A Spectral Method of Moments for Hierarchical Imitation Learning ⋆

Nguyen Nguyen * **Timothy L. Molloy** ** **Girish N. Nair** ***
Ioannis Ch. Paschalidis *

* *College of Engineering and Hariri Institute for Computing & Computational Science & Engineering, Boston University, 8 St. Mary's St., Boston, MA 02215, USA, E-mail:* {nguyenpn, yannisp}@bu.edu
** *Australian National University, Canberra, Australia, E-mail:* timothy.molloy@anu.edu.au
*** *University of Melbourne, Melbourne, Australia, E-mail:* gnair@unimelb.edu.au

Abstract: Recent empirical success has led to a rise in popularity of the options framework for Hierarchical Reinforcement Learning (HRL). This framework tackles the scalability problem in Reinforcement Learning (RL) by introducing a layer of abstraction (i.e. high-level options) over the (low-level) decision process. Hierarchical Imitation Learning (HIL) is the problem of learning low-level and high-level policies within HRL from expert demonstrations consisting only of the low-level actions and states, with the high-level options being hidden (or latent). Due to the latent options, recent work on HIL has focused on the development of Expectation-Maximization (EM) algorithms inspired by approaches such as the celebrated Baum-Welch algorithm for hidden Markov models (HMMs). In this work, we take a different approach and derive a new HIL framework inspired by the spectral method of moments for HMMs. The method of moments offers global and consistent convergence under mild regulatory conditions, whilst only requiring one sweep through the data set of state and action pairs, giving it a competitive run time.

Keywords: Learning for control, Machine learning, Markov Decision Processes, Options, Imitation Learning, Method of Moments.

1. INTRODUCTION

Hierarchical Reinforcement Learning (HRL) seeks to address the scalability problem of Reinforcement Learning (RL) by introducing layers of abstraction over the decision process, enabling general sweeping decisions over large epochs and smaller specific decisions on finer (more granular) epochs (Sutton et al., 1999; Barto and Mahadevan, 2003). The success of HRL relies on discovering suitable abstractions. In the literature, the problem of discovering suitable abstractions has been tackled both separately and in conjunction (in a single end-to-end process) with learning the optimal policy (Barto and Mahadevan, 2003). In specific instances where expert demonstrations are available, the process of discovering abstractions and learning optimal policies can be accelerated via *Hierarchical Imitation Learning* (HIL). Specifically, HIL involves computing a hierarchy of policies from expert demonstrations and is the extension of *Imitation Learning* (IL) to HRL. In this paper, we develop a novel HIL approach for the HRL with options framework of (Sutton et al., 1999).

⋆ Research partially supported by the NSF under grants CCF-2200052, IIS-1914792, and DMS-1664644, by the ONR under grants N00014-19-1-2571 and N00014-21-1-2844, by the NIH under grants R01 GM135930 and UL54 TR004130, by the Australian government via grant AUSMURIB000001, and by the Boston University Kilachand Fund for Integrated Life Science and Engineering.

The HRL with options framework proposed by Sutton et al. (1999) involves a two-tiered hierarchy of policies, with a high-level policy governing "options" or decision as to which of a finite set of low-level policies are used to select actions. A key challenge of HIL in this options framework is that in practice, only (low-level) states and actions are directly observed through expert demonstrations, not the (high-level) options. The options thus constitute hidden (or latent) variables, and so recent HIL works have drawn inspiration from Expectation-Maximization (EM) techniques for learning Hidden Markov models (HMMs) and other latent variable models (Daniel et al., 2016; Zhang and Paschalidis, 2021; Giammarino and Paschalidis, 2021). These EM techniques process state-action pairs from expert demonstrations with a Bayesian smoother to compute a surrogate function for the (log)likelihood, and subsequent maximization of this surrogate function over the policy space. Whilst local-convergence theoretical guarantees have recently been shown for such an EM approach in the context of HIL (Zhang and Paschalidis, 2021), the nature of EM techniques as local-search procedures means that they are prone to convergence to local (non-global) maxima, and slow convergence with associated high computational expense.

In HMMs and other specific classes of latent variable models, *methods of moments* have been developed to overcome convergence issues inherent with EM techniques

(Hsu et al., 2012; Hsu and Kakade, 2013; Mattila et al., 2020, 2015, 2017; Anandkumar et al., 2014; Parikh et al., 2012). These moment methods are free of local convergence problems (Mattila et al., 2020; Anandkumar et al., 2014), and often offer much faster practical convergence with less computational expense (Mattila et al., 2015, 2017). Moment methods have therefore been used both by themselves and as initialization algorithms for EM techniques (cf. (Zhang et al., 2016)). Nevertheless, moment methods have not previously been investigated for HIL in the options framework.

The key contribution of this paper is the development of a new method of moments for HIL in the HRL options framework of (Sutton et al., 1999). Inspired by the method of moments for HMMs developed in (Hsu et al., 2012), our method of moments for HIL offers global convergence under mild regularity and non-degeneracy conditions, and has the practical advantage of only requiring a single pass through the expert demonstrations. It therefore serves as both a useful alternative and complementary technique to the previously developed but locally-convergent EM algorithms of (Daniel et al., 2016; Giammarino and Paschalidis, 2021; Zhang and Paschalidis, 2021).

Notation: Uppercase letters denote random variables, lowercase letters denote realizations. Uppercase bold letters denote matrices, lowercase bold letters denote vectors. Superscript on a quantity acts like a label in case there are many quantities with the same symbol. Subscript on a quantity denotes it being a subclass of the original quantity. The Kronecker product $\otimes$ is defined as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & \dots & a_{1n}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\boldsymbol{B} & \dots & a_{mn}\boldsymbol{B} \end{bmatrix},$$

where $\boldsymbol{A}$ is a $m \times n$ matrix, $\boldsymbol{B}$ is a $p \times q$ matrix, and $\boldsymbol{A} \otimes \boldsymbol{B}$ is a $mp \times nq$ matrix. The Hadamard (element-wise) product $\circ$ is defined as

$$\boldsymbol{A} \circ \boldsymbol{B} = \begin{bmatrix} a_{11}b_{11} & \dots & a_{1n}b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1}b_{m1} & \dots & a_{mn}b_{mn} \end{bmatrix},$$

where $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{A} \circ \boldsymbol{B}$ are $m \times n$ matrices. Furthermore, $\mathbf{I}_m$ denotes an $m \times m$ identity matrix, $\mathbf{1}_{m\times n}$ denotes an $m \times n$ matrix with all of its entries equal to one, $\mathbf{0}_{m\times n}$ denotes an $m \times n$ matrix with all of its entries equal to zero, and $\boldsymbol{e}_j$ denotes the j^{th} unit vector. The Moore–Penrose inverse of a matrix $\boldsymbol{A}$ will be denoted $\boldsymbol{A}^+$ and its transpose by $\boldsymbol{A}^{\mathrm{T}}$.

