A Proof of Theorem 4.1

For convenience, we use G_i to represent the i-th factor graph and its adjacency matrix. Also, we denote the number of nodes in G_i as K_i and an identity matrix with K_i diagonal elements as I_{K_i}.

Proof. The normalized laplacian matrix of the Kronecker product of n factor graphs $\otimes_{i=1}^{n} G_i$ can be written as:

$$\mathcal{L}_{\otimes_{i=1}^{n} G_i} = \quad \otimes_{i=1}^{n} I_{K_i} - (\otimes_{i=1}^{n} D_{G_i}^{-\frac{1}{2}})(\otimes_{i=1}^{n} G_i)(\otimes_{i=1}^{n} D_{G_i}^{-\frac{1}{2}}). \tag{8}$$

Using the property of the Kronecker product of matrices, $(A \otimes B)(C \otimes D) = AC \otimes BD$, we can obtain that:

$$\begin{aligned} \mathcal{L}_{\otimes_{i=1}^{n} G_i} &= \otimes_{i=1}^{n} I_{K_i} - \otimes_{i=1}^{n} (D_{G_i}^{-\frac{1}{2}} G_i D_{G_i}^{-\frac{1}{2}}) \\ &= \otimes_{i=1}^{n} I_{K_i} - \otimes_{i=1}^{n} (I_{K_i} - \mathcal{L}_{G_i}). \end{aligned} \tag{9}$$

Let $\{\lambda_{k_1}^{G_1}\}, \{\lambda_{k_2}^{G_2}\}, \ldots, \{\lambda_{k_n}^{G_n}\}$ be the eigenvalues of matrices $\mathcal{L}_{G_1}, \mathcal{L}_{G_2}\ , \ldots, \mathcal{L}_{G_n}$, with the corresponding orthonormal eigenvectors $\{v_{k_1}^{G_1}\}, \{v_{k_2}^{G_2}\}, \ \ldots, \{v_{k_n}^{G_n}\}$, where $k_i = 1, 2, \ldots, K_i$. Also, denote the diagonal matrices, whose diagonal elements are the values $\{1 - \lambda_{k_1}^{G_1}\}, \{1 - \lambda_{k_2}^{G_2}\}, \ldots, \{1 - \lambda_{k_n}^{G_n}\}$, as $\Lambda_{G_1}, \Lambda_{G_2}, \ldots, \Lambda_{G_n}$, and the square matrices containing the eigenvectors $\{v_{k_1}^{G_1}\}, \{v_{k_2}^{G_2}\}, \ldots, \{v_{k_n}^{G_n}\}$ as the column vectors as $V_{G_1}, V_{G_2}, \ldots, V_{G_n}$. Using the spectral decomposition of the matrix $I_{K_i} - \mathcal{L}_{G_i}$ $(i = 1, \ldots, n)$, we can obtain that:

$$\begin{aligned} \mathcal{L}_{\otimes_{i=1}^{n} G_i} &= \otimes_{i=1}^{n} I_{K_i} - \otimes_{i=1}^{n} (V_{G_i} \Lambda_{G_i} V_{G_i}^T) \\ &= \otimes_{i=1}^{n} I_{K_i} - (\otimes_{i=1}^{n} V_{G_i})(\otimes_{i=1}^{n} \Lambda_{G_i})(\otimes_{i=1}^{n} V_{G_i})^T \\ &= (\otimes_{i=1}^{n} V_{G_i})(\otimes_{i=1}^{n} I_{K_i} - \otimes_{i=1}^{n} \Lambda_{G_i})(\otimes_{i=1}^{n} V_{G_i})^T, \end{aligned} \tag{10}$$

since $\otimes_{i=1}^{n} I_{K_i} = \otimes_{i=1}^{n} [(V_{G_i})(V_{G_i})^T] = (\otimes_{i=1}^{n} V_{G_i})(\otimes_{i=1}^{n} V_{G_i})^T$. This implies that $\mathcal{L}_{\otimes_{i=1}^{n} G_i}$ has eigenvalues $\{[1 - \prod_{i=1}^{n}(1 - \lambda_{k_i}^{G_i})]\}$ and corresponding eigenvectors $\{\otimes_{i=1}^{n} v_{k_i}^{G_i}\}$.

