Hodge Laplacian Quasi-Harmonic Flows for Option Discovery

Andreas Theophilou
University of Bath
United Kingdom
ajt80@bath.ac.uk

Özgür Şimşek
University of Bath
United Kingdom
o.simsek@bath.ac.uk

Abstract

A central challenge in reinforcement learning is discovering meaningful skills that improve exploration and learning. Prior Laplacian-based option discovery methods build options from 0-form Laplacians defined on nodes [20, 12, 19]. In contrast, 1-form Laplacians defined on edges, particularly the Hodge Laplacian [3, 6, 17], represent interpretable, topology-aware edge flows and naturally capture cyclic behaviour through kernel eigenvectors known as harmonic flows. Despite this appeal, harmonic flows have not previously been explored as a basis for skill discovery in reinforcement learning. This is what we do here. We first introduce harmonic-flow skills and show that, on typical state graphs, purely harmonic flows can become nearly inactive in tree-like "dead-end" appendages, which can limit coverage. To address this, we propose quasi-harmonic flow skills: skill-inducing edge flows that preserve harmonic cycles where harmonic activity is strong, while injecting drift into appendages using the lowest-magnitude non-harmonic eigenvector of the Hodge Laplacian where harmonic activity is weak. Our approach is eigenvector-only, relying solely on spectral components of the Hodge Laplacian. We compare quasi-harmonic flow skills against a harmonic-flow skills baseline as well as the related Laplacian-based option discovery methods from the literature. We demonstrate empirical gains and analytical improvements on hitting-time based metrics.

1 Introduction

Reinforcement learning [32] studies how an autonomous agent can, from its own experience, learn how to act in order to maximize long-term accumulation of rewards. Here we address the case where the agent knows how states are connected to each other but not where the goal or rewards are. This case motivates a coverage question: how can the agent construct reusable behaviours that allow it to navigate a known space efficiently, so that it can efficiently discover where the rewards are – in other words, a set of generally useful behaviours that are task-agnostic. This setting has been considered earlier in closely related graph-spectral option work, including diffusion options [2]; it is not the same problem as frontier exploration in an unknown environment. A useful illustrative example is an agent that already knows the layout of a house but does not know where the keys have been placed; and the relevant question is how to cover this known house efficiently.

A standard way to improve exploration is to augment primitive actions with temporally extended behaviours, also called skills, which can be represented as *options* [33, 25]. Well-chosen options can reduce effective planning depth, improve credit assignment, and change the visitation distribution in ways that accelerate learning. A particularly attractive direction is to discover options directly from the *structure* of the environment. Laplacian eigenvectors have therefore become a useful and promising direction for representation learning and option discovery [30, 21, 22, 24, 19, 20, 12, 2].

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Existing Laplacian option methods use eigenvectors of a 0-form Laplacian, known as the *Graph Laplacian,* where eigenvectors are mutually orthogonal, form a basis for functions on the vertices, and are ordered by their smoothness.

In practice, these methods construct hill-climbing eigenvector-based gradients to form policies. The resulting policies flow toward individual extrema, isolated regions of the state space, and boundary states. This behaviour can be useful for exploration because it helps the agent move toward far-away regions, around blocks, or through bottlenecks that random walks of primitive actions may rarely reach. However, it has clear limitations in pushing the agent to only certain regions. Emphasizing an agent's transitions toward one region necessarily underemphasizes its transitions toward other regions. Visitation probability mass cannot increase everywhere at once. As a result, options have the potential improve access to a small number of boundary or terminal regions while making other states harder to hit, even worse than when following a random walk.

Here we investigate a different Laplacian construction: the 1-form Hodge Laplacian . Unlike the 0-form Laplacian, which acts on nodes, the 1-form Hodge Laplacian acts directly on edge flows. Its energy decomposes into divergence and curl terms, so its kernel consists of flows that are simultaneously divergence-free and curl-free. These kernel eigenvectors, called *harmonic flows*, do not create gradients into any single isolated state or region. Instead, they capture global circulation around topological cycles, making the resulting flows topology-aware, highly interpretable and desirable for coverage because a circulating flow does not concentrate flow toward any specific state.

Pure harmonic flows are not sufficient on their own. They are active around topological cycles, but they can be nearly inactive in dead-end regions, cul-de-sacs, or tree-like appendages attached to a cyclic core. This limitation is natural: entering a dead end requires some amount of divergence, because flow must increase toward a terminal or boundary-like part of the graph. We therefore introduce *quasi-harmonic flow options* (QHFO), which combine harmonic kernel eigenvectors with the non-harmonic eigenvector associated with the lowest eigenvalue of the 1-form Hodge Laplacian. The non-harmonic component supplies the smallest available divergence-and-curl solution outside the harmonic subspace, and we inject it only where harmonic activity is weak.

Our contributions are as follows.

1. We identify why policies obtained from the eigenvectors of a 0-form Laplacian can concentration visitation unevenly, and why harmonic flows avoid this in harmonic regions via zero divergence and local curl.
2. We formulate harmonic flow options from the kernel of the 1-form Hodge Laplacian, giving a topology-aware alternative to gradient-based 0-form Laplacian options.
3. We identify why pure harmonic flows fail to cover dead-end appendages.
4. We introduce QHFO, a one-shot eigenvector blend that adds controlled non-harmonic drift where harmonic activity is weak. We evaluate QHFO against related laplacian-based option methds across large structured domains and against analytical hitting-time metrics.

2 Preliminaries

A finite Markov decision process (MDP) is a 5-tuple $(\mathcal{S}, \mathcal{A}, P, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is a finite set of states, $\mathcal{A}$ a finite set of actions, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ the transition matrix (indicating the probability of transitioning to state s' upon taking action a in state s), $\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ the reward function (indicating the expected reward when transitioning to state s' upon taking action a in state s), and $\gamma \in (0, 1)$ the discount factor. At decision stage $t \geq 0$, the agent observes state $s_t \in \mathcal{S}$, executes action $a_t \in \mathcal{A}$, then transitions into state $s_{t+1} \in \mathcal{S}$, receiving reward $r_{t+1} \in \mathbb{R}$. The *return* at decision stage t is the discounted sum of future rewards, $\sum_{i \geq 0} \gamma^i r_{t+i}$. The objective is to learn a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ that maximizes expected return. The *stationary distribution* of an MDP under policy π is a probability distribution over states that remains unchanged as decision stages progress, satisfying $\sigma = \sigma P^\pi$, where P^π is the transition matrix for policy π. The *expected hitting time*, is the expected number of steps that a stochastic Markov chain starting at state i arrives at state j for the first time, $h_{ij} = \mathbb{E}[min\{t \geq 0 | X_t = j,\ X_0 = i\}]$, where $\mathbb{E}[.]$ denotes expectation.

Using the MDP actions $\mathcal{A}$, also called the *primitive actions*, executed in a single decision stage, one can define *temporally-extended actions* using the options framework [33, 25]. An *option* o is a triple

$(\mathcal{I}_o, \pi_o, \beta_o)$, where $\mathcal{I}_o \subseteq \mathcal{S}$ is the initiation set, denoting the set of states in which the option can start execution; $\pi_o : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the option policy, specifying a probability distribution over the primitive actions for each state; and $\beta_o : \mathcal{S} \to [0, 1]$ is the termination condition, specifying the probability with which the option terminates at each state.

Associated with MDP $(\mathcal{S}, \mathcal{A}, P, \mathcal{R}, \gamma)$, we define a *state transition graph* (V, E), with vertices corresponding to the states of the MDP and edges corresponding to possible transitions between states through primitive actions: $V = \mathcal{S}$, $E = \{(s \to s') \mid s \in \mathcal{S},\ s' \in \mathcal{S},\ \exists a\ \mathcal{P}(s, a, s') > 0\}$.

The *incidence matrix* of directed graph (V, E) is a matrix of size $|V| \times |E|$, where the entry in row v and column e is: -1 if edge e is outgoing from vertex v, $+1$ if edge e is incoming into vertex v, and 0 if edge e is not attached to vertex v. The *orientation* of an undirected graph is an assignment of a direction to each edge, turning the initial undirected graph into a directed graph. The *oriented incidence matrix* of an undirected graph is the incidence matrix of any orientation of the graph.

A *0-form* on a graph is a function that assigns a scalar value to each vertex of the graph. Given incidence matrix $\mathbf{B}_1$ and 0-form ϕ, the product $\mathbf{B}_1^\top \phi \in \mathbb{R}^{|E|}$ assigns one scalar to each edge. For edge $a \to b$, that scalar is $\phi(b) - \phi(a)$. Thus $\mathbf{B}_1^\top \phi$ measures how much ϕ changes across each edge and is the discrete *gradient* of ϕ.

0-form graph Laplacian $\mathbf{L}_0$ is an operator acting on functions defined on the vertices of a graph. For an undirected graph, $\mathbf{L}_0$ can be computed via the oriented incidence matrix $\mathbf{B}_1$: $\mathbf{L}_0 = \mathbf{B}_1 \mathbf{B}_1^\top$. It maps node signals to node signals, with $(\mathbf{L}_0\phi)(v) = \sum_{u \sim v}(\phi(v) - \phi(u))$, where $u \sim v$ means that u is adjacent to v. Hence $\mathbf{L}_0\phi$ gives a local disagreement value at each node. The corresponding scalar smoothness energy is $\phi^\top \mathbf{L}_0 \phi = \|\mathbf{B}_1^\top \phi\|_2^2 = \sum_{\forall (a \to b)}(\phi(b) - \phi(a))^2$, which is small when neighboring states have similar values.

A *1-form* on a graph is a function that assigns a scalar value to each oriented edge. We call it an *edge flow.* For edge $a \to b$, the flow points from a to b if the scalar is positive, from b to a if the scalar is negative. Given incidence matrix $\mathbf{B}_1$ and 1-form $\mathbf{x}$, the product $\mathbf{B}_1\mathbf{x} \in \mathbb{R}^{|V|}$ gives the signed node imbalance, the difference between the incoming and outgoing flow to the node. With our sign convention, this is net inflow. We call this signed node imbalance the *discrete divergence*. A flow is divergence-free when $\mathbf{B}\mathbf{x} = 0$, that is, if incoming and outgoing flow balances at every node.

A *filled face* is an enclosed region of a graph bounded by edges and vertices. Each face is given an arbitrary fixed boundary orientation, clockwise or counterclockwise. Given a set of oriented faces F and a set of oriented edges E, let matrix $\mathbf{B}_2$, with $|E|$ rows and $|F|$ columns, record oriented face boundaries such that the matrix entry for edge e and face f is: $+1$ if the orientations of edge e and face f agree, -1 if they oppose, and 0 if e is not part of the boundary of f. Given 1-form $\mathbf{x}$, $\mathbf{B}_2^\top \mathbf{x} \in \mathbb{R}^{|F|}$ gives the *discrete curl,* or signed circulation, around each filled face. The curl for face f is the sum of signed flow on its boundary edges. Its magnitude measures the strength of local rotation around f; its sign gives the rotation direction relative to the chosen boundary orientation.

The 1-form graph Laplacian, $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top$, known as the Hodge 1-Laplacian, operates on functions defined on edges, capturing structural properties related to cycles and flows. Its scalar energy is $\mathbf{x}^\top \mathbf{L}_1 \mathbf{x} = \|\mathbf{B}_1\mathbf{x}\|_2^2 + \|\mathbf{B}_2^\top \mathbf{x}\|_2^2$, so $\mathbf{L}_1$ measures squared divergence plus squared curl. The kernel (nullspace) of $\mathbf{L}_1$, $ker(\mathbf{L}_1) = \{\mathbf{x} : \mathbf{L}_1\mathbf{x} = 0\}$, therefore contains exactly the flows satisfying $\mathbf{B}_1\mathbf{x} = 0$ and $\mathbf{B}_2^\top \mathbf{x} = 0$. These zero-divergence, zero-curl edge flows are harmonic flows. They have no local sources or sinks and no local rotation around filled faces, but they still circulate around unfilled holes. Notably, the dimension of the kernel of $\mathbf{L}_1$ equals the first Betti number, the number of independent holes or cycles.

For any symmetric positive semidefinite Laplacian $\mathbf{L}$, a unit eigenvector z satisfying $\mathbf{L}z = \lambda z$ has energy $z^\top \mathbf{L} z = \lambda$. Thus small eigenvalues correspond to low-energy vectors. For $\mathbf{L}_0$, the lowest nonconstant eigenvectors are the smoothest nontrivial node signals. For $\mathbf{L}_1$, kernel eigenvectors have zero divergence and zero curl, while the lowest non-kernel eigenvectors are the unit edge flows, orthogonal to $\ker(\mathbf{L}_1)$, with minimum nonzero divergence-plus-curl energy.

3 Harmonic and Quasi-Harmonic Flow Options

For our construction, we start with the state transition graph. The Laplacians used here are symmetric operators so we first symmetrize the state transition graph by converting all directed edges into

undirected edges: that is, two vertices are now connected by an undirected edge if a transition is possible in at least one direction between the corresponding states. We then orient the graph by assigning each edge an arbitrary direction. We are given (or construct) a set of filled faces F. On graphs we experiment with, each filled face is a cycle of four vertices, such as $(v_{00} \to v_{10} \to v_{11} \to v_{01} \to v_{00})$, where v_{ij} refers to the grid square at coordinates i and j.

We use edge flows to define option policies. Given edge flow $\mathbf{x}$, let $x_{s \to v}$ denote the oriented flow along edge $s \to v$, which can be read from $\mathbf{x}$ with sign determined by the canonical edge orientation. Let $\mathcal{N}(s)$ denotes the neighbors of node s. For state s, the option policy $\pi_{\mathbf{x}}(v \mid s)$ is zero for any node $v \notin \mathcal{N}(s)$, and otherwise is computed using a Boltzmann distribution over the neighbors:

$$\begin{aligned} \pi_{\mathbf{x}}(v \mid s) &:= \frac{e^{\mu\, x(s,v)/c_{\mathbf{x}}(s)}}{\sum_{y \in \mathcal{N}(s)} e^{\mu\, x(s,y)/c_{\mathbf{x}}(s)}}, \qquad v \in \mathcal{N}(s), \\ c_{\mathbf{x}}(s) &:= \begin{cases} \max_{y \in \mathcal{N}(s)} |x(s,y)|, & \max_{y \in \mathcal{N}(s)} |x(s,y)| > \varepsilon, \\ 1, & \text{otherwise.} \end{cases} \end{aligned} \tag{1}$$

where ε is set to a very small number (we use 10^{-12}) and parameter $\mu > 0$ controls how sharply the policy follows the flow direction.

Harmonic Flow Options (HFO) are constructed directly from the harmonic subspace. Let $\{\mathbf{h}_k\}_{k=1}^{r}$ be an orthonormal basis of $\ker(\mathbf{L}_1)$, the harmonic eigenvectors. Each $\mathbf{h}_k$ defines an option policy $\pi_{\mathbf{h}_k}$ as defined in Equation (1). We additionally include $-\mathbf{h}_k$ as a separate option when we want both directions. The option's initiation set is the entire state space, $\mathcal{I}_{\mathbf{h}_k} = \mathcal{S}$. The termination condition is $\beta_{\mathbf{h}_k} : \mathcal{S} \to \beta_c$, allowing the option to terminate at every state with fixed probability $\beta_c \in (0, 1)$.

Figure 1 shows HFO in the `four-rooms` domain, where there is one topological hole, thus the kernel dimension is 1, resulting in two options (forward and reverse). Figure 1 (c) demonstrates a limitation of options that induce direct paths (such as shortest paths) toward a single termination state. In particular, states not in the direct paths, or those far from the termination state can have substantially lower visitation probability (measured here by the stationary distribution).

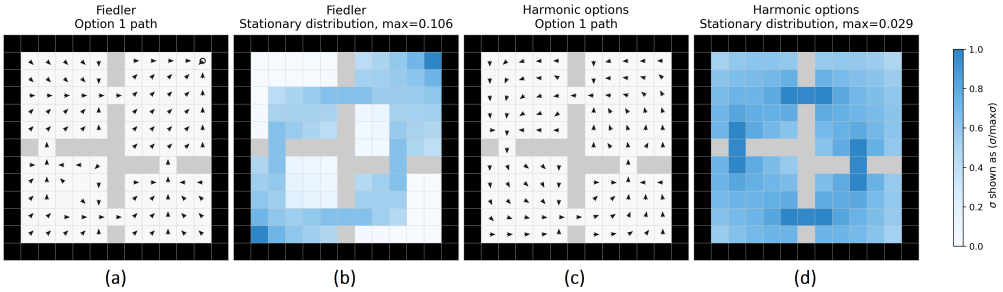

Figure 1: In the `four-rooms` domain: Plot (a) shows the policy of an option which takes a shortest path to a local maxima of the Fiedler vector (depicted by the small circle in the top-right corner) before terminating (equivalent to the one of the first two reversible covering options). Plot (c) shows the policy of one of the two reversible harmonic flow options. Plots (b) & (d) show the stationary distribution (σ normalized between zero and one) of a walker taking the option policies in (a) & (c) and their reverse with no other primitive actions.

Quasi-Harmonic Flow Options (QHFO) are constructed by modifying each harmonic flow to improve coverage in appendages and dead-ends while preserving cyclic structure in the core. The key design principle is to inject drift only where harmonic drift activity is weak, using a soft weight derived from the harmonic subspace (as described below and defined in Equation 2) which we use to blend a sign positive or sign negative version of the smoothest non-harmonic eigenvector in low-activity (drift) regions (see: 3). We are specific in selecting the sign of the smoothest non-harmonic eigenvector as matching drift directions is desirable prior to blending the eigenvectors. The option is available in all states. We allow the option to terminate in all states with probability $\beta_c \in (0, 1)$.

Soft drift weights from harmonic activity. We quantify where harmonic flows are inactive by aggregating harmonic magnitudes incident to each node. For node v, we define the node *drift weight*

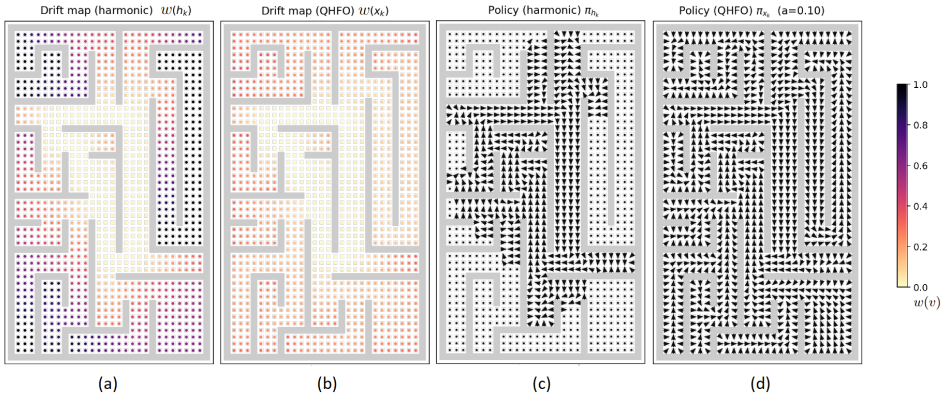

Figure 2: **QHFO drift injection.** Here the domain in (a)-(d) is a partial graph of the double large maze we use for evaluations. From left to right: (a) harmonic drift weights $w(\mathbf{h}_k)$, showing weak activity in appendages, (b) QHFO drift weights $w(\mathbf{x}_k)$ after injection, (c) harmonic policy $\pi_{\mathbf{h}_k}$ induced by (1), (d) quasi-harmonic policy $\pi_{\mathbf{x}_k}$, showing improved directed behaviour in appendages while preserving cyclic structure in the core.

