subseteq \mathbb{S}^m$$

containing M and a function

$$\widehat{Q}^\star_{o,a} : \mathcal{U}_{o,a} \to \mathbb{R}$$

such that

$$\widehat{Q}^\star_{o,a}(N) = \bar{Q}^\star(o, N, a)$$

for every relevant $N \in \mathcal{U}_{o,a}$. The map $\widehat{Q}^\star_{o,a}$ is Fréchet differentiable at M. Therefore there exists a bounded linear functional

$$D\widehat{Q}^\star_{o,a}(M) : \mathbb{S}^m \to \mathbb{R}$$

such that

$$\widehat{Q}^\star_{o,a}(M + K) = \widehat{Q}^\star_{o,a}(M) + D\widehat{Q}^\star_{o,a}(M)[K] + r(K)$$

for all sufficiently small $K \in \mathbb{S}^m$, where the remainder $r(K)$ satisfies

$$\frac{|r(K)|}{\|K\|_F} \to 0 \qquad \text{as } \|K\|_F \to 0.$$

Equivalently,

$$r(K) = o(\|K\|_F) \qquad \text{as } \|K\|_F \to 0.$$

The vector space $\mathbb{S}^m$, equipped with the Frobenius inner product, is a finite-dimensional Hilbert space. Hence every bounded linear functional on $\mathbb{S}^m$ has a unique Riesz representative in $\mathbb{S}^m$. Therefore there exists a unique matrix

$$W_{o,M,a} \in \mathbb{S}^m$$

such that

$$D\widehat{Q}^\star_{o,a}(M)[K] = \langle W_{o,M,a}, K \rangle \qquad \text{for every } K \in \mathbb{S}^m.$$

Substituting this representation into the Fréchet expansion gives

$$\widehat{Q}^\star_{o,a}(M + K) = \widehat{Q}^\star_{o,a}(M) + \langle W_{o,M,a}, K \rangle + r(K).$$

Now take

$$K = H_0 + H_1.$$

Whenever $\|H_0\|_F + \|H_1\|_F$ is sufficiently small and $M + H_0 + H_1$ lies in the relevant domain, we have $M + H_0 + H_1 \in \mathcal{U}_{o,a}$ and

$$\bar{Q}^\star(o, M + H_0 + H_1, a) = \widehat{Q}^\star_{o,a}(M + H_0 + H_1).$$

Thus

$$\bar{Q}^\star(o, M + H_0 + H_1, a) = \widehat{Q}^\star_{o,a}(M) + \langle W_{o,M,a}, H_0 + H_1 \rangle + r(H_0 + H_1).$$

Since $\widehat{Q}^\star_{o,a}(M) = \bar{Q}^\star(o, M, a)$, this becomes

$$\bar{Q}^\star(o, M + H_0 + H_1, a) = \bar{Q}^\star(o, M, a) + \langle W_{o,M,a}, H_0 + H_1 \rangle + r(H_0 + H_1).$$

The remainder satisfies

$$r(H_0 + H_1) = o(\|H_0 + H_1\|_F) \qquad \text{as } \|H_0 + H_1\|_F \to 0.$$

Therefore

$$\bar{Q}^\star(o, M + H_0 + H_1, a) = \bar{Q}^\star(o, M, a) + \langle W_{o,M,a}, H_0 + H_1 \rangle + o(\|H_0 + H_1\|_F).$$

This proves the first claimed expansion.

We now prove the approximate additivity of value increments. Define, for sufficiently small H,

$$\Delta(H) := \bar{Q}^\star(o, M + H, a) - \bar{Q}^\star(o, M, a).$$

Using the expansion above with $K = H$, we have

$$\Delta(H) = \langle W_{o,M,a}, H \rangle + r(H), \qquad r(H) = o(\|H\|_F).$$

Hence

$$\Delta(H_0 + H_1) = \langle W_{o,M,a}, H_0 + H_1 \rangle + r(H_0 + H_1),$$

while

$$\Delta(H_0) = \langle W_{o,M,a}, H_0 \rangle + r(H_0)$$

and

$$\Delta(H_1) = \langle W_{o,M,a}, H_1 \rangle + r(H_1).$$

Because the Frobenius inner product is linear in its second argument,

$$\langle W_{o,M,a}, H_0 + H_1 \rangle = \langle W_{o,M,a}, H_0 \rangle + \langle W_{o,M,a}, H_1 \rangle.$$