2. PROBLEM FORMULATION

In this section, we introduce the HRL with options framework (Sutton et al., 1999; Barto and Mahadevan, 2003) and formulate the associated HIL problem.

2.1 HRL with Options Framework

The HRL with options framework corresponds to the Bayesian network shown in Fig. 1 where O_t, S_t, and A_t denote the option, the state, and the action at time $t \geq 1$, respectively. The triple (O_t, S_t, A_t) forms a discrete-time

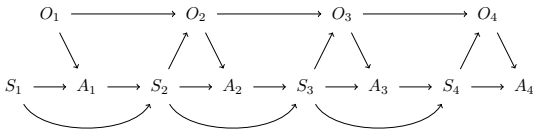

Fig. 1. Bayesian network of the HRL option framework.

Markov chain with O_t, S_t, and A_t defined over the finite spaces $\mathcal{O}$, $\mathcal{S}$, and $\mathcal{A}$, respectively. We denote the cardinality of these spaces as $|\mathcal{O}| = \omega$, $|\mathcal{S}| = \zeta$, and $|\mathcal{A}| = \alpha$.

The initial option and state pair (o_1, s_1) is sampled from an initial distribution $\pi_1(\cdot, \cdot)$. For $t \geq 1$, to advance one time step starting from the current pair (o_t, s_t), the action a_t is sampled from a low-level policy $\pi_{lo}(\cdot|s_t, o_t)$. Then, the resulting state s_{t+1} is sampled from an environment transition probability distribution $\Phi(\cdot|s_t, a_t)$. Finally, the next option o_{t+1} is sampled based on the new state and the previous option from the high-level policy $\pi_{hi}(\cdot|o_t, s_{t+1})$. The HRL with options framework is thus characterized by the policies π_{hi} and π_{lo}, and the transition distribution Φ.

Remark 2.1. The framework we consider differs slightly from that in Sutton et al. (1999) in that:

(1) The termination random variable, along with its decision policy is omitted, with the option transition based solely on the high-level policy π_{hi}. This is due to the fact that the termination factor is only involved in the transition between options, without directly affecting any observables in any way. For the sake of simplicity, the termination policy is folded into the high-level policy as one single object.
(2) The process starts with the pair (o_1, s_1) instead of (o_0, s_1). This difference is inconsequential as the resulting extra transition would be canceled out during the operations below.

2.2 The HIL Problem

Suppose that an expert uses the HRL with options framework to generate a sequence of states and actions $\{(s_t, a_t)\}_{t=1}^T$. In the HIL problem, we seek to use this sequence to learn the expert's underlying low and high-level policies π_{lo} and π_{hi}. The associated options $\{o_t\}_{t=1}^T$ are not observed and constitute hidden (or latent) variables. The HIL problem is thus an instance of learning in the presence of latent variables which has motivated its solution via EM approaches in (Daniel et al., 2016; Zhang and Paschalidis, 2021; Giammarino and Paschalidis, 2021). Due to local convergence issues inherent in EM approaches, we shall take a different approach and develop a method of moments for HIL inspired by the method of moments for HMMs developed in (Hsu et al., 2012).

To develop our method, we define the following matrices.

Definition 2.1. For $s \in \mathcal{S}$, define $\boldsymbol{\Pi}_s^{lo} \in \mathbb{R}^{\omega\times\alpha}$ with

$$\boldsymbol{\Pi}_s^{lo}[o, a] = \pi_{lo}(A_t = a|O_t = o, S_t = s)$$

as the matrix representation of π_{lo} under the state s.

Definition 2.2. For $s \in \mathcal{S}$, define $\boldsymbol{\Pi}_s^{hi} \in \mathbb{R}^{\omega\times\omega}$ with

$$\boldsymbol{\Pi}_s^{hi}[o, o'] = \pi_{hi}(O_{t+1} = o'|O_t = o, S_{t+1} = s)$$

as the matrix representation of π_{hi} under the state s.

Definition 2.3. For $a \in \mathcal{A}$, define $\mathbf{\Phi}_a^A \in \mathbb{R}^{\zeta \times \zeta}$ with

$$\mathbf{\Phi}_a^A[s, s'] = \Phi(S_{t+1} = s' | S_t = s, A_t = a)$$

as matrix representations of the transition dynamics.

Definition 2.4. For $s' \in \mathcal{S}$, define $\mathbf{\Xi}_{s'} \in \mathbb{R}^{\zeta \times \omega}$ with

$$\mathbf{\Xi}_{s'}[s, o] = P(S_t = s, O_t = o, S_{t+1} = s').$$

We also require the following mild regulatory assumptions.

Assumption 1. (Option-Action Identifiability). Under the same state, no two options contain the same policy for choosing an action, i.e., $\mathbf{\Pi}_s^{lo}$ has full row rank $\forall s \in \mathcal{S}$.

Assumption 2. (Option-Option Identifiability). Under the same state, no two options give the same policy for choosing the next option, i.e., $\mathbf{\Pi}_s^{hi}$ has full rank $\forall s \in \mathcal{S}$.

Assumption 3. $\mathbf{\Xi}_s$ has full column rank $\forall s \in \mathcal{S}$.

Assumption 4. All actions have a non-zero chance of transitioning a state to all of its neighboring states and one state is another state's neighbor if there exists an action under which the probability of transitioning from the latter to the former is non-zero, i.e., for any $s, s' \in \mathcal{S}$, if there exists $a \in \mathcal{A}$ such that $\mathbf{\Phi}_a^A[s, s'] > 0$, then $\mathbf{\Phi}_{a'}^A[s, s'] > 0 \ \forall a' \in \mathcal{A}$.