$w(v)$ as follows:

$$w(v) \;=\; \left(1 - \frac{\sum_{e\ni v}\sum_{k=1}^{r}|\mathbf{h}_k(e)|}{\max_{v'\in V}\sum_{e\ni v'}\sum_{k=1}^{r}|\mathbf{h}_k(e)|}\right)^2 \in [0,1]. \tag{2}$$

We lift these weights to edges by taking the mean at endpoints: for edge $e = \{a, b\}$, we define $w_e = \frac{1}{2}\big(w(a) + w(b)\big)$. For the set of all edges, we have $\mathbf{w} = (w_e)_{e\in E}$.

Intuitively, $w(v) \approx 0$ in the cyclic core where harmonic activity is large, and $w(v) \approx 1$ in dead-end-like regions where harmonic activity is weak, see Figure 2 then plot (a) & (c) to see the drift weights for the harmonic eigenvectors in two domains.

A single smooth non-harmonic mode. Let $\mathbf{u}$ be the smoothest non-harmonic eigenvector of $\mathbf{L}_1$, that is, the eigenvector associated with the smallest eigenvalue above 0. Empirically, $\mathbf{u}$ (and $-\mathbf{u}$) captures the dominant "gradient into appendages" behaviour, so QHFO adds only this single non-harmonic mode to the kernel eigenvectors in HFO.

Blending the flows. QHFO adds the weighted correction $\mathbf{w}\odot\mathbf{u}$ to each harmonic flow and normalizes the result. Because the harmonic basis is orthonormal, $\|\mathbf{h}_k\|_2 = 1$ and the blend strength is shared across all k. Let $\bar{w} = |E|^{-1}\sum_{e\in E} w_e$ be the mean edge weight. We set

$$\alpha = \min\left\{1, \frac{\bar{w}+\varepsilon}{\|\mathbf{w}\odot\mathbf{u}\|_2 + \varepsilon}\right\}, \tag{3}$$

$$s_k = \arg\max_{s\in\{-1,+1\}} s\,\langle \mathbf{w}\odot\mathbf{h}_k, \mathbf{u}\rangle, \qquad \mathbf{x}_k = \frac{\mathbf{h}_k + s_k\alpha(\mathbf{w}\odot\mathbf{u})}{\|\mathbf{h}_k + s_k\alpha(\mathbf{w}\odot\mathbf{u})\|_2}. \tag{4}$$

We take $s_k = +1$ in a tie. The weights determine where the correction is applied, α limits its magnitude, and s_k aligns it with $\mathbf{h}_k$. The sign rule is equivalent to maximizing the weighted cosine similarity: its denominator is positive and independent of s. The resulting unit flow $\mathbf{x}_k$ defines the option policy through (1); setting $\alpha = 0$ recovers HFO.

Figure 2 illustrates the construction: the harmonic activity weights identify weakly active regions in panel (a), and the correction supplies additional drift there in panels (b) and (d).

QHFO (simple blend) We additionally evaluate a simple blend version of QHFO where we set $\mathbf{w} = \mathbf{1}$ and $\alpha = 1$ in (4).

QHFO (source). As an alternative to the first non-harmonic eigenvector of the Hodge Laplacian, we use the eigenvector f_{src} associated with the smallest eigenvalue of the source Laplacian $L_{\mathrm{src}} = I + \Gamma - P^{rw}$ [35], where P^{rw} is the random walk transition matrix with $\Gamma_{vv} = 0.0001\,w(v)$. The weights (sink-weights) come from the drift-weight map created by the harmonics, visualized

Algorithm 1 Quasi-Harmonic Flow Options (QHFO)

Input: transition graph $G = (V, E)$, faces F, temperature μ
Build $\mathbf{B}_1, \mathbf{B}_2$ and $\mathbf{L}_1 = \mathbf{B}_1^\top \mathbf{B}_1 + \mathbf{B}_2 \mathbf{B}_2^\top$
Compute harmonic eigenvectors $\{\mathbf{h}_k\}_{k=1}^r$ and the smoothest non-harmonic eigenvector $\mathbf{u}$
Compute weights $w(v)$ by (2) and $w_e = \frac{1}{2}(w(a) + w(b))$; let $\mathbf{w} = (w_e)_{e \in E}$
Compute $\bar{w} = |E|^{-1} \sum_{e \in E} w_e$ and the shared strength α by (3)
for $k = 1$ to r **do**
    Choose the alignment sign s_k by (4)
    Form the quasi-harmonic flow $\mathbf{x}_k$ by (4)
    Define option $o_k = (\mathcal{S}, \pi_{\mathbf{x}_k}, \beta_o)$ where $\pi_{\mathbf{x}_k}$ is given by (1)
    Add reverse option $o_k^{\text{rev}} = (\mathcal{S}, \pi_{-\mathbf{x}_k}, \beta_o)$
end for
Return: the set of QHFO options

in Figure 2(a). Where harmonic activity is high, $w(v)$ and the sink-weights are near zero; where harmonic activity is low, $w(v)$ approaches one and the sink-weights approach 0.0001. This encourages eigenvector values/energy to concentrate in regions of high harmonic activity, that are less penalized by the sink-weights, with values attenuating into regions where it is weak, such as appendages. The resulting gradient can thus be followed from the region where harmonic activity is high, toward the more strongly sink-weighted regions and appendages. We discuss the resulting flow in appendix G Note: Since the drift map can include zero values, we do not use NEO's fractional-power scaling. After obtaining f_{src}, we use a simple blend ($w = 1, a = 1$) to blend it with the harmonic vector(s) as in 4. We show the policy flow of both the QHFO (simple blend) and QHFO (source) in appendix C.

4 Related work

Our work is most closely related to other Laplacian option discovery methods that explicitly target coverage and hitting-time criteria. Eigenoptions [20] construct options from eigenvectors of a (node) graph Laplacian via policy iteration; related Laplacian option/representation work includes proto-value functions and Laplacian-based abstractions [21, 22, 24, 19]. Covering options similarly use spectral structure to reduce cover times, producing shortest-path behaviours to termination states [12]. Diffusion-based option constructions also target hitting-time behaviour via finding diffusely far away local minima [2], after which they solve shortest paths to obtain policies. In contrast, our options are built from edge-flow eigenvectors of the 1-form Hodge Laplacian and induce policies directly from those flows. Related to the 0-form Laplacian methods is the successor representation [5, 26]. Graph-based option discovery also include structural heuristics methods such as options constructed by betweenness centrality [33] and bridge centrality[23]) and clustering or hierarchy constructions [16]. DIAYN is a canonical example of diversity-driven skill learning without reward shaping [8]. Count-based exploration and novelty signals are also widely used in online exploration [34, 31, 40].

Novel exploration via orthogonality (NEO) [35] also employs spectral representations, and connect to earlier novelty/subgoal work [29, 30, 28], but are designed for online settings where novelty/visitation signals shape the representations constructed at time-step intervals as the domain is discovered pushing the agent to the frontier of states that have been discovered. Covering eigen options (CEO) [18] are also constructed online from the current successor representation of the fixed learning policy [26] and push the agent toward the frontier. Because NEO requires an explicit novelty signal and CEO uses a fixed policy for it's successor representation construction, the learned eigenvectors of these approach are not purely intrinsic to the environment geometry, as such, they can work very well in online settings, but their frozen option sets are less suited to the task-agnostic settings and all state-pair analytical hitting-time criterion we considered here. In the appendix 3. We empirically evaluate NEO and CEO in the tabular domains during forward forward tasks, and after freezing the learned options, evaluate them in reverse the tasks where they competitively do not perform as well. Additionally we show analytical hitting time results in appendix 4 5.

Our work is also tied closely to Laplacian representation learning. Recent work studies how to obtain the eigenvectors of the 0-form Laplacian in Hilbert space when Laplacian representations capture reachability and how to scale them to better reflect inter-state distances [36, 10, 14, 37, 39]. Spectral

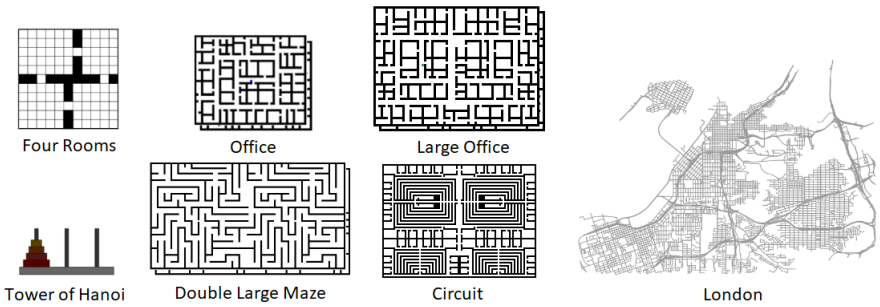

Figure 3: **Evaluation domains.** These domains span small grids, structured indoor layouts, dead-end-heavy environments, and a real-world road-network style graph.

Exterior Calculus [4] gives theory for constructing frames of differential forms and Hodge–de Rham operators directly from the 0-form Laplacian eigenvectors.

The use of the harmonic kernel of the Hodge Laplacian has previously been operationalized in machine learning, for instance, to identify cyclic signal-transmission paths brain networks [1].The use of low-frequency Hodge coordinates has also been used for trajectory outlier detection and cyclic structure [9], other application include Hodge-compositional edge Gaussian processes [41] and simplicial neural networks [7, 27].

5 Performance Evaluation

Here we analyses the performance of HFO and QHFO against related laplacian-based option discovery methods: eigenoptions, covering options and diffusion options empirically with a reinforcement learning agent and analytical through exact matrix inverse on hitting time metrics. While covering options were originally designed for single-state initialization (called point-options), restricting their initiation to only one state can impair performance in larger problems with a limited time horizon, as has been previously discussed [2] [13]. Therefore, we enable option initiation in any state for the empirical evaluations, for covering options and all other methods. When no option is active, with *option initialization probability* $\rho \in (0, 1)$, we trigger an option in the set of options $\mathbf{O}$ being initiated; an option in the set $\mathbf{O}$ is then chosen with equal probability $(1/|\mathbf{O}|)$ and initiated. For HFO and QHFO, we map flows to policies using Equation (1) with $\mu = 5$ and use termination probability $\beta_c = 0.05$. In addition to the standard QHFO options we also evaluate the variants QHFO (simple blend) and QHFO (source). We evaluate on seven domains spanning different sizes and connectivity structures: `office` (2558 nodes), `circuit` (3850), `large office` (4492), `double large maze` (10652), `London` (5000), `four-rooms` (104), and `tower of Hanoi` (729).