Then, we let $\Lambda = \otimes_{i=1}^{n} I_{K_i} - \otimes_{i=1}^{n} \Lambda_{G_i}$ and $D = \otimes_{i=1}^{n} D_{G_i}$. Since the normalized Laplacian could be expressed in terms of Laplacian matrix as $\mathcal{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$, we can get $L_{\otimes_{i=1}^{n} G_i}(\otimes_{i=1}^{n} V_{G_i}) = D^{\frac{1}{2}} \mathcal{L}_{\otimes_{i=1}^{n} G_i} D^{\frac{1}{2}} (\otimes_{i=1}^{n} V_{G_i})$. By making assumption (used and testified in [24, 42]) that $D_{G_i}^{\frac{1}{2}} V_{G_i} \approx V_{G_i} D_{G_i}^{\frac{1}{2}}$, for $i = 1, 2, \ldots, n$, we can derive that:

$$\begin{aligned} L_{\otimes_{i=1}^{n} G_i}(\otimes_{i=1}^{n} V_{G_i}) &\approx D^{\frac{1}{2}} \mathcal{L}_{\otimes_{i=1}^{n} G_i}(\otimes_{i=1}^{n} V_{G_i}) D^{\frac{1}{2}} \\ &= D^{\frac{1}{2}} \Lambda (\otimes_{i=1}^{n} V_{G_i}) D^{\frac{1}{2}}. \end{aligned} \tag{11}$$

After applying the same assumption again, we finally obtain that:

$$L_{\otimes_{i=1}^{n} G_i}(\otimes_{i=1}^{n} V_{G_i}) \approx (D\Lambda)(\otimes_{i=1}^{n} V_{G_i}). \tag{12}$$

Based on Equation (12), we can get an approximation of the Laplacian spectrum, including the eigenvalues and corresponding eigenvectors, of the Kronecker product of n factor graphs, shown as Theorem 4.1.

Next, we will prove that the estimated eigenvalues $\mu_{k_1 k_2, \ldots, k_n}$ in Theorem 4.1 are non-negative. It is obvious that $d_{k_i}^{G_i}$ and $\prod_{i=1}^{n} d_{k_i}^{G_i}$ are non-negative. Then, we need to prove $[1 - \prod_{i=1}^{n}(1 - \lambda_{k_i}^{G_i})]$ is non-negative. We know that if λ is an eigenvalue of a normalized Laplacian matrix, we can get $0 \leq \lambda \leq 2$. Hence, $-1 \leq 1 - \lambda_{k_i}^{G_i} \leq 1$, for $i = 1, 2, \ldots, n$. Based on this, we can get that $\left| \prod_{i=1}^{n}(1 - \lambda_{k_i}^{G_i}) \right| \leq 1$ and thus $[1 - \prod_{i=1}^{n}(1 - \lambda_{k_i}^{G_i})]$ is non-negative. □

B Basic Conceptions and Notations

Markov Decision Process (MDP): The RL problem can be described with an MDP, denoted by $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition function, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to R^1$ is the reward function, and $\gamma \in (0, 1]$ is the discount factor.

State transition graph in an MDP: The state transitions in $\mathcal{M}$ can be modelled as a state transition graph $G = (V_G, E_G)$, where V_G is a set of vertices representing the states in $\mathcal{S}$, and E_G is a set of undirected edges representing state adjacency in $\mathcal{M}$. We note that:

Remark B.1. There is an edge between state s and s' (i.e., s and s' are adjacent) if and only if $\exists\, a \in \mathcal{A},\ s.t.\ \mathcal{P}(s, a, s') > 0 \ \vee\ \mathcal{P}(s', a, s) > 0$.

The adjacency matrix A of G is an $|\mathcal{S}| \times |\mathcal{S}|$ matrix whose (i, j) entry is 1 when s_i and s_j are adjacent, and 0 otherwise. The degree matrix D is a diagonal matrix whose entry (i, i) equals the number of edges incident to s_i. The Laplacian matrix of G is defined as $L = D - A$. Its second smallest eigenvalue $\lambda_2(L)$ is called the algebraic connectivity of the graph G, and the corresponding normalized eigenvector is called the Fiedler vector [4]. Last, the normalized Laplacian matrix is defined as $\mathcal{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$.

