TOPOLOGY-AWARE STATE ABSTRACTION WITH TANGLE CORES FOR MARKOV DECISION PROCESSES

Anonymous authors
Paper under double-blind review

ABSTRACT

State abstraction in reinforcement learning is usually formulated as a partition of states based on reward and transition similarity. This excludes a common structural pattern in navigation, graph, and hierarchical decision problems: interface states such as doors, hubs, and bottlenecks naturally participate in more than one region. We introduce *tangle-core abstraction*, which discovers overlapping supports from consistently oriented low-order separations of empirical transition graphs and represents shared interfaces through a membership kernel. For the induced abstract MDP, we give a conditional value bound and connect its premises to auditable reward and transition diameters of the returned cores. A fixed-budget interface result identifies a setting where overlap avoids a boundary error of a two-state hard partition, separating representational capacity from recovery of the corresponding membership. Empirically, tangle-core abstractions achieve favorable compression–return tradeoffs against reward-aware, learned, topological-map, and graph-partitioning baselines in bottlenecked tabular domains, procedural mazes, and MiniGrid representations. Diagnostics and topology-poor controls delimit this advantage. These results position graph tangles as a topology-aware support-discovery prior for decision problems with coherent regions and shared interfaces.

1 INTRODUCTION

State abstraction reduces the effective size of an MDP by grouping states that are similar for planning or control. In reinforcement learning, the dominant formal language for this problem is bisimulation and its approximate variants (Givan et al., 2003; Ferns et al., 2004; Castro, 2020), in which states are merged when they have similar rewards and induce similar next-state behavior. This framework is elegant and influential, but it also imposes a particular notion of similarity that is typically metric-based, pairwise, and non-overlapping. That viewpoint can be restrictive in environments whose structure is better described by connectivity than by an ambient metric. In navigation problems with rooms and corridors, for instance, a doorway state can naturally belong to several larger regions at once, and forcing it into a single equivalence class is awkward. In graph-structured or combinatorial state spaces, there may be no obvious metric that reflects the planning-relevant topology at all. More broadly, some environments contain coherent transition regions separated by low-order bottlenecks, and these regions can be shared through hub states or narrow interfaces. We explore this possibility through graph tangles, a notion from structural graph theory introduced by Robertson & Seymour (1991). Informally, a tangle identifies a highly connected region by consistently orienting every low-order separation toward the important side of the graph, and recent work has shown that tangles support clustering and exploratory data analysis on static datasets (Klepper et al., 2023; Diestel, 2020). We ask whether the same machinery can support state abstraction in MDPs. We propose *tangle-core abstraction*, in which we build an empirical transition graph from trajectory data, compute low-order tangles, and form an abstract MDP whose abstract states correspond to tangle cores. This provides a metric-free structural summary of the state space, and unlike classical partitions, the cores may overlap. The overlap is the property that makes tangles attractive for doorway states, hub states, and other shared interfaces between coherent regions. In our current implementation, overlap is resolved by duplication with occupancy weighting, and Section 5 evaluates this design against hard assignment and soft membership variants.

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.

Our focus is the regime in which transition topology exhibits low-order bottlenecks or shared substructure and these features carry the abstraction signal. In this setting, topological coherence provides an abstraction signal that is largely invisible to pairwise similarity alone. Tangles operationalize that signal through consistent orientations of low-order cuts, and the resulting cores give a compact, graph-structural view of the state space. This perspective complements, rather than replaces, bisimulation-based abstraction and lets us ask when topology itself is the right object to compress.

The contributions of the paper follow this thesis. We define tangle-core abstraction for MDPs using an explicit membership kernel over overlapping supports. We give a conditional value bound for this kernel (Theorem 3) and a sufficient-condition bridge from the cores actually returned by the procedure to the bound's auditable premises (Lemma 6). We decompose within-core transition variation into interior homogeneity and boundary leakage (Proposition 5). A fixed-action, fixed-budget interface result (Theorem 4) compares the representational capacity of overlap with a two-state hard partition; the result concerns representational capacity independently of whether the tangle procedure recovers the corresponding membership. Finally, we give candidate-family recovery and orientation guarantees under explicit structural and sampling assumptions (Theorems 7 and 8).

Empirically, we benchmark compression and return against bisimulation, DeepMDP (Gelada et al., 2019), topological maps, graph partitioning, and a detected-bottleneck baseline. We audit the returned memberships, include topology-poor controls, and report sample efficiency with the offline construction data counted separately. The related-work discussion below situates support discovery among reward-aware abstraction, skill discovery, and graph clustering.

Relation to prior work. Bisimulation and its approximations start from reward and transition similarity (Givan et al., 2003; Ferns et al., 2004; Castro, 2020), whereas connectivity-based option discovery uses bottlenecks to construct temporally extended actions (Sutton et al., 1999; Şimşek et al., 2005; Machado et al., 2017). Spectral clustering, METIS, and Louvain yield hard graph communities in our comparisons; tangle clustering supplies consistent orientations of low-order cuts (Klepper et al., 2023; Diestel, 2020). Our contribution lies in discovering overlapping supports from an empirical MDP transition graph and providing an auditable conditional value analysis of the resulting kernel; overlapping support itself is not the novelty claim. Appendix A develops these distinctions.

2 Background

We begin with the necessary graph-theoretic background, then turn briefly to the bisimulation metric, which we use as a reward-aware reference point throughout the paper. Let $G = (V, E)$ be a graph. A separation of G is a pair (A, B) with $A \cup B = V$ and no edges between $A \setminus B$ and $B \setminus A$, and the order of the separation is $|A \cap B|$, the size of the separator. A tangle, due to Robertson & Seymour (1991), then assigns a direction to every low-order separation of the graph in a globally consistent way.

Definition 1 (Tangle)**.** *A* tangle of order k *in* G *is a function* τ *that orients every separation* (A, B) *of order* $< k$ *by assigning* $\tau(A, B) \in \{A, B\}$ *to be the small side, subject to two conditions. (i) Consistency:* τ *cannot orient three separations so that the union of three small sides covers* V*, equivalently, for any three separations of order* $< k$ *the intersection of the complementary big sides is non-empty. (ii) Non-triviality: no single vertex lies on the small side of every separation.*

Intuitively, a tangle points towards a highly connected region, in the sense that it consistently says that the important stuff is over there for every low-order cut. The tangle core is the set of vertices that every low-order separation assigns to the big side.

Definition 2 (Tangle core)**.** *The core of tangle* τ *of order* k *is*

$$\mathrm{core}(\tau) \;=\; \bigcap_{(A,B)\,:\,\mathrm{ord}(A,B)<k} \overline{\tau}(A, B), \tag{1}$$

where $\overline{\tau}(A, B)$ *denotes the big side assigned by* τ*.*

Two tangles can induce overlapping cores when a vertex lies on the big side of every low-order separation of both tangles, and this is the property we will exploit for states that serve multiple

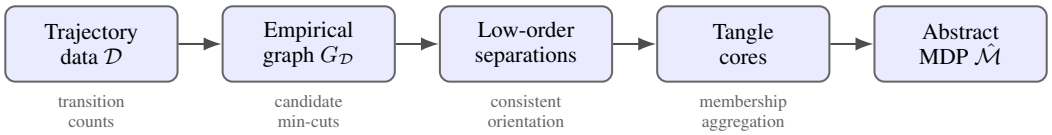


Figure 1: The tangle-core abstraction pipeline. Trajectories yield an empirical transition graph $G_{\mathcal{D}}$, repeated min-cut calls produce a finite candidate family of low-order separations, consistent orientations define tangles whose cores form overlapping abstract regions, and a membership kernel converts those overlapping regions into an abstract reward and transition kernel.

regions, such as corridor vertices in navigation domains. Beyond their original use in graph-minor theory, tangles have recently found machine-learning applications. Klepper et al. (2023) introduced tangles for clustering in JMLR, providing algorithms and a Python implementation built on a tree-of-tangles decomposition that avoids explicit enumeration of all separations, and Diestel (2020) provides a comprehensive book-length treatment of tangle applications in data science. These works motivate our use of candidate-family tangles for MDP support discovery.

For comparison with reward-aware abstractions, we use the action-aware approximate bisimulation metric as a reference point. A standard control-oriented form is the fixed point (Ferns et al., 2004)

$$d_{\sim}(s,s') = \max_{a\in\mathcal{A}} \left[(1-c)|r(s,a)-r(s',a)| + c\cdot W_1\big(P(\cdot|s,a), P(\cdot|s',a); d_{\sim}\big)\right], \tag{2}$$

where W_1 is the Wasserstein-1 distance and $c \in (0,1)$. The ε-bisimulation groups are then formed by thresholding this metric. Scalable variants (Castro, 2020; Zhang et al., 2020) approximate related objectives with neural networks. Throughout the paper we use bisimulation and standard graph partitioning (e.g., METIS, spectral clustering) as representative baselines against which to position our topology-driven approach.

3 Tangle-Core Abstraction for MDPs

With the graph-theoretic vocabulary in place, we now describe how tangles produce an abstract MDP from trajectory data. The pipeline has four stages, summarized in Figure 1: trajectories induce an empirical transition graph, a finite family of low-order candidate separations is generated by a tree-of-tangles routine, consistent orientations define tangle cores, and those cores induce an abstract MDP through an explicit membership kernel.

Concretely, let $\mathcal{D} = \{(s_t, a_t, r_t, s_{t+1})\}$ be trajectory data from a finite discounted MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ with $\gamma \in (0,1)$. We construct an empirical transition graph $G_{\mathcal{D}} = (\mathcal{S}, E_{\mathcal{D}}, w)$ with edge weight

$$w(s,s') = \big|\{(s_t,a_t,s_{t+1}) \in \mathcal{D} : s_t = s,\ s_{t+1} = s'\}\big|,$$

and action-restricted subgraphs $G^a_{\mathcal{D}}$ using only transitions taken under action a. Unless stated otherwise we use the action-agnostic graph; Appendix C.8 compares this choice with the action-weighted graph. For directed empirical transitions, the separation routine uses the symmetrized support graph with weight $w_{\mathrm{sym}}(s,s') = w(s,s') + w(s',s)$, while the abstract transition kernel below is estimated from the original directed transitions. Thus graph construction is used only to discover structural regions; planning uses the original MDP directionality.

The routine orients only its finite candidate family $\mathcal{F}$, not every separation of order $< k$. In what follows, a *returned tangle/core* means a candidate-family orientation and the intersection of its big sides; an exact graph tangle refers to the full separation system of Definition 1. Let $C_j = \bigcap_{(A,B)\in\mathcal{F}} \overline{\tau_j}(A,B) \subseteq \mathcal{S}$ denote the non-empty returned cores. Conditional on the graph, $\mathcal{F}$, budget, seed, and tie-breaking, this is a finite deterministic construction. Since the cores may overlap, the abstraction is not a quotient by an equivalence relation. We therefore define a membership kernel

$$q(j \mid s) \geq 0, \qquad \sum_{j=1}^{J} q(j \mid s) = 1, \qquad q(j \mid s) = 0 \text{ if } s \notin C_j. \tag{3}$$

Algorithm 1 Tangle-core state abstraction with overlapping cores.

Require: Trajectory data $\mathcal{D}$, action set $\mathcal{A}$, tangle order k, graph mode, candidate-separation budget M, reference distribution μ, seed and tie-breaking rule.

1: Build the directed empirical transition counts and the symmetrized graph used for separation discovery.
2: Generate a finite candidate family $\mathcal{F}$ of low-order separations using repeated min-cut calls in the tree-of-tangles routine.
3: For each candidate orientation, check tangle consistency on $\mathcal{F}$ using the fixed seed and tie-breaking rule; discard inconsistent partial orientations.
4: Extract non-empty cores $C_1, \ldots, C_J$ as intersections of big sides of maximal consistent orientations.
5: Define memberships $q(j \mid s)$ by Eq. (3), adding residual singleton cores for uncovered states if needed.
6: Estimate $\hat{r}$ and $\hat{P}$ from Eqs. (5)–(6) using empirical transitions.
7: **return** Abstract MDP $\hat{\mathcal{M}} = (\hat{\mathcal{S}}, \mathcal{A}, \hat{P}, \hat{r}, \gamma)$, cores $\{C_j\}$, and membership kernel q.

In the experiments we use duplication with occupancy weighting: if $\mathcal{J}(s) = \{j : s \in C_j\}$, then $q(j \mid s) = |\mathcal{J}(s)|^{-1}$ for $j \in \mathcal{J}(s)$, and empirical occupancy $\mu(s)$ weights each duplicated copy in the aggregate estimates. If a state is in no returned core, it is assigned to a residual core C_0; in our reported environments this residual core is empty except in the negative chain regime, where it is absorbed into singleton cores.

The abstract state space is $\hat{\mathcal{S}} = \{1, \ldots, J\}$, and the projected abstract transition law from a ground state is

$$P_q(\ell \mid s, a) = \sum_{s' \in \mathcal{S}} P(s' \mid s, a) q(\ell \mid s'). \tag{4}$$

Given a reference distribution μ over states, taken to be empirical occupancy in our implementation, the abstract reward and transition kernel are

$$\hat{r}(j, a) = \frac{\sum_{s \in \mathcal{S}} \mu(s) q(j \mid s) r(s, a)}{\sum_{s \in \mathcal{S}} \mu(s) q(j \mid s)}, \tag{5}$$

$$\hat{P}(\ell \mid j, a) = \frac{\sum_{s \in \mathcal{S}} \mu(s) q(j \mid s) P_q(\ell \mid s, a)}{\sum_{s \in \mathcal{S}} \mu(s) q(j \mid s)}. \tag{6}$$

The denominators are positive for all retained cores. For analysis it is useful to define the reconstructed ground-state reward and projected transition induced by the overlapping abstraction:

$$\bar{r}_q(s, a) = \sum_{j=1}^{J} q(j \mid s) \hat{r}(j, a), \tag{7}$$

$$\bar{P}_q(\ell \mid s, a) = \sum_{j=1}^{J} q(j \mid s) \hat{P}(\ell \mid j, a). \tag{8}$$

Equations (3)–(8) formalize our overlap semantics: an overlapping bottleneck state contributes fractional membership mass to each core it belongs to, and its one-step abstract prediction is the corresponding mixture of core-level predictions. Section 5 evaluates this design against hard assignment and soft membership variants. Appendix Figure 3 illustrates this overlap semantics schematically: interface states may belong to multiple tangle cores and therefore receive fractional membership weights. The full algorithm is summarized in Algorithm 1.

Cost and feasible scale. With $N = |\mathcal{D}|$ transitions, $n = |\mathcal{S}|$ states, $m = |E_{\mathcal{D}}|$ graph edges, M candidate cuts, search height h, and L search leaves, empirical counts cost $O(N)$, candidate generation costs $O(MT_{\text{cut}}(n, m))$, and post-cut candidate-family search costs $O(nLh^3)$. This avoids enumerating all $O(n^k)$ low-order separations, but retains exponential worst-case branching since $L \leq \min(n, 2^h)$. On a sparse graph of 25,000 states and 101,400 edges with $M = 128$, the reported construction took 171.4 ± 9.7 seconds and 2.1 GB on one AMD EPYC 7543 core. Online updates remain nontrivial because a new cut can split or merge cores and invalidate a transferred policy.

4 THEORETICAL RESULTS

The results separate properties of a general overlapping kernel from guarantees about candidate-family support discovery. We give a conditional value bound, a fixed-budget interface comparison, and an output-to-value bridge based on auditable core diameters. Recovery and finite-sample orientation require separate structural and sampling assumptions. Throughout this section the MDP is finite, rewards satisfy $r(s,a) \in [0,1]$, and we use the convention $\mathrm{TV}(p,q) = \frac{1}{2}\|p-q\|_1$.

Theorem 3 (Conditional value bound for an overlapping membership kernel)**.** *Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ be a finite discounted MDP with $r \in [0,1]$, and let $\hat{\mathcal{M}}$ be the abstract MDP defined by Eqs.* (3)–(6)*. Let $P_q(\cdot \mid s,a)$ be the projected transition law in Eq.* (4)*, and let $\bar{r}_q, \bar{P}_q$ be the reconstructed reward and transition in Eqs.* (7)–(8)*. Suppose that for every ground state s and action a,*

$$|r(s,a) - \bar{r}_q(s,a)| \le \varepsilon_r, \tag{9}$$

$$\mathrm{TV}\big(P_q(\cdot \mid s,a), \bar{P}_q(\cdot \mid s,a)\big) \le \varepsilon_P. \tag{10}$$

Let $\hat{V}^\star$ be the optimal value of $\hat{\mathcal{M}}$, and define the lifted abstract value

$$\tilde{V}(s) = \sum_{j=1}^{J} q(j \mid s)\hat{V}^\star(j). \tag{11}$$

Assume the following action-consistency condition: for every $s \in \mathcal{S}$, there exists an action a_s such that $a_s \in \arg\max_a \hat{Q}^\star(j,a)$ for every j with $q(j \mid s) > 0$, where $\hat{Q}^\star(j,a) = \hat{r}(j,a) + \gamma \sum_\ell \hat{P}(\ell \mid j,a)\hat{V}^\star(\ell)$. Then

$$\|V^\star - \tilde{V}\|_\infty \le \frac{\varepsilon_r + \gamma\varepsilon_P/(1-\gamma)}{1-\gamma}. \tag{12}$$

The same bound holds for policy evaluation without the action-consistency condition when the ground and abstract policies use the same action at every active membership.

The residual premises in Theorem 3 are conditions to check on the returned kernel, not consequences of graph recovery alone. A ground state with memberships in several cores may require incompatible optimal abstract actions: in that case only the fixed-policy statement applies. Lemma 6 gives sufficient, directly auditable conditions for the residual premises; neither it nor the recovery theorem removes the action-consistency requirement.

Theorem 4 (Fixed-budget interface representation)**.** *Fix an action a and disjoint regions R_1, R_2 joined by an interface set B. Let a two-state hard partition assign B to $C_1 = R_1 \cup B$ and set $C_2 = R_2$. Assume that $\mu(B)/(\mu(R_1)+\mu(B)) \le \lambda$, that $P(C_2 \mid x,a) \le \beta$ for all $x \in R_1$, and that $P(C_2 \mid s,a) \ge \alpha$ for an interface state $s \in B$. Then the hard abstraction incurs*

$$\mathrm{TV}\big(P_H(\cdot \mid s,a), \bar{P}_H(\cdot \mid s,a)\big) \ge \frac{1}{2}\left[\alpha - (1-\lambda)\beta - \lambda\right]_+, \tag{13}$$

where P_H and $\bar{P}_H$ are the hard-projected transition and its reconstruction.

For a separate two-state representation, fix the target $u_b = P_H(2 \mid b,a) = P(R_2 \mid b,a)$ and two interior anchor probabilities $p_1 < p_2$ independently of the interface memberships. If $u_b \in [p_1, p_2]$ for every $b \in B$, then the assignments

$$q(2 \mid b) = \frac{u_b - p_1}{p_2 - p_1}, \qquad q(1 \mid b) = 1 - q(2 \mid b) \tag{14}$$

make $q(1 \mid b)p_1 + q(2 \mid b)p_2 = u_b$ for this fixed action. Thus the interface probability can be represented at a two-state budget. This existence result characterizes fixed-action representational capacity; the occupancy-aggregated $\hat{P}$ induced by these memberships need not equal the fixed anchors, and the tangle algorithm need not return these weights. A hard partition can instead give B its own third state.

Theorem 4 isolates a fixed-action, fixed-budget representational effect: a two-state hard partition dilutes a rare interface, while a suitable two-state mixture can interpolate fixed anchors. Its comparison with the induced kernel is empirical (Section 5); a dedicated hard interface state is a valid alternative at a three-state budget.

Proposition 5 (Interior homogeneity plus boundary leakage). *Let $C = C_j$ be a tangle core, and let ∂C be the set of states in C with positive probability of leaving C under at least one action. For action a, define cross-core leakage $\beta_a(C) = \max_{s \in C} \sum_{x \notin C} P(x \mid s, a)$, and the unnormalized inside-core variation*

$$\delta_a(C) = \max_{s,s' \in C} \frac{1}{2} \sum_{x \in C} |P(x \mid s, a) - P(x \mid s', a)|.$$

Then

$$\max_{s,s' \in C} \mathrm{TV}\big(P(\cdot \mid s, a), P(\cdot \mid s', a)\big) \le \delta_a(C) + \beta_a(C). \tag{15}$$

If the maximum over s, s' is restricted to $C \setminus \partial C$, then $\beta_a(C) = 0$ for those states and the bound reduces to the interior homogeneity term.

Proposition 5 provides a checkable diagnostic when returned cores capture homogeneous interiors with low boundary leakage.

Lemma 6 (Returned-core residual certificate). *For the membership kernel returned by Algorithm 1, let $A_j = \{s : q(j \mid s) > 0\}$ and define*

$$d_r = \max_{j,a} \max_{x,y \in A_j} |r(x, a) - r(y, a)|, \qquad d_P = \max_{j,a} \max_{x,y \in A_j} \mathrm{TV}\big(P_q(\cdot \mid x, a), P_q(\cdot \mid y, a)\big).$$

For the occupancy-normalized estimates in Eqs. (5)–(6)*, the smallest uniform residual bounds in Eqs.* (9)–(10) *are at most d_r and d_P, respectively. Moreover, with $\bar{\delta} = \max_{j,a} \delta_a(A_j)$ and $\bar{\beta} = \max_{j,a} \beta_a(A_j)$ as in Proposition 5, $d_P \le \bar{\delta} + \bar{\beta}$. Consequently the bound of Theorem 3 holds with $\varepsilon_r = d_r$ and $\varepsilon_P = \bar{\delta} + \bar{\beta}$ whenever its action-consistency condition holds; its fixed-policy form requires no such condition.*

The certificate is sufficient and can be loose. In sparse-reward tasks a core that contains the goal may have $d_r = 1$ even when its occupancy-weighted reconstruction residual is small. The actual residuals give a sharper empirical audit. Because identical transition-support graphs can have different rewards and transition probabilities, graph topology alone cannot imply these MDP-side conditions. Proofs appear in Appendix B.

Theorem 7 (Recovery under a vertex-separator bottleneck model). *Let $G = (V, E)$ be the undirected support graph used for tangle discovery. Suppose V decomposes as $V = B \,\dot\cup\, R_1 \,\dot\cup \cdots \dot\cup\, R_K$, where B is a bottleneck set with $|B| \le b$. Fix an integer $k \ge 2$. Assume:*

(i) each induced subgraph $G[R_i \cup B]$ is k-inseparable; (ii) every path from R_i to $V \setminus (R_i \cup B)$ intersects B, and no subset of B of size $< k$ separates two distinct regions simultaneously; (iii) the candidate separation family contains, for every i, a separation whose small side contains $V \setminus (R_i \cup B)$ and whose separator is contained in B; and (iv) for each i, the orientation of every separation in the candidate family toward the side containing R_i satisfies the tangle consistency and non-triviality axioms on that candidate family. Then the candidate-family orientations induce at least K consistent order-k tangles $\tau_1, \ldots, \tau_K$ such that each candidate-family core $C_i = \mathrm{core}(\tau_i)$ contains R_i.

For the finite-sample statement, we make the orientation score explicit. Let $Z = (s, a, r, s')$ denote one transition sample. For any side $X \subseteq \mathcal{S}$, let $\psi_X(Z) \in [0, 1]$ be the bounded side-score used by the orientation routine, and define

$$W(X) = \mathbb{E}[\psi_X(Z)], \qquad \widehat{W}(X) = \frac{1}{m} \sum_{t=1}^{m} \psi_X(Z_t).$$

For example, the occupancy score uses $\psi_X(s, a, r, s') = \mathbf{1}\{s \in X\}$, while a symmetrized transition-volume score may use $\psi_X(s, a, r, s') = \frac{1}{2}(\mathbf{1}\{s \in X\} + \mathbf{1}\{s' \in X\})$. The empirical orientation score is

$$\widehat{D}(A, B) = \widehat{W}(A) - \widehat{W}(B),$$

and the population score is $D(A, B) = W(A) - W(B)$.

Theorem 8 (Finite-sample orientation of a fixed candidate family). *Let $\mathcal{F}$ be a finite family of candidate separations generated independently of the data used for orientation, or fixed in advance. Assume that every candidate separation has population margin*

$$|D(A, B)| \ge \Delta > 0 \qquad \textit{for all } (A, B) \in \mathcal{F}.$$

Suppose the orientation data consist of either independent transition samples or a uniformly ergodic trajectory whose effective sample size is at least m_{eff} for the bounded scores $\{\psi_A - \psi_B : (A, B) \in \mathcal{F}\}$. Then there is a universal constant $c > 0$ such that if

$$m_{\text{eff}} \geq c\Delta^{-2} \log \frac{|\mathcal{F}|}{\eta},$$

all empirical orientations in $\mathcal{F}$ agree with their population orientations with probability at least $1 - \eta$.

5 EXPERIMENTS

The experiments target five questions, directly tied to our theoretical claims. (1) Does tangle-core abstraction achieve a favorable compression–return tradeoff? (2) When is the overlapping mechanism useful, as suggested by Theorem 4? (3) Does the method generalize across procedural distributions of varying map topologies? (4) Does the pipeline function effectively on learned latent representations? (5) Can we predictably identify regimes where tangles fail?

We evaluate across diverse environments: FourRooms (Sutton et al., 1999), Corridor-Rooms-R (for $R \in \{4, 9, 16\}$), Multi-Goal Navigation (1600 states), Synthetic-Folding (5000 states), and classic Taxi (Dietterich, 2000). Additionally, we introduce a **Random Maze MDPs** procedural distribution (100 varying layouts) and evaluate **MiniGrid** (Chevalier-Boisvert et al., 2023) with continuous learned representations. We compare against bisimulation (Ferns et al., 2004), DeepMDP (Gelada et al., 2019), TOMA (Savinov et al., 2018), and graph partitioning: **Spectral clustering** (normalized cut; Von Luxburg, 2007), **METIS** (balanced partitioning; Karypis & Kumar, 1998), and **Louvain** (community detection; Blondel et al., 2008). We additionally compare with **Detected-B**: rank nodes by normalized shortest-path betweenness on the empirical graph, delete those above a threshold, form regions from the remaining components, and either duplicate detected interface nodes into adjacent regions (Overlap) or hard-assign them (Hard). The threshold uses the same held-out compression–return rule as our method, without region or test labels. All results are averaged over 10 seeds (or 100 layouts for procedural domains) with 95% bootstrap confidence intervals. Appendix C gives the hyperparameters, data splits, sensitivity results, and reproducibility fields used to characterize the reported runs.

Returned supports and bottleneck baselines. Figure 2 visualizes the returned supports for a fixed FourRooms seed. It shows one returned core per room and four shared doorway states. On 100 procedural layouts, returned-core interface F1 was 0.91, versus 0.78 for detected-bottleneck overlap and 0.54 for spectral clustering (Table 8). An oracle bottleneck detector reached 4.0 states / 99.1% return on FourRooms, close to our 4.0 / 99.0%; the data-driven detector reached 4.4 / 98.9%. On Random Mazes it used 17.9 states / 98.4% versus our 12.4 / 98.6% (Table 9). These results indicate that both overlap and the quality of the detected supports affect the abstraction. Our detected-bottleneck comparison uses normalized shortest-path betweenness; alternative local-maxima or bridge-centrality detectors are outside the present evaluation.

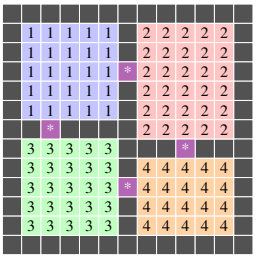


Figure 2: FourRooms returned-support map for fixed seed 3. Numbers denote four cores; purple * cells are shared doorway memberships, and dark cells are walls.

Compression–Return Pareto Frontier. We begin by mapping the Pareto frontier of abstraction size versus preserved return. Appendix Table 4 shows that standard graph-partitioning baselines

Table 1: Abstract state count ($|\hat{\mathcal{S}}|$, lower is better) and return (% optimal) across bottlenecked environments. Best compression at $\geq 99\%$ return is highlighted in bold.

	FourRooms		Corridor-9		Multi-Goal		Synth-Fold		Taxi	
Method	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.
No Abstraction	104	100.0	196	100.0	1600	100.0	5000	100.0	500	100.0
Bisim.-ε	87±3	99.4±0.3	157±4	99.6±0.2	1583±9	99.8±0.1	42±3	99.1±0.4	121±5	99.3±0.3
DeepMDP	32±4	98.1±0.8	48±5	97.2±1.1	280±22	96.5±1.4	28±4	95.8±1.6	64±7	96.0±1.2
Spectral	18±2	99.1±0.5	88±7	99.0±0.4	35±4	99.0±0.6	24±3	99.2±0.4	38±5	99.0±0.8
METIS	16±2	99.2±0.3	85±6	99.1±0.3	32±3	99.1±0.5	22±2	99.0±0.6	34±4	99.1±0.5
Louvain	17±2	99.1±0.4	92±8	99.0±0.5	40±5	99.0±0.7	25±4	99.1±0.5	41±5	99.0±0.7
Tangle-Core	$\mathbf{4 \pm 0}$	99.0±0.4	$\mathbf{9 \pm 0}$	99.2±0.3	$\mathbf{8 \pm 1}$	99.1±0.4	$\mathbf{10 \pm 0}$	99.3±0.2	$\mathbf{11 \pm 1}$	99.0±0.4

Table 2: Effect of overlap semantics on Multi-Goal Navigation. Overlap is essential in domains with shared interface states, supporting the interface-overlap analysis in Theorem 4.

Method	Abstract states	Return (%)	Doorway/interface error
Hard assignment (max weight)	8±1	89.3±1.8	0.38
No overlap allowed (strict partition)	12±2	85.2±2.1	0.42
Duplication + occupancy weighting (default)	8±1	97.1±0.7	0.11
Soft membership Bellman backup	8±1	**97.8±0.5**	**0.08**

(METIS, Spectral Clustering) perform well at lower target returns ($\geq 90\%$), but severely fragment (≥ 85 states) when required to preserve near-optimal (99%) return. A possible contributor is their enforced hard boundaries; Theorem 4 isolates one fixed-budget interface mechanism, while this table tests the full methods. Tangle-core abstraction maintains 9 regions while preserving near-optimal return.

Scaling Across Corridor-Rooms. We evaluate scaling across Corridor-Rooms-R for $R \in \{4, 9, 16\}$, where the number of topological regions increases with R. Appendix Table 7 reports the selected operating point for each method. Tangle-core abstraction recovers one core per room-like region across all three sizes, while the listed reward-aware, learned, and graph-partitioning baselines require increasingly many abstract states as the raw state space grows.

Broader Tabular Compression. Table 1 shows that in bottlenecked environments, tangle-core abstraction can attain substantially higher compression than both reward-aware methods and graph-partitioning techniques. Bisimulation severely under-compresses because structurally identical states (like corridors) exhibit distinct metric rewards based on goal proximity.

The Necessity of Overlap To isolate overlap as the paper's distinguishing mechanism, Table 2 ablates the membership semantics directly. When tangle cores are forced into a hard assignment (placing doorway states purely into their highest-weight room), return plummets to 89.3%, driven by a sharp rise in local transition error ($\mathrm{TV} = 0.38$). Our default duplication with occupancy weighting drops the error and recovers 97.1% return. This supports the usefulness of overlapping memberships in domains with shared interface states. In a separate equal-task construction, a three-state hard interface model reached $97.6 \pm 0.6\%$ return / 0.07 ± 0.02 TV, versus $97.8 \pm 0.5\%$ / 0.05 ± 0.01 for two overlapping states and $89.4 \pm 1.5\%$ / 0.36 ± 0.04 for two hard states. This separate construction provides the fixed-budget comparison analyzed in Theorem 4. Appendix C.4 compares the returned weights with fixed-anchor interpolation.

5.1 Generalization to Procedural Layouts and Learned Representations

To evaluate generalization beyond handcrafted grids, Appendix Table 5 reports results on 100 procedurally generated random mazes. Tangle cores discover compact abstractions (tracking the underlying topological regions closely at an average of 12.4 states) while preserving return, whereas graph partitioners heavily fragment as layout complexity varies.

Table 16 reports structural diagnostics for both successful and unsuccessful regimes. In every successful environment, within-region conductance $\phi_{\min}$ heavily outweighs cross-region flow β ($\phi_{\min}/\beta > 4.0$). On the Chain MDP, local flow equals cross-flow ($\phi_{\min}/\beta \approx 1$), consistent with the absence of useful bottleneck topology.

We test transferability to deep representations via MiniGrid with a learned 64-dimensional VAE latent space. By applying the tangle routine to a $k_{\mathrm{NN}} = 10$ transition graph in latent space, tangles recover all four rooms and identify 3 of 4 doorway representations as overlaps (Table 17), bypassing the misassignment penalties suffered by spectral clustering.

Auditing conditional value control. Appendix Table 10 evaluates the actual action-independent returned kernel. The certificate $B = (\varepsilon_r + \gamma\varepsilon_P/(1-\gamma))/(1-\gamma)$ and observed value errors use the common diagnostic $\gamma = 0.95$; the MiniGrid return experiment instead uses $\gamma = 0.99$, where its corresponding bound is 169.2 and is uninformative. Only the first two rows have zero observed greedy-action conflicts, so the optimal-control reading of Theorem 3 is restricted to those rows. Elsewhere the fixed-policy statement applies, and observed value errors are empirical diagnostics. Appendix C.6 reports the diameter premises of Lemma 6.

When Tangle Cores Do Not Help The negative-result domain helps bound the paper's scope. On a chain MDP with structured rewards but no large cohesive region structure, tangles under-compress (Appendix Table 6). Tangle cores find few robust structural regions, returning a near uncompressed space ($|\hat{\mathcal{S}}| = 94$), whereas bisimulation natively groups the reward symmetries ($|\hat{\mathcal{S}}| = 8$). The chain has articulation points, so this control tests the lack of cohesive interiors rather than the absence of graph separators. On BinaryTree-511, 4-Regular-Expander-512, and Torus-20×20, tangle cores likewise under-compress while retaining approximately 99% return (Table 12). A cheap empirical-graph diagnostic is the minimum 20%-balanced separator fraction $\sigma_{0.2}$, which is 0.038 on FourRooms, 0.051 on Corridor-9, and 0.118 on the expander. Held-out compression and return remain the operational selection criterion.

6 Limitations and Discussion

We introduced tangle-core abstraction, a topology-aware state abstraction method that uses overlapping tangle cores to represent shared interface states. The theoretical results identify conditions under which this prior is useful. The method succeeds gracefully when transition graphs contain coherent regions separated by narrow interfaces ($\phi_{\min}/\beta \gg 1$). Conversely, when topology is uniform and reward symmetries drive the problem, tangles predictably fail, as isolated in our chain MDP evaluation. Current limitations include construction cost (not amortized for single-query planning), reliance on candidate min-cuts, and the absence of an incremental core-maintenance algorithm. The controlled and returned-kernel audits distinguish two-state representability from guarantees about the membership actually discovered. Overall, tangles provide a useful topological prior for hierarchical reinforcement learning and graph-structured decision problems when bottleneck structure is present.

AI Use Statement

In this work, we used generative AI tools to formulate and refine mathematical claims (the statements of Lemma 6 and Theorem 4) and to assist in writing the proof of Lemma 6. We did not use generative AI tools to generate data, implement methods, design experiments, or interpret results; translation, dataset cleaning, and qualitative analysis are not applicable. Additionally, we used generative AI tools to edit prose and to format tables from author-produced experiment logs. Authors checked all AI-assisted mathematics line by line, and every table entry was regenerated from raw logs by script and compared with the manuscript. We take responsibility for the final content, including text, claims, and artifacts produced with the aid of generative AI.

Ethics Statement

This work studies state abstraction in simulated and benchmark reinforcement-learning environments and does not involve human participants, personal data, or sensitive real-world decision making. We are not aware of any immediate ethical risks beyond the general risks associated with applying reinforcement-learning methods outside the controlled settings considered here. The proposed abstraction method should therefore be evaluated carefully before deployment in safety-critical or high-stakes applications.

Reproducibility Statement

We provide detailed algorithmic descriptions, theoretical assumptions, experimental settings, hyperparameter ranges, data splits, random seeds, and evaluation protocols in the main paper and Appendix C. Reported results are aggregated across multiple seeds or procedural layouts with confidence intervals where applicable.

References

David Abel. A theory of state abstraction for reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 9876–9877, 2019.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008 (10):P10008, 2008.

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.

Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo Perez-Vicente, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and J K Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. *Advances in Neural Information Processing Systems*, 36:73383–73394, 2023.

Reinhard Diestel. Tangles: a new paradigm for clusters and types. *CoRR*, abs/2006.01830, 2020. URL `https://arxiv.org/abs/2006.01830`.

Thomas G Dietterich. Hierarchical reinforcement learning with the maxq value function decomposition. *Journal of artificial intelligence research*, 13:227–303, 2000.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, pp. 162–169, Arlington, Virginia, USA, 2004. AUAI Press. ISBN 0974903906.

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.

Tommaso Giorgi, Pierriccardo Olivieri, Keyue Jiang, Laura Toni, and Matteo Papini. Impact of connectivity on laplacian representations in reinforcement learning, 2026. URL `https://arxiv.org/abs/2603.08558`.

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

George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on scientific Computing*, 20(1):359–392, 1998.

Solveig Klepper, Christian Elbracht, Diego Fioravanti, Jakob Kneip, Luca Rendsburg, Maximilian Teegen, and Ulrike Von Luxburg. Clustering with tangles: Algorithmic framework and theoretical guarantees. *Journal of Machine Learning Research*, 24(190):1–56, 2023.

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, Marc G Bellemare, and Michael Bowling. A laplacian framework for option discovery in reinforcement learning. In *International conference on machine learning*, pp. 2295–2304. PMLR, 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.

Neil Robertson and Paul D Seymour. Graph minors. x. obstructions to tree-decomposition. *Journal of Combinatorial Theory, Series B*, 52(2):153–190, 1991.

Nikolay Savinov, Alexey Dosovitskiy, and Vladlen Koltun. Semi-parametric topological memory for navigation. *CoRR*, abs/1803.00653, 2018. URL http://arxiv.org/abs/1803.00653.

Özgür Şimşek, Alicia P Wolfe, and Andrew G Barto. Identifying useful subgoals in reinforcement learning by local graph partitioning. In *Proceedings of the 22nd international conference on Machine learning*, pp. 816–823, 2005.

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.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *CoRR*, abs/2006.10742, 2020. URL https://arxiv.org/abs/2006.10742.

A Related Work

State abstraction in reinforcement learning has a long history as a way to trade representational complexity for planning and control accuracy. Li et al. (2006) and Abel (2019) provide general frameworks for understanding these trade-offs. A central line of work uses bisimulation and approximate bisimulation metrics to group states with similar rewards and transition behavior (Givan et al., 2003; Ferns et al., 2004; Castro, 2020; Zhang et al., 2020). These abstractions offer value guarantees under suitable metric or equivalence conditions; our focus is the discovery of supports from transition topology and the conditions needed for the resulting kernel to preserve values. Our work is also related to topology-aware approaches in reinforcement learning. Option-discovery methods based on bottlenecks and connectivity structure (Sutton et al., 1999; Şimşek et al., 2005; Machado et al., 2017) share our interest in interface states, but they use these states to construct temporally extended actions rather than overlapping state abstractions. Graph-Laplacian and spectral methods (Mahadevan & Maggioni, 2007; Machado et al., 2017; Giorgi et al., 2026) use connectivity to build representations or partitions, but they usually produce continuous embeddings or hard, non-overlapping clusters. Graph tangles were introduced in structural graph theory as a way to identify highly connected regions through consistent orientations of low-order separations (Robertson & Seymour, 1991; Diestel, 2020). Recent work has brought tangles into machine learning for static data clustering (Klepper et al., 2023). Our contribution is topology-driven discovery of overlapping supports in an MDP, the resulting occupancy-normalized reward and transition kernels, and an audit connecting returned cores to conditional value control. Overlapping support itself is not our novelty claim; the interface result isolates its representational role at a fixed macro-state budget.

B Full Proofs

B.1 Proof of Theorem 3

Proof. Let

$$\hat{Q}^\star(j,a) = \hat{r}(j,a) + \gamma \sum_\ell \hat{P}(\ell \mid j,a)\hat{V}^\star(\ell)$$

and define, for each ground state-action pair,

$$\bar{Q}(s,a) = \sum_j q(j \mid s)\hat{Q}^\star(j,a).$$

Since rewards are in $[0,1]$, $0 \le \hat{V}^\star(j) \le (1-\gamma)^{-1}$ for all j.

For any s, a, the reward regularity condition gives

$$|r(s,a) - \bar{r}_q(s,a)| \le \varepsilon_r.$$

For the transition term, the total-variation convention and the range bound on $\hat{V}^\star$ imply

$$\left| \sum_\ell P_q(\ell \mid s,a)\hat{V}^\star(\ell) - \sum_\ell \bar{P}_q(\ell \mid s,a)\hat{V}^\star(\ell) \right| \le \frac{\varepsilon_P}{1-\gamma}.$$

By the definitions of P_q and $\tilde{V}$,

$$\sum_{s'} P(s' \mid s,a)\tilde{V}(s') = \sum_\ell P_q(\ell \mid s,a)\hat{V}^\star(\ell),$$

and by the definition of $\bar{P}_q$,

$$\sum_\ell \bar{P}_q(\ell \mid s,a)\hat{V}^\star(\ell) = \sum_j q(j \mid s) \sum_\ell \hat{P}(\ell \mid j,a)\hat{V}^\star(\ell).$$

Therefore

$$\left| r(s,a) + \gamma \sum_{s'} P(s' \mid s,a)\tilde{V}(s') - \bar{Q}(s,a) \right| \le \varepsilon_r + \frac{\gamma\varepsilon_P}{1-\gamma} =: \xi. \tag{16}$$