Subtracting the expressions gives

$$\Delta(H_0 + H_1) - \Delta(H_0) - \Delta(H_1) = r(H_0 + H_1) - r(H_0) - r(H_1).$$

Let

$$s(H_0, H_1) := \|H_0\|_F + \|H_1\|_F.$$

We claim that

$$r(H_0 + H_1) - r(H_0) - r(H_1) = o(s(H_0, H_1)) \qquad \text{as } s(H_0, H_1) \to 0.$$

Let $\varepsilon > 0$. Since $r(H) = o(\|H\|_F)$, there exists $\delta > 0$ such that

$$\|H\|_F < \delta \quad \Longrightarrow \quad |r(H)| \le \varepsilon \|H\|_F.$$

If

$$s(H_0, H_1) = \|H_0\|_F + \|H_1\|_F < \delta,$$

then

$$\|H_0\|_F < \delta, \qquad \|H_1\|_F < \delta,$$

and, by the triangle inequality,

$$\|H_0 + H_1\|_F \le \|H_0\|_F + \|H_1\|_F = s(H_0, H_1) < \delta.$$

Therefore

$$|r(H_0)| \le \varepsilon \|H_0\|_F, \qquad |r(H_1)| \le \varepsilon \|H_1\|_F,$$

and

$$|r(H_0 + H_1)| \le \varepsilon \|H_0 + H_1\|_F \le \varepsilon(\|H_0\|_F + \|H_1\|_F).$$

Hence

$$\begin{aligned}|r(H_0+H_1)-r(H_0)-r(H_1)| &\le |r(H_0+H_1)|+|r(H_0)|+|r(H_1)| \\ &\le 2\varepsilon(\|H_0\|_F+\|H_1\|_F).\end{aligned}$$

Since $\varepsilon > 0$ was arbitrary,

$$r(H_0+H_1)-r(H_0)-r(H_1)=o(\|H_0\|_F+\|H_1\|_F).$$

Returning to Δ, we obtain

$$\Delta(H_0+H_1)=\Delta(H_0)+\Delta(H_1)+o(\|H_0\|_F+\|H_1\|_F).$$

Expanding the definition of Δ gives

$$\begin{aligned}&\bar{Q}^\star(o,M+H_0+H_1,a)-\bar{Q}^\star(o,M,a) \\ &\quad=\big[\bar{Q}^\star(o,M+H_0,a)-\bar{Q}^\star(o,M,a)\big]+\big[\bar{Q}^\star(o,M+H_1,a)-\bar{Q}^\star(o,M,a)\big] \\ &\qquad+o(\|H_0\|_F+\|H_1\|_F).\end{aligned}$$

This is the claimed first-order additivity of Bellman-value increments in matrix coordinates. The centered approximation in the theorem is the same identity specialized to the base point $M = 0$, with the common base value subtracted once. □

F.9 Proof of Corollary 4.7

Proof. Fix $o \in \mathcal{O}$ and $a \in \mathcal{A}$. Assume that 0, H_0, H_1, and H_0+H_1 lie in the relevant matrix domain, and that the increments are sufficiently small for the local expansion in Theorem 4.6 to apply at the reference matrix $M = 0$. The centered value is defined by

$$\widetilde{Q}^\star(o,M,a)=\bar{Q}^\star(o,M,a)-\bar{Q}^\star(o,0,a).$$

Applying Theorem 4.6 with reference point $M = 0$ gives

$$\begin{aligned}&\bar{Q}^\star(o,H_0+H_1,a)-\bar{Q}^\star(o,0,a) \\ &\quad=\big[\bar{Q}^\star(o,H_0,a)-\bar{Q}^\star(o,0,a)\big]+\big[\bar{Q}^\star(o,H_1,a)-\bar{Q}^\star(o,0,a)\big] \\ &\qquad+o(\|H_0\|_F+\|H_1\|_F).\end{aligned}$$