C Finding the Fiedler vector for the illustrative example shown in Figure 1(a)

(1) Compute the normalized Laplacian matrix of G_1 and G_2, namely $\mathcal{L}_1$ and $\mathcal{L}_2$:

$$\mathcal{L}_1 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix},\ \mathcal{L}_2 = \begin{bmatrix} 1 & -\frac{1}{\sqrt{2}} & 0 & 0 \\ -\frac{1}{\sqrt{2}} & 1 & -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 1 & -\frac{1}{\sqrt{2}} \\ 0 & 0 & -\frac{1}{\sqrt{2}} & 1 \end{bmatrix}. \tag{13}$$

(2) Compute the eigenvalues and eigenvectors of $\mathcal{L}_1$ and $\mathcal{L}_2$:

$$\lambda_1^{G_1} = 0,\ \lambda_2^{G_1} = 2,\ v_{1:2}^{G_1} = \frac{1}{\sqrt{2}} \left[\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 1 \end{bmatrix} \right]. \tag{14}$$

$$\lambda_1^{G_2} = 0,\ \lambda_2^{G_2} = 0.5,\ \lambda_3^{G_2} = 1.5,\ \lambda_4^{G_2} = 2, \tag{15}$$

$$v_{1:4}^{G_2} = \frac{1}{\sqrt{3}} \left[\begin{bmatrix} \frac{1}{\sqrt{2}} \\ 1 \\ 1 \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \begin{bmatrix} -1 \\ -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \\ 1 \end{bmatrix}, \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -1 \\ 1 \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \right]. \tag{16}$$

(3) Compute the degree list of G_1 and G_2 (sorted in ascending order), namely d^{G_1} and d^{G_2}:

$$d^{G_1} = [1,\ 1]^T,\ d^{G_2} = [1,\ 1,\ 2,\ 2]^T. \tag{17}$$

(4) According to Theorem 4.1, we can get two approximations of the Fiedler vector:

$$v_{11} = v_1^{G_1} \otimes v_1^{G_2} = \frac{1}{\sqrt{6}} \left[\frac{1}{\sqrt{2}},\ 1,\ 1,\ \frac{1}{\sqrt{2}},\ \frac{1}{\sqrt{2}},\ 1,\ 1,\ \frac{1}{\sqrt{2}} \right]^T, \tag{18}$$

$$v_{24} = v_2^{G_1} \otimes v_4^{G_2} = \frac{1}{\sqrt{6}} \left[-\frac{1}{\sqrt{2}},\ 1,\ -1,\ \frac{1}{\sqrt{2}},\ \frac{1}{\sqrt{2}},\ -1,\ 1,\ -\frac{1}{\sqrt{2}} \right]^T. \tag{19}$$

D Pseudo Code of Multi-agent Covering Option Discovery

Algorithm 1 Multi-agent Covering Option Discovery

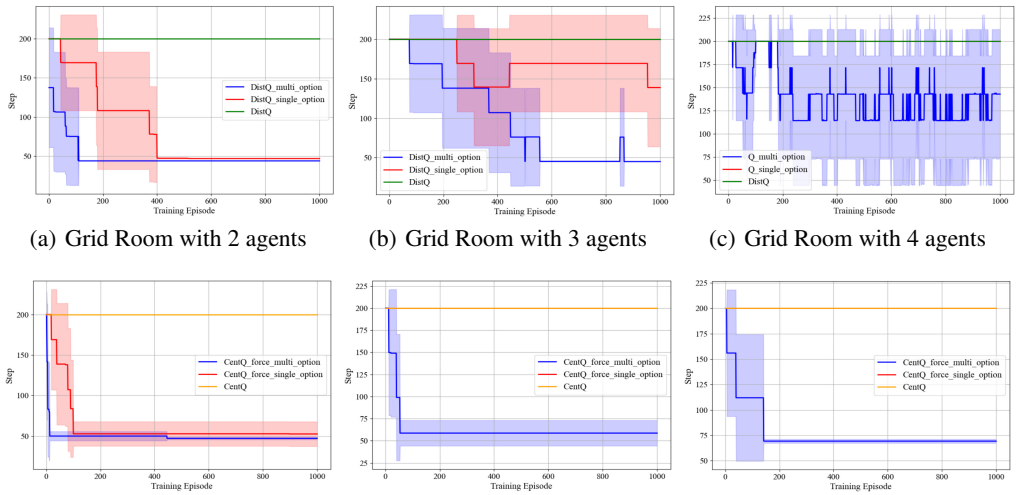
Input: # of agents n, list of adjacency matrices $A_{1:n}$, # of options to generate tot_num
Output: list of multi-agent options Ω
$\Omega \leftarrow \emptyset$, $cur\_num \leftarrow 0$
while $cur\_num < tot\_num$ **do**
    Collect the degree list $D_{1:n}$ of each individual state transition graph according to $A_{1:n}$
    Obtain the list of normalized laplacian matrices $\mathcal{L}_{1:n}$ corresponding to $A_{1:n}$
    Calculate the eigenvalues U_i and corresponding eigenvectors V_i for each $\mathcal{L}_i$ and collect them as $U_{1:n}$ and $V_{1:n}$
    Obtain the Fielder vector F of the joint state space using Theorem 4.1 with $D_{1:n}$, $U_{1:n}$, $V_{1:n}$
    Collect the list of joint states corresponding to the minimum or maximum in F, named MIN and MAX respectively
    Convert each joint state s_{joint} in MIN and MAX to $(s_1, \cdots, s_n)$, where s_i is the corresponding individual state of agent i, based on the equation:
    $ind(s_{joint}) = ((ind(s_1) * dim(A_2) + \cdots + ind(s_{n-1})) * dim(A_n) + ind(s_n)$
    where $dim(A_i)$ is the dimension of A_i, $ind(s_i)$ is the index of s_i (indexed from 0) in the state space of agent i
    Generate a new list of options Ω' through *GenerateOptions*
    $\Omega \leftarrow \Omega \cup \Omega'$, $cur\_num \leftarrow cur\_num + len(\Omega')$
    Update $A_{1:n}$ through *UpdateAdjacencyMatrices*
end while
Return Ω

