Randomized Spectral Options: Option Discovery via Random Reward Probing

Yiming Fei
College of Computer Science and Technology
Zhejiang University
Hangzhou, China
yimingfei@zju.edu.cn

Lang Qin
College of Computer Science and Technology
Zhejiang University
Hangzhou, China
qinl@zju.edu.cn

Rui Yan
College of Computer Science and Technology
Zhejiang University of Technology
Hangzhou, China
ryan@zjut.edu.cn

Huajin Tang*
College of Computer Science and Technology
Zhejiang University
Hangzhou, China
htang@zju.edu.cn

***Abstract*—Discovering diverse and useful options is key to hierarchical reinforcement learning, yet many spectral approaches rely on Laplacian decomposition and eigenfunction learning, making it costly to scale option sets and difficult to reuse the learned representations. We propose *Randomized Spectral Options* (RSO), an RL-native alternative that generates smooth option-discovery potentials by randomized reward probing through successor representations/features. RSO samples a small batch of orthogonal random reward probes via QR decomposition, learns a successor representation backbone under an exploration policy, and obtains a family of randomized value-function potentials through a single readout from the learned representation. Each potential is then turned into a potential-based shaping reward to train signed options that ascend/descend the induced potential field. On MiniGrid, RSO achieves state coverage and downstream performance competitive with Laplacian-representation-based options, and Atari 2600 games visualizations suggest that the learned potentials emphasize semantically meaningful bottlenecks.**

***Index Terms*—hierarchical reinforcement learning, option discovery, randomized spectral options, successor representation.**

I. Introduction

Despite major progress, reinforcement learning (RL) still struggles in problems with sparse rewards and long-horizon credit assignment [1]–[6]. Hierarchical reinforcement learning mitigates this difficulty by introducing temporal abstraction, so that exploration and control operate over temporally extended behaviors rather than primitive actions alone [7]–[9]. A standard abstraction is the option framework [10], which also underlies end-to-end methods such as option-critic [11]. The central challenge is therefore *option discovery*: learning a set of options that is both diverse and useful. Existing methods mainly exploit state-space structure [12], [13] or optimize explicit criteria such as planning and cover time [14]–[17].

Among structure-based methods, Laplacian option discovery extracts smooth directions from the eigenvectors or eigenfunctions of a transition graph and converts them into intrinsic rewards for exploration [18]–[25]. Although effective, this line of work has two practical limitations. First, adding new options typically requires additional spectral computation or auxiliary eigenfunction-learning machinery, which can be expensive in large or continuous settings [26]. Second, the learned spectral representation is often used only to derive intrinsic rewards, limiting its reuse within the broader RL pipeline.

We propose *Randomized Spectral Options* (RSO), an RL-native alternative motivated by the connection between Bellman equations and Laplacian operators. Instead of solving for new eigenvectors, RSO samples orthogonal random reward probes and obtains randomized value-function potentials through a successor representation/feature backbone [27]–[31]. These potentials induce shaping rewards for learning signed options, making option expansion cheap while enabling tighter reuse of the learned representation; the backbone itself is learned with TD-style updates [32]. Empirically, RSO achieves competitive state coverage and downstream performance while avoiding explicit spectral decomposition.

II. Related Work

Spectral option discovery builds a state-transition graph and extracts smooth potentials from Laplacian eigenvectors/eigenmaps that capture large-scale geometry and bottlenecks [18], [19], [33]. These potentials can be converted into intrinsic rewards to learn exploration options and improve state-space coverage [20], [21]. However, maintaining a graph (or its approximation) and performing eigen-decomposition can be expensive to scale [21], [26].

To improve scalability, eigenoption discovery connects Laplacian structure to successor representations, reducing reliance on explicit transition graphs [22], [27]. More recently, Laplacian-style representations have been learned directly from experience using scalable approximations and improved objectives [24], [25], [34], and auxiliary-task formulations

* Corresponding author.

such as Proto-Value Networks emphasize representation reuse beyond intrinsic-reward construction [20], [35], [36].

Deep Laplacian-based Options (DLO) further integrates representation learning with option training in an online closed loop to enable temporally extended exploration in high-dimensional environments [23], [25]. Despite these advances, expanding the option set can remain non-trivial and is often coupled to the learned representation dimensionality or to remaining spectral machinery [22]–[24]. These limitations motivate our Randomized Spectral Options (RSO), which replaces explicit spectral learning with TD-learned successor representations/features and uses random reward probing to generate option-design potentials with low marginal cost.