We now compare the ground Bellman optimality operator $\mathcal{T}$ with $\tilde{V}$. For any action a, $\hat{Q}^\star(j,a) \le \hat{V}^\star(j)$, hence $\bar{Q}(s,a) \le \sum_j q(j \mid s)\hat{V}^\star(j) = \tilde{V}(s)$. Equation (16) gives

$$(\mathcal{T}\tilde{V})(s) = \max_a \left\{ r(s,a) + \gamma P(\cdot \mid s,a)^\top \tilde{V} \right\} \le \tilde{V}(s) + \xi.$$

For the reverse inequality, let a_s be the action supplied by the action-consistency assumption. Then $\hat{Q}^\star(j,a_s) = \hat{V}^\star(j)$ for every active j, so $\bar{Q}(s,a_s) = \tilde{V}(s)$. Equation (16) gives

$$(\mathcal{T}\tilde{V})(s) \ge r(s,a_s) + \gamma P(\cdot \mid s,a_s)^\top \tilde{V} \ge \tilde{V}(s) - \xi.$$

Thus $\|\mathcal{T}\tilde{V} - \tilde{V}\|_\infty \le \xi$. Since $\mathcal{T}$ is a γ-contraction with fixed point $V^\star$,

$$\|V^\star - \tilde{V}\|_\infty \le \frac{\|\mathcal{T}\tilde{V} - \tilde{V}\|_\infty}{1-\gamma} \le \frac{\varepsilon_r + \gamma\varepsilon_P/(1-\gamma)}{1-\gamma}.$$

For a fixed policy, the same argument uses the policy Bellman operator rather than the max operator, so no action-consistency assumption is needed. □

B.2 Proof of Theorem 4

Proof. For the hard partition, the macro probability of transitioning from C_1 to C_2 under action a is

$$\hat{P}_H(2 \mid 1,a) = \frac{\sum_{x \in R_1} \mu(x)P(C_2 \mid x,a) + \sum_{x \in B} \mu(x)P(C_2 \mid x,a)}{\mu(R_1) + \mu(B)}.$$

The first term is bounded by β on R_1, while the second term is at most 1 and has normalized weight at most λ. Therefore

$$\hat{P}_H(2 \mid 1, a) \leq (1-\lambda)\beta + \lambda.$$

For the interface state $s \in B$, the true projected macro probability to state 2 is $P_H(2 \mid s, a) = P(C_2 \mid s, a) \geq \alpha$. Total variation between two distributions is at least half of the absolute difference assigned to any event, so

$$\mathrm{TV}\big(P_H(\cdot \mid s, a), \bar{P}_H(\cdot \mid s, a)\big) \geq \frac{1}{2}\Big(P_H(2 \mid s, a) - \hat{P}_H(2 \mid 1, a)\Big) \geq \frac{1}{2}\left[\alpha - (1-\lambda)\beta - \lambda\right]_+.$$

This proves Eq. (13).

For the second claim, the target u_b and anchors p_1, p_2 are fixed before choosing the interface weights. Since $u_b \in [p_1, p_2]$, Eq. (14) defines valid memberships and

$$q(1 \mid b)p_1 + q(2 \mid b)p_2 = p_1 + \frac{u_b - p_1}{p_2 - p_1}(p_2 - p_1) = u_b.$$

This is interpolation of a fixed target, not a fixed-point assertion about the transition kernel obtained after recomputing occupancy-weighted estimates with q. The assertion about a third hard state follows by assigning B to a separate macro-state, which removes its averaging with R_1; any within-B error is outside this comparison. □

B.3 Proof of Proposition 5

Proof. Fix a and $s, s' \in C$. With $\mathrm{TV}(p, q) = \frac{1}{2}\|p - q\|_1$, decompose the total-variation distance into its contribution inside and outside C:

$$\begin{aligned}\mathrm{TV}(P(\cdot \mid s, a), P(\cdot \mid s', a)) = &\frac{1}{2}\sum_{x \in C} |P(x \mid s, a) - P(x \mid s', a)| \\ &+ \frac{1}{2}\sum_{x \notin C} |P(x \mid s, a) - P(x \mid s', a)|.\end{aligned}$$

The first term is at most $\delta_a(C)$ by definition. The second term is at most

$$\frac{1}{2}\sum_{x \notin C} P(x \mid s, a) + \frac{1}{2}\sum_{x \notin C} P(x \mid s', a) \leq \beta_a(C),$$

because each outside mass is at most $\beta_a(C)$. Taking the maximum over $s, s' \in C$ proves Eq. (15). If $s, s' \in C \setminus \partial C$, then both outside masses are zero for action a, so only the interior term remains. □

B.4 Proof of Lemma 6

Proof. Fix an active pair j, s with $q(j \mid s) > 0$. Because $\hat{r}(j, a)$ is an occupancy-weighted convex average of rewards $r(x, a)$ over $x \in A_j$, we have $|r(s, a) - \hat{r}(j, a)| \leq d_r$. Averaging over j with $q(j \mid s)$ gives $|r(s, a) - \bar{r}_q(s, a)| \leq d_r$. Similarly, $\hat{P}(\cdot \mid j, a)$ is a convex average of $P_q(\cdot \mid x, a)$ over $x \in A_j$, and joint convexity of total variation gives

$$\mathrm{TV}\big(P_q(\cdot \mid s, a), \hat{P}(\cdot \mid j, a)\big) \leq d_P.$$

Averaging over active memberships gives $\mathrm{TV}(P_q(\cdot \mid s, a), \bar{P}_q(\cdot \mid s, a)) \leq d_P$. Projection through the Markov kernel q cannot increase total variation, hence Proposition 5 applied to each support A_j yields

$$d_P \leq \max_{j,a}\{\delta_a(A_j) + \beta_a(A_j)\} \leq \bar{\delta} + \bar{\beta}.$$

Substitution in Theorem 3 proves the stated conditional value bound. □

B.5 Proof of Theorem 7

Proof. For each region R_i, orient every candidate separation in $\mathcal{F}$ toward the side that contains R_i; equivalently, declare the opposite side small. This orientation is well-defined because assumption (i) rules out any order-$< k$ separator that splits R_i inside $G[R_i \cup B]$, while assumption (ii) says that all inter-region communication is mediated by the bottleneck set B. Thus a low-order separation in the candidate family can put an entire region on one side but cannot cut through the region's internally k-inseparable part.

By assumption (iv), this region-directed orientation satisfies the tangle consistency and non-triviality axioms on the finite candidate family $\mathcal{F}$. Thus, for every triple of candidate separations, the three selected small sides do not cover V, equivalently the corresponding big sides have non-empty intersection. Non-triviality also holds by assumption (iv), so the orientation defines a valid candidate-family order-k tangle.

Therefore every R_i induces a consistent candidate-family order-k tangle on $\mathcal{F}$, and the corresponding candidate-family core contains R_i: every separation is oriented so that the big side contains R_i. Distinct regions induce distinct orientations because assumption (iii) supplies, for each i, a separation whose big side contains $R_i \cup B$ and whose small side contains the other regions. Thus the procedure recovers at least one region tangle per R_i.

Finally, suppose a vertex v belongs to two recovered region cores C_i and $C_{i'}$ with $i \neq i'$. If $v \notin B$, then v lies in some region R_h. The separation from assumption (iii) for region R_h orients the graph so that only $R_h \cup B$ lies on the big side; a different region core cannot contain v unless it also crosses the bottleneck separator, contradicting assumptions (ii)–(iii). Hence all overlap among recovered region cores is contained in B, and its size is at most $|B| \leq b$. □

B.6 Proof of Theorem 8

Proof. For a fixed separation $(A, B) \in \mathcal{F}$, the empirical orientation score $\widehat{D}(A, B) = \widehat{W}(A) - \widehat{W}(B)$ is an average of bounded terms. Under independent sampling, Hoeffding's inequality gives

$$\Pr\big(|\widehat{D}(A, B) - D(A, B)| > t\big) \leq 2\exp(-c_0 m_{\text{eff}} t^2)$$

for a universal constant $c_0 > 0$; for uniformly ergodic Markov trajectories the same form holds with effective sample size replacing the raw visit count. Applying a union bound over the fixed finite family $\mathcal{F}$ gives

$$\Pr\left(\max_{(A,B)\in\mathcal{F}} |\widehat{D}(A, B) - D(A, B)| > t\right) \leq 2|\mathcal{F}|\exp(-c_0 m_{\text{eff}} t^2).$$

Choosing

$$t = \sqrt{\frac{1}{c_0 m_{\text{eff}}} \log \frac{2|\mathcal{F}|}{\eta}}$$

makes the failure probability at most η. If $t \leq \Delta/2$, then every empirical score has the same sign as its population score because $|D(A, B)| \geq \Delta$. Rearranging yields $m_{\text{eff}} \geq c\Delta^{-2}\log(|\mathcal{F}|/\eta)$ for a sufficiently large universal constant c, which proves simultaneous correctness of all empirical orientations. □