function *GenerateOptions*(MIN, MAX)
    $\Omega' \leftarrow \emptyset$
    for $s = (s_1, \cdots, s_n)$ in $(MIN \cup MAX)$ **do**
        Define the initiation set I_ω as the joint states in the known region of the joint state space
        Define the termination condition as:
        $$\beta_\omega(s_{cur}) \leftarrow \begin{cases} 1 & if\ (s_{cur} == s)\ or\ (s_{cur}\ is\ unknown) \\ 0 & otherwise \end{cases}$$
        where s_{cur} is the current joint state
        Train the intra-option policy $\pi_\omega = (\pi_\omega^1, \cdots, \pi_\omega^n)$, where π_ω^i maps the individual state of agent i to its action aiming at leading agent i from any state in its initiation set to its termination state s_i
        $\Omega' \leftarrow \Omega' \cup \{< I_\omega, \pi_\omega, \beta_\omega >\}$
    end for
    Return Ω'
end function

function *UpdateAdjacencyMatrices*(MIN, MAX)
    for $s_{min} = (s_{min}^1, \cdots, s_{min}^n)$ in MIN **do**
        for $s_{max} = (s_{max}^1, \cdots, s_{max}^n)$ in MAX **do**
            for $i = 1$ to n **do**
                $A_i[ind(s_{min}^i)][ind(s_{max}^i)] = 1$
                $A_i[ind(s_{max}^i)][ind(s_{min}^i)] = 1$
            end for
        end for
    end for
end function

E Additional Evaluation Results

E.1 n-agent Grid Room Task



(a) Grid Room with 2 agents (b) Grid Room with 3 agents (c) Grid Room with 4 agents

(d) Grid Room with 2 agents (e) Grid Room with 3 agents (f) Grid Room with 4 agents

Figure 10: Evaluation on n-agent Grid Room tasks: (a)-(c) using Distributed Q-Learning as the high-level policy. The performance improvement of our approach is more significant as the number of agents increases. (d)-(f) using Centralized Q-Learning + Force as the high-level policy. Agents with single-agent options start to fail since the 3-agent case. Also, it can be observed that the centralized way to utilize the n-agent options leads to faster convergence.