Assumption 5. (Stationary). The process (O_t, S_t) starts with the stationary distribution, that is $\boldsymbol{\pi}_s^1[o] = \boldsymbol{\pi}_s^\infty[o]$ where $\boldsymbol{\pi}_s^t \in \mathbb{R}^\omega$ with $\boldsymbol{\pi}_s^t[o] = P(O_t = o, s_t = s)$ for $s \in \mathcal{S}$.

Remark 2.2. Assumptions 1, 2, and 3 follow the same line of reasoning as Condition 1 of Hsu et al. (2012); they remove malicious instances that can cause learning to confuse options that have the same transition/action probability. Assumption 4 is needed as the method of moments relies on the cancellation of certain terms across all actions, it can be interpreted as an action emission noise in the expert or a transition noise in the environment.

3. SPECTRAL METHOD OF MOMENTS

In this section, we develop our method of moments for HIL. We specifically identify observable moments of the states and actions, and show that they enable recovery of the low-policy π_{lo} via matrix diagonalization and the high-level policy π_{hi} via simple matrix algebra.

3.1 Moments in HIL

We note that under Assumption 5, the moments of the states, options, and actions are time-invariant. Thus, without loss of generality, we consider the moments $\boldsymbol{M}_a \in \mathbb{R}^{\zeta\zeta \times \zeta\alpha}$ for $a \in \mathcal{A}$ with

$$\begin{aligned}\boldsymbol{M}_a[s_2\zeta + s_1, s_3\alpha + a_3] \\ = \ P(S_2 = s_2, S_1 = s_1, A_2 = a, S_3 = s_3, A_3 = a_3),\end{aligned}$$

and $\boldsymbol{K}_s \in \mathbb{R}^{\zeta \times \alpha}$ for $s \in \mathcal{S}$ with

$$\boldsymbol{K}_s[s_1, a_2] = P(S_1 = s_1, S_2 = s, A_2 = a_2).$$

Our goal now is to construct an expression of these observable moments that allows the recovery of the low-level policy via matrix diagonalization.

3.2 Diagonalizable Forms

We first examine the properties of $\boldsymbol{K}_s$ for $s \in \mathcal{S}$. Specifically, let $\boldsymbol{V}_s \in \mathbb{R}^{\alpha \times \omega}$ for $s \in \mathcal{S}$ be a matrix of right singular vectors corresponding to the ω largest singular values of $\boldsymbol{K}_s$. We then have the following lemma.

Lemma 3.1. Define the block-diagonal matrices

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{V}_1 & & \\ & \ddots & \\ & & \boldsymbol{V}_\zeta \end{bmatrix} \text{ and } \mathbf{\Pi}^{lo} = \begin{bmatrix} \mathbf{\Pi}_1^{lo} & & \\ & \ddots & \\ & & \mathbf{\Pi}_\zeta^{lo} \end{bmatrix}. \quad (1)$$

Then the product matrix

$$\mathbf{\Pi}^{lo}\boldsymbol{V} = \begin{bmatrix} \mathbf{\Pi}_1^{lo}\boldsymbol{V}_1 & & \\ & \ddots & \\ & & \mathbf{\Pi}_\zeta^{lo}\boldsymbol{V}_\zeta \end{bmatrix}$$

is invertible.

Proof. We have

$$\begin{aligned}\boldsymbol{K}_s[s_1, a_2] &= \sum_{o_1} \sum_{o_2} P(O_1 = o_1, S_1 = s_1, S_2 = s) \\ &\quad \times \pi_{hi}(O_2 = o_2 | O_1 = o_1, S_2 = s) \\ &\quad \times \pi_{lo}(A_2 = a_2 | O_2 = o_2, S_2 = s) \\ &= \left\{\mathbf{\Xi}_s \mathbf{\Pi}_s^{hi} \mathbf{\Pi}_s^{lo}\right\}[s_1, a_2]. \quad (2)\end{aligned}$$

This implies $\text{rowspan}(\boldsymbol{K}_s) \subseteq \text{rowspan}(\mathbf{\Pi}_s^{lo})$. In addition, because $\mathbf{\Xi}_s$ is full column rank and $\mathbf{\Pi}_s^{hi}$ is full rank (Assumption 2 and 3),

$$\mathbf{\Pi}_s^{lo} = (\mathbf{\Xi}_s \mathbf{\Pi}_s^{hi})^+ \boldsymbol{K}_s,$$

which implies $\text{rowspan}(\mathbf{\Pi}_s^{lo}) \subseteq \text{rowspan}(\boldsymbol{K}_s)$. Thus,

$$\text{rowspan}(\boldsymbol{V}_s^{\mathrm{T}}) = \text{rowspan}(\boldsymbol{K}_s) = \text{rowspan}(\mathbf{\Pi}_s^{lo}).$$

Therefore, $\mathbf{\Pi}_s^{lo}\boldsymbol{V}_s$ is invertible. Since s is chosen arbitrarily, this applies for all s. Because $\mathbf{\Pi}_s^{lo}\boldsymbol{V}_s$ is invertible for all s, it follows that $\mathbf{\Pi}^{lo}\boldsymbol{V}$ is also invertible. □

We next examine the properties of the moments $\boldsymbol{M}_a$. Before doing so, note that the moments $\boldsymbol{M}_a$ involve the transition dynamics $\mathbf{\Phi}_a^A$ as well as the underlying low- and high-level policies we are interested in. To remove the influence of the transition dynamics on $\boldsymbol{M}_a$, let us define the kernel matrix $\mathbf{\Psi} \in \mathbb{R}^{\zeta \times \zeta}$ with

$$\mathbf{\Psi}[s_2, s_3] = \begin{cases} \psi_{s_2 s_3}, & \text{if } \mathbf{\Phi}_a^A[s_2, s_3] > 0 \ \forall a \in \mathcal{A}, \\ 0, & \text{otherwise}, \end{cases} \quad (3)$$

and normalizer matrices $\mathbf{\Psi} \in \mathbb{R}^{\zeta \times \zeta}$ for $a \in \mathcal{A}$ with

$$\boldsymbol{N}_a[s_2, s_3] = \begin{cases} \dfrac{1}{\mathbf{\Phi}_a^A[s_2, s_3]}, & \text{if } \mathbf{\Phi}_a^A[s_2, s_3] > 0, \\ 0, & \text{otherwise}, \end{cases} \quad (4)$$

where $\psi_{s_2 s_3}$ are constants of choice such that $\mathbf{\Psi}$ is full rank (such constants will always exist under Assumption 4). We then may define the surrogate moments

$$\hat{\boldsymbol{M}}_a = (\mathbf{\Psi} \otimes \mathbf{1}_{\zeta \times \alpha}) \circ (\boldsymbol{N}_a \otimes \mathbf{1}_{\zeta \times \alpha}) \circ \boldsymbol{M}_a, \quad (5)$$

for $a \in \mathcal{A}$ and

$$\hat{\boldsymbol{M}} = \sum_{a \in \mathcal{A}} \hat{\boldsymbol{M}}_a \quad (6)$$

that do not depend on the transition dynamics where $\hat{\boldsymbol{M}}_a$ and $\hat{\boldsymbol{M}}$ have the same dimensions as $\boldsymbol{M}_a$.

These surrogate moments combined with Lemma 3.1 lead to the following theorem that establishes that the observable moments allow the recovery of the low-level policy via matrix diagonalization.

Theorem 3.2. The product $\boldsymbol{V}^{\mathrm{T}} \hat{\boldsymbol{M}}^+ \hat{\boldsymbol{M}}_a \boldsymbol{V}$ admits the factorization:

$$\boldsymbol{V}^{\mathrm{T}} \hat{\boldsymbol{M}}^+ \hat{\boldsymbol{M}}_a \boldsymbol{V} = \boldsymbol{B}^{-1} \mathbf{\Lambda}_a \boldsymbol{B}, \quad (7)$$

where

$$\mathbf{\Lambda}_a = \begin{bmatrix} \mathrm{diag}(\mathbf{\Pi}_1^{lo}\boldsymbol{e}_a) & & \\ & \ddots & \\ & & \mathrm{diag}(\mathbf{\Pi}_\zeta^{lo}\boldsymbol{e}_a) \end{bmatrix}. \tag{8}$$

Proof. Let

$$\mathbf{\Xi} = \begin{bmatrix} \mathbf{\Xi}_1 & & \\ & \ddots & \\ & & \mathbf{\Xi}_\zeta \end{bmatrix} \text{ and } \mathbf{\Pi}^{hi} = \begin{bmatrix} \mathbf{\Pi}_1^{hi} & & \\ & \ddots & \\ & & \mathbf{\Pi}_\zeta^{hi} \end{bmatrix}.$$

Then,

$$\begin{aligned} &\boldsymbol{M}_a\left[s_2\zeta + s_1, s_3\alpha + a_3\right] \\ &= \sum_{s_2', o_1} \sum_{s_2'', o_2} \sum_{s_2''', o_2'} \sum_{s_3, o_2'} \sum_{s_3', o_2''} \\ &\qquad P(O_1 = o_1, S_1 = s_1, S_2 = s_2 = s_2') \\ &\qquad \times \pi_{hi}(O_2 = o_2, S_2 = s_2'' | O_1 = o_1, S_2 = s_2') \\ &\qquad \times \pi_{lo}(A_2 = a, S_2 = s_2''', O_2 = o_2' | O_2 = o_2, S_2 = s_2'') \\ &\qquad \times P(S_3 = s_3, O_2 = o_2'' | A_2 = a, S_2 = s_2''', O_2 = o_2') \\ &\qquad \times \pi_{hi}(O_3 = o_3, S_3 = s_3' | O_2 = o_2'', S_3 = s_3) \\ &\qquad \times \pi_{lo}(A_3 = a_3, S_3 = s_3'' | O_3 = o_3, S_3 = s_3') \\ &= \left\{\mathbf{\Xi}\mathbf{\Pi}^{hi}\mathbf{\Lambda}_a(\mathbf{\Phi}_a^A \otimes \mathbf{I}_\omega)\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo}\right\}\left[s_2\zeta + s_1, s_3\alpha + a_3\right]. \end{aligned}$$

Consider the $\zeta \times \alpha$ submatrix

$$\begin{aligned} &\boldsymbol{U}^a_{s_2 s_3}[s_1, a_3] = \boldsymbol{M}_a[s_2\zeta + s_1, s_3\alpha + a_3] \\ &\quad \Leftrightarrow \boldsymbol{U}^a_{s_2 s_3} = \mathbf{\Xi}_{s_2}\mathbf{\Pi}^{hi}_{s_2}\mathrm{diag}(\mathbf{\Pi}^{lo}_{s_2}\boldsymbol{e}_a)\mathbf{\Phi}^A_a[s_2, s_3]\mathbf{\Pi}^{hi}_{s_3}\mathbf{\Pi}^{lo}_{s_3}. \end{aligned}$$

Notice that $\mathbf{\Phi}^A_a[s_2, s_3]$ is a real number that can be estimated using observable data. We define

$$\begin{aligned} \hat{\boldsymbol{U}}^a_{s_2 s_3} &= \frac{1}{\mathbf{\Phi}^A_a[s_2, s_3]}\boldsymbol{U}^a_{s_2 s_3} \\ &= \mathbf{\Xi}_{s_2}\mathbf{\Pi}^{hi}_{s_2}\mathrm{diag}(\mathbf{\Pi}^{lo}_{s_2}\boldsymbol{e}_a)\mathbf{\Pi}^{hi}_{s_3}\mathbf{\Pi}^{lo}_{s_3} \end{aligned} \tag{9}$$

for all $s_2, s_3 \in \mathcal{S}$ such that $\mathbf{\Phi}^A_a[s_2, s_3] > 0\ \forall a \in \mathcal{A}$, and $\hat{\boldsymbol{U}}^a_{s_2 s_3} = \mathbf{0}_{\zeta\times\alpha}$ otherwise.

By the Definitions (5, 9)

$$\begin{aligned} \hat{\boldsymbol{M}}_a &= (\mathbf{\Psi} \otimes \mathbf{1}_{\zeta\times\alpha}) \circ \begin{bmatrix} \hat{\boldsymbol{U}}^a_{11} & \dots & \hat{\boldsymbol{U}}^a_{1\zeta} \\ \vdots & \ddots & \vdots \\ \hat{\boldsymbol{U}}^a_{\zeta 1} & \dots & \hat{\boldsymbol{U}}^a_{\zeta\zeta} \end{bmatrix} \\ &= \mathbf{\Xi}\mathbf{\Pi}^{hi}\mathbf{\Lambda}_a(\mathbf{\Psi} \otimes \mathbf{I}_\omega)\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo}. \end{aligned}$$

By the Definition (6)

$$\begin{aligned} \hat{\boldsymbol{M}} &= \sum_{a\in\mathcal{A}} \hat{\boldsymbol{M}}_a \\ &= \mathbf{\Xi}\mathbf{\Pi}^{hi}\left(\sum_{a\in\mathcal{A}} \mathbf{\Lambda}_a\right)(\mathbf{\Psi} \otimes \mathbf{I}_\omega)\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo} \\ &= \mathbf{\Xi}\mathbf{\Pi}^{hi}(\mathbf{\Psi} \otimes \mathbf{I}_\omega)\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo}. \end{aligned}$$

Finally, we can write Equation (7) as

$$\begin{aligned} &\boldsymbol{V}^{\mathrm{T}}\hat{\boldsymbol{M}}^{+}\hat{\boldsymbol{M}}_a\boldsymbol{V} \\ &= \left(\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo}\boldsymbol{V}\right)^{-1}\left(\mathbf{\Psi}^{-1} \otimes \mathbf{I}_\omega\right)\mathbf{\Lambda}_a(\mathbf{\Psi} \otimes \mathbf{I}_\omega)\mathbf{\Pi}^{hi}\mathbf{\Pi}^{lo}\boldsymbol{V}, \end{aligned} \tag{10}$$

and the proof is complete. □

In order to compute the eigenbasis that jointly diagonalize (7) for all $a \in \mathcal{A}$, we find a vector $\boldsymbol{\eta} \in \mathbb{R}^\alpha$ such that the eigenvalues of

$$\sum_{a\in\mathcal{A}} \boldsymbol{\eta}_a \boldsymbol{V}^{\mathrm{T}}\hat{\boldsymbol{M}}^{+}\hat{\boldsymbol{M}}_a\boldsymbol{V} = \boldsymbol{B}\left(\sum_{a\in\mathcal{A}} \boldsymbol{\eta}_a\mathbf{\Lambda}_a\right)\boldsymbol{B}^{-1} \tag{11}$$

are well spread. In other words, we find $\boldsymbol{\eta}$ such that the values $\boldsymbol{e}_o^{\mathrm{T}}\mathbf{\Pi}^{lo}_s\boldsymbol{\eta}$ are distinct and non-zero for all $(o, s) \in \mathcal{O}\times\mathcal{S}$. As suggested in Hsu and Kakade (2013), this can be satisfied in most cases if $\boldsymbol{\eta}$ is sampled uniformly from the surface of a unit sphere in $\mathbb{R}^\alpha$.

The eigen-decomposition will yield an eigenbasis up to a permutation $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{\omega\times\zeta}$ of the pair $(o, s) \in \mathcal{O}\times\mathcal{S}$. To put it differently, the diagonal matrix obtained from diagonalizing $\boldsymbol{V}^{\mathrm{T}}\hat{\boldsymbol{M}}^{+}\hat{\boldsymbol{M}}_a\boldsymbol{V}$ using this basis will be of the form $\boldsymbol{\mathcal{P}}\mathbf{\Lambda}_a\boldsymbol{\mathcal{P}}^{\mathrm{T}}$. With some further processing, an order up to a permutation $\hat{\boldsymbol{\mathcal{P}}} \in \mathbb{R}^\omega$ of $o \in \mathcal{O}$ can be recovered, meaning the diagonal matrix obtained will be of the form $(\mathbf{I}_\zeta \otimes \hat{\boldsymbol{\mathcal{P}}})\mathbf{\Lambda}_a(\mathbf{I}_\zeta \otimes \hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}})$. This ordering corresponds to the relabeling of the options. Because this recovery process, while necessary, does not represent the main contribution of this work, it will be elaborated in the Appendix.

After obtaining the low-level policy matrices $\hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{lo}_s$, the high-level policy matrices can be computed by the following theorem, up to the permutation $\hat{\boldsymbol{\mathcal{P}}}$ of the options.

Theorem 3.3.

$$\hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{hi}_{s'}\hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}} = \sum_s \boldsymbol{w}_{s'}[s]\left(\hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{lo}_s\boldsymbol{K}_s^{+}\hat{\boldsymbol{K}}_{ss'}\mathbf{\Pi}^{lo^+}_{s'}\hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}}\right), \tag{12}$$

where:

- $\hat{\boldsymbol{K}}_{ss'}$ is a $\zeta \times \alpha$ submatrix of $\hat{\boldsymbol{M}}$ defined by
$$\hat{\boldsymbol{K}}_{ss'}[s'', a] = \hat{\boldsymbol{M}}[s\zeta + s'', s'\alpha + a]. \tag{13}$$
- $\boldsymbol{w}_{s'}$ are length ζ weight vectors of choice subject to
$$\boldsymbol{w}_i^{\mathrm{T}}\mathbf{\Psi}\boldsymbol{e}_i^{\mathrm{T}} = 1,\ \forall i \in \mathcal{S}. \tag{14}$$

Proof. According to Definition (13) and Equation (2), $\boldsymbol{K}_s$ and $\hat{\boldsymbol{K}}_{ss'}$ can be written as following:

$$\begin{aligned} \boldsymbol{K}_s &= \mathbf{\Xi}_s\mathbf{\Pi}^{hi}_s\mathbf{\Pi}^{lo}_s, \\ \hat{\boldsymbol{K}}_{ss'} &= \mathbf{\Psi}[s, s']\left(\mathbf{\Xi}_s\mathbf{\Pi}^{hi}_s\mathbf{\Pi}^{hi}_{s'}\mathbf{\Pi}^{lo}_{s'}\right). \end{aligned}$$

Therefore,

$$\begin{aligned} &\mathbf{\Pi}^{lo}_s\boldsymbol{K}_s{}^{+}\hat{\boldsymbol{K}}_{ss'}\mathbf{\Pi}^{lo^+}_{s'} \\ &= \mathbf{\Psi}[s, s']\left(\mathbf{\Pi}^{lo}_s\mathbf{\Pi}^{lo^+}_s\mathbf{\Pi}^{hi^{-1}}_s\mathbf{\Xi}_s{}^{+}\mathbf{\Xi}_s\mathbf{\Pi}^{hi}_s\mathbf{\Pi}^{hi}_{s'}\mathbf{\Pi}^{lo}_{s'}\mathbf{\Pi}^{lo^+}_{s'}\right) \\ &= \mathbf{\Psi}[s, s']\mathbf{\Pi}^{hi}_{s'}. \end{aligned}$$

With that, we contract the left hand side of Equation (12)

$$\begin{aligned} &\sum_s \boldsymbol{w}_{s'}[s]\left(\hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{lo}_s\boldsymbol{K}_s^{+}\hat{\boldsymbol{K}}_{ss'}\mathbf{\Pi}^{lo^+}_{s'}\hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}}\right) \\ &= \sum_s \boldsymbol{w}_{s'}[s]\mathbf{\Psi}[s, s']\left(\hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{hi}_{s'}\hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}}\right) \\ &= \hat{\boldsymbol{\mathcal{P}}}\mathbf{\Pi}^{hi}_{s'}\hat{\boldsymbol{\mathcal{P}}}^{\mathrm{T}}. \end{aligned}$$

The last equality holds because we chose $\boldsymbol{w}_{s'}$ such that $\sum_s \boldsymbol{w}_{s'}[s]\mathbf{\Psi}[s, s'] = 1$. Analysis of the choice of $\boldsymbol{w}_{s'}$ will be reserved for future work. □

3.3 Proposed Method of Moments for HIL

Given the observed sequence $\{(s_t, a_t)\}_{t=1}^T$, our method of moments to learn the policies π_{lo} and π_{hi} is:

Step 1: Estimate $\boldsymbol{M}_a$, $\boldsymbol{K}_s$, and $\boldsymbol{\Phi}_a^A$ from data via:

$$\boldsymbol{M}_a\left[s_2\zeta + s_1, s_3\alpha + a_3\right] = \frac{\sum_{t=1}^{T-2} \mathbb{I}_{\{s_t=s_1, s_{t+1}=s_2, a_{t+1}=a, s_{t+2}=s_3, a_{t+2}=a_3\}}}{T-2},$$

$$\boldsymbol{K}_s\left[s_1, a_2\right] = \frac{\sum_{t=1}^{T-1} \mathbb{I}_{\{s_t=s_1, s_{t+1}=s, a_{t+1}=a_2\}}}{T-1},$$

$$\boldsymbol{\Phi}_a^A[s, s'] = \frac{\sum_{t=1}^{T-1} \mathbb{I}_{\{s_t=s, a_t=a, s_{t+1}=s'\}}}{\sum_{t=1}^{T-1} \mathbb{I}_{\{s_t=s, a_t=a\}}}.$$

Step 2: Compute the surrogate moments $\hat{\boldsymbol{M}}_a$ and $\hat{\boldsymbol{M}}$ according to Equations (5) and (6).

Step 3: Perform SVD on $\boldsymbol{K}_s$, and construct the matrix $\boldsymbol{V}$ according to Equation (1).

Step 4: Compute the joint eigenbasis $\boldsymbol{B}$ using Equation (11). Then, recover the order of its column using the algorithm discussed in the Appendix.

Step 5: Recover $\boldsymbol{\Pi}^{lo}$ using the diagonals that result from diagonalizations according to Equation (7).

Step 6: Compute $\boldsymbol{\Pi}^{hi}$ with Equation (12).

3.4 Performance discussion

The algorithm consists of two parts, data collection with complexity $\mathcal{O}(T)$, and data processing with complexity $\mathcal{O}(\zeta^4\alpha\omega)$, dominated by the cost of computing $\boldsymbol{V}^{\mathrm{T}}\hat{\boldsymbol{M}}^{+}\hat{\boldsymbol{M}}_a\boldsymbol{V}$ and its eigenbasis. This gives us the total time complexity of $\mathcal{O}(T + \zeta^4\alpha\omega)$.

Comparison to the EM methods presented in Zhang and Paschalidis (2021) and Giammarino and Paschalidis (2021), which has time complexity of $\mathcal{O}(T\omega^2)$ and $\mathcal{O}(\zeta\alpha\omega^3)$ per iteration respectively, can be difficult. This is due to the fact that they have different bottlenecks, along with the fact that the method of moments is parameterless while EM methods need initialization. However, a general rule is that the larger the number of samples is relative to the number of states and actions, the better the method of moments performs compared to EM.

Another thing to note is that the techniques mentioned can be synergistic, with the output of the method of moments being good initialization for EM methods to refine.

4. EXPERIMENT

In this section, we examine the proposed algorithm in numerical experiments. We will use a similar setup to Zhang and Paschalidis (2021) to test our model. Let there be a finite state machine with four states and the following parameters:

$$\boldsymbol{\Pi}_1^{hi} = \begin{bmatrix} 0.67 & 0.33 \\ 0.16 & 0.84 \end{bmatrix}, \quad \boldsymbol{\Pi}_2^{hi} = \begin{bmatrix} 0.88 & 0.12 \\ 0.16 & 0.84 \end{bmatrix},$$
$$\boldsymbol{\Pi}_3^{hi} = \begin{bmatrix} 0.84 & 0.16 \\ 0.12 & 0.88 \end{bmatrix}, \quad \boldsymbol{\Pi}_4^{hi} = \begin{bmatrix} 0.84 & 0.16 \\ 0.33 & 0.67 \end{bmatrix}.$$

$$\boldsymbol{\Pi}_1^{lo} = \begin{bmatrix} 0.6 & 0.4 \\ 0.1 & 0.9 \end{bmatrix}, \quad \boldsymbol{\Pi}_2^{lo} = \begin{bmatrix} 0.7 & 0.3 \\ 0.15 & 0.85 \end{bmatrix},$$
$$\boldsymbol{\Pi}_3^{lo} = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}, \quad \boldsymbol{\Pi}_4^{lo} = \begin{bmatrix} 0.9 & 0.1 \\ 0.35 & 0.65 \end{bmatrix}.$$

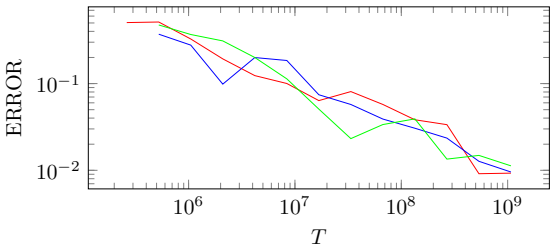

Fig. 2. Log-log plot of the error versus the number of sample points for several realizations of the problem.

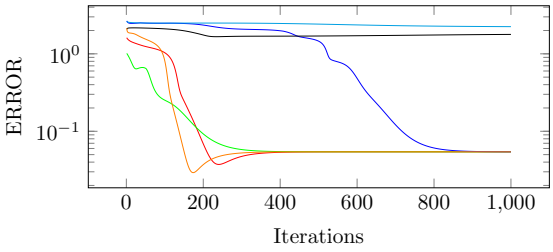

Fig. 3. Iterations versus error of EM runs with various initializations, some of which do not converge.

$$\boldsymbol{\Phi}_1^A = \begin{bmatrix} 0.7 & 0.1 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.1 & 0.1 \\ 0.3 & 0.3 & 0.3 & 0.1 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{bmatrix},$$

$$\boldsymbol{\Phi}_2^A = \begin{bmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.1 & 0.3 & 0.3 & 0.3 \\ 0.1 & 0.1 & 0.4 & 0.4 \\ 0.1 & 0.1 & 0.1 & 0.7 \end{bmatrix}.$$

The error will be measured by:

$$\text{ERROR} = \sqrt{\left\|\boldsymbol{\Pi}^{lo} - \overline{\boldsymbol{\Pi}}^{lo}\right\|_2^2 + \left\|\boldsymbol{\Pi}^{hi} - \overline{\boldsymbol{\Pi}}^{hi}\right\|_2^2},$$

where $\overline{\boldsymbol{\Pi}}^{lo}, \overline{\boldsymbol{\Pi}}^{hi}$ are the predicted values of $\boldsymbol{\Pi}^{lo}, \boldsymbol{\Pi}^{hi}$.

For intuition, we can think of the states as locations on a number line (i.e., states with larger index are further right), the actions are $\mathcal{A} = \{\text{move-left}, \text{move-right}\}$, and the options are $\mathcal{O} = \{\text{tend-to-move-left}, \text{tend-to-move-right}\}$. Looking at the numbers, we can see that the agent wants to alternately move from left to right and right to left.

In Fig. 2 we plot the error versus the number of samples for a few runs of our method in log scale. It can be seen that the error is polynomial relative to the number of samples.

For comparison purposes, Fig. 3 depicts a few EM runs with randomized initialization and a sample size of 3×10^5. It can be seen that initialization have a significant effect on EM's rate of convergence and whether or not it arrives at the correct optima. In contrast, the proposed method of moments does not require initialization.

5. CONCLUSIONS AND FUTURE WORK

We developed a novel method of moments for Hierarchical Imitation Learning (HIL) that offers global convergence under mild regulatory conditions. Our method of moments for HIL is based on similar methods for HMMs and other

latent variable models, and avoids the local convergence issues inherent in previous Expectation-Maximization (EM) approaches to HIL. Future work could include further relaxation of the conditions under which the method holds and examining its extension to situations in which the options form a semi-Markov (rather than a Markov) process.

Appendix A. ORDER RECOVERY PROCESS

From Equation (10), we know the eigenbasis $\boldsymbol{B} \in \mathbb{R}^{\zeta\omega\times\zeta\omega}$ is of the form

$$\boldsymbol{B} = \left(\boldsymbol{\Pi}^{hi}\boldsymbol{\Pi}^{lo}\boldsymbol{V}\right)^{-1}\left(\boldsymbol{\Psi}^{-1}\otimes\mathbf{I}_{\omega}\right). \tag{A.1}$$

The eigen-decomposition (7) will introduce an unknown scaling and permutation to the columns of the basis:

$$\hat{\boldsymbol{B}} = \boldsymbol{B}\mathrm{diag}(\boldsymbol{c})\boldsymbol{\mathcal{P}}, \tag{A.2}$$

where $\boldsymbol{c}$ is a vector that corresponds to the scaling of each column, and $\boldsymbol{\mathcal{P}}$ is a permutation operator on the columns.

For convenience, define the following shorthands:

$$\boldsymbol{X}_s = \left(\boldsymbol{\Pi}^{hi}_s\boldsymbol{\Pi}^{lo}_s\boldsymbol{V}_s\right)^{-1},$$

$$\boldsymbol{X} = \left(\boldsymbol{\Pi}^{hi}\boldsymbol{\Pi}^{lo}\boldsymbol{V}\right)^{-1} = \mathrm{diag}(\boldsymbol{X}_1,\ldots,\boldsymbol{X}_{\zeta}),$$

and $\boldsymbol{\Gamma} = \boldsymbol{\Psi}^{-1}$ with elements γ_{ij}. Rewrite (A.1) as $\boldsymbol{B} = \boldsymbol{X}\left(\boldsymbol{\Gamma}\otimes\mathbf{I}_{\omega}\right)$. Recalling Equation (A.2), we have that the structure of the sub-matrices $\hat{\boldsymbol{J}}_s \in \mathbb{R}^{\omega\times\zeta\omega}$ of the basis $\hat{\boldsymbol{B}}$ is $\hat{\boldsymbol{J}}_s = [\gamma_{s1}\boldsymbol{X}_s \ \ldots \ \gamma_{s\zeta}\boldsymbol{X}s]\,\mathrm{diag}(\boldsymbol{c})\boldsymbol{\mathcal{P}}$. It is easily seen that each of these sub-matrices contains ω sets of ζ linearly dependent vectors, and that the grouping of these linearly dependent columns are identical due to them sharing the same permutation $\boldsymbol{\mathcal{P}}$. We separate and represent these sets as $\zeta\omega\times\zeta$ matrices $\hat{\boldsymbol{Q}}_o$ given by

$$\hat{\boldsymbol{Q}}_o = \begin{bmatrix}\gamma_{11}\boldsymbol{X}_1\boldsymbol{e}_o & \ldots & \gamma_{1\zeta}\boldsymbol{X}_1\boldsymbol{e}_o\\ \vdots & \ddots & \vdots\\ \gamma_{\zeta 1}\boldsymbol{X}_{\zeta}\boldsymbol{e}_o & \ldots & \gamma_{\zeta\zeta}\boldsymbol{X}_{\zeta}\boldsymbol{e}_o\end{bmatrix}\mathrm{diag}(\boldsymbol{c}_o)\boldsymbol{\mathcal{P}}_o, \tag{A.3}$$

where $\boldsymbol{c}_o$ and $\boldsymbol{\mathcal{P}}_o$ are unknown scaling factor and permutation corresponding to the group o.