Tabular empirical evaluation. All tabular settings use the same Q-learning [38] training loop with ϵ-greedy exploration ($\epsilon = 0.1$), learning rate $\alpha = 0.4$, discount $\gamma = 0.99$, and option initialization probability $\rho = 0.1$. These settings follow closely related novelty/eigenvector exploration protocols [35]. The reward is sparse: 100 for transitions to a goal state, 0 otherwise. For each run, we sample a goal state using a run-specific random seed, then choose the start state as the node with the largest shortest-path distance to the goal. This process produces challenging start-goal pairs, although it can favour methods whose terminations lie near boundaries, such as diffusion and cover options. We apply a backward Q-update pass in addition to forward updates to improve learning efficiency, matching the empirical protocol used in closely related option-learning baselines. We allow options to be used as learnable actions, enabling the agent to exploit them whenever advantageous. Specifically, we do a semi-Markov decision process backups for options.

We run 25 instances per domain. The time horizon (maximum number of steps per episode) is 500 for `four-rooms`, 1500 for `double large maze`, and 1000 for all other domains. Every 2000 decision stages, which we define as an *epoch*, we evaluate the learning agent by running an evaluation episode without exploration actions ($\epsilon = 0$) and record the sum of the rewards (return without discounting) which we plot in Figure 4 in the y-axis under the label *return* with a 2σ error width.

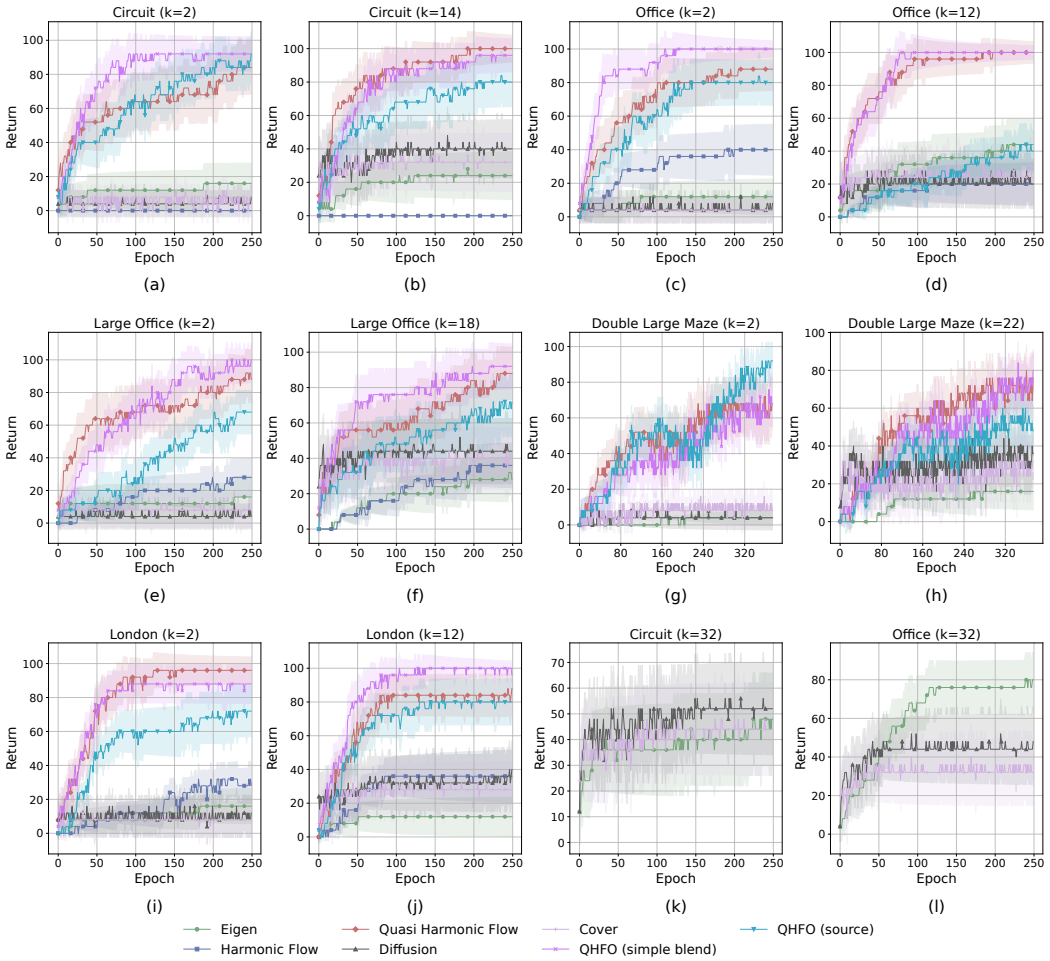

Figure 4: **Empirical learning curves.** Each panel shows average undiscounted return versus epoch for a given domain and option budget k (number of options). Here quasi harmonic flow refers to the standard blend QHFO option method whilst QHFO (simple blend) and QHFO (source) are dicussed in section 3.

Figure 4 shows performance comparisons between harmonic flow options, quasi-harmonic flow options, and the strongest competing baselines (diffusion, cover, and eigenoptions). The most significant separation appears in Circuit (plot (a)). With only $k = 2$ options (plot (a)), the simple blend variant of QHFO approaches near-100 average return by 100 epochs, whilst prior baselines are still obtaining less than 20 average return, indicating that the agent solves almost all start/goal instances within the training distribution using only two options effectively. In the same setting, the competing methods (including HFO) remain below roughly 20 average return, i.e., they solve only a small fraction of instances. For Circuit at $k = 14$ (plot (b)), we choose $k = 2r$ where the harmonic kernel dimension is $r = 7$ (Betti number 7) and we include both flow directions; QHFO again outperforms baselines. At $k = 32$ (plot (k)), the baselines improve but still remain limited (typically ≤ 50 average return). This can be potentially explained by the fact that Circuit contains many distinct boundary/appendage regions; in contrast, QHFO already achieves strong performance with $k = 2$ as it's option policies tend to flow better into the appendages. Whilst there is variation in the performance we see each of the QHFO variants performs well against baselines.

We see similar behaviour in `office`, `large office`, `London`, and `double large maze` (plots (c)–(j)). Eigenoptions become competitive at larger option budgets (e.g., `office` at $k = 32$), but at small budgets ($k = 2$) all baselines perform poorly relative to QHFO. In `double large maze`, we do not report eigenoptions because the required policy-iteration computations were not tractable at this scale; Overall, QHFO consistently provides competitive exploration performance with very few

options, while remaining fully eigenvector-based and topology-aware. Q-learning was unable to find any goal state, therefore we did not include it in plots.

In appendix 5, we show aggregate visitation heatmaps, which summarize the visitation under all state/goal pair episodes into a single map. These visitation maps show that QHFO options obtain a better coverage over all states against the competing Laplacian approaches.

Extending to function approximation. We extend our method to deep RL by constructing a high-level graph and acting on it as in the tabular setting. We pre-train five ant agents with Truncated Quantile Critics (TQC) which uses soft-actor critic (SAC) [11, 15] in an ant-maze domain for 5 million steps each, where each episode starts from a random initial state and assigns a nearby random goal for 5000 time-steps. We use the resulting trajectories, represented as $3 \times 64 \times 64$ colour top-down ant images, to train an ALLO network [10]. After $500k$ training iterations, we sample 1024 states from the SAC-TQC replay buffer, rescale the eigenvectors by dividing by the square root of their eigenvalues, giving a commute time-based scaling [36], and apply spectral clustering to identify distinct regions. We treat each cluster centroid as a graph node and connect centroids whose clusters are immediately adjacent. By immediately adjacent we mean that, for an agent in one cluster, it is the next cluster they enter after interacting in the domain. This yields a state-transition graph G_{ALLO} , on which we build the 1-form Hodge Laplacian and compute its eigenvectors to obtain flow option policies (shown in the appendix 7 for the large maze ant domains).

Each graph-level option is treated as a high-level policy whose action specifies the next goal for the lower-level primitive policy, implemented by the pretrained SAC-TQC agent; details are given in Appendix F. In the ant versions of `four-rooms` and `Large-AntMaze`, shown in Figure 3, we discretize the floor into cells corresponding to grid-world nodes. The ant starts in cell $[0, 0]$, the top-left corner, and explores using the high-level graph, thereby mimicking the tabular setting: when an option moves the high-level agent from one node to an adjacent node, that adjacent node is passed as a goal to the SAC-TQC policy.

Analytical evaluation. Table 1 reports hitting-time statistics between all state pairs under options with an option initiation probability $\rho = 0.1$ via exact matrix inversion. QHFO consistently reduces mean and median hitting times relative to HFO in environments with substantial appendage structure (e.g., `office`), indicating that the injected drift improves access to hard-to-reach regions. Several baselines exhibit heavy-tailed hitting-times, reflected by very large means/maxima compared to medians, suggesting that their induced dynamics can make some start–goal pairs extremely unlikely to connect within a truncation horizon. We report in Table 6 in the appendix the analytical results for QHFO (simple blend) and QHFO (source) .

Table 2 compares estimated hitting times for all option policies solved and executed on the high-level graph, with node-to-node transitions carried out by the SAC-TQC primitive policy. For each of the five trained agent instances, we run the agent for 100000 steps and record all-pairs cell hitting times, i.e., how long it takes to move from any cell to any other cell. There are 174 cells in `Large-AntMaze` and 104 in `ant-four-rooms`. QHFO again performs favourably compared to the other exploration option methods. Appendix F shows examples of the resulting G_{ALLO} graphs and QHFO flows, and reports option performance using the cell decomposition as an oracle grid.

6 Conclusion and limitations

We introduced Harmonic Flow Options (HFO) and Quasi-Harmonic Flow Options (QHFO) from the 1-form Hodge Laplacian. HFO provides topology-aware cyclic options from harmonic kernel eigenvectors. QHFO preserves these cycles where harmonic activity is strong, while injecting drift in dead-end appendages using the smoothest non-harmonic eigenvector, guided by a soft harmonic-activity weight and implemented as a one-shot eigenvector blend. Across empirical navigation tasks and analytical hitting-time metrics, QHFO improves coverage relative to harmonic-only baselines and compares favorably to established Laplacian option methods. A limitation of our work is the symmetric assumption, which is both a limitation of ALLO and the reversibility assumption of eigenvectors of our method and main baselines we compare against.

Table 1: Analytical hitting-time statistics using exact matrix inversion under option-augmented dynamics with option initiation probability $\rho = 0.1$ (matching the empirical runs). Each row reports the number of options k used by the method, and the mean/median/max/standard deviation of hitting times aggregated over the evaluated start–goal pairs. We use a cap of 10^{10} in the largest domain analytically evaluated circuit and 10^7 for all other domains. For covering options, the suffix (point) refers to the options using point-option initialization.

Circuit

Method	k	μ	Mean	Median	Max	Std.
Random walk	0	–	340,744	320,876	929,647	186,667
Covering	4	–	8.46e9	581	1.00e10	3.54e9
Covering (point)	4	–	326,083	237,927	1.05e6	192,869
Diffusion options	4	–	8.01e9	9.64e9	1.00e10	3.92e9
Eigenoptions	4	–	5.01e9	4.82e9	1.00e10	4.78e9
QHFO	4	1	131,759	66,268	884,816	156,446
QHFO	4	5	252,547	51,750	7.57e6	716,829
Covering	8	–	7.26e9	1,630	1.00e10	4.36e9
Covering (point)	8	–	301,745	227,343	1.07e6	197,202
Diffusion options	8	–	6.03e9	9.64e9	1.00e10	4.80e9
Eigenoptions	8	–	3.10e9	3.48e7	1.00e10	4.21e9
QHFO	8	1	70,556	48,146	477,854	70,962
QHFO	8	5	62,950	38,133	386,119	62,861
Covering	14	–	6.59e9	9.64e9	1.00e10	4.64e9
Covering (point)	14	–	268,207	164,743	1.12e6	200,914
Diffusion options	14	–	5.20e9	3,020	1.00e10	4.89e9
Eigenoptions	14	–	3.35e9	3.43e6	1.00e10	4.41e9
HFO	14	1	177,660	128,954	722,199	142,413
QHFO	14	1	69,586	46,059	454,494	68,102
QHFO	14	5	61,866	39,442	590,291	70,943

Office

Method	k	μ	Mean	Median	Max	Std.
Random walk	0	–	87,290	86,094	203,972	37,619
Covering (point)	2	–	81,893	81,653	179,531	35,157
QHFO	2	1	45,763	20,499	288,487	44,003
Covering	4	–	8.84e6	9.75e6	1.00e7	3.08e6
Covering (point)	4	–	77,160	75,274	179,452	33,411
Diffusion options	4	–	8.41e6	9.75e6	1.00e7	3.38e6
Eigenoptions	4	–	376,083	61,777	7.32e6	843,509
HFO	4	1	490,026	36,654	4.98e6	952,589
QHFO	4	1	34,232	24,940	144,337	28,702
HFO	4	5	1.92e6	120,639	9.78e6	3.00e6
QHFO	4	5	42,241	28,308	297,010	37,185
Covering	12	–	6.65e6	2,837	1.00e7	4.62e6
Covering (point)	12	–	67,817	71,332	156,131	27,711
Diffusion options	12	–	6.08e6	35,764	1.00e7	4.65e6
Eigenoptions	12	–	138,119	31,761	2.48e6	299,520
HFO	12	1	316,043	28,676	3.66e6	644,681
QHFO	12	1	25,983	21,083	118,574	20,579
HFO	12	5	1.20e6	49,207	9.13e6	2.20e6
QHFO	12	5	29,886	24,368	179,184	26,385

Four Rooms

Method	k	μ	Mean	Median	Max	Std.
Random walk	0	–	616	672	1,149	261
Covering	2	–	196,195	11,508	2.99e6	437,269
Covering (point)	2	–	472	422	1,268	259
Diffusion options	2	–	16,160	994	167,088	31,381
Eigenoptions	2	–	1,956	394	19,111	3,271
QHFO	2	1	219	199	574	111
Covering	4	–	272	148	1,897	270
Diffusion options	4	–	320	142	2,539	375
Eigenoptions	4	–	330	222	1,268	256

Tower of Hanoi

Method	k	μ	Mean	Median	Max	Std.
Random walk	0	–	14,172	15,195	29,713	5,922
Cover	2	–	6.89e6	9.75e6	1.00e7	4.28e6
Cover (point)	2	–	12,281	11,013	31,695	5,131
Diffusion options	2	–	6.89e6	9.75e6	1.00e7	4.28e6
Eigenoptions	2	–	2.07e6	156,577	9.90e6	3.21e6
HFO	2	1	7,473	6,056	25,391	4,341
QHFO	2	1	5,264	4,677	19,693	2,620
HFO	2	5	9,719	7,675	51,524	6,964
QHFO	2	5	6,543	5,518	33,100	4,618
Cover	4	–	4.29e6	1.56e6	1.00e7	4.53e6
Cover (point)	4	–	10,164	10,342	15,795	3,715
Eigenoptions	4	–	82,123	3,327	3.45e6	325,845
HFO	4	1	7,896	7,554	21,752	3,784
QHFO	4	1	5,468	4,999	18,429	2,683
HFO	4	5	9,803	8,654	43,994	5,758
QHFO	4	5	4,810	4,131	14,889	2,755

Table 2: In-environment Ant evaluations by option-generation method. Values are mean ± standard deviation over five seeds, with median hitting time shown in parentheses. Bold indicates the lowest mean hitting time within each evaluation setting.

Setting	Method	Coverage	Hit time, mean (median)
Ant-FourRooms			
2 opt.	Cover	0.633 ± 0.035	$37{,}223 \pm 3{,}424$ (532 ± 75)
	Diffusion	0.165 ± 0.007	$83{,}499 \pm 676$ ($100{,}000 \pm 0$)
	Eigenoptions	0.544 ± 0.019	$46{,}078 \pm 1{,}844$ ($1{,}971 \pm 1{,}716$)
	QHFO	0.987 ± 0.019	$\mathbf{3{,}444 \pm 1{,}717}$ (967 ± 147)
Large-AntMaze			
2 opt.	Cover	0.522 ± 0.016	$48{,}008 \pm 1{,}509$ ($2{,}025 \pm 1{,}305$)
	Diffusion	0.476 ± 0.013	$52{,}598 \pm 1{,}294$ ($100{,}000 \pm 0$)
	Eigenoptions	0.553 ± 0.055	$45{,}025 \pm 5{,}568$ ($20{,}595 \pm 44{,}389$)
	QHFO	0.807 ± 0.068	$\mathbf{22{,}893 \pm 6{,}862}$ ($2{,}843 \pm 1{,}543$)
6 opt.	Cover	0.498 ± 0.010	$50{,}382 \pm 945$ ($61{,}335 \pm 52{,}948$)
	Diffusion	0.474 ± 0.010	$52{,}725 \pm 958$ ($100{,}000 \pm 0$)
	Eigenoptions	0.585 ± 0.028	$41{,}933 \pm 2{,}661$ (760 ± 92)
	QHFO	0.715 ± 0.075	$\mathbf{32{,}058 \pm 6{,}866}$ ($3{,}741 \pm 1{,}489$)

Acknowledgements

We would like to thank the members of the Bath Reinforcement Learning Laboratory for their comments and useful suggestions.

References

[1] D. Vijay Anand and Moo K. Chung. Hodge laplacian of brain networks. *IEEE Transactions on Medical Imaging*, 42(5):1563–1573, 2023.

[2] Amitay Bar, Ronen Talmon, and Ron Meir. Option discovery in the absence of rewards with manifold analysis. In *International Conference on Machine Learning*, pages 664–674. PMLR, 2020.

[3] Eckmann Beno. Harmonische funktionen und randwertaufgaben in einem komplex. *Commentarii Mathematici Helvetici*, 17(1):240–255, 12 1944.

[4] Tyrus Berry and Dimitrios Giannakis. Spectral exterior calculus. *Communications on Pure and Applied Mathematics*, 73(4):689–770, 2020.

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

[6] Jozef Dodziuk. Finite-difference approach to the hodge theory of harmonic forms. *American Journal of Mathematics*, 98(1):79–104, 1976.

[7] Stefania Ebli, Michaël Defferrard, and Gard Spreemann. Simplicial neural networks. *CoRR*, abs/2010.03633, 2020.

[8] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning diverse skills without a reward function. 2018.

[9] Florian Frantzen, Jean-Baptiste Seby, and Michael T. Schaub. Outlier detection for trajectories via flow-embeddings. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pages 1568–1572, 2021.

[10] Diego Gomez, Michael Bowling, and Marlos C. Machado. Proper laplacian representation learning. In *The Twelfth International Conference on Learning Representations*, 2024.

[11] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *CoRR*, abs/1801.01290, 2018.

[12] Yuu Jinnai, Jee Won Park, David Abel, and George Konidaris. Discovering options for exploration by minimizing cover time. In *International Conference on Machine Learning*, pages 3130–3139. PMLR, 2019.

[13] Yuu Jinnai, Jee Won Park, Marlos C. Machado, and George Konidaris. Exploration in reinforcement learning with deep covering options. In *Proceedings of the Eighth International Conference on Learning Representations (ICLR)*, 2020.

[14] Martin Klissarov and Marlos C Machado. Deep Laplacian-based options for temporally-extended exploration. In *International Conference on Machine Learning*, pages 17198–17217. PMLR, 2023.

[15] Arsenii Kuznetsov, Pavel Shvechikov, Alexander Grishin, and Dmitry P. Vetrov. Controlling overestimation bias with truncated mixture of continuous distributional quantile critics. *CoRR*, abs/2005.04269, 2020.

[16] Andrew Levy, George Konidaris, Robert Platt, and Kate Saenko. Learning multi-level hierarchies with hindsight. In *Proceedings of International Conference on Learning Representations*, 2019.

[17] Lek-Heng Lim. Hodge laplacians on graphs. *SIAM Review*, 62(3):685–715, 2020.

[18] Marlos C Machado, Andre Barreto, Doina Precup, and Michael Bowling. Temporal abstraction in reinforcement learning with the successor representation. *Journal of Machine Learning Research*, 24(80):1–69, 2023.

[19] Marlos C Machado, Marc G Bellemare, and Michael Bowling. A laplacian framework for option discovery in reinforcement learning. In *International Conference on Machine Learning*, pages 2295–2304. PMLR, 2017.

[20] Marlos C Machado, Clemens Rosenbaum, Xiaoxiao Guo, Miao Liu, Gerald Tesauro, and Murray Campbell. Eigenoption discovery through the deep successor representation. In *International Conference on Learning Representations*, 2018.

[21] Sridhar Mahadevan. Proto-value functions: Developmental reinforcement learning. In *Proceedings of the Twenty-Second International Conference on Machine Learning*, pages 553–560, 2005.

[22] 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.

[23] Parham Moradi, Mohammad Ebrahim Shiri, and Negin Entezari. Automatic skill acquisition in reinforcement learning agents using connection bridge centrality. In Tai-hoon Kim, Thanos Vasilakos, Kouichi Sakurai, Yang Xiao, Gansen Zhao, and Dominik Ślęzak, editors, *Communication and Networking*, pages 51–62, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[24] Marek Petrik. An analysis of laplacian methods for value function approximation in mdps. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2574–2579, 2007.

[25] Doina Precup. *Temporal abstraction in reinforcement learning*. University of Massachusetts Amherst, 2000.

[26] Rahul Ramesh, Manan Tomar, and Balaraman Ravindran. Successor options: An option discovery framework for reinforcement learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3304–3310, 2019.

[27] T. Mitchell Roddenberry, Nicholas Glaze, and Santiago Segarra. Principled simplicial neural networks for trajectory prediction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9020–9029. PMLR, 18–24 Jul 2021.

[28] Özgür Şimşek and Andrew Barto. Skill characterization based on betweenness. *Advances in Neural Information Processing Systems*, 21, 2008.

[29] Özgür Şimşek and Andrew G Barto. Using relative novelty to identify useful temporal abstractions in reinforcement learning. In *Proceedings of the Twenty-First International Conference on Machine Learning*, page 95, 2004.

[30] Özgür Şimşek, Alicia P Wolfe, and Andrew G Barto. Identifying useful subgoals in reinforcement learning by local graph partitioning. In *Proceedings of the Twenty-Second International Conference on Machine Learning*, pages 816–823, 2005.

[31] Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

[32] Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1.

[33] 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.

[34] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #Exploration: A study of count-based exploration for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 30, 2017.

[35] Andreas Theophilou and Özgür Şimşek. Novel exploration via orthogonality. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[36] Kaixin Wang, Kuangqi Zhou, Jiashi Feng, Bryan Hooi, and Xinchao Wang. Reachability-aware laplacian representation in reinforcement learning. In *Proceedings of the 40th International Conference on Machine Learning*, pages 36670–36693, 2023.

[37] Kaixin Wang, Kuangqi Zhou, Qixin Zhang, Jie Shao, Bryan Hooi, and Jiashi Feng. Towards better Laplacian representation in reinforcement learning with generalized graph drawing. In *International Conference on Machine Learning*, pages 11003–11012. PMLR, 2021.

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

[39] Yifan Wu, George Tucker, and Ofir Nachum. The Laplacian in RL: Learning representations with efficient approximations. In *International Conference on Learning Representations*, 2019.

[40] Kai Yang, Jian Tao, Jiafei Lyu, and Xiu Li. Exploration and anti-exploration with distributional random network distillation. In *Proceedings of the 41st International Conference on Machine Learning*, pages 56397–56421, 2024.

[41] Maosheng Yang, Viacheslav Borovitskiy, and Elvin Isufi. Hodge-compositional edge Gaussian processes. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3754–3762. PMLR, 02–04 May 2024.

A Appendix: Computational resources

For all experiments we used a 9900k Intel CPU with two 4080 GPUs, 16GB of Ram and a 256GB SSD hard-drive with a 500GB HDD Drive.

B Appendix: Aggregate visitation heatmaps

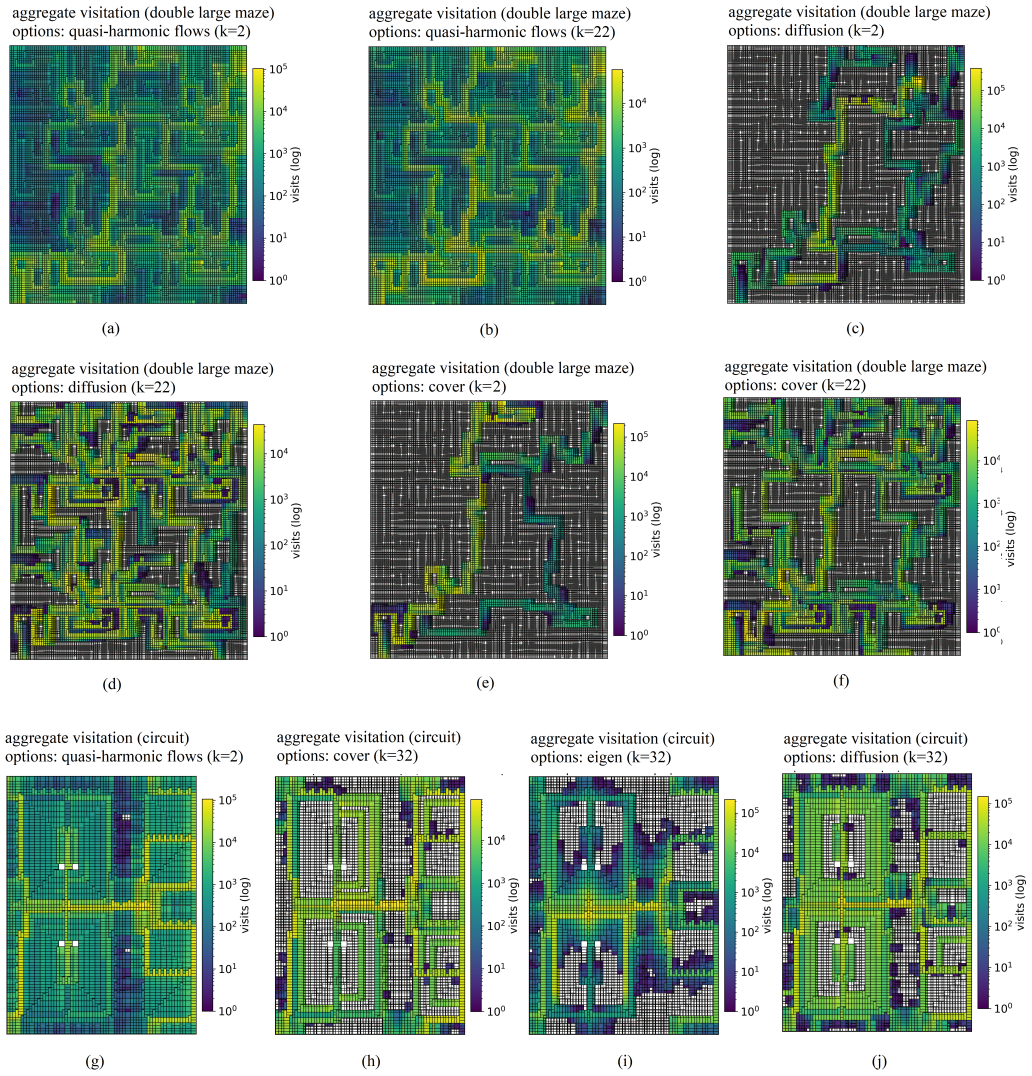

Figure 5: **Aggregate visitation heatmaps (log scale) comparing option families.** Each panel shows the (log) number of state visits accumulated under an option-augmented ϵ-greedy agent, aggregated over runs; brighter colors indicate more frequent visitation. Grey cells are walls/unreachable. The parameter k denotes the number of options available to the agent. **Double Large Maze:** (a) QHFO (k=2), (b) QHFO (k=22), (c) Diffusion (k=2), (d) Diffusion (k=22), (e) Cover (k=2), (f) Cover (k=22). **Circuit:** (g) QHFO (k=2), (h) Cover (k=32), (i) Eigenoptions (k=32), (j) Diffusion (k=32).

Discussion. These heatmaps visualize how each option family changes the agent's *state visitation distribution*. With a very small option budget (k=2), QHFO already produces broadly distributed visitation in Double Large Maze (Fig. 5a), whereas Cover and Diffusion with k=2 concentrate visitation along a relatively narrow corridor-like structure (Fig. 5c,e). As the option budget increases (e.g., k=22), both Cover and Diffusion become more diffuse (Fig. 5d,f), but still exhibit pronounced concentration on a subset of preferred routes compared to QHFO (Fig. 5b), which remains comparatively spread across the accessible space. In Circuit, QHFO with only k=2 (Fig. 5g) already distributes visitation across major regions, while the k=32 baselines (Fig. 5h–j) show stronger route/region bias and more uneven coverage in several sub-areas. Overall, these plots support the hypothesis that quasi-harmonic flow options encourage movement that spreads visitation well within domains.

C QHFO variant comparison

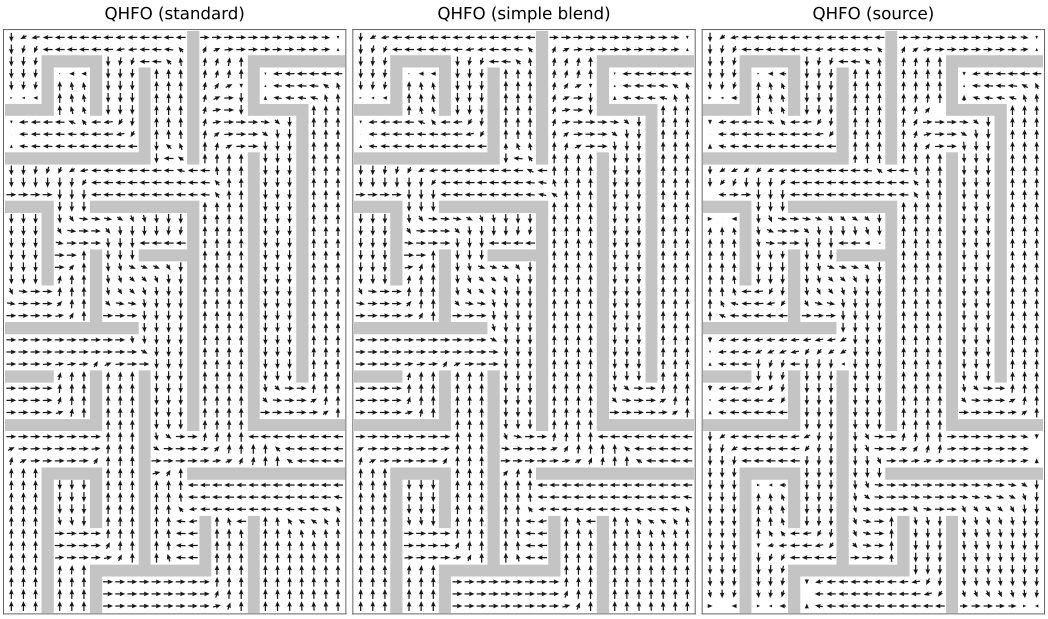

Figure 6: As in Figure 2, we plot the policy flows for the standard QHFO flow in plot (a), and comparatively plot the policy flows for QHFO using the simple blend approach and using the source Laplacian for the non-harmonic in plot (c).

In Figure 6, we can see that each of the QHFO variants produce a flow where the harmonic flow is mostly intact around the topological hole. An observation will show a clear different for the QHFO (source) approach, however, as all of the flow is driving into appendages whereas for standard QHFO and QHFO (simple blend) to flow is moving inward to several appendages and outwards to others. This is not a subtle difference, for QHFO (source), there's more drive into appendages, yet, an active option policy under the QHFO (source) flow cannot move from the end of one appendage into the end of another. This could conceptually be a limitation in certain settings such as with low termination probabilities, yet it may also act as an advantage (for instance, if all appendages were equally rewarding and the agent started within an active harmonic region).

D Comparison with NEO and CEO

NEO and CEO are both learned in an online setting in which the current start state and either the online SR or visitation distribution is available – here the learned options are focused on expanding the explored frontier. They can work very well on the frontier exploration tasks, yet, are not as suitable for the fully task-agnostic case (where the most recent relevant baseline is diffusion options). To illustrate this, we implemented NEO and CEO in every tabular domain used for our analytical results. We follow their original methods while using the same SMDP/Q-learning protocol as QHFO and NEO: the same learning rate and discount factor, option-initiation probability $\rho = 0.1$, and the backward SMDP updates. For each of 25 seeds, we sample a goal and choose a maximally distant start state. Each method may learn up to 64 options on this forward task. We then freeze the option set, reset the task learner, exchange the start and goal states, and evaluate reuse on the reverse task. One epoch is 2,000 decision stages. Table 3 reports mean return over 25 seeds. [1]

[1]The source code is available at `https://github.com/AndreasTheo/QHFO`

Table 3: **Forward and frozen-reverse return** (mean $\pm$ 95% CI half-width; 25 seeds). "Final" is epoch 125 for Four Rooms and epoch 250 otherwise. Higher is better.

Environment	Method	Forward: epoch 120	Forward: final	Frozen reverse: epoch 120	Frozen reverse: final
Circuit	NEO	40.00 ± 19.20	80.00 ± 15.68	0.00 ± 0.00	0.00 ± 0.00
Circuit	CEO	12.00 ± 12.74	44.00 ± 19.46	0.00 ± 0.00	0.00 ± 0.00
Four Rooms	NEO	80.00 ± 15.68	80.00 ± 15.68	100.00 ± 0.00	100.00 ± 0.00
Four Rooms	CEO	100.00 ± 0.00	100.00 ± 0.00	80.00 ± 15.68	80.00 ± 15.68
Office	NEO	100.00 ± 0.00	100.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
Office	CEO	72.00 ± 17.60	100.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
Tower of Hanoi	NEO	96.00 ± 7.68	92.00 ± 10.63	4.00 ± 7.68	4.00 ± 7.68
Tower of Hanoi	CEO	100.00 ± 0.00	100.00 ± 0.00	4.00 ± 7.68	8.00 ± 10.63

The forward results confirm that this online setting can be effective for both methods. NEO and CEO are particularly effective on the tasks where its frontier options were learned. The frozen reverse task asks the **task-agnostic** transfer question directly: once discovery stops, are the same options useful for a different, in this case, reversed task?

D.1 NEO & CEO analytical results

To test the suitability of NEO and CEO for hitting-time coverage more directly, we take the newest k frozen options available before the end of learning and apply the same exact analytical all-pairs computation used for QHFO, over the same domains and every matched option budget. This directly asks analytically whether the frozen options remain useful from every state to every other state. We use $\rho = 0.1$. QHFO and random-walk values are from the submission; caps are 10^{10} for Circuit and 10^7 otherwise.

Table 4: **Exact all-pairs mean hitting time.** Lower is better.

Environment	k	Random walk	QHFO ($\mu = 1$)	NEO	CEO
Circuit	4	340,744	131,759	3.08×10^9	5.76×10^9
Circuit	8	340,744	70,556	2.44×10^9	3.57×10^9
Circuit	14	340,744	69,586	2.22×10^9	1.60×10^9
Four Rooms	2	616	219	928	718
Four Rooms	4	616	–	660	707
Office	2	87,290	45,763	5.84×10^6	5.05×10^6
Office	4	87,290	34,232	3.29×10^6	4.17×10^6
Office	12	87,290	25,983	2.94×10^6	3.68×10^6
Tower of Hanoi	2	14,172	5,264	364,564	625,184
Tower of Hanoi	4	14,172	5,468	279,552	616,662

Table 5: **Exact all-pairs maximum hitting time.** Lower is better.

Environment	k	Random walk	QHFO ($\mu = 1$)	NEO	CEO
Circuit	4	929,647	884,816	1.00×10^{10}	9.97×10^9
Circuit	8	929,647	477,854	1.00×10^{10}	9.90×10^9
Circuit	14	929,647	454,494	1.00×10^{10}	9.84×10^9
Four Rooms	2	1,149	574	2,492	2,808
Four Rooms	4	1,149	–	1,580	2,736
Office	2	203,972	288,487	1.00×10^7	1.00×10^7
Office	4	203,972	144,337	1.00×10^7	1.00×10^7
Office	12	203,972	118,574	1.00×10^7	9.90×10^6
Tower of Hanoi	2	29,713	19,693	4.34×10^6	3.22×10^6
Tower of Hanoi	4	29,713	18,429	5.40×10^6	2.69×10^6

This does not mean that NEO and CEO are ineffective in their intended online setting; rather, it shows that their frozen option sets are not designed for the same **task-agnostic** all-pairs criterion as the other baselines and our proposed methods.

E Appendix: Extended exact Analytical results

Table 6: Analytical hitting-time statistics for QHFO (simple blend) and QHFO (source), using the settings in Table 1.

Circuit

Method	k	μ	Mean	Median	Max	Std.
QHFO (simple blend)	4	1	101,880	55,063	1.04e6	114,487
QHFO (source)	4	1	244,106	110,854	1.66e6	302,819
QHFO (simple blend)	4	5	114,272	40,050	2.48e6	279,004
QHFO (source)	4	5	128,438	66,050	923,768	142,540
QHFO (simple blend)	8	1	75,899	51,309	576,641	74,119
QHFO (source)	8	1	95,759	55,080	621,144	96,012
QHFO (simple blend)	8	5	100,001	38,222	2.46e6	234,829
QHFO (source)	8	5	84,594	47,027	513,579	87,789
QHFO (simple blend)	14	1	72,861	53,689	530,956	68,078
QHFO (source)	14	1	87,721	65,116	652,773	82,046
QHFO (simple blend)	14	5	98,699	38,404	2.23e6	222,508
QHFO (source)	14	5	69,653	45,721	371,514	66,068

Office

Method	k	μ	Mean	Median	Max	Std.
QHFO (simple blend)	2	1	34,769	29,029	156,897	26,956
QHFO (source)	2	1	43,758	27,530	263,702	41,376
QHFO (simple blend)	2	5	37,673	25,357	313,604	39,083
QHFO (source)	2	5	98,289	35,964	1.01e6	147,407
QHFO (simple blend)	4	1	29,988	27,274	127,677	21,019
QHFO (source)	4	1	34,560	24,864	172,847	30,419
QHFO (simple blend)	4	5	35,861	23,263	265,089	35,591
QHFO (source)	4	5	36,440	26,589	185,824	31,000
QHFO (simple blend)	12	1	27,731	24,776	127,742	20,719
QHFO (source)	12	1	27,243	24,562	102,960	18,223
QHFO (simple blend)	12	5	28,951	20,749	144,619	24,626
QHFO (source)	12	5	33,674	26,754	186,933	26,771

Four Rooms

Method	k	μ	Mean	Median	Max	Std.
QHFO (simple blend)	2	1	415	268	2,569	402
QHFO (source)	2	1	292	267	880	153
QHFO (simple blend)	2	5	767	282	7,429	1,218
QHFO (source)	2	5	347	296	1,214	251

Tower of Hanoi

Method	k	μ	Mean	Median	Max	Std.
QHFO (simple blend)	2	1	5,086	2,118	89,868	6,777
QHFO (source)	2	1	4,026	2,901	42,817	3,939
QHFO (simple blend)	2	5	4,468	1,267	163,652	8,214
QHFO (source)	2	5	6,321	2,664	153,282	10,975
QHFO (simple blend)	4	1	4,994	2,104	88,432	6,666
QHFO (source)	4	1	4,034	2,996	40,667	3,829
QHFO (simple blend)	4	5	4,379	1,314	162,733	8,169
QHFO (source)	4	5	3,396	2,290	50,684	3,797

Stochastic wind robustness. We also evaluated a stochastic Office variant with upward wind probability $1/8$, analogous to the wind stochasticity used in diffusion-option evaluations [2]. Table 7 reports mean hitting times. QHFO remains substantially better than the baselines under this stochastic perturbation.

Table 7: Mean hitting times in Office with upward wind probability $1/8$.

Method	$k = 2$	$k = 4$	$k = 12$
Random walk	1.33e9	1.33e9	1.33e9
Cover	7.77e9	6.70e9	5.51e9
Diffusion	7.33e9	5.03e9	3.01e9
Eigenoptions	6.06e8	1.62e7	2.42e6
HFO ($\mu = 1$)	9.93e6	9.44e6	1.80e6
QHFO ($\mu = 1$)	1.70e5	1.43e5	1.01e5

F Appendix: Function approximation

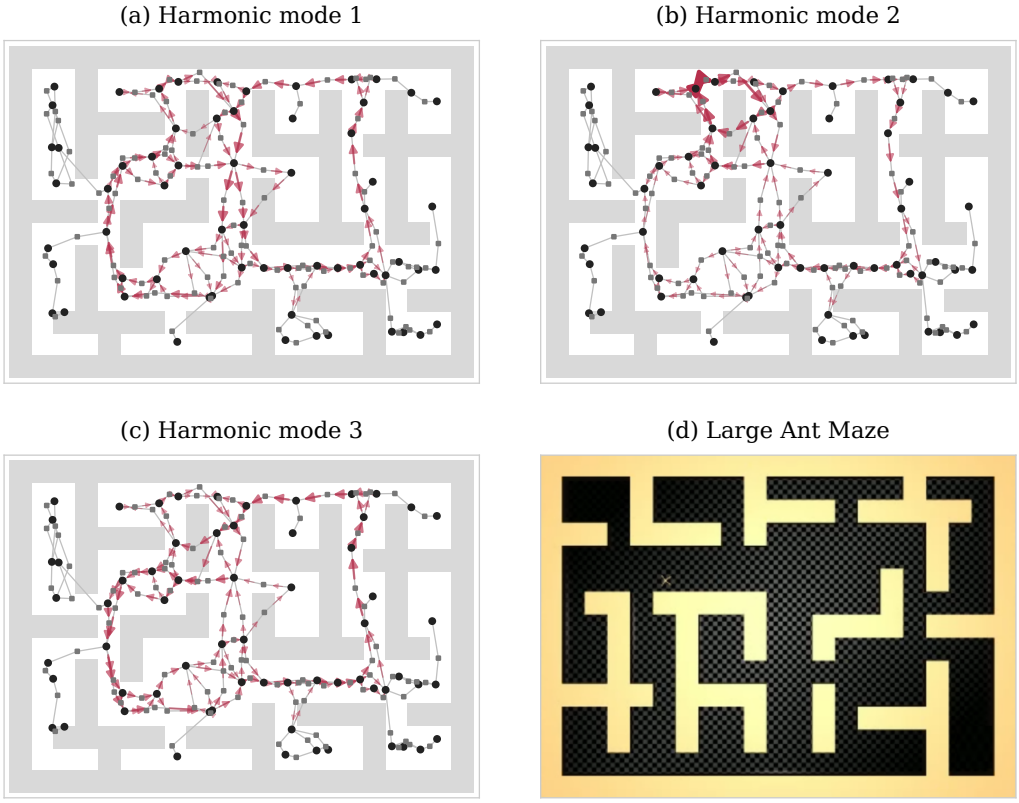

Figure 7: Harmonic flows on the `Large Ant Maze` graph, constructed from learned eigenvectors and eigenvalues. Panels (a–c) show its three harmonic modes; (d) shows the Ant domain.

Table 8: In-environment hitting-time results using exact grid cells instead of a learnt graph. Values are mean ± standard deviation over five seeds. Hitting times use a 100,000-step cap for unobserved pairs; median hitting time is reported in parentheses. Bold indicates the lowest mean hitting time within each evaluation setting.

Evaluation setting	Method	Coverage	Hitting time, mean (median)
Large-AntMaze			
Exact cell (2 options)	Cover	0.653 ± 0.020	$34{,}846 \pm 1{,}956$ (176 ± 3)
	Diffusion	0.760 ± 0.023	$24{,}132 \pm 2{,}275$ (141 ± 3)
	Eigenoptions	0.914 ± 0.020	$8{,}906 \pm 2{,}028$ (167 ± 7)
	QHFO	0.979 ± 0.019	$\mathbf{4{,}433 \pm 2{,}088}$ (621 ± 42)
Exact cell (6 options)	Cover	0.664 ± 0.016	$33{,}742 \pm 1{,}604$ (179 ± 7)
	Diffusion	0.767 ± 0.020	$23{,}465 \pm 1{,}981$ (144 ± 2)
	Eigenoptions	0.916 ± 0.018	$8{,}756 \pm 1{,}847$ (169 ± 14)
	QHFO	0.987 ± 0.017	$\mathbf{3{,}219 \pm 1{,}962}$ (582 ± 27)

Table 9: Implementation details for the image-based ALLO representation model and Ant TQC controller.

Component	Configuration
Image-Based ALLO	
Image observations	RGB images of shape $[3, 64, 64]$; dark floor and bright ant rendering.
Convolutional encoder	Three $\mathrm{Conv2d}$ layers with channel sizes $3 \to 16 \to 16 \to 16$; kernel size 3, stride 2, padding 1; ReLU after each layer.
Fully connected encoder	Flatten ($16 \times 8 \times 8 = 1024$); MLP dimensions $1024 \to 256 \to 256 \to 256 \to d$; ReLU after each hidden layer.
Representation dimension	$d = 50$.
Training settings	$500{,}000$ iterations; batch size 1024; sampling discount 0.9.
Optimization	Adam optimizer; learning rate 1×10^{-4}; dual learning rate 1×10^{-4}.
Barrier parameters	Initial coefficient 0.1; coefficient bounds $[0, 10000]$.
Dual parameters	Initial value -1.0; bounds $[-100, 100]$.
Ant TQC Controller	
Algorithm and policy	SB3-contrib TQC with `MultiInputPolicy`; goal-conditioned observations.
Observation and action	Observation dimension 29; goal dimension 2; action dimension 8, bounded by $[-1, 1]$.
Actor and critic networks	Two hidden layers of 256 units each; ReLU activation; Adam optimizer; learning rate 3×10^{-4} for both networks.
Distributional critics	2 critics; 25 quantiles per critic.
RL hyperparameters	Discount factor $\gamma = 0.99$; Polyak coefficient $\tau = 0.005$; fixed entropy coefficient 0.05; batch size 256.
Update frequency	Train frequency 1; gradient steps 1.
Reward and normalization	Sparse reward with success bonus 1.0; normalized xy coordinates.

G QHFO (source) coverage

Corollary 1 (Source gradients into weak-harmonic regions)**.** *Let $G = (V, E)$ be a finite connected undirected graph, and let P be its random walk transition matrix. Let the drift map values $w(v) \in [0, 1]$ be nonconstant. Using the source Laplacian [35] $L = I + \Gamma - P$ with $\Gamma = \mathrm{diag}(0.0001\, w(v))$, let (λ, f) be its lowest eigenpair, with f chosen strictly positive. Then every node v with $\Gamma_{vv} > \lambda$ is reachable from a node i with $\Gamma_{ii} < \lambda$ along a path strictly decreasing in f.*

Proof. First, we start with the argument of Theorem 3.3 from the NEO paper, specialised to the undirected setting. The eigenvalue equation gives $f_i = \frac{1}{1+\Gamma_{ii}-\lambda} \sum_j P_{ij} f_j$. For a node v with $\Gamma_{vv} > \lambda$, the factor $1/(1 + \Gamma_{vv} - \lambda)$ is less than 1. Since $f > 0$ and $\sum_j P_{vj} = 1$, there must therefore be an adjacent node with a strictly larger eigenvector value.
Starting from v, extend a strictly increasing path as far as possible. Since the graph is finite, this path ends at a node i with no adjacent node having a larger eigenvector value. The preceding node has a strictly smaller value and, since the graph is undirected, is also adjacent to i. Hence $(1+\Gamma_{ii}-\lambda)f_i = \sum_j P_{ij} f_j < f_i$, which implies $\Gamma_{ii} < \lambda$. It follows that, since we take the undirected assumption, there exists a reverse path. □

Consequently, if we take a non-deterministic (softmax) policy over all adjacent decreases in f, there is a non-zero probability of following each such decreasing path. Indeed, every adjacent transition has positive probability, Thus every node v satisfying $\Gamma_{vv} > \lambda$ can be reached with non-zero probability along a strictly decreasing path from a corresponding node i satisfying $\Gamma_{ii} < \lambda$.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.
Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.
The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in the abstract and introduction do accurately reflect the paper. We perform multi-seed experiments to confirm results and claims.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in the conclusion section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors

should reflect on how these assumptions might be violated in practice and what the implications would be.

- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: We only provide a Corollary beyond prior results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the information needed and supplementary sour code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same

dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Yes, we provide the supplementary material for the main experiments.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, we provide the parameters of the experiments.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the errors bars with experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We mention the computation used in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research conforms with the respective code and ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: We do not consider the research to have distinguishable social impact beyond the research of artificial intelligence as a whole.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: Existing assets are not used in this research.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide a link to the source code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: We do not do crowdsourcing nor research with human subjects in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: We do not do crowdsourcing nor research with human subjects in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were not used for any method development.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.