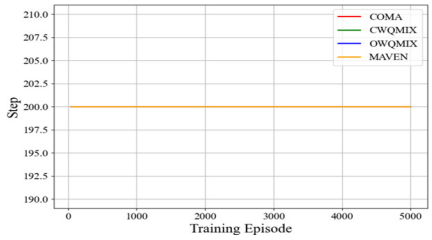
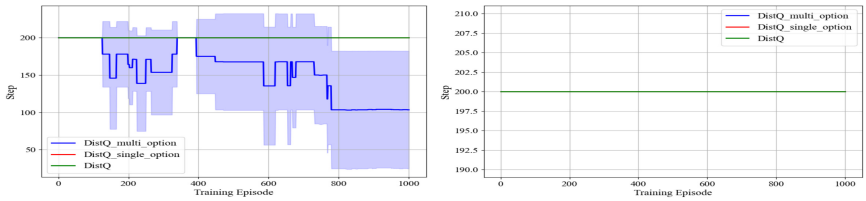
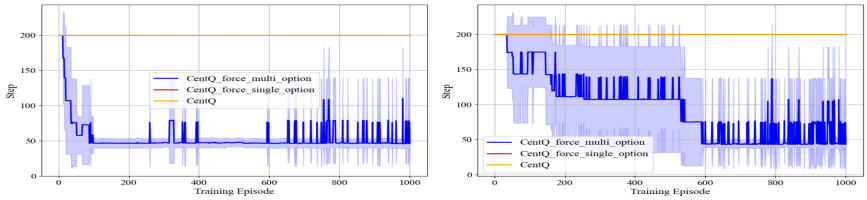


Figure 11: Performance of SOTA MARL algorithms: COMA, MAVEN, Weighted QMIX, on the 4-agent Grid Maze Task (Figure 5(c)5(f)). For each algorithm, the experiment is repeated three times with different random seeds (codes are available in the provided link). On this discrete problem setting, these SOTA algorithms do not show better performance than the tabular Q-learning we use as baselines. Also, our method performs much better on the same task.

E.2 $m \times n$-agent Grid Room Task



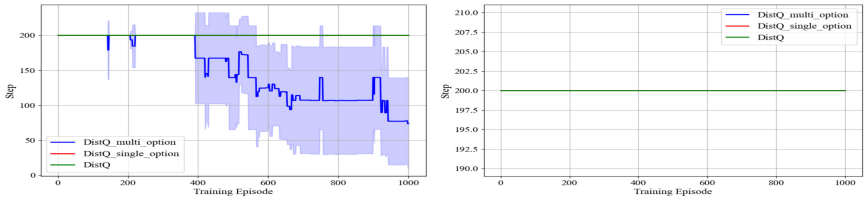
(a) 2×2 agents (b) 3×2 agents



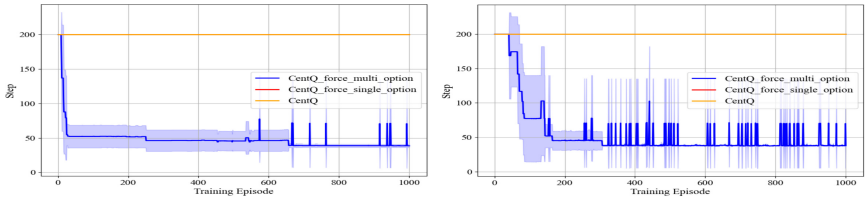
(c) 2×2 agents (d) 3×2 agents

Figure 12: Comparisons on the $m \times n$ Grid Room tasks: (a)(b) Distributed Q-Learning; (c)(d) Centralized Q-Learning + Force.

E.3 n-agent Grid Room Task with random grouping



(a) Grid Room with 4 agents (b) Grid Room with 6 agents



(c) Grid Room with 4 agents (d) Grid Room with 6 agents

Figure 13: Comparisons on the n-agent Grid Room tasks with random grouping: (a)(b) Distributed Q-Learning; (c)(d) Centralized Q-Learning + Force.

E.4 A quantitative study on the approximation error of the joint transition graph with Kronecker-product approximation

In this section, we evaluate the approximation error when we use $\otimes_{i=1}^{n} G_i$ as a factorized approximation of $\widetilde{G}$, regarding option discovery. We test on a simplified Grid Room task shown as Figure 14, where two agents are represented as triangles and the goal area is labelled as circles. The time complexity to compute the groundtruth of the Laplacian spectrum of the joint state transition graph is cubic with the number of the joint states which grows exponentially with the number of agents. For example, there are 74 states for each agent in Figure 8, and the computation complexity is already $\mathcal{O}(10^{11})$ (i.e., $(74^2)^3$).

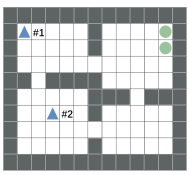
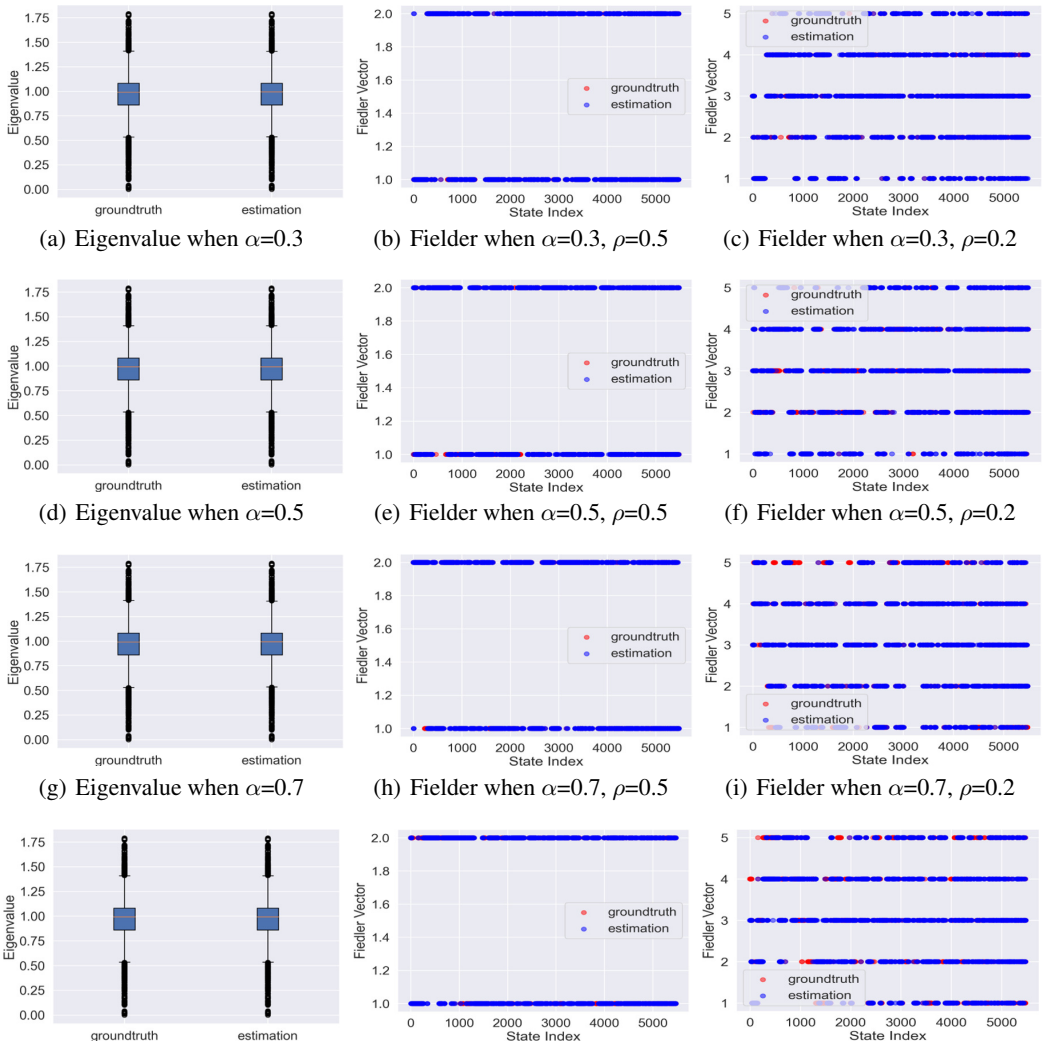


Figure 14: Simulator



(a) Eigenvalue when α=0.3 (b) Fielder when α=0.3, ρ=0.5 (c) Fielder when α=0.3, ρ=0.2

(d) Eigenvalue when α=0.5 (e) Fielder when α=0.5, ρ=0.5 (f) Fielder when α=0.5, ρ=0.2

(g) Eigenvalue when α=0.7 (h) Fielder when α=0.7, ρ=0.5 (i) Fielder when α=0.7, ρ=0.2

(j) Eigenvalue when α=0.9 (k) Fielder when α=0.9, ρ=0.5 (l) Fielder when α=0.9, ρ=0.2

Figure 15: Comparison between the groundtruth and estimation of the Laplacian spectrum of the joint state transition graph as transition influence increases. The first column shows the distribution of the eigenvalues, from which we can see the distribution of the estimated eigenvalues is very close to the groundtruth. The second column shows the Fiedler vector on the joint state space, where we partition the states into 2 clusters (i.e., $\rho = 0.5$) and the states with the value in the Fiedler vector that is lower than the median is labelled as 1 and the others are labelled as 2. Similarly, in the third column, the states are partitioned into 5 clusters (i.e., $\rho = 0.2$), where the value of the states labeled as i is between the $(i-1)$-th and i-th quintile of the values in the Fielder vector. In the third column, the number of the unmatched groundtruth (i.e., red points) goes up as α increases, showing that approximation error increases with α.