We define $\boldsymbol{d}_o = \left[\boldsymbol{X}_1^{\mathrm{T}}\boldsymbol{e}_o^{\mathrm{T}} \ \ldots \ \boldsymbol{X}_{\zeta}^{\mathrm{T}}\boldsymbol{e}_o^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{\zeta\omega}$. Then, (A.3) can be rewritten as $\hat{\boldsymbol{Q}}_o = \mathrm{diag}(\boldsymbol{d}_o)(\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega})\mathrm{diag}(\boldsymbol{c}_o)\boldsymbol{\mathcal{P}}_o$. In order to recover the original ordering of the columns of $\hat{\boldsymbol{Q}}_o$, we need to somehow match them with the columns of $\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega}$, which is a known quantity.

Before proceeding, let us introduce the element-wise inverse operator $\oslash$ such that $\boldsymbol{A}^{\oslash}$ corresponds to the matrix formed by inverting each element of the matrix $\boldsymbol{A}$, and so that $(\boldsymbol{A}\boldsymbol{\mathcal{P}})^{\oslash} = \boldsymbol{A}^{\oslash}\boldsymbol{\mathcal{P}}$, and $[\mathrm{diag}(\boldsymbol{u})\boldsymbol{A}\mathrm{diag}(\boldsymbol{v})]^{\oslash} = \mathrm{diag}(\boldsymbol{u})^{-1}\boldsymbol{A}^{\oslash}\mathrm{diag}(\boldsymbol{v})^{-1}$.. Let's assume that $\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega}$ has no zero entries. If there are zero entries, we can further partition $\hat{\boldsymbol{Q}}_o$ and $\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega}$ into column groups that has the same rows with zero entries, remove those rows, and use the following procedure on each of the groups before combining the result.

We have the following reduction:

$$\begin{aligned}\hat{\boldsymbol{Q}}_o^{\oslash}\hat{\boldsymbol{Q}}_o^{\mathrm{T}} &= \mathrm{diag}(\boldsymbol{d}_o)^{-1}(\boldsymbol{\Gamma}^{\oslash}\boldsymbol{\Gamma}^{\mathrm{T}}\otimes\mathbf{1}_{\omega}\mathbf{1}_{\omega}^{\mathrm{T}})\mathrm{diag}(\boldsymbol{d}_o)\\ &= (\boldsymbol{d}_o^{\oslash}\boldsymbol{d}_o^{\mathrm{T}})\circ(\boldsymbol{\Gamma}^{\oslash}\boldsymbol{\Gamma}^{\mathrm{T}}\otimes\mathbf{1}_{\omega}\mathbf{1}_{\omega}^{\mathrm{T}}).\end{aligned}$$

Therefore $\hat{\boldsymbol{Q}}_o^{\oslash}\hat{\boldsymbol{Q}}_o^{\mathrm{T}}\circ(\boldsymbol{\Gamma}^{\oslash}\boldsymbol{\Gamma}^{\mathrm{T}}\otimes\mathbf{1}_{\omega}\mathbf{1}_{\omega}^{\mathrm{T}})^{\oslash} = \boldsymbol{d}_o^{\oslash}\boldsymbol{d}_o^{\mathrm{T}}$. It is easily verifiable that $\mathrm{diag}(\boldsymbol{d}_o^{\oslash}\boldsymbol{d}_o^{\mathrm{T}}\boldsymbol{e}_i)\mathrm{diag}(\boldsymbol{d}_o) = \boldsymbol{d}_o[i]\mathbf{I}$.

We have $\mathrm{diag}(\boldsymbol{d}_o^{\oslash}\boldsymbol{d}_o^{\mathrm{T}}\mathbf{1}_{\zeta\omega})\hat{\boldsymbol{Q}}_o = (\mathbf{1}_{\zeta\omega}^{\mathrm{T}}\boldsymbol{d}_o)(\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega})\mathrm{diag}(\boldsymbol{c}_o)\boldsymbol{\mathcal{P}}_o$. Notice that the columns of this matrix are just multiples of the columns of $(\boldsymbol{\Gamma}\otimes\mathbf{1}_{\omega})$. As such a matching can be computed by checking linear dependence between the columns of the two matrices.

REFERENCES

Anandkumar, A., Ge, R., Hsu, D., Kakade, S.M., and Telgarsky, M. (2014). Tensor decompositions for learning latent variable models. *Journal of machine learning research*, 15, 2773–2832.

Barto, A.G. and Mahadevan, S. (2003). Recent advances in hierarchical reinforcement learning. *Discrete event dynamic systems*, 13(1), 41–77.

Daniel, C., Van Hoof, H., Peters, J., and Neumann, G. (2016). Probabilistic inference for determining options in reinforcement learning. *Machine Learning*, 104(2), 337–357.

Giammarino, V. and Paschalidis, I.C. (2021). Online Baum-Welch algorithm for hierarchical imitation learning. In *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE.

Hsu, D. and Kakade, S.M. (2013). Learning mixtures of spherical Gaussians: moment methods and spectral decompositions. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, 11–20.

Hsu, D., Kakade, S.M., and Zhang, T. (2012). A spectral algorithm for learning hidden Markov models. *Journal of Computer and System Sciences*, 78(5), 1460–1480.

Mattila, R., Rojas, C., Moulines, E., Krishnamurthy, V., and Wahlberg, B. (2020). Fast and consistent learning of hidden Markov models by incorporating non-consecutive correlations. In H.D. III and A. Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 6785–6796. PMLR.

Mattila, R., Rojas, C.R., Krishnamurthy, V., and Wahlberg, B. (2017). Identification of hidden Markov models using spectral learning with likelihood maximization. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 5859–5864.

Mattila, R., Rojas, C.R., and Wahlberg, B. (2015). Evaluation of spectral learning for the identification of hidden Markov models. *IFAC-PapersOnLine*, 48(28), 897–902. 17th IFAC Symposium on System Identification SYSID 2015.

Parikh, A.P., Song, L., Ishteva, M., Teodoru, G., and Xing, E.P. (2012). A spectral algorithm for latent junction trees. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, 675–684.

Sutton, R.S., Precup, D., and Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1), 181–211.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M.I. (2016). Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. *The Journal of Machine Learning Research*, 17(1), 3537–3580.

Zhang, Z. and Paschalidis, I. (2021). Provable hierarchical imitation learning via em. In *International Conference on Artificial Intelligence and Statistics*, 883–891. PMLR.