The left-hand side is exactly

$$\widetilde{Q}^\star(o,H_0+H_1,a).$$

The first bracket on the right-hand side is

$$\bar{Q}^\star(o,H_0,a)-\bar{Q}^\star(o,0,a)=\widetilde{Q}^\star(o,H_0,a),$$

and the second bracket is

$$\bar{Q}^\star(o,H_1,a)-\bar{Q}^\star(o,0,a)=\widetilde{Q}^\star(o,H_1,a).$$

Substituting these three identities into the theorem's expansion yields

$$\widetilde{Q}^\star(o,H_0+H_1,a)=\widetilde{Q}^\star(o,H_0,a)+\widetilde{Q}^\star(o,H_1,a)+o(\|H_0\|_F+\|H_1\|_F).$$

This proves the centered compositional value law. □

F.10 Additional Matrix-Value Consequences

Proposition F.1 (Local value stability and value-null directions)**.** *Fix (o,a) and a reference matrix M. Under Assumption 4.4, there exist a Frobenius neighborhood $\mathcal{U}$ of M in the relevant matrix domain and a constant $L_{o,M,a} < \infty$ such that $|\bar{Q}^\star(o,N_1,a)-\bar{Q}^\star(o,N_0,a)| \le L_{o,M,a}\|N_1-N_0\|_F$ for all $N_0, N_1 \in \mathcal{U}$. Moreover, if $H \in \mathbb{S}^m$ is such that $M+H \in \mathcal{U}$, then $\bar{Q}^\star(o,M+H,a)-\bar{Q}^\star(o,M,a) = \langle W_{o,M,a}, H\rangle + o(\|H\|_F)$. In particular, any perturbation direction H satisfying $\langle W_{o,M,a}, H\rangle = 0$ is invisible to first order: $\bar{Q}^\star(o,M+H,a)-\bar{Q}^\star(o,M,a)=o(\|H\|_F)$.*

Applied to a grounded transition structure G with descriptor $M(G)$, Proposition F.1 says that exact matrix-preserving transformations leave value unchanged exactly, while families of deformations with small induced matrix perturbation that is Frobenius-orthogonal to the local gradient are value-invisible to first order. Equivalently, the matrix directions aligned with $W_{o,M,a}$ are the locally value-relevant structural features, while Frobenius-orthogonal directions are first-order nuisance directions.

In particular, if T is any transformation on grounded transition structures such that $M(TG) = M(G)$, then $\bar{Q}^\star(o, M(TG), a) = \bar{Q}^\star(o, M(G), a)$. If instead $\{T_\varepsilon\}_{\varepsilon>0}$ is a family of transformations such that $M(T_\varepsilon G) = M(G) + H_\varepsilon, \quad H_\varepsilon \to 0$, and $\langle W_{o,M(G),a}, H_\varepsilon \rangle = 0 \quad$ for all ε, then $\bar{Q}^\star(o, M(T_\varepsilon G), a) - \bar{Q}^\star(o, M(G), a) = o(\|H_\varepsilon\|_F)$. No first-order claim is intended for discrete edits whose induced matrix change is not small in Frobenius norm.

Proposition F.2 (Population sufficiency of the matrix state for value regression)**.** *Fix an environment e and any distribution over tuples $(o_t, \tau_{0:t-1}, a)$ induced by interaction with e. Let $X := (o_t, \tau_{0:t-1}, a), \quad Z := (o_t, M(\tau_{0:t-1}), a), \quad Y := Q^\star_e(o_t, \tau_{0:t-1}, a)$. Under Assumption 4.1, $Y = \bar{Q}^\star(Z)$ almost surely. Hence $\mathbb{E}[Y \mid X] = \mathbb{E}[Y \mid Z] = \bar{Q}^\star(Z)$ almost surely, and, for squared loss, $\inf_f \mathbb{E}\big[(Y - f(X))^2\big] = \inf_g \mathbb{E}\big[(Y - g(Z))^2\big] = 0$, where the infima range over measurable predictors f and g. Therefore, at the population level, access to the full transition history cannot improve optimal value prediction beyond the matrix state.*

Proposition F.2 is a population sufficiency statement, not a finite-sample generalization bound. Its role is to justify why one should prefer a matrix-conditioned value model to an unrestricted history-to-value map when the matrix-conditioned value law is the intended inductive bias.