III. Preliminaries

A. Bellman equations, options, and eigenoptions

We consider a discounted MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ and a stationary policy π. Its state-value function is

$$V_\pi(s) \doteq \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, r(S_t, A_t, S_{t+1}) \;\middle|\; S_0 = s\right]. \quad (1)$$

It satisfies the Bellman expectation equation

$$V_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s) \sum_{s'\in\mathcal{S}} \mathcal{P}(s'|s,a)\Big(r(s,a,s') + \gamma V_\pi(s')\Big). \quad (2)$$

For finite state spaces, define the policy-averaged transition matrix $[P_\pi]_{ij} \doteq \sum_{a\in\mathcal{A}} \pi(a \mid s_i)\, \mathcal{P}(s_j \mid s_i, a)$ and the expected reward vector $[\mathbf{r}_\pi]_i \doteq \sum_{a\in\mathcal{A}} \pi(a \mid s_i) \sum_{s'\in\mathcal{S}} \mathcal{P}(s' \mid s_i, a)\, r(s_i, a, s')$. Then

$$(I - \gamma P_\pi)\, \mathbf{v}_\pi = \mathbf{r}_\pi, \qquad \mathbf{v}_\pi = (I - \gamma P_\pi)^{-1} \mathbf{r}_\pi. \quad (3)$$

An option is a temporally extended action $o = (\mathcal{I}_o, \pi_o, \beta_o)$ with initiation set, intra-option policy, and termination rule [10]. Spectral option methods construct a transition graph with weighted adjacency matrix W and degree matrix D, and use the random-walk Laplacian

$$L_{\mathrm{rw}} \doteq I - D^{-1}W. \quad (4)$$

Small non-zero eigenvectors of L_{rw} (denoted as u_i) vary smoothly within well-connected regions and change sharply across bottlenecks [18], [19]. Eigenoption methods convert such potentials into intrinsic rewards and learn options that move along $\pm u_i$ [20], [22].

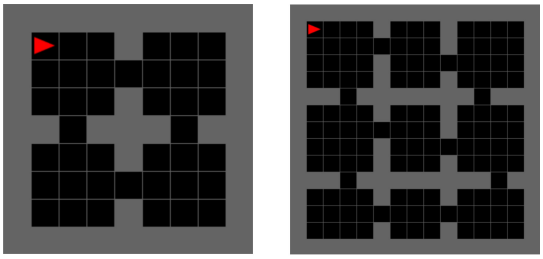
B. Orthogonal random probes and successor representations

RSO uses orthogonal random reward probes. Let $n = |\mathcal{S}|$ and sample $G \in \mathbb{R}^{n\times k}$ with

$$G_{s,i} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2). \quad (5)$$

A thin QR factorization $G = QR$ yields orthonormal columns in Q, and the i-th probe is

$$r_i(s) \doteq Q_{s,i}, \qquad \text{equivalently} \qquad \mathbf{r}_i = Q\, \mathbf{e}_i. \quad (6)$$



(a) MiniGrid-FourRooms (b) MiniGrid-Maze

Fig. 1. MiniGrid environments.

This removes within-batch redundancy while preserving isotropy in expectation [37]. Successor representations (SR) capture discounted future state occupancy under a policy [27]:

$$M^\pi(s, s') \doteq \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, \mathbf{1}\{S_t = s'\} \;\middle|\; S_0 = s\right]. \quad (7)$$

For a reward vector $\mathbf{r}$, the value function factorizes as

$$V_\pi(s) = \sum_{s'\in\mathcal{S}} M^\pi(s, s')\, r(s') = \big(M^\pi \mathbf{r}\big)(s). \quad (8)$$

With a feature map $\phi(s) \in \mathbb{R}^d$ and linear reward model $r(s) = w^\top \phi(s)$, successor features (SF) [28] are

$$\psi^\pi(s) \doteq \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t\, \phi(S_t) \;\middle|\; S_0 = s\right], \quad (9)$$

which give

$$V_\pi(s) = w^\top \psi^\pi(s). \quad (10)$$

These identities let orthogonal random rewards generate a family of value-function probes through SR/SF, forming the basis of RSO.