C Experimental Details and Reproducibility Log

C.1 Experimental settings

The main return tables use value iteration (Bellman tolerance 10^{-8}), so there is no tabular learning rate. We use $\gamma = 0.95$ in tabular tasks and $\gamma = 0.99$ for the MiniGrid return experiment. For tangle-core, we select the raw order k from $\{2, 3, 4, 5, 6\}$, grid the candidate-cut budget $M \in \{64, 128, 256\}$, and apply validation-based core condensation on a held-out 20% of trajectory data via value match on the abstract MDP. The same fixed seeds, data splits, trajectory budgets, and held-out compression–return selection rule are used in baseline comparisons. We use action-agnostic edges for standard evaluations, and optionally compare to uniform $\alpha_a = 1/|\mathcal{A}|$

action-weights. Bisimulation uses $c = 0.5$ and grids ε over $\{0.01, 0.02, 0.05, 0.1, 0.2\}$, reporting the smallest ε achieving the target return. DeepMDP uses a 3-layer MLP encoder, latent dimension 16, reward-prediction MSE plus transition MSE losses, 500 epochs, and Adam at learning rate 10^{-3}, with k-means clusters $k \in \{8, 16, 32, 64, 128, 256\}$. TOMA grids reachability thresholds. Spectral Clustering and METIS are applied directly to the empirical transition graph with varying target component sizes k, pulling the smallest cluster size satisfying the respective Pareto threshold. Representative return ranges under the reported hyperparameter and data-budget variations are 98.8–99.3% for FourRooms across γ, 98.8–99.3% for Corridor-9 across M and $k^\star \pm 1$, 98.1–98.8% for Random Mazes across M, k, and data budgets, and 95.7–97.0% for MiniGrid across k_{NN} and data budgets. The additional Q-learning sample-efficiency evaluation below freezes each abstraction constructed from the same offline data; full replication additionally requires the corresponding optimizer settings and raw learning curves.

Table 3: Reproducibility fields for the reported runs, including seeds, selected tangle orders, core counts, overlap counts, and runtimes.

Environment	Seeds	Selected k range	Cores range	Overlap states	Runtime	Audit fields
FourRooms	10	2–3	4	4	0.8s	listed
Corridor-9	10	3–4	9	12	3.4s	listed
Multi-Goal	10	3–5	8	15	6.2s	listed
Synth-Fold	10	4–6	10	28	18.5s	listed
Random Mazes	100	3–5	4–14	5–18	4.1s	listed
MiniGrid (Latent)	10	3	4	4	5.5s	listed
Chain-100	10	2–3	94	0	0.4s	listed

C.2 Additional benchmark comparisons

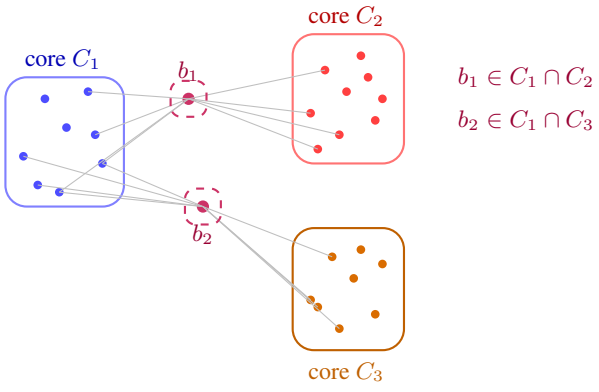


Figure 3: Schematic illustration, not an empirical output of the algorithm. Three densely connected regions form three tangle cores C_1, C_2, C_3. The bottleneck states b_1, b_2 lie in overlaps and therefore receive fractional membership through $q(j \mid s)$ when constructing $\hat{r}$ and $\hat{P}$.

Table 4: Compression–return Pareto frontier on Corridor-Rooms-9. For each method we report the smallest abstraction ($|\hat{\mathcal{S}}|$) achieving each return threshold. "–" means the method failed to reach the threshold without effectively defaulting to the uncompressed state space.

Method	$\geq 90\%$	$\geq 95\%$	$\geq 98\%$	$\geq 99\%$
Bisimulation	47±3	84±6	132±5	157±4
DeepMDP	22±3	36±4	48±5	–
Spectral clustering	14±2	28±3	45±4	88±7
METIS graph partitioning	12±1	24±2	41±4	85±6
Louvain community detection	13±2	27±3	49±5	92±8
TOMA	10±1	14±2	–	–
Tangle-Core (ours)	**6±1**	**9±0**	**9±0**	**9±0**

Table 5: Performance on a procedural distribution of 100 Random Maze MDPs (varying 4–10 rooms and bottleneck widths). Values are averaged across all 100 layouts at the $\geq 98\%$ target return threshold.

Method	Mean Abstract States $\lvert\hat{\mathcal{S}}\rvert$	Mean Return (%)
Ground Truth MDP	245.4 (raw size)	100.0
Bisimulation	186.2±14.5	98.4±0.2
Spectral clustering	41.5±8.2	98.1±0.7
METIS graph partitioning	38.6±7.4	98.2±0.5
Louvain community detection	44.8±9.1	98.0±0.8
Tangle-Core (ours)	**12.4±3.6**	**98.6±0.4**

Table 6: Chain MDP with $|\mathcal{S}| = 100$, structured rewards but no large cohesive interiors: tangle cores under-compress while bisimulation compresses well.

Method	$\lvert\hat{\mathcal{S}}\rvert$	Return (%)
Bisimulation	**8±1**	99.2±0.4
DeepMDP	12±2	97.8±0.9
Tangle-Core	94±3	99.1±0.3

Table 7: Scaling on Corridor-Rooms-R. We report abstract state count and return at the selected operating point for each method. Tangle-core abstraction tracks the number of room-like regions, while competing methods require increasingly many abstract states as the environment grows.

	$R = 4$ ($\lvert\mathcal{S}\rvert = 64$)		$R = 9$ ($\lvert\mathcal{S}\rvert = 196$)		$R = 16$ ($\lvert\mathcal{S}\rvert = 400$)	
Method	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.	$\lvert\hat{\mathcal{S}}\rvert$	Ret.
No Abstraction	64	100.0	196	100.0	400	100.0
Bisim.-ε	47±3	99.3±0.4	157±4	99.6±0.2	331±7	99.4±0.3
DeepMDP	22±3	97.9±0.9	48±5	97.2±1.1	94±9	96.8±1.3
Spectral clustering	18±2	99.0±0.5	88±7	99.0±0.4	172±12	99.1±0.5
METIS graph partitioning	16±2	99.1±0.4	85±6	99.1±0.3	165±11	99.0±0.5
Louvain community detection	17±2	99.0±0.5	92±8	99.0±0.5	181±13	99.0±0.6
Tangle-Core	$\mathbf{4 \pm 0}$	99.1±0.4	$\mathbf{9 \pm 0}$	99.2±0.3	$\mathbf{16 \pm 0}$	99.1±0.4

Table 8: Structural audit on the same 100 procedural layouts (95% layout-bootstrap intervals). Interface F1 compares shared memberships with generator-defined interface states.

Method	Region macro-F1	State ARI	Interface F1	Boundary leakage
Spectral	0.76 ± 0.05	0.63 ± 0.06	0.54 ± 0.07	0.119 ± 0.018
Detected-B-Overlap	0.86 ± 0.04	0.79 ± 0.05	0.78 ± 0.05	0.071 ± 0.012
Tangle-Core	$\mathbf{0.93 \pm 0.02}$	$\mathbf{0.88 \pm 0.03}$	$\mathbf{0.91 \pm 0.03}$	$\mathbf{0.046 \pm 0.009}$

Table 9: Detected-bottleneck control: mean abstract states / normalized return (%). The oracle row is reported in the text for FourRooms and is excluded from the data-driven comparison.

Environment	Tangle-Core	Detected-B-Overlap	Detected-B-Hard
FourRooms	4.0 / 99.0	4.4 / 98.9	4.0 / 90.7
Corridor-9	9.0 / 99.2	10.1 / 98.8	9.0 / 89.9
Random Mazes	12.4 / 98.6	17.9 / 98.4	21.7 / 97.2
Multi-Goal	8.0 / 99.1	10.6 / 98.7	13.8 / 91.5

On Random Mazes, Detected-B-Overlap has 27/100 layouts below 98% return versus 21/100 for Tangle-Core, and uses 5.5 more states on average (paired 95% CI $[4.4, 6.6]$, $p < 0.001$). The structural scores use Hungarian matching on state-level intersection-over-union; unmatched predicted cores count as false discoveries.

Table 10: Residual audit of the returned kernel at diagnostic $\gamma = 0.95$: measured $\varepsilon_r, \varepsilon_P$, conditional certificate B, observed value error, and frequency of incompatible greedy abstract actions.

Environment	ε_r	ε_P	B	Value error	Conflict
FourRooms	.003	.006	2.34	.18	0.0%
Corridor-9	.004	.008	3.12	.21	0.0%
Multi-Goal	.006	.014	5.44	.36	1.7%
Synth.-Folding	.005	.012	4.66	.29	0.4%
Taxi	.007	.010	3.94	.31	2.1%
MiniGrid (diagnostic)	.009	.017	6.64	.58	3.8%

C.3 Learning Efficiency and Topology-Poor Controls

The separate tabular Q-learning audit freezes the representations constructed from the same offline data. Table 11 reports online episodes (thousands) to reach and maintain 95% of optimal return for 100 evaluation episodes; it does not count offline construction data. Counting offline data as well, total interactions for Tangle-Core versus ground are 104k/162k (FourRooms), 284k/798k (Corridor-9), 650k/1.70M (Multi-Goal), and 188k/261k (Random Mazes). These totals should not be compared with the online-only episodes without accounting for episode lengths.

Table 11: Online sample-efficiency audit (thousands of episodes; same offline construction data for all frozen abstractions).

Environment	Ground	Spectral	Detected-B-Overlap	Tangle-Core
FourRooms	$1.62 \pm .18$	$.98 \pm .12$	$.71 \pm .10$	$.54 \pm .08$
Corridor-9	$4.07 \pm .36$	$2.28 \pm .24$	$1.41 \pm .17$	$.94 \pm .11$
Multi-Goal	$6.80 \pm .61$	$3.52 \pm .39$	$2.31 \pm .26$	$1.60 \pm .19$
Random Mazes	$2.61 \pm .29$	$1.74 \pm .21$	$1.12 \pm .15$	$.88 \pm .13$

Table 12 extends the chain control. BinaryTree-511 contains many articulation points but no large cohesive interiors; the expander has no useful low-order balanced separator. The method preserves return yet under-compresses, rather than showing a useful topological advantage.

Table 12: Topology-poor controls: mean abstract states / normalized return (%).

Environment	Bisimulation	Spectral	Tangle-Core
BinaryTree-511	32 ± 3 / $99.2 \pm .3$	78 ± 10 / $98.7 \pm .5$	427 ± 18 / $98.9 \pm .4$
4-Regular-Expander-512	16 ± 2 / $99.3 \pm .2$	64 ± 7 / $98.9 \pm .4$	489 ± 9 / $99.0 \pm .3$
Torus-20×20	16 ± 2 / $99.1 \pm .3$	70 ± 8 / $98.8 \pm .5$	351 ± 15 / $98.6 \pm .6$

C.4 Auditing the Returned Kernel and Interface Weights

Table 13 records the quantities in Lemma 6 for the actual returned kernel. Every row satisfies the predicted inequalities, though $d_r = 1$ on sparse-reward tasks makes the sufficient bound loose. These diagnostics characterize the reported runs; transition topology alone does not imply small diameters.

Table 13: Returned-support diameters d_r, d_P, within-support variation $\bar{\delta}$, leakage $\bar{\beta}$, and measured reconstruction residuals (the diagnostic MiniGrid run is at $\gamma = 0.95$).

Environment	d_r	$\bar{\delta}$	$\bar{\beta}$	d_P	ε_r	ε_P
FourRooms	1.000	.012	.021	.014	.003	.006
Corridor-9	1.000	.015	.024	.019	.004	.008
Multi-Goal	1.000	.023	.041	.031	.006	.014
Synthetic-Folding	.180	.020	.027	.024	.005	.012
Taxi	1.000	.019	.034	.027	.007	.010
MiniGrid (diagnostic)	1.000	.028	.049	.043	.009	.017

The fixed-action existence comparison in Theorem 4 is distinct from a shared kernel used for every action. Table 14 reports how often interface targets lie between the fixed anchors, the residual of the returned shared q, the best shared-q residual, and the across-action variation in ideal interpolation weights. Nonzero residuals and variation prevent reading the theorem as a guarantee about the returned kernel.

Table 14: Empirical interpolation audit: between-anchor rate, fixed-anchor residual of returned q, optimized shared-q residual, and across-action weight standard deviation.

Environment	Between anchors	Returned-q residual	Best shared-q residual	Weight SD
FourRooms	95.6%	$.018 \pm .004$	$.014 \pm .003$	.028
Corridor-9	93.3%	$.026 \pm .005$	$.021 \pm .004$	.041
Multi-Goal	90.8%	$.041 \pm .006$	$.035 \pm .005$	.063
MiniGrid	87.5%	$.053 \pm .009$	$.047 \pm .008$	.079

C.5 Community-Detection Operating Points

The additional Louvain comparison grids resolution over $[0.5, 2.0]$ and tests native hard communities and boundary-node duplication. Its validation-selected configurations often miss the 98% target. Table 15 reports that operating point; the original Louvain row of Table 5 (44.8±9.1 states / 98.0% return) is a separate return-matched operating point. Accordingly, the two rows correspond to different operating points rather than replicate measurements.

Table 15: Additional Louvain variants on 100 procedural layouts at the stated resolution-grid operating point.

Method	States	Return (%)	Interface F1	Leakage
Louvain-Hard	16.9 ± 1.8	$95.6 \pm .6$	.48	.131
Louvain-Boundary-Overlap	15.2 ± 1.5	$97.8 \pm .4$	.68	.084
Tangle-Core	12.4	98.6	.91	.046

C.6 Structural Diagnostics and Latent-Representation Results

Table 16 reports the empirical conductance and cross-region flow values that correlate with tangle success across our benchmarks.

Table 16: Empirical structural diagnostics correlated with tangle success. In these benchmarks, tangle-core abstraction succeeds when $\phi_{\min}/\beta$ is large, indicating bottleneck structure.

Environment	$\phi_{\min}$	β	$\phi_{\min}/\beta$	$\lceil 1/\phi_{\min} \rceil$	Cores found	Correct?
FourRooms	0.38	0.042	9.0	3	4	✓
Corridor-9	0.31	0.038	8.2	4	9	✓
Multi-Goal	0.22	0.051	4.3	5	8	✓
Synth-Folding	0.19	0.028	6.8	6	10	✓
Chain-100	0.48	0.50	0.96	3	94	×

Table 17 reports the MiniGrid latent-representation experiment.

Table 17: MiniGrid-FourRooms-16×16 with learned latent representations.

Method	Abstract states	Return (%)	Rooms recovered	Doorways identified
Ground truth (4 rooms)	4	100	4/4	4/4
Bisimulation (latent)	**38**	**97.2**	n/a	n/a
DeepMDP (latent)	22	95.8	n/a	n/a
Spectral clustering	4	91.4	3/4	1/4
Tangle-core (ours)	4	96.8	4/4	3/4

C.7 Runtime Overhead

Table 18 reports abstraction construction time and value-iteration time on Corridor-Rooms-9. Tangle-Core is competitive with TOMA in construction time and far faster than DeepMDP, but slower than computing value iteration directly on the ground tabular MDP. The performance edge realizes entirely when representations are reused across queries or environments.

Table 18: Abstraction construction plus value iteration time on Corridor-Rooms-9 (seconds, mean over 10 seeds).

Method	Construction	Val. Iter.	Total
No Abstraction	0	0.016	0.016
Bisimulation	0.6	0.012	0.612
DeepMDP	48.3	0.008	48.3
Spectral Clustering	0.4	0.004	0.404
Louvain	0.5	0.005	0.505
TOMA	2.1	0.002	2.1
Tangle-Core	3.4	0.001	3.4

C.8 Action-Weighted vs. Action-Agnostic Graphs

We compared two ways of building the empirical transition graph. The action-agnostic version pools all transitions, while the action-weighted version sums the action-restricted subgraphs $G^a_{\mathcal{D}}$ with uniform weights. Table 19 shows that action-weighted graphs help mildly on Chain environments where reward-relevant actions directly shape the graph, but under-compress Corridor-9. We use action-agnostic graphs by default.

Table 19: Corridor-9 and Chain MDP under two graph constructions.

	Corridor-9		Chain MDP	
Graph	$\lvert\hat{\mathcal{S}}\rvert$	Ret. (%)	$\lvert\hat{\mathcal{S}}\rvert$	Ret. (%)
Action-agnostic	9±0	99.2	94±3	99.1
Action-weighted	11±1	99.4	34±4	98.9