F.11 Proof of Proposition F.1

Proof. Fix $o \in \mathcal{O}$, $a \in \mathcal{A}$, and a reference matrix M. By Assumption 4.4, there exist an open set

$$\mathcal{U}_{o,a} \subseteq \mathbb{S}^m$$

containing M and a function

$$\widehat{Q}^\star_{o,a} : \mathcal{U}_{o,a} \to \mathbb{R}$$

such that

$$\widehat{Q}^\star_{o,a}(N) = \bar{Q}^\star(o, N, a)$$

for every relevant $N \in \mathcal{U}_{o,a}$. Moreover, $D\widehat{Q}^\star_{o,a}$ is locally bounded in the operator norm induced by the Frobenius norm.

Choose $r > 0$ such that the closed Frobenius ball

$$\overline{B}_r(M) := \{N \in \mathbb{S}^m : \|N - M\|_F \le r\}$$

is contained in $\mathcal{U}_{o,a}$. For each $N \in \overline{B}_r(M)$, local boundedness of $D\widehat{Q}^\star_{o,a}$ gives an open neighborhood $\mathcal{V}_N \subseteq \mathcal{U}_{o,a}$ and a constant $L_N < \infty$ such that

$$\|D\widehat{Q}^\star_{o,a}(P)\|_{\mathrm{op},F} \le L_N \qquad \text{for all } P \in \mathcal{V}_N,$$

where $\|\cdot\|_{\mathrm{op},F}$ denotes the operator norm induced by $\|\cdot\|_F$. The family $\{\mathcal{V}_N : N \in \overline{B}_r(M)\}$ is an open cover of the compact set $\overline{B}_r(M)$, so it admits a finite subcover $\mathcal{V}_{N_1}, \dots, \mathcal{V}_{N_K}$. Let

$$L_{o,M,a} := \max_{1 \le j \le K} L_{N_j} < \infty.$$

Then

$$\|D\widehat{Q}^\star_{o,a}(P)\|_{\mathrm{op},F} \le L_{o,M,a} \qquad \text{for all } P \in \overline{B}_r(M).$$

Now let N_0, N_1 lie in the relevant matrix domain and satisfy $\|N_i - M\|_F < r$. Since Frobenius balls are convex, the segment

$$N_\theta := (1-\theta)N_0 + \theta N_1, \qquad \theta \in [0,1],$$

lies in $\overline{B}_r(M)$. Define

$$g(\theta) := \widehat{Q}^\star_{o,a}(N_\theta).$$

Because $\widehat{Q}^\star_{o,a}$ is Fréchet differentiable, g is differentiable on $[0,1]$ and the chain rule gives

$$g'(\theta) = D\widehat{Q}^\star_{o,a}(N_\theta)[N_1 - N_0].$$

Therefore

$$|g'(\theta)| \le \|D\widehat{Q}^\star_{o,a}(N_\theta)\|_{\mathrm{op},F}\, \|N_1 - N_0\|_F \le L_{o,M,a}\|N_1 - N_0\|_F.$$

The fundamental theorem of calculus yields

$$|\widehat{Q}^\star_{o,a}(N_1) - \widehat{Q}^\star_{o,a}(N_0)| = \left|\int_0^1 g'(\theta)\, d\theta\right| \le \int_0^1 |g'(\theta)|\, d\theta \le L_{o,M,a}\|N_1 - N_0\|_F.$$

Since $\widehat{Q}^\star_{o,a}$ agrees with $\bar{Q}^\star(o,\cdot,a)$ on the relevant domain, this proves

$$|\bar{Q}^\star(o,N_1,a) - \bar{Q}^\star(o,N_0,a)| \le L_{o,M,a}\|N_1 - N_0\|_F$$

for all N_0, N_1 in the domain with $\|N_i - M\|_F < r$. This is the first claim, with $\mathcal{U} := \{N : \|N - M\|_F < r\}$ intersected with the relevant matrix domain.

For the second claim, Theorem 4.6 gives

$$\bar{Q}^\star(o, M+H, a) - \bar{Q}^\star(o, M, a) = \langle W_{o,M,a}, H\rangle + o(\|H\|_F)$$

as $\|H\|_F \to 0$. If $\langle W_{o,M,a}, H\rangle = 0$, the leading first-order term vanishes and therefore

$$\bar{Q}^\star(o, M+H, a) - \bar{Q}^\star(o, M, a) = o(\|H\|_F).$$

This proves the stated first-order invisibility of value-null directions. □

F.12 Proof of Proposition F.2

Proof. Fix an environment e and any distribution over tuples $(o_t, \tau_{0:t-1}, a)$ induced by interaction with e. Define

$$X := (o_t, \tau_{0:t-1}, a), \qquad Z := (o_t, M(\tau_{0:t-1}), a), \qquad Y := Q^\star_e(o_t, \tau_{0:t-1}, a).$$

Assumption 4.1 states pointwise that

$$Y = \bar{Q}^\star(Z).$$

Hence Y is a measurable function of Z, and since Z itself is a measurable function of X, the random variable Y is measurable with respect to both $\sigma(Z)$ and $\sigma(X)$. A basic property of conditional expectation is that if a random variable is measurable with respect to the conditioning sigma-algebra, then its conditional expectation given that sigma-algebra equals the variable itself almost surely. Therefore

$$\mathbb{E}[Y \mid X] = Y \qquad \text{almost surely}$$

and

$$\mathbb{E}[Y \mid Z] = Y \qquad \text{almost surely.}$$

Using $Y = \bar{Q}^\star(Z)$, both identities become

$$\mathbb{E}[Y \mid X] = \mathbb{E}[Y \mid Z] = \bar{Q}^\star(Z) \qquad \text{almost surely.}$$

Now consider squared loss. For any measurable predictor h,

$$(Y - h)^2 \ge 0 \qquad \text{pointwise,}$$

so

$$\mathbb{E}[(Y-h)^2] \ge 0.$$

Choosing the history-based predictor

$$f^\star(X) := \bar{Q}^\star(Z) = Y$$

gives

$$\mathbb{E}[(Y - f^\star(X))^2] = 0.$$

Hence

$$\inf_f \mathbb{E}[(Y - f(X))^2] = 0.$$

Likewise, choosing the matrix-based predictor

$$g^\star(Z) := \bar{Q}^\star(Z) = Y$$

gives

$$\mathbb{E}[(Y - g^\star(Z))^2] = 0,$$

and therefore

$$\inf_g \mathbb{E}[(Y - g(Z))^2] = 0.$$

Combining the two equalities proves

$$\inf_f \mathbb{E}[(Y - f(X))^2] = \inf_g \mathbb{E}[(Y - g(Z))^2] = 0.$$

Thus, at the population level, full history information offers no improvement over the matrix state for predicting the optimal value. □

Justification of the transformation paragraph. If $M(TG) = M(G)$, then

$$\bar{Q}^\star(o, M(TG), a) = \bar{Q}^\star(o, M(G), a)$$

by direct substitution into the matrix-conditioned value law. For the family $\{T_\varepsilon\}_{\varepsilon>0}$, define

$$H_\varepsilon := M(T_\varepsilon G) - M(G).$$

Proposition F.1 gives

$$\bar{Q}^\star(o, M(G) + H_\varepsilon, a) - \bar{Q}^\star(o, M(G), a) = \langle W_{o,M(G),a}, H_\varepsilon \rangle + o(\|H_\varepsilon\|_F).$$

If $\langle W_{o,M(G),a}, H_\varepsilon \rangle = 0$ for all ε, then

$$\bar{Q}^\star(o, M(T_\varepsilon G), a) - \bar{Q}^\star(o, M(G), a) = o(\|H_\varepsilon\|_F),$$

which is the claimed first-order invariance statement.