As mentioned in Section 4.2, the approximation error occurs when the state transitions of an agent are influenced by others. However, the state transition influence among agents, e.g., collisions and blocking, would most likely result in local perturbations of the transition graph and thus is inconsequential to global properties of $\widetilde{G}$. Therefore, approximating $\widetilde{G}$ by $\otimes_{i=1}^{n} G_i$ allows efficient option discovery. In Figure 8, we have evaluated on the case where an agent's state transitions will be influenced by the others' states (i.e., blocking by other agents when going ahead). However, the transition influence for an agent may also come from the action choices of the other agents. Thus, in this scenario (i.e., Figure 14), we set Agent #1 as the leading agent and Agent #2 will follow the moving direction of Agent #1 with the probability α, so the state transition of Agent #2

α	0.3	0.5	0.7	0.9
Algebraic Connectivity ($\times 10^{-3}$)	8.0988	8.1153	8.0996	7.9763
Estimation Accuracy of Fielder when ρ=0.5 (%)	100	100	100	100
Estimation Accuracy of Fielder when ρ=0.2 (%)	99.9	96.2	89.8	79.4

Table 1: Numeric results on the groundtruth of the algebraic connectivity and accuracy of the Fielder estimation, as the transition influence increases.

can be influenced by the action choice of Agent #1. With a certain α, we collect a million state transitions (i.e., $\{(s, a, s')\}$) through Monte Carlo sampling, based on which we can build the joint state transition graph $\widetilde{G}$ and the individual state transition graphs G_i ($i = 1, 2$) and then get $\otimes_{i=1}^{2} G_i$.

As shown in Figure 15 and Table 1, we set α as 0.3, 0.5, 0.7, and 0.9, respectively, to show the approximation error as the transition influence goes up. For the covering option discovery, we only care about the Laplacian spectrum of the state transition graph, especially the algebraic connectivity and Fielder vector. We validate through experiments that the approximation error on the algebraic connectivity and Fielder vector caused by the transition influence among the agents is minor, and thus we can still accurately identify multi-agent options.

In the first column of Figure 15, we visualize the distribution of the eigenvalues corresponding to the Laplacian matrix of $\widetilde{G}$ (i.e., groundtruth) and $\otimes_{i=1}^{2} G_i$ (i.e., estimation). It can be observed that the estimated distribution is very close to the groundtruth. Further, we show the algebraic connectivity of $\widetilde{G}$ when setting α as 0.3, 0.5, 0.7, 0.9 in Table 1. The algebraic connectivity of our estimation $\otimes_{i=1}^{2} G_i$ is 8.1131×10^{-3} (invariant to α), which is close to the groundtruth values.

In the second and third column of Figure 15, we compare the estimated Fiedler vector with the groundtruth. As mentioned in Section 4.2, we only need to identify areas in the state space with relatively low or high values in the Fielder vector and connect them with options. Thus, we partition the states into 2 clusters (i.e., $\rho = 1/2 = 0.5$) according to the median of the values in the Fiedler vector, or partition them into 5 clusters (i.e., $\rho = 1/5 = 0.2$) based on the quintile. We use the Fiedler vector of $\widetilde{G}$ as the groundtruth and compare it with the estimated Fiedler vectors, by comparing the label (i.e., which cluster it belongs to) of each state. It can be observed that the number of the unmatched groundtruth (shown as red points) increases with α. Further, we note that the states with the lowest or highest value in the Fiedler vector (i.e., MIN or MAX) are the subgoals based on which we define the options, so the estimation accuracy of these states are directly related to the option discovery. In Table 1, we show the estimation accuracy of the subgoals. The third row corresponds to defining the states with the lowest or highest 20% values in the Fielder vector as the subgoals, which is also the setup we use for option discovery. It can be observed that even if in conditions where the state transition influence is heavy (i.e., $\alpha = 0.9$), we can still estimate about 80% of the subgoals correctly and build options toward them accordingly.

These results empirically validate our statement that approximating $\widetilde{G}$ with $\otimes_{i=1}^{n} G_i$ allows efficient option discovery in cases where transition influence exists. We will consider a theoretical characterization of the impact of approximation errors in the future work.