IV. Randomized Spectral Options

This section presents *Randomized Spectral Options* (RSO), based on two observations. **(i) Operator similarity:** Bellman systems and graph Laplacians both take the form "identity minus a Markov-like operator," linking value functions to Laplacian-style potentials. **(ii) Randomized spectral probing:** instead of explicitly computing eigenvectors, one can probe such operators with randomized right-hand sides to obtain smooth directions [38], [39]. RSO instantiates this idea by sampling orthogonal random rewards, learning a successor representation/features backbone under an exploration policy, and obtaining randomized value functions as linear readouts $V_i = M\, \mathbf{r}_i$. These value functions are then converted into shaping rewards to train signed options. In MiniGrid, the overall pipeline is shown in Fig. 2.

A. From Laplacian solutions to value functions: similarity and mismatch

Many option-discovery methods use Laplacian eigenvectors as smooth potentials over a transition graph. For the random-walk Laplacian $L_{\mathrm{rw}} = I - D^{-1}W$, a discrete Poisson equation is

$$L_{\mathrm{rw}}\, u = b, \quad (11)$$

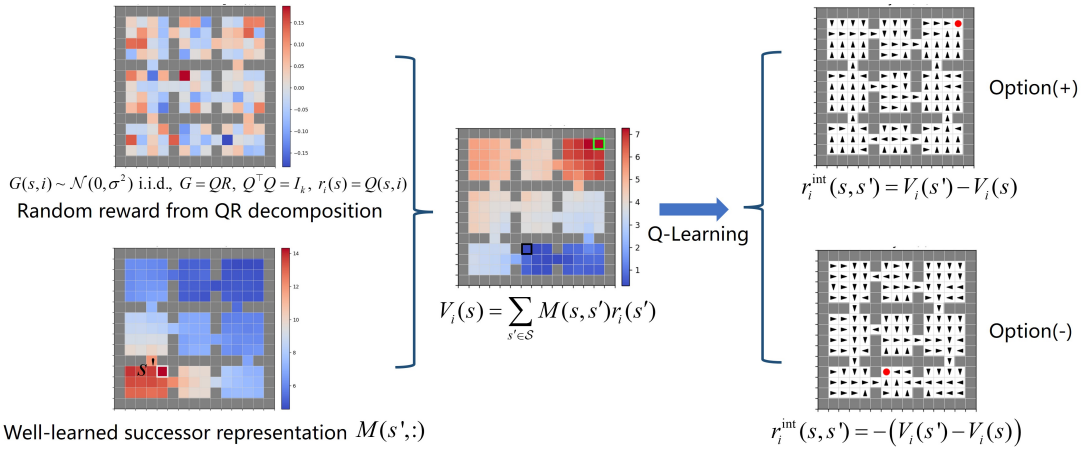


Fig. 2. Pipeline of Randomized Spectral Options (RSO). (1) Sample k orthogonal random reward probes $\{\mathbf{r}_i\}_{i=1}^k$ via thin QR factorization. (2) Under an exploration policy, learn the successor representation M with TD learning. (3) Compute randomized value functions (potentials) $V_i = M\,\mathbf{r}_i$. (4) Convert each V_i into a potential-based intrinsic reward and train two signed options that ascend/descend the induced potential field toward local maxima/minima.

and Laplacian eigenvectors solve $L_{\text{rw}} u = \lambda u$. For a fixed policy π, the Bellman equation is

$$(I - \gamma P_\pi)\,\mathbf{v}_\pi = \mathbf{r}_\pi, \qquad \mathbf{v}_\pi = (I - \gamma P_\pi)^{-1}\mathbf{r}_\pi. \tag{12}$$

If the graph is built from the policy-induced random walk so that $L_{\text{rw}} = I - P_\pi$, then under the usual reversibility/symmetrization assumption L_{rw} and $I - \gamma P_\pi$ share eigenvectors. Hence, for any reward vector $\mathbf{r}$,

$$V = (I - \gamma P_\pi)^{-1}\mathbf{r} = \sum_k \frac{\langle \mathbf{r}, \varphi_k \rangle}{1 - \gamma \eta_k}\,\varphi_k = \sum_k \frac{\langle \mathbf{r}, \varphi_k \rangle}{1 - \gamma(1 - \mu_k)}\,\varphi_k, \tag{13}$$

where $P_\pi \varphi_k = \eta_k \varphi_k$, and therefore $L_{\text{rw}} \varphi_k = (1 - \eta_k)\varphi_k = \mu_k \varphi_k$ with μ_k denoting the corresponding eigenvalue of the random-walk Laplacian. Thus Bellman solutions act as low-pass filtered graph potentials: random rewards generate smooth value-function probes without explicit eigendecomposition. The analogy is not exact—L_{rw} depends on a chosen graph, P_π depends on the policy and environment, and γ sets a temporal scale—but it motivates a randomized alternative to spectral option discovery.

B. Random-reward probing as randomized spectral decomposition

Equation (13) suggests a simple strategy: sample diverse rewards and map them through the Bellman/SR operator to obtain smooth potentials. In the tabular case, the SR gives

$$V_\pi(\cdot; \mathbf{r}) = M^\pi \mathbf{r}. \tag{14}$$

Accordingly, RSO samples k orthogonal rewards $\{\mathbf{r}_i\}_{i=1}^k$ by thin QR decomposition ((6)) and defines

$$V_i^\pi \doteq V_\pi(\cdot; \mathbf{r}_i) = M^\pi \mathbf{r}_i, \qquad V_i^\pi(s) = \sum_{s'} M^\pi(s,s')\, r_i(s'). \tag{15}$$

Each V_i^π is a smoothed version of a random signal, with smoothing determined by the transition dynamics under π. In this sense, orthogonal random rewards provide a randomized basis of option-discovery potentials without explicit spectral computation.

C. Option learning from randomized potentials

We convert each randomized value function into a potential $\Phi_i(s) \doteq V_i^\pi(s)$ and define a potential-based intrinsic reward [41]

$$r_{\text{pb}}^{(i)}(s,a,s') \doteq \gamma\,\Phi_i(s') - \Phi_i(s). \tag{16}$$

This reward encourages motion toward higher potential. To obtain bidirectional behaviors, we learn two signed options from each probe:

$$\begin{aligned} r_{\text{pb}}^{(i,+)}(s,a,s') &= \gamma\Phi_i(s') - \Phi_i(s), \\ r_{\text{pb}}^{(i,-)}(s,a,s') &= \Phi_i(s) - \gamma\Phi_i(s'). \end{aligned} \tag{17}$$

Each option can be trained with a standard value-based RL algorithm on the intrinsic return induced by $r_{\text{pb}}^{(i,\pm)}$. We use the full state space as the initiation set, and in the tabular case terminate when the option value becomes non-positive and with an additional geometric stopping probability. Fig. 3 shows the resulting RSO options in MiniGrid-FourRooms.

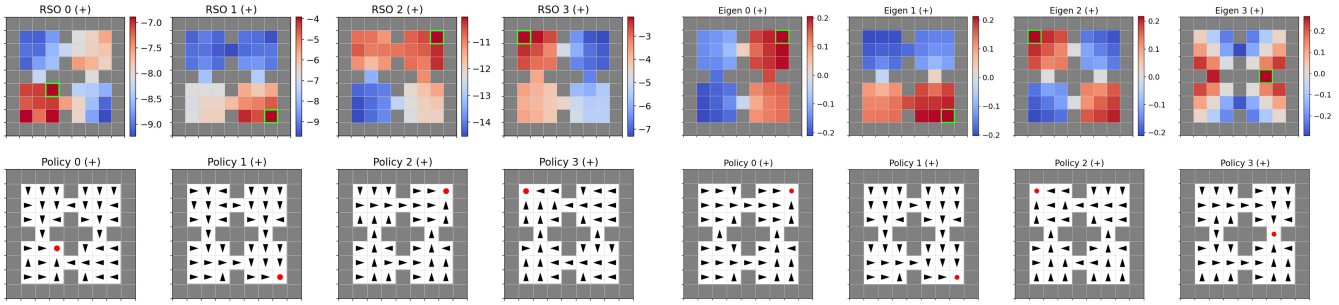
D. Function approximation with successor features and QR rewards

In high-dimensional or continuous domains, we replace the tabular SR with successor features. With feature map $\phi(s) \in \mathbb{R}^d$ and linear reward model $r(s) = w^\top \phi(s)$, the induced value is

$$V_\pi(s; w) = w^\top \psi^\pi(s), \tag{18}$$

where ψ^π is defined in (9). We sample k random reward weights and orthogonalize them by QR: if $G \in \mathbb{R}^{d \times k}$ has i.i.d. Gaussian entries and $G = QR$, then

$$w_i \doteq Q\,\mathbf{e}_i \in \mathbb{R}^d. \tag{19}$$



(a) RSO × 4 in MiniGrid-FourRooms **(b) Eigenoption × 4 in MiniGrid-FourRooms**

Fig. 3. Comparison between RSO and eigenoptions in MiniGrid-FourRooms. (a) Top row: heatmaps of randomized value functions V_i induced by orthogonal random reward probes; bottom row: the corresponding RSO policies trained to reach the maxima of V_i. (b) Top row: heatmaps of the Laplacian eigenvectors with the smallest 4 non-zero eigenvalues; bottom row: the corresponding eigenoption policies that ascend the induced potential fields toward their maxima.

Each w_i defines a feature-linear reward and corresponding potential

$$V_i^{\pi}(s) \doteq w_i^{\top} \psi^{\pi}(s). \tag{20}$$

We then reuse the same shaping construction in (16)–(17). The SF backbone is learned online with TD regression,

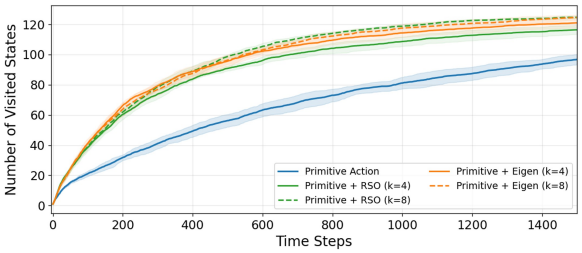
$$\mathcal{L}_{\mathrm{SF}}(\theta) = \mathbb{E}_{(s,s')\sim\mathcal{D}}\Big[\big\|\psi_\theta^{\pi}(s) - \big(\phi(s) + \gamma\, \psi_\theta^{\pi}(s')\big)\big\|_2^2\Big], \tag{21}$$

where $\mathcal{D}$ is a replay buffer collected under the exploration policy. Once ψ_θ^{π} is learned, adding a new option only requires sampling a new w_i and applying a linear readout, avoiding repeated spectral optimization and reducing marginal cost.

V. Experiments

We evaluate three aspects of RSO: exploration and downstream reward collection against Eigenoptions [21], the role of SR as an RL-friendly intermediate representation, and qualitative Atari visualizations in high-dimensional domains.

(a) State coverage curves in MiniGrid-Maze



(b) State coverage curves in MiniGrid-FourRooms

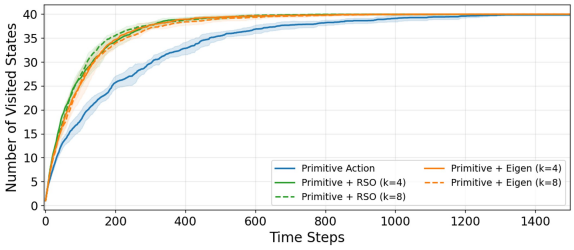


Fig. 4. State coverage curves in different MiniGrid environments.

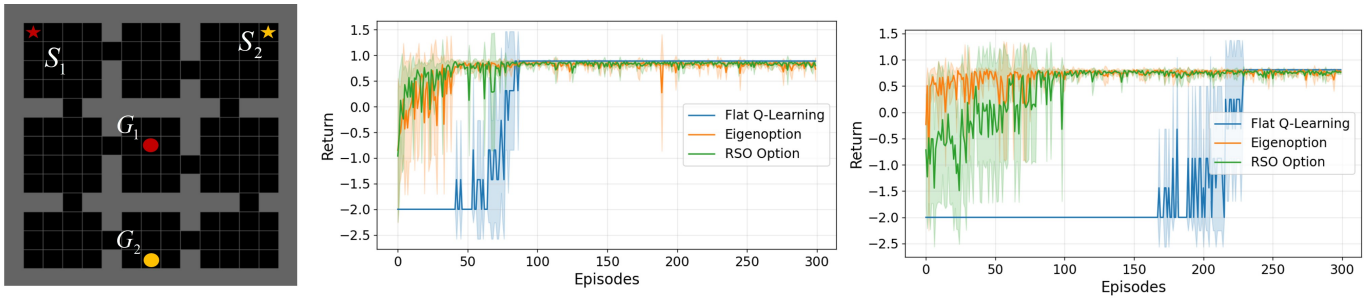
A. Evaluation in MiniGrid

We evaluate on the two MiniGrid environments in Fig. 1, using tabular state encodings and four primitive actions {up, down, left, right}. For each environment, we collect transitions for 5000 episodes of maximum length 500 from uniformly sampled non-wall start states while simultaneously learning the SR and estimating the random-walk Laplacian. We then train Q-learning option policies on the replay buffer, learning $k \in \{4, 8\}$ signed option pairs for each method. Unless otherwise noted, we use 5 random seeds, learning rate 0.05, discount factor 0.999 for SR learning, and discount factor 0.9 for option learning.

a) State-coverage efficiency.: Fig. 4 reports option-augmented random walks in MiniGrid-FourRooms and MiniGrid-Maze. At each step, the agent selects a primitive action with probability 0.9 and otherwise executes a uniformly sampled option; options terminate geometrically with mean horizon 15 or when $\max_a Q(s,a) \le 0$. For both $k = 4$ and $k = 8$, eigenoptions and RSO achieve similar coverage and clearly outperform primitive-action random walks.

b) Reward-collection performance.: Fig. 5 shows goal-reaching performance for two (start, goal) settings when tabular Q-learning uses the learned options only for exploration. The agent receives $+1$ on reaching the goal and -0.01 otherwise, and uses the same option termination rule as above; under ϵ-greedy exploration, an option is executed with probability 0.05. Both eigenoptions and RSO improve reward collection over primitive-action exploration, with the relative advantage depending on the start-goal configuration.

c) Using SR to improve RSO learning.: We further test SR-based intrinsic exploration in MiniGrid-Maze using SPIE [42]. Fig. 6 shows that SR-based exploration improves early coverage and accelerates SR convergence, highlighting that the exploration policy strongly affects both SR and Laplacian estimation. Since SR also supports limited transfer [30], [43], it may additionally provide useful warm-start signals for later option learning.



(a) (Statrt,Goal) setting **(b) Return curves for (S_1, G_1)** **(c) Return curves for (S_2, G_2)**

Fig. 5. Goal-reaching return curves in MiniGrid-Maze for two (start, goal) settings.

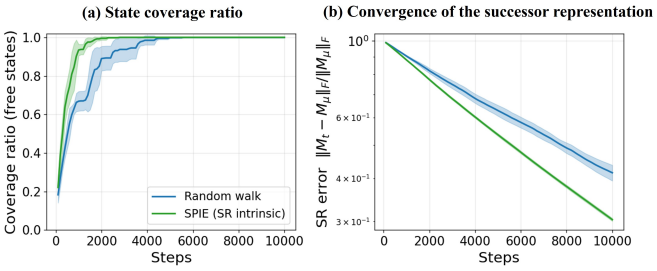


Fig. 6. SR-based intrinsic exploration improves state coverage and SR convergence.

B. RSO visualization in Atari 2600 games

For a qualitative test in high-dimensional domains, we evaluate RSO on *Montezuma's Revenge* and *Pitfall* from ALE [44]. We train a successor-feature network on stacks of four frames using RND for exploration [6], and use PPO with 8 workers, rollout length 128, and 10,000 rollouts [45]. After training, we sample 8 orthogonal reward directions, compute the corresponding value functions, and visualize representative maxima over 10 random-walk episodes in Fig. 7. The selected frames suggest that RSO highlights bottleneck-like states, including ropes, ladders, and life-loss moments in *Montezuma's Revenge* and room transitions in *Pitfall*.

VI. Conclusion

In this paper, we introduced *Randomized Spectral Options* (RSO), a simple RL-native approach to option discovery that replaces explicit spectral decomposition with randomized reward probing and successor representations/features. By sampling orthogonal random reward probes and mapping them through an SR/SF backbone, RSO produces a family of smooth value-function potentials at essentially the cost of a linear readout, enabling inexpensive and flexible expansion of the option set. These potentials naturally yield potential-based shaping rewards, from which we learn signed options that promote efficient exploration.

Empirically, RSO attains state-coverage and downstream performance comparable to eigenoption-based methods in MiniGrid, while offering a lower marginal cost for adding options and tighter integration with standard RL pipelines. Moreover, the Atari visualizations suggest that the randomized potentials can highlight semantically meaningful bottlenecks, supporting the view that SR/SF representations provide reusable structure beyond intrinsic-reward construction.

Several directions are promising for future work. First, it would be valuable to develop principled criteria for selecting and refreshing probe directions online as the policy and representation evolve. Second, integrating RSO with end-to-end option-critic training—potentially warm-starting option critics from SF readouts—could further improve data efficiency. Finally, extending the approach to partial observability and continuous control remains an open challenge.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant Nos. 32441113 and 62236007). The authors would also like to acknowledge the anonymous reviewers and chairs for their insightful comments, which helped improve this work.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.
[2] A. Ecoffet, J. Huizinga, J. Lehman, K. O. Stanley, and J. Clune, "First return, then explore," *Nature*, vol. 590, no. 7847, pp. 580–586, 2021.
[3] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel, "Vime: Variational information maximizing exploration," *Advances in neural information processing systems*, vol. 29, 2016.
[4] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *International conference on machine learning*. PMLR, 2017, pp. 2778–2787.
[5] G. Ostrovski, M. G. Bellemare, A. Oord, and R. Munos, "Count-based exploration with neural density models," in *International conference on machine learning*. PMLR, 2017, pp. 2721–2730.
[6] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," *arXiv preprint arXiv:1810.12894*, 2018.
[7] S. Pateria, B. Subagdja, A.-h. Tan, and C. Quek, "Hierarchical reinforcement learning: A comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 5, pp. 1–35, 2021.
[8] O. Nachum, H. Tang, X. Lu, S. Gu, H. Lee, and S. Levine, "Why does hierarchy (sometimes) work so well in reinforcement learning?" *arXiv preprint arXiv:1909.10618*, 2019.
[9] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum, "Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation," *Advances in neural information processing systems*, vol. 29, 2016.

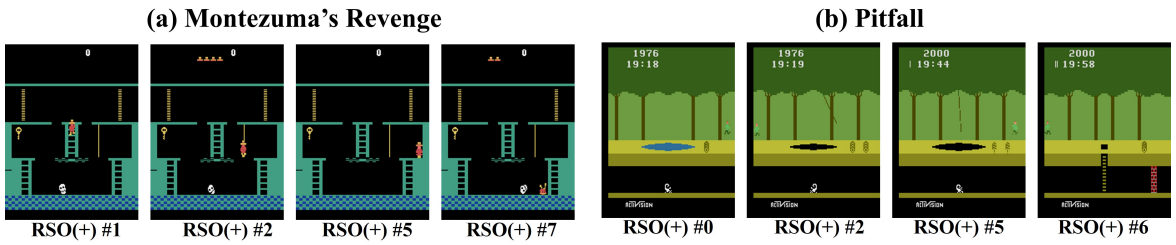


Fig. 7. Four representative randomized spectral options in Atari 2600 games.

[10] R. S. Sutton, D. Precup, and S. Singh, "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning," *Artificial intelligence*, vol. 112, no. 1-2, pp. 181–211, 1999.

[11] P.-L. Bacon, J. Harb, and D. Precup, "The option-critic architecture," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[12] M. C. Machado, M. G. Bellemare, and M. Bowling, "A laplacian framework for option discovery in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2295–2304.

[13] M. Klissarov and M. C. Machado, "Deep laplacian-based options for temporally-extended exploration," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[14] A. Solway, C. Diuk, N. Córdova, D. Yee, A. G. Barto, Y. Niv, and M. M. Botvinick, "Optimal behavioral hierarchy," *PLoS computational biology*, vol. 10, no. 8, p. e1003779, 2014.

[15] Y. Jinnai, D. Abel, D. Hershkowitz, M. Littman, and G. Konidaris, "Finding options that minimize planning time," in *International Conference on Machine Learning*. PMLR, 2019, pp. 3120–3129.

[16] Y. Jinnai, J. W. Park, D. Abel, and G. Konidaris, "Discovering options for exploration by minimizing cover time," in *International Conference on Machine Learning*. PMLR, 2019, pp. 3130–3139.

[17] A. Ivanov, A. Bagaria, and G. Konidaris, "Discovering options that minimize average planning time," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 17, 2025, pp. 17 573–17 581.

[18] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[19] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[20] S. Mahadevan and M. Maggioni, "Proto-value functions: A laplacian framework for learning representation and control in markov decision processes." *Journal of Machine Learning Research*, vol. 8, no. 10, 2007.

[21] M. C. Machado, M. G. Bellemare, and M. Bowling, "A laplacian framework for option discovery in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2295–2304.

[22] M. C. Machado, C. Rosenbaum, X. Guo, M. Liu, G. Tesauro, and M. Campbell, "Eigenoption discovery through the deep successor representation," *arXiv preprint arXiv:1710.11089*, 2017.

[23] M. Klissarov and M. C. Machado, "Deep laplacian-based options for temporally-extended exploration," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.

[24] Y. Wu, G. Tucker, and O. Nachum, "The laplacian in rl: Learning representations with efficient approximations," in *International Conference on Learning Representations*, 2019.

[25] D. Gomez, M. Bowling, and M. C. Machado, "Proper laplacian representation learning," in *International Conference on Learning Representations*, 2024.

[26] D. A. Spielman and S.-H. Teng, "Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 3, pp. 835–885, 2014.

[27] P. Dayan, "Improving generalization for temporal difference learning: The successor representation," *Neural computation*, vol. 5, no. 4, pp. 613–624, 1993.

[28] T. D. Kulkarni, A. Saeedi, S. Gautam, and S. J. Gershman, "Deep successor reinforcement learning," *arXiv preprint arXiv:1606.02396*, 2016.

[29] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver, "Successor features for transfer in reinforcement learning," *Advances in neural information processing systems*, vol. 30, 2017.

[30] A. Barreto, D. Borsa, J. Quan, T. Schaul, D. Silver, M. Hessel, D. Mankowitz, A. Zidek, and R. Munos, "Transfer in deep reinforcement learning using successor features and generalised policy improvement," in *International Conference on Machine Learning*. PMLR, 2018, pp. 501–510.

[31] M. C. Machado, A. Barreto, D. Precup, and M. Bowling, "Temporal abstraction in reinforcement learning with the successor representation," *Journal of machine learning research*, vol. 24, no. 80, pp. 1–69, 2023.

[32] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine learning*, vol. 3, no. 1, pp. 9–44, 1988.

[33] F. R. Chung, *Spectral graph theory*. American Mathematical Soc., 1997, vol. 92.

[34] K. Wang, K. Zhou, Q. Zhang, J. Shao, B. Hooi, and J. Feng, "Towards better laplacian representation in reinforcement learning with generalized graph drawing," in *International Conference on Machine Learning*. PMLR, 2021, pp. 11 003–11 012.

[35] J. Farebrother, J. Greaves, R. Agarwal, C. Le Lan, R. Goroshin, P. S. Castro, and M. G. Bellemare, "Proto-value networks: Scaling representation learning with auxiliary tasks," in *International Conference on Learning Representations*, 2023.

[36] C. Lyle, M. Rowland, G. Ostrovski, and W. Dabney, "On the effect of auxiliary tasks on representation dynamics," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 1–9.

[37] F. Mezzadri, "How to generate random matrices from the classical compact groups," *arXiv preprint math-ph/0609050*, 2006.

[38] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM review*, vol. 53, no. 2, pp. 217–288, 2011.

[39] D. P. Woodruff *et al.*, "Sketching as a tool for numerical linear algebra," *Foundations and Trends® in Theoretical Computer Science*, vol. 10, no. 1–2, pp. 1–157, 2014.

[40] M. Chevalier-Boisvert, B. Dai, M. Towers, R. Perez-Vicente, L. Willems, S. Lahlou, S. Pal, P. S. Castro, and J. Terry, "Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks," in *Advances in Neural Information Processing Systems 36, New Orleans, LA, USA*, December 2023.

[41] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward transformations: Theory and application to reward shaping," in *Icml*, vol. 99. Citeseer, 1999, pp. 278–287.

[42] C. Yu, N. Burgess, M. Sahani, and S. J. Gershman, "Successor-predecessor intrinsic exploration," in *Advances in neural information processing systems*, 2023.

[43] L. Lehnert, S. Tellex, and M. L. Littman, "Advantages and limitations of using successor features for transfer in reinforcement learning," *arXiv preprint arXiv:1708.00102*, 2017.

[44] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The arcade learning environment: An evaluation platform for general agents," *Journal of artificial intelligence research*, vol. 47, pp. 253–279, 2013.

[45] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.