Learning Robot Manipulation Tasks with Task-Parameterized Semi-Tied Hidden Semi-Markov Model

Ajay Kumar Tanwani[1,2], Sylvain Calinon[1]

Abstract—In this paper, we investigate the semi-tied Gaussian mixture models for robust learning and adaptation of robot manipulation tasks. We make use of the spatial and temporal correlation in the data by tying the covariance matrices of the mixture model with common synergistic directions/basis vectors, instead of estimating full covariance matrices for each cluster in the mixture. This allows the reuse of the discovered synergies in different parts of the task having similar coordination patterns. We extend the approach to task-parameterized and hidden semi-Markov models for autonomous adaptation to changing environmental situations. The planned movement sequence from the model is smoothly followed with a finite horizon linear quadratic tracking controller. Experiments to encode whole body motion data in simulation, followed by valve opening and pick-and-place via obstacle avoidance tasks with the Baxter robot, show improvement over standard Gaussian mixture models with much less parameters and better generalization ability.

Index Terms—Learning and Adaptive Systems, Dexterous Manipulation, Probability and Statistical Methods, Telerobotics and Teleoperation

I. INTRODUCTION

PROBABILISTIC clustering models, such as Gaussian mixture model (GMM), are widely used to encode local trends in the data for classification or regression. For the set of T observations $\{\boldsymbol{\xi}_t\}_{t=1}^T$ with $\boldsymbol{\xi}_t \in \mathbb{R}^D$, the probability density function $\mathcal{P}$ of GMM with K mixture components is represented as

$$\mathcal{P}(\boldsymbol{\xi}_t|\theta) = \sum_{i=1}^K \pi_i \, \mathcal{N}(\boldsymbol{\xi}_t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \qquad (1)$$

where $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is the multivariate Gaussian distribution with prior π_i, mean $\boldsymbol{\mu}_i$, and covariance matrix $\boldsymbol{\Sigma}_i$. $\theta = \{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^K$ are the set of parameters to be estimated in the density function.

Note that the number of parameters in the covariance matrix $\boldsymbol{\Sigma}_i$ grows quadratically with the dimension of datapoints D. With the burgeoning of high-dimensional sensory data and multimodal interfaces for skill acquisition in robotics, an open challenge is to compactly represent the high-dimensional data
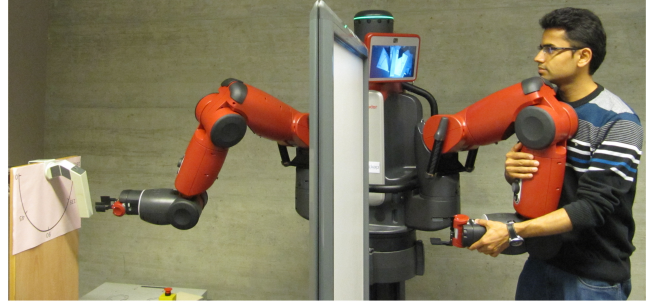
Fig. 1: Baxter robot learns to open/close the valve semi-autonomously via teleoperation.

with fewer parameters, albeit having effective generalization in similar unseen contexts.

Subspace clustering methods address this challenge by using a parsimonious model to reduce the number of parameters [1]. One way to reduce the number of parameters would be to constrain the covariance structure to a diagonal or spherical/isotropic matrix, thereby, restricting the number of parameters to grow linearly or up to a constant at the cost of treating each dimension separately. Such decoupling, however, cannot encode the important motor control principles of coordination, synergies and action-perception couplings [2].

A. Proposed Approach

In this paper, we seek out a latent feature space in the high-dimensional data to reduce the number of model parameters that can be robustly estimated. We exploit a technique called the semi-tied Gaussian mixture model, that associates or ties the covariance matrices of mixture model with a common latent space [3]. The latent space decorrelates the data and the mixture components map the data onto the corresponding subspaces to cope with perturbations. The task-parameterized version of the model adapts the model parameters to new environmental situations in a probabilistic manner. For reactive autonomous behaviour, we encapsulate the spatial and state duration information in the data with a hidden semi-Markov model (HSMM) [4]. Learning of the model parameters is performed with expectation-maximization (EM) [5], and the generated movement sequence from the model is tracked with a finite-horizon linear quadratic tracking (LQT) controller. The proposed framework combines subspace clustering, task

adaptability and optimal control for learning manipulation tasks in robotics.

B. Related Work

Most of the work on clustering models has focused on global dimensionality reduction methods as a pre-processing step. Notable examples include principal component analysis, factor analysis and linear discriminant analysis. Subspace clustering models, in contrast, learn multiple subspaces to encode the data according to their local trend, i.e., they perform segmentation and dimensionality reduction simultaneously. A broad range of these models encompasses sparse subspace clustering [6], DP-space clustering [7], parsimonious models [8], mixture of factor analyzers (MFA) [9] or mixture of probabilistic principal component analyzers (MPPCA) [10]. For example, MFA performs subspace clustering by assuming the structure of covariance matrix of the form

$$\boldsymbol{\Sigma}_i = \boldsymbol{\Lambda}_i \boldsymbol{\Lambda}_i^\top + \boldsymbol{\Psi}_i, \tag{2}$$

where $\boldsymbol{\Lambda}_i \in \mathbb{R}^{D \times d}$ is the *factor loadings matrix* with $d < D$ for parsimonious representation of the data, and $\boldsymbol{\Psi}_i$ is the diagonal noise matrix. Coordinated MFA has found its application in robotics in tracking 3D human movement from motion capture data [11], and more recently for learning trajectories in robot programming by demonstration framework [12].

Note that each covariance matrix of the mixture component has its own subspace spanned by the basis vectors of $\boldsymbol{\Sigma}_i$. As the number of components increase to encode more complex skills, an increasing large number of potentially redundant parameters are used to fit the data. Consequently, there is a need to share the basis vectors across the mixture components. The concept was first exploited in speech processing where the covariance matrices in output state sequence of a Hidden Markov Model (HMM) were tied to a common linear transform [3]. Parameter tying, for example, has been used to robustly estimate the density parameters with thousands of states in a HMM for building phone models [13].

To the best of our knowledge, the concept of tying covariance matrices in mixture models to encode manipulation skills in robotics has not been used. We extend the method here to a task-parameterized model [14] and encode the state duration and transition with a hidden semi-Markov model to enable the handling of previously unseen situations in an autonomous manner.

C. Application Scenario

We apply the proposed approach to teleoperation with the Baxter robot where the robot is required to execute manipulation tasks semi-autonomously in the presence of communication latencies (see Fig. 1 for the teleoperation mock-up). The operator controls/teleoperates the right arm with a simulated delay using the left arm by taking feedback from the camera mounted on the end-effector of the right arm. In case of communication disruptions, the robot recognises the current state of the task being performed by the teleoperator and reproduces the movement on the remote side autonomously until further communication is established. This form of teleoperation is used to acquire manipulation skills within the DexROV project (dexterous undersea inspection and maintenance in presence of communication latencies) [15]. Here, we perform experiments to learn the challenging tasks of opening/closing a valve and pick-place an object via obstacle avoidance with the Baxter robot. Our results show that the proposed model efficiently learns the key movement directions to construct the latent space that are exploited to encode the task with a fewer number of model parameters, while subsequently generalizing to unseen configurations of the task.

II. SEMI-TIED GAUSSIAN MIXTURE MODEL

When the covariance matrices of the mixture model share the same set of parameters for the latent feature space, we call the model a *semi-tied Gaussian mixture model*. The main idea behind semi-tied GMMs is to decompose the covariance matrix $\boldsymbol{\Sigma}_i$ into two terms: a common latent feature matrix $\boldsymbol{H} \in \mathbb{R}^{D \times D}$ and a component-specific diagonal matrix $\boldsymbol{\Sigma}_i^{(\mathrm{diag})} \in \mathbb{R}^{D \times D}$, i.e.,

$$\boldsymbol{\Sigma}_i = \boldsymbol{H} \boldsymbol{\Sigma}_i^{(\mathrm{diag})} \boldsymbol{H}^\top. \tag{3}$$

The latent feature matrix encodes the locally important synergistic directions represented by D non-orthogonal basis vectors that are shared across all the mixture components, while the diagonal matrix selects the appropriate subspace of each mixture component as convex combination of a subset of the basis vectors of $\boldsymbol{H}$. Depending upon the sparsity of the convex combination, there are multiple subspaces to choose. In other words, we search for a global linear transformation of the data such that the transformed data can be modelled by a mixture of diagonal covariance matrices only.[1]

In high-dimensional spaces, Gaussian mixture components with full covariance matrices tend to over-fit the training data when the data is noisy and/or the number of datapoints is insufficient. By tying the covariance matrices, the mixture components are forced to align along a set of common coordination patterns. This is also in line with biological motor control where the central nervous system (CNS) is believed to generate complex movements by temporal modulation of postural synergies [16]. The implementation of postural synergies corresponds here to the basis vectors of $\boldsymbol{H}$, while the diagonal matrix of each mixture component $\boldsymbol{\Sigma}_i^{(\mathrm{diag})}$ modulates the basis vectors in time for efficient encoding of complex tasks.

A. Conceptual Example

Consider multiple demonstrations of a 3-dimensional Z-shaped movement as shown in Fig. 2. Encoding with semi-tied GMM reveals the locally important basis vectors comprising the latent feature space $\boldsymbol{H}$. In contrast, PCA here would yield orthogonal basis vectors along the directions of largest variance globally. Note that the basis vectors are not required to be orthogonal in the semi-tied GMM. It can be seen in Fig. 2 that the basis vector in red is shared across the first and

---

[1]Note that the eigen decomposition of $\boldsymbol{\Sigma}_i = \boldsymbol{U}_i \boldsymbol{\Sigma}_i^{(\mathrm{diag})} \boldsymbol{U}_i^\top$ contains D basis vectors of $\boldsymbol{\Sigma}_i$ in $\boldsymbol{U}_i$. In comparison, semi-tied mixture model gives D globally representative basis vectors that are shared across all the mixture components.
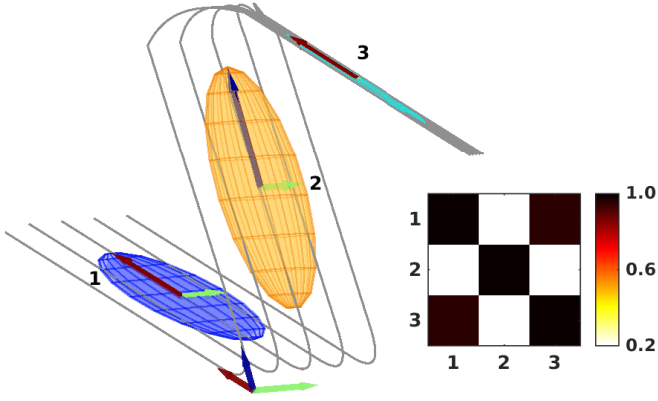
Fig. 2: *(left)* Semi-tied mixture model encoding of Z-shaped data with 3 components and basis vectors shown at the origin, *(right)* pairwise correlation among the mixture components of semi-tied GMM (see section II-D for details).

the third mixture component, while the basis vector in green is shared across the first and the second mixture component. The basis vector in blue is tied only to the second mixture component. This yields high correlation between the first and the third mixture component, and low correlation of the second Gaussian component with other mixture components (see right of Fig. 2).

B. Maximum Likelihood Parameter Estimation

We are interested in maximum likelihood estimates of the parameters of semi-tied GMM, $\theta = \{\{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i^{(\text{diag})}\}_{i=1}^K, \boldsymbol{H}\}$. Given the initial set of parameters $\hat{\theta}$, substituting the expression for $\boldsymbol{\Sigma}_i$ from Eq. (3) in the auxiliary function [5] yields,

$$\mathcal{Q}(\theta, \hat{\theta}) \approx \frac{1}{2} \sum_{t=1}^T \sum_{i=1}^K h_{t,i}^{\hat{\theta}} \left(\log \left(\pi_i^2 - |\boldsymbol{\Sigma}_i| \right) - \boldsymbol{\xi}_t^{i\top} \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\xi}_t^i \right),$$

$$\approx \frac{1}{2} \sum_{t=1}^T \sum_{i=1}^K h_{t,i}^{\hat{\theta}} \left(2 \log(\pi_i) - \log \left(\frac{|\boldsymbol{\Sigma}_i^{(\text{diag})}|}{|\boldsymbol{B}|^2} \right) - \boldsymbol{\xi}_t^{i\top} \boldsymbol{B}^\top \boldsymbol{\Sigma}_i^{(\text{diag})-1} \boldsymbol{B} \boldsymbol{\xi}_t^i \right),$$

$$(4)$$

where $\boldsymbol{B} = \boldsymbol{H}^{-1}, \boldsymbol{\xi}_t^i = \boldsymbol{\xi}_t - \boldsymbol{\mu}_i$, and $h_{t,i}^{\hat{\theta}} = p(i|\boldsymbol{\xi}_t, \hat{\theta})$ is the probability of data point $\boldsymbol{\xi}_t$ to belong to i-th Gaussian component at time t. Setting $\frac{\partial \mathcal{Q}(\theta, \hat{\theta})}{\partial \boldsymbol{B}}$ and $\frac{\partial \mathcal{Q}(\theta, \hat{\theta})}{\partial \boldsymbol{\Sigma}_i^{(\text{diag})}}$ equal to 0, and solving for $\boldsymbol{B}$ and $\boldsymbol{\Sigma}_i^{(\text{diag})}$ respectively results in an expectation-maximization procedure to compute the maximum likelihood estimate of parameters (see [17] for details). Following this, we get a row-by-row optimisation of $\boldsymbol{B}$, with $\boldsymbol{b}_d$ (d-th row of $\boldsymbol{B}$) related to all other rows by the cofactor of $\boldsymbol{B}$

$$\boldsymbol{b}_d = \boldsymbol{c}_d \boldsymbol{G}_d^{-1} \sqrt{\frac{\sum_{t=1}^T \sum_{i=1}^K h_{t,i}^{\hat{\theta}}}{\boldsymbol{c}_d \boldsymbol{G}_d^{-1} \boldsymbol{c}_d^\top}}, \quad (5)$$

where $\boldsymbol{c}_d$ is the d-th row of cofactors of $\boldsymbol{B}$ with $\boldsymbol{C} = \text{cof}(\boldsymbol{B})$ recomputed after each update of $\boldsymbol{b}_d$

$$\boldsymbol{C} = (\boldsymbol{B}^\top)^{-1} |\boldsymbol{B}|, \quad (6)$$

$$\boldsymbol{G}_d = \sum_{i=1}^K \frac{1}{\Sigma_{i,d}^{(\text{diag})}} \boldsymbol{S}_i \sum_{t=1}^T h_{t,i}^{\hat{\theta}}, \quad (7)$$

where $\Sigma_{i,d}^{(\text{diag})}$ is the d-th diagonal element of the i-th Gaussian, and $\boldsymbol{S}_i$ is the full sample covariance matrix given by

$$\boldsymbol{S}_i = \frac{\sum_{t=1}^T h_{t,i}^{\hat{\theta}} (\boldsymbol{\xi}_t^i)(\boldsymbol{\xi}_t^i)^\top}{\sum_{t=1}^T h_{t,i}^{\hat{\theta}}}. \quad (8)$$

The corresponding maximum likelihood estimate of $\boldsymbol{\Sigma}_i^{(\text{diag})}$ is computed as

$$\boldsymbol{\Sigma}_i^{(\text{diag})} = \text{diag} \left(\boldsymbol{B} \boldsymbol{S}_i \boldsymbol{B}^\top \right). \quad (9)$$

Note the variational nature of optimisation where the current estimate of $\boldsymbol{\Sigma}_i^{(\text{diag})}$ is dependent on $\boldsymbol{B}$ and vice versa. Both $\boldsymbol{B}$ and $\boldsymbol{\Sigma}_i^{(\text{diag})}$ are iteratively improved in each EM step and the likelihood is guaranteed to increase at each step.

C. Generalized Formulation

The mixture components of a semi-tied GMM tend to align themselves towards the basis vectors of $\boldsymbol{H}$. To analyze the impact of this alignment on the encoding of movement synergies, we introduce a *tying factor* $\alpha \in [0, 1]$ that controls the degree of tying of the full covariance matrices with the semi-tied covariance matrices, i.e.,

$$\boldsymbol{\Sigma}_i = \alpha \boldsymbol{H} \boldsymbol{\Sigma}_i^{(\text{diag})} \boldsymbol{H}^\top + (1 - \alpha) \boldsymbol{S}_i, \quad (10)$$

where $\alpha = 1$ gives a semi-tied GMM, $\alpha = 0$ leads to a standard GMM, and $(0 < \alpha < 1)$ yields a family of models with intermediate tying of the basis vectors. The overall algorithm is shown in Alg. 1. Investigation of other promising approaches to study the effect of number of basis vectors including extended maximum likelihood linear transform (EMLLT) [18], multiple linear transforms [19], or DP-space [7] is subject to future work.

D. Analysis of Semi-Tied Gaussian Mixture Model

1) Number of Parameters N_p: The number of parameters for K covariance matrices in semi-tied GMM is smaller than the number of parameters for full covariance matrices in GMM ($D^2 + KD$ compared to $\frac{KD(D+1)}{2}$ of GMM respectively). The decrease in number of parameters is accompanied with additional computational cost of finding $\boldsymbol{B}$ and $\boldsymbol{\Sigma}_i^{(\text{diag})}$ in semi-tied GMM. Compared to semi-tied GMM, standard GMM only requires the estimate of $\boldsymbol{S}_i$ in Eq. (8) for the covariance matrix update in each M step. More importantly, semi-tied GMM reveals the latent structure in the data and can be exploited to deal with noisy/insufficient data.

Algorithm 1 *Semi-Tied Gaussian mixture model*

Input: $\{\{\boldsymbol{\xi}_t\}_{t=1}^T, K, \alpha\}$

    procedure EM Semi-Tied_Gaussian_Mixture_Model

1: Initialize the parameters: $\{\{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^K, \boldsymbol{B}\}$

2: **repeat**

3:      $h_{t,i}^{\hat{\theta}} := \dfrac{\pi_i \, \mathcal{N}(\boldsymbol{\xi}_t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)}{\sum_{k=1}^K \pi_k \, \mathcal{N}(\boldsymbol{\xi}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$

4:      $\pi_i := \dfrac{\sum_{t=1}^T h_{t,i}^{\hat{\theta}}}{T}, \qquad \boldsymbol{\mu}_i := \dfrac{\sum_{t=1}^T h_{t,i}^{\hat{\theta}} \boldsymbol{\xi}_t}{\sum_{t=1}^T h_{t,i}^{\hat{\theta}}}$

5:      Compute $\boldsymbol{S}_i$ using Eq. (8)

6:      **repeat**

7:          Compute $\boldsymbol{\Sigma}_i^{(\text{diag})}$ using Eq. (9)

8:          **for** $d := 1$ **to** D **do**

9:              Compute $\boldsymbol{C}$ using Eq. (6)

10:         Compute $\boldsymbol{G}_d$ using Eq. (7)

11:         Compute $\boldsymbol{b}_d$ using Eq. (5)

12:         **end for**

13:      **until** $\boldsymbol{B}$ converges

14:      $\boldsymbol{H} := \boldsymbol{B}^{-1}$, compute $\boldsymbol{\Sigma}_i$ using Eq. (10)

15: **until** $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\xi}) := \sum_{t=1}^T \log\left(\sum_{i=1}^K \pi_i \, \mathcal{N}(\boldsymbol{\xi}_t|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\right)$ converges with $\theta \approx \theta^*$

16: **return** $\theta^* := \{\pi_i^*, \boldsymbol{\mu}_i^*, \boldsymbol{\Sigma}_i^*\}_{i=1}^K$

---

2) Correlation of Mixture Components: To analyse the encoding of semi-tied GMMs, we define $\boldsymbol{M}_c \in \mathbb{R}^{K \times K}$ as the correlation matrix that gives pairwise correlation coefficient between each pair of covariance matrices in the mixture model, i.e.,

$$\boldsymbol{M}_c = \text{corr}\left(\textbf{vec}(\boldsymbol{\Sigma}_1) \quad \textbf{vec}(\boldsymbol{\Sigma}_2) \quad \cdots \quad \textbf{vec}(\boldsymbol{\Sigma}_K)\right), \quad (11)$$

where $\textbf{vec}(\boldsymbol{\Sigma}_i)$ above corresponds to the elements of $\boldsymbol{\Sigma}_i$ in vector form, and $m_c(i,j)$ defines the correlation between the corresponding pair of mixture components. The metric is based on the observation that correlation among the mixture components is higher if they share the same subspace as in semi-tied GMM.

III. Task-Parameterized Semi-Tied GMM

With increasing functional and behavioural expectations of robots, it has become imperative to encode manipulation tasks such that the robots are able to execute them in previously unseen contexts. Task-parameterized models provide a probabilistic formulation to deal with different real world situations by adapting the model parameters, instead of hard coding the solution for each new situation or handling it in an ad hoc manner [20], [21].

Task-parameterized GMMs in [14], [22], [23] can handle new environmental situations by defining external frames of reference. For example, a frame can be attached to an object whose position and orientation may change during the task. When a different situation occurs (position/orientation of the object changes), changes in the task parameters or reference frames are used to modulate the model parameters in order to adapt the robot movement to the new situation.

We represent the task parameters with P coordinate systems, defined by the frames $\{\boldsymbol{A}_j, \boldsymbol{b}_j\}_{j=1}^P$, where $\boldsymbol{A}_j$ denotes the orientation of the frame as a rotation matrix and $\boldsymbol{b}_j$

represents the origin of the frame.[2] The observations $\boldsymbol{\xi}_t$ are observed from different frames of reference forming a third order tensor dataset $\{\boldsymbol{\xi}_t^{(j)}\}_{t,j=1}^{T,P}$ with $\boldsymbol{\xi}_t^{(j)} = \boldsymbol{A}_j^{-1}(\boldsymbol{\xi}_t - \boldsymbol{b}_j)$. The parameters of the task-parameterized GMM are defined by $\theta_p = \{\pi_i, \{\boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)}\}_{j=1}^P\}_{i=1}^K$, where $\boldsymbol{\mu}_i^{(j)}$ and $\boldsymbol{\Sigma}_i^{(j)}$ define the mean and covariance matrix of i-th mixture component in frame P. The probability of data point $\boldsymbol{\xi}_t$ to belong to the i-th Gaussian component at time t (E-step) in the task-parameterized formulation is given by

$$h_{t,i}^{\hat{\theta}_p} = \frac{\pi_i \prod\limits_{j=1}^P \mathcal{N}(\boldsymbol{\xi}_t^{(j)}| \boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)})}{\sum_{k=1}^K \pi_k \prod\limits_{j=1}^P \mathcal{N}(\boldsymbol{\xi}_t^{(j)}| \boldsymbol{\mu}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)})}, \quad (12)$$

where $\prod\limits_{j=1}^P \mathcal{N}(\boldsymbol{\xi}_t^{(j)}| \boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)})$ represents the product of the probabilities of the datapoint observed in P frames to belong to i-th Gaussian in the corresponding frame. Here, we introduce the idea of tying the mixture components in task-parameterized GMM, thereby, tying the i-th covariance matrix in the j-th frame with latent feature space $\boldsymbol{H}^{(j)}$ and diagonal matrix $\boldsymbol{\Sigma}_i^{(j)(\text{diag})}$, i.e., $\boldsymbol{\Sigma}_i^{(j)} = \alpha \boldsymbol{H}^{(j)} \boldsymbol{\Sigma}_i^{(j)(\text{diag})} \boldsymbol{H}^{(j)\top} + (1 - \alpha) \boldsymbol{S}_i^{(j)}$. Maximum likelihood estimates of the parameters remain the same as described in Alg. 1 except the computation is repeated with respect to P different frames and the E-step $h_{t,i}^{\hat{\theta}_p}$ is evaluated using Eq. (12). Similar to semi-tied GMM, increasing α from 0 to 1 increases the effect of tying the mixture components in the task-parameterized formulation.

After the training phase, the task-parameterized semi-tied GMM parameters can be adapted to new environmental situations. For a new unseen situation represented by candidate frames $\{\tilde{\boldsymbol{A}}_j, \tilde{\boldsymbol{b}}_j\}_{j=1}^P$, the new model parameters $\{\tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i\}$ for the i-th mixture component correspond to the product of the linearly transformed i-th Gaussian components in P frames

$$\mathcal{N}(\tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i) \propto \prod_{j=1}^P \mathcal{N}\left(\tilde{\boldsymbol{A}}_j \boldsymbol{\mu}_i^{(j)} + \tilde{\boldsymbol{b}}_j, \tilde{\boldsymbol{A}}_j \boldsymbol{\Sigma}_i^{(j)} \tilde{\boldsymbol{A}}_j^\top\right). \quad (13)$$

Evaluating the product of Gaussian yields

$$\tilde{\boldsymbol{\Sigma}}_i = \left(\sum_{j=1}^P \left(\tilde{\boldsymbol{A}}_j \boldsymbol{\Sigma}_i^{(j)} \tilde{\boldsymbol{A}}_j^\top\right)^{-1}\right)^{-1},$$

$$\tilde{\boldsymbol{\mu}}_i = \tilde{\boldsymbol{\Sigma}}_i \sum_{j=1}^P \left(\tilde{\boldsymbol{A}}_j \boldsymbol{\Sigma}_i^{(j)} \tilde{\boldsymbol{A}}_j^\top\right)^{-1} \left(\tilde{\boldsymbol{A}}_j \boldsymbol{\mu}_i^{(j)} + \tilde{\boldsymbol{b}}_j\right),$$

$$\text{with } \boldsymbol{\Sigma}_i^{(j)} = \alpha \boldsymbol{H}^{(j)} \boldsymbol{\Sigma}_i^{(j)(\text{diag})} \boldsymbol{H}^{(j)\top} + (1 - \alpha) \boldsymbol{S}_i^{(j)}. \quad (14)$$

A. Extension to Hidden Semi-Markov Model (HSMM)

In many robotics applications, it is desirable to encode tasks with an efficient representation capable of handling movements with variable durations, recurring patterns, options in the movement, or partial/unaligned demonstrations [24]. This allows the robot to resume the desired behaviour from

---

[2]Without loss of generality, the frames can be time-varying defined at time t by $\{\boldsymbol{A}_{t,j}, \boldsymbol{b}_{t,j}\}_{j=1}^P$.

situations such as manually stopping the robot in the course of the task. Here, we encapsulate the spatio-temporal information in the data with an explicit duration HSMM that replaces the self-transition probabilities of staying in a state with an explicit model of state duration [4].

In the resulting task-parameterized semi-tied HSMM with K states, each state i output distribution is described by a single Gaussian corresponding to the product of P Gaussians. The parameters of task-parameterized semi-tied HSMM are described by $\theta_h = \left\{ \Pi_i, \{a_{i,m}\}_{m=1}^{K}, \{\boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)}\}_{j=1}^{P}, \mu_i^{D}, \Sigma_i^{D} \right\}_{i=1}^{K}$, where Π_i is the initial state probability, $a_{i,m}$ is the transition probability of moving from state i to state m, $\{\boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)}\}_{j=1}^{P}$ is the parameter set for emission probability of state i, and $\{\mu_i^{D}, \Sigma_i^{D}\}$ represents the mean and the standard deviation of staying d consecutive time steps in state i estimated by a Gaussian $\mathcal{N}(d|\mu_i^{D}, \Sigma_i^{D})$. This duration model is used as a replacement of the self-transition probabilities $a_{i,i}$. Parameters $\left\{ \Pi_i, \{a_{i,m}\}_{m=1}^{K}, \{\boldsymbol{\mu}_i^{(j)}, \boldsymbol{\Sigma}_i^{(j)}\}_{j=1}^{P} \right\}_{i=1}^{K}$ are estimated using an EM algorithm [25] with tying in the i-th covariance matrix of the j-th frame given by Eq. (14). Parameters $\{\mu_i^{D}, \Sigma_i^{D}\}_{i=1}^{K}$ are estimated empirically from the data after training.

Given the new model parameters $\{\tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i\}_{i=1}^{K}$ estimated for a given situation using Eq. (14), the probability of datapoint $\boldsymbol{\xi}_t$ to be in state i at time t is recursively computed using the forward variable $\alpha_{t,i}^{\text{HSMM}} \triangleq P(s_t = i, \boldsymbol{\xi}_1 \ldots \boldsymbol{\xi}_t | \theta_h)$ of explicit duration HSMM [4], [26],

$$\alpha_{t,i}^{\text{HSMM}} = \sum_{j=1}^{K} \sum_{d=1}^{\min(d^{\max}, t-1)} \alpha_{t-d,j}^{\text{HSMM}} \, a_{j,i} \, \mathcal{N}(d|\mu_i^{D}, \Sigma_i^{D})$$
$$\prod_{s=t-d+1}^{t} \mathcal{N}(\boldsymbol{\xi}_s | \tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i), \quad (15)$$

with initialization given by $\alpha_{1,i}^{\text{HSMM}} = \frac{\pi_i \mathcal{N}(\boldsymbol{\xi}_1 | \tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\xi}_1 | \tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\Sigma}}_k)}$. The forward variable is used here to evaluate the current state of the task $\boldsymbol{\xi}_1$ and subsequently plan the movement sequence for the next T steps with $t = 1 \ldots T$. Note that during reproduction, we only use the transition matrix and the duration model to plan the future evolution of the initial/current state $\boldsymbol{\xi}_1$ and omit the influence of the spatial data, i.e., $\mathcal{N}(\boldsymbol{\xi}_t | \tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i) = 1$ for $t > 1$. This is used to retrieve a step-wise reference trajectory $\mathcal{N}(\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\Sigma}}_t)$ from a given state sequence q_t computed from the forward variable, with

$$q_t = \arg\max_i \alpha_{t,i}^{\text{HSMM}}, \quad \hat{\boldsymbol{\mu}}_t = \tilde{\boldsymbol{\mu}}_{q_t}, \quad \hat{\boldsymbol{\Sigma}}_t = \tilde{\boldsymbol{\Sigma}}_{q_t}. \quad (16)$$

B. Trajectory Reproduction with Linear Quadratic Tracking

Consider a double integrator system as an analogue of a unit mass attached to the datapoint $\boldsymbol{\xi}_t$. The desired step-wise reference trajectory $\mathcal{N}(\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\Sigma}}_t)$ can be smoothly tracked by minimizing the cost function

$$c_t(\boldsymbol{\xi}_t, \boldsymbol{u}_t) = \sum_{t=1}^{T} (\boldsymbol{\xi}_t - \hat{\boldsymbol{\mu}}_t)^\top \boldsymbol{Q}_t (\boldsymbol{\xi}_t - \hat{\boldsymbol{\mu}}_t) + \boldsymbol{u}_t^\top \boldsymbol{R}_t \boldsymbol{u}_t, \quad (17)$$
$$\text{s.t.} \quad \dot{\boldsymbol{\xi}}_t = \boldsymbol{A}_d \boldsymbol{\xi}_t + \boldsymbol{B}_d \boldsymbol{u}_t,$$

starting from the initial state $\boldsymbol{\xi}_1$. Let $\boldsymbol{\xi}_t = [\boldsymbol{x}_t^\top \ \dot{\boldsymbol{x}}_t^\top]^\top$, $\hat{\boldsymbol{\mu}}_t = [\hat{\boldsymbol{\mu}}_t^{x\top} \ \hat{\boldsymbol{\mu}}_t^{\dot{x}\top}]^\top$ where $\boldsymbol{x}, \dot{\boldsymbol{x}}$ represent the position and velocity of the double integrator system. Setting $\boldsymbol{Q}_t = \hat{\boldsymbol{\Sigma}}_t^{-1} \succeq 0$, $\boldsymbol{R}_t \succ 0$, the control input $\boldsymbol{u}_t^*$ that minimizes the cost function is given by

$$\begin{aligned} \boldsymbol{u}_t^* &= -\boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{P}_t (\boldsymbol{\xi}_t - \hat{\boldsymbol{\mu}}_t) + \boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{d}_t, \quad (18) \\ &= \boldsymbol{K}_t^{\mathcal{P}} (\hat{\boldsymbol{\mu}}_t^x - \boldsymbol{x}_t) + \boldsymbol{K}_t^{\mathcal{V}} (\hat{\boldsymbol{\mu}}_t^{\dot{x}} - \dot{\boldsymbol{x}}_t) + \boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{d}_t, \end{aligned}$$

where $[\boldsymbol{K}_t^{\mathcal{P}}, \boldsymbol{K}_t^{\mathcal{V}}] = \boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{P}_t$ are the full stiffness and damping matrices, $\boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{d}_t$ is the feedforward term, and $\boldsymbol{P}_t, \boldsymbol{d}_t$ are the solutions of the following differential equations

$$\begin{aligned} -\dot{\boldsymbol{P}}_t &= \boldsymbol{A}_d^\top \boldsymbol{P}_t + \boldsymbol{P}_t \boldsymbol{A}_d - \boldsymbol{P}_t \boldsymbol{B}_d \boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{P}_t + \boldsymbol{Q}_t, \quad (19) \\ -\dot{\boldsymbol{d}}_t &= \boldsymbol{A}_d^\top \boldsymbol{d}_t - \boldsymbol{P}_t \boldsymbol{B}_d \boldsymbol{R}_t^{-1} \boldsymbol{B}_d^\top \boldsymbol{d}_t + \boldsymbol{P}_t \hat{\boldsymbol{\mu}}_t - \boldsymbol{P}_t \boldsymbol{A}_d \hat{\boldsymbol{\mu}}_t, \end{aligned}$$

with terminal conditions set to $\boldsymbol{P}_T = 0$ and $\boldsymbol{d}_T = 0$. Note that the gains can be precomputed before simulating the system if the reference trajectory and/or the task parameters do not change during the reproduction of the task. The resulting trajectory $\boldsymbol{\xi}_t^*$ smoothly tracks the step-wise reference trajectory $\hat{\boldsymbol{\mu}}_t$ and the gains $\boldsymbol{K}_t^{\mathcal{P}}, \boldsymbol{K}_t^{\mathcal{V}}$ stabilize $\boldsymbol{\xi}_t$ along $\boldsymbol{\xi}_t^*$ in accordance with the precision required during the task.

IV. EXPERIMENTS, RESULTS AND DISCUSSION

In this section, we first evaluate the performance of semi-tied GMM to encode a high-dimensional whole body movement (chicken dance from the CMU motion capture database [27]). Following this, we show how the proposed task learning method enables the Baxter robot to open/close a valve and pick-and-place an object by avoiding an obstacle from different environmental situations. Performance setting in all our experiments is as follows: $\{\pi_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{K}$ are initialized using k-means clustering algorithm, $\boldsymbol{B} = 0.1\boldsymbol{I}$, $\boldsymbol{R} = 9\boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix.

A. Whole Body Motion Capture Data - Chicken Dance

The dataset consists of two subjects performing the chicken dance. The dance involves rapid and brisk whole body limb movements with $D = 94$ corresponding to the recorded timestamps ($T \approx 11$ seconds) and the 3-dimensional position of 31 joints for one subject, thereby, making it a challenging problem for the algorithm.

Result of the regenerated dance movement sequence with 75 mixture components and 500 downsampled datapoints is shown in Fig. 3. Plots on bottom right show a generic trend where semi-tied GMM ($\alpha = 1$) requires more mixture components to model the training data in comparison to a standard GMM ($\alpha = 0$). Decreasing the tying factor in a semi-tied GMM gradually pushes the solution towards a standard GMM as seen with $\alpha = 0.6$ and the resulting MSE curve. The number of parameters, however, remain order of magnitudes less for a semi-tied GMM ($15,886$ only in comparison to $334,875$ for a standard GMM with 75 mixture components). Pairwise correlation comparison in Fig. 4 reveals that the correlation among the mixture components as defined in Eq. 11 increases with the semi-tied GMM in comparison to the correlation observed with the standard GMM.
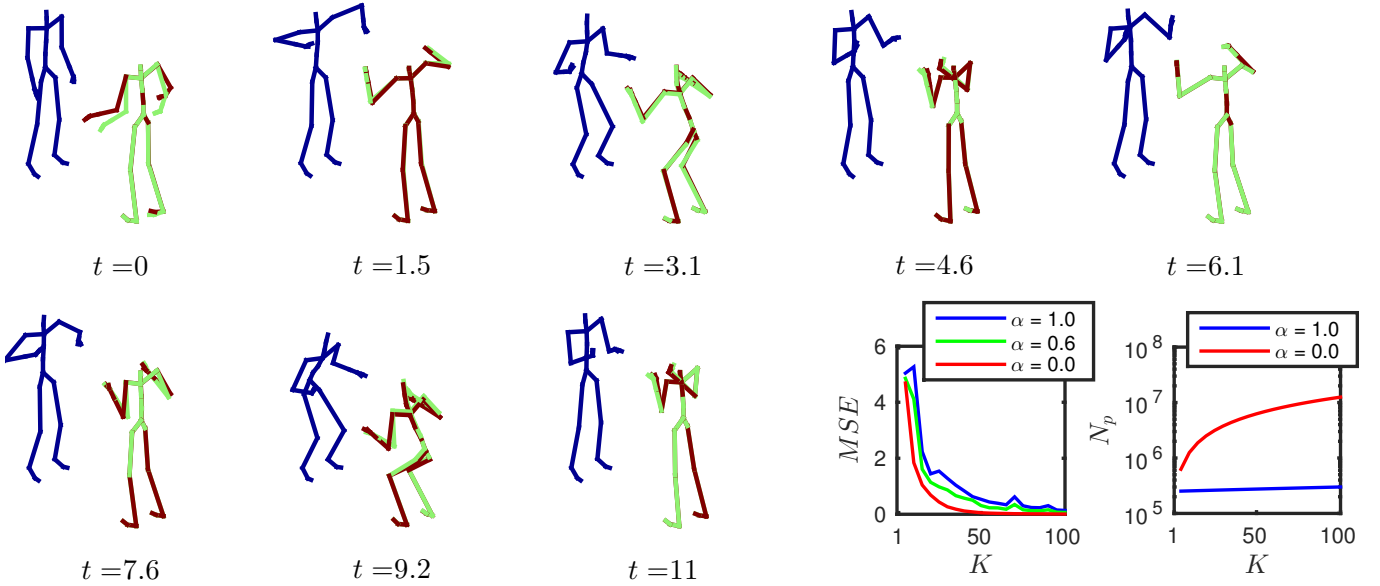
Fig. 3: Chicken dance movement for the two subjects is shown in blue and red. Regenerated movement for the subject in red is shown in green using Gaussian mixture regression. Two plots on bottom right show comparison of mean squared error (MSE) and the number of parameters N_p of covariance matrix in $\log 10$ scale with increasing number of mixture components K. Time is in seconds, $\alpha = 1$ represents semi-tied GMM, whereas $\alpha = 0$ corresponds to a standard GMM.
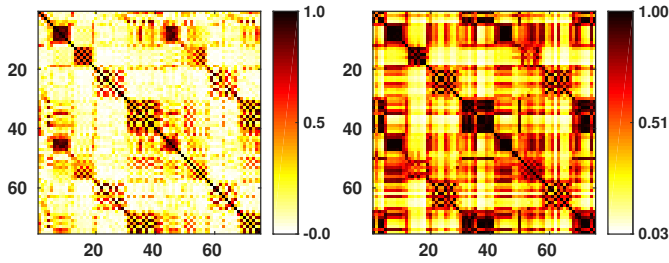

Fig. 4: Pairwise correlation comparison among the mixture components for whole body motion capture data: *(left)* training with standard GMM, *(right)* training with semi-tied GMM.

Fig. 5: Baxter valve opening movement reproduction for an unseen valve configuration: *(left)* encoding with task-parameterized HSMM ($\alpha = 0$), *(right)* encoding with task parameterized semi-tied HSMM ($\alpha = 1$). Note that the mixture components are better aligned and scaled in task-parameterized semi-tied HSMM.

B. Baxter Valve Opening

Valve opening task is a standard benchmark in robotics because it can be applied to a wide range of environments and applications. The goal is to bring the valve in an open position from different initial configurations of the valve using the torque-controlled Baxter robot in our teleoperation setup shown in Fig. 1. Note that the task allows the reuse of synergistic directions such as when reaching the valve and when coming back to a neutral joint angle configuration (home position).

The adaptive aspect of the task requires to ascertain where to grasp the valve and where to stop turning it. Consequently, we attach two frames, one with the observed initial configuration of valve $\{A_1, b_1\}$ and other with the desired end configuration of the valve $\{A_2, b_2\}$ (marked with a visual tag of 0 degree around the valve). We record eight kinesthetic demonstrations with the initial configuration of the valve corresponding to $\{180, 135, 90, 45, 157.5, 112.5, 67.5, 22.5\}$ degrees with the
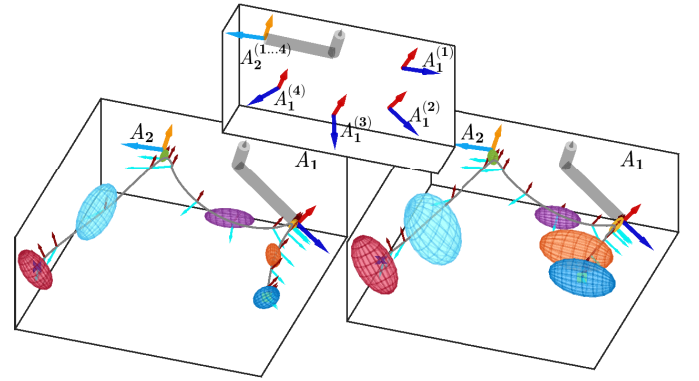
horizontal in the successive demonstrations, $n = 1 \dots 8$. The first 4 demonstrations are used for the training test, while the remaining 4 are used for the test set. Each observation comprises of the end-effector Cartesian position $x_t^p \in \mathbb{R}^3$, quaternion orientation $\varepsilon_t^o \in \mathbb{R}^4$, linear velocity $\dot{x}_t^p \in \mathbb{R}^3$, and quaternion derivative (estimated from angular velocity) $\dot{\varepsilon}_t^o \in \mathbb{R}^4$ for a total of 14 dimensions per sample. Each demonstration is further downsampled to a total of 200 datapoints. For notational convenience, we define $\boldsymbol{\xi}_t = [x_t^\top \ \dot{x}_t^\top]^\top$ with $x_t = \left[x_t^{p\top} \ \varepsilon_t^{o\top} \right]^\top$ and $\dot{x}_t = \left[\dot{x}_t^{p\top} \ \dot{\varepsilon}_t^{o\top} \right]^\top$, and represent the
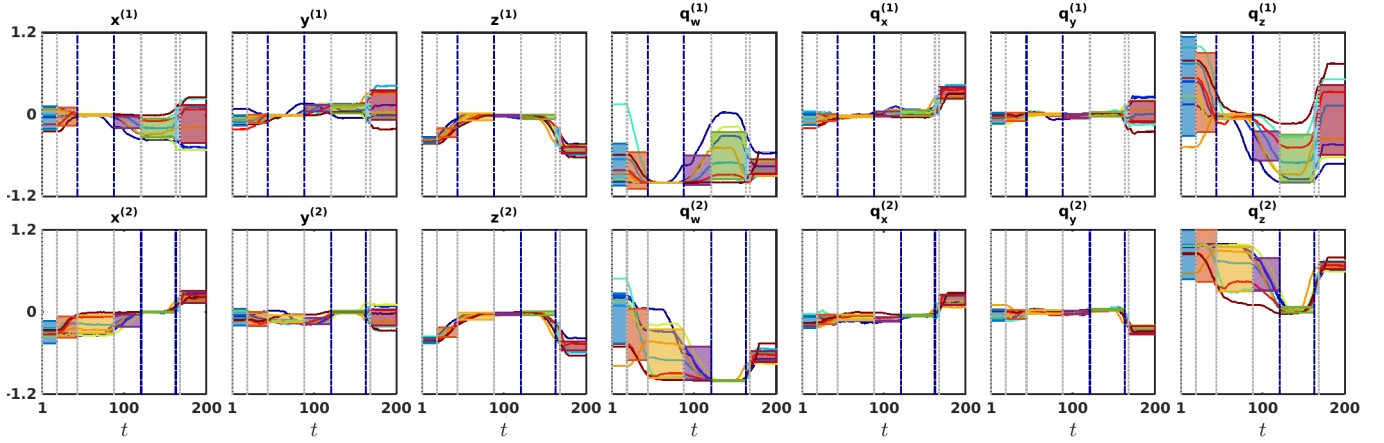
Fig. 6: Variance of the learned model along position and orientation variables in frame 1 *(top)* and frame 2 *(bottom)*. Invariant phase across all demonstrations (highlighted in blue) is observed for components 3 and 5 in frame 1 and frame 2 respectively.
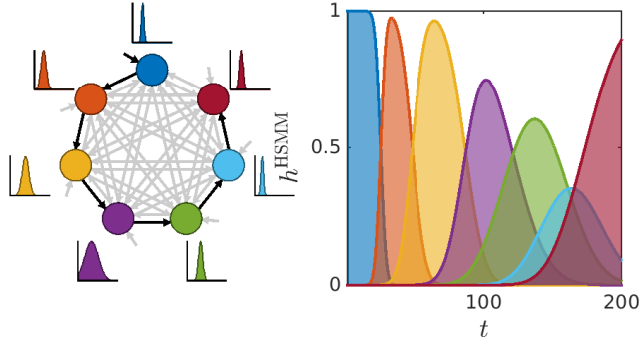

Fig. 7: HSMM encoding of the baxter valve opening task: *(left)* resulting left-right HSMM with duration model shown next to each state ($d^{\max} = 100$), *(right)* rescaled forward variable, $h_{t,i}^{\mathrm{HSMM}} = \frac{\alpha_{t,i}^{\mathrm{HSMM}}}{\sum_{k=1}^{K} \alpha_{t,k}^{\mathrm{HSMM}}}$, evolution with time.

TABLE I: Performance analysis of tying factor α in task-parameterized semi-tied HSMM with training MSE, testing MSE, number of covariance matrix parameters using 7 mixture components and 2 frames, and time required for training the model in seconds.

α	Training MSE	Testing MSE	Number of Parameters	Training Time (s)
valve opening				
0.0	**0.0021**	0.0146	1470	**2.45**
0.5	0.0038	0.0119	1470	5.40
1.0	0.0040	**0.0119**	**588**	9.78
pick-and-place via obstacle avoidance				
0.0	**0.0023**	0.0138	1470	**2.21**
0.5	0.0028	0.0129	1470	4.73
1.0	0.0033	**0.0127**	**588**	10.21

frame as

$$
A_j^{(n)} = \begin{bmatrix} R_j^{(n)} & 0 & 0 & 0 \\ 0 & \mathcal{E}_j^{(n)} & 0 & 0 \\ 0 & 0 & R_j^{(n)} & 0 \\ 0 & 0 & 0 & \mathcal{E}_j^{(n)} \end{bmatrix}, b_j^{(n)} = \begin{bmatrix} p_j^{(n)} \\ 0 \\ 0 \\ 0 \end{bmatrix},
$$

(20)

where $p_j^{(n)} \in \mathbb{R}^3, R_j^{(n)} \in \mathbb{R}^{3\times3}, \mathcal{E}_j^{(n)} \in \mathbb{R}^{4\times4}$ denote the Cartesian position, the rotation matrix and the quaternion matrix of the j-th frame in the n-th demonstration respectively. A sketch of different frames in the demonstrations can be seen in top zoomed portion of Fig. 5. Note that we do not consider time as an explicit variable as the duration model in HSMM encapsulates the timing information locally.

Results of regenerating the movements with 7 mixture components are shown in Fig. 5. The number of Gaussians are empirically selected in this experiment based on the important phases in the task such as reaching, grasping, turning etc. Alternatively, a Bayesian information criterion, or a non-parametric approach based on Dirichlet processes can also be used for model selection. For a given initial configuration of the valve, the model parameters are adapted by evaluating

the product of Gaussians for a new frame configuration. The reference trajectory is then computed from the initial position of the robot arm using the forward variable (see Fig. 7 for HSMM encoding) and tracked using LQT. The robot arm moves from its initial configuration to align itself with the first frame $\{A_1, b_1\}$ to grasp the valve, and follows it with the turning movement to align with the second frame $\{A_2, b_2\}$ before returning back to the home position. Table I quantifies the encoding results with different values of α. We can see that the task-parameterized semi-tied HSMM ($\alpha = 1$) drastically reduces the number of parameters and yields better testing error than training error compared to task-parameterized HSMM with $\alpha = 0$. Fig. 6 shows that the task-parameterized formulation exploits variability in the observed demonstrations to statistically encode different phases of the task. Here, reaching the valve and coming back to home position have higher variability in the demonstrations, whereas aligning with the frames for grasping/turning and stopping the valve have no observed variations in their respective coordinate systems. Consequently, the robot arm is able to reach the valve from different initial configurations, grasp the valve and turn it to the desired position.
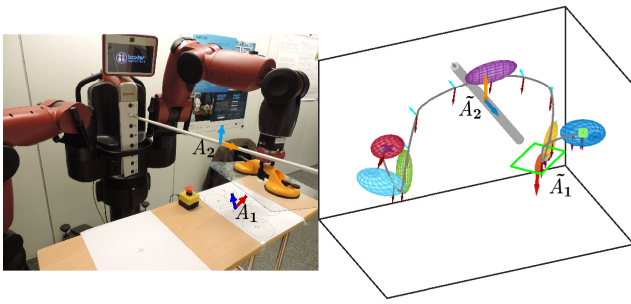
Fig. 8: *(left)* Baxter robot picks the glass plate with a suction lever and places it on the cross after avoiding an obstacle of varying height, *(right)* reproduction for test set demonstration corresponding to maximum obstacle height.

C. Baxter Pick-and-Place via Obstacle Avoidance

The objective in this task is to place the object in a desired target position by picking it from different initial positions and orientations of the object, while adapting the movement to avoid the obstacle (see Fig. 8 for setup with the Baxter robot). We describe the task with two frames, one for the object initial configuration with $\{A_1, b_1\}$ as defined in Eq. (20) and other for the obstacle $\{A_2, b_2\}$ with $A_2 = I$ and b_2 to specify the centre of the obstacle. We collect 8 kinesthetic demonstrations with different initial configurations of the object and the obstacle successively displaced upwards as marked with the visual tags in the figure. Alternate demonstrations $\{1, 3, 5, 7\}$ are used for the training set, while the rest are used for the test set. During evaluation of the learned task-parameterized semi-tied HSMM, we can see that the robot arm is able to generalize effectively by following a similar pattern to the recorded demonstrations in picking and placing the object (see right of Fig. 8 for an example of reproduction requiring extrapolation of the training data). Table I depicts a similar trend to the valve opening task, thereby verifying the efficacy of the proposed method for learning manipulation tasks.

V. CONCLUSION

In this paper, we have presented a technique to tie the covariance matrices of a mixture model with a shared set of basis vectors. The approach is based on the hypothesis that similar coordination patterns occur at different phases in a manipulation task. While feature extraction remains one of the central problems in machine learning, semi-tied GMM yields interesting features in the latent space that can be reused in other parts of the skill. We have shown that the proposed task-parameterized semi-tied HSMM encoding enables the robot to autonomously deal with different situations in manipulation tasks. This has enabled the Baxter robot to tackle valve opening and pick-and-place via obstacle avoidance problems from previously unseen configurations of the environment.

REFERENCES

[1] P. D. McNicholas and T. B. Murphy, "Parsimonious Gaussian mixture models," *Statistics and Computing*, vol. 18, no. 3, pp. 285–296, September 2008.

[2] D. M. Wolpert, J. Diedrichsen, and J. R. Flanagan, "Principles of sensorimotor learning," *Nature Reviews*, vol. 12, pp. 739–751, 2011.

[3] M. J. F. Gales, "Semi-tied covariance matrices for hidden markov models." *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 272–281, 1999.

[4] S.-Z. Yu, "Hidden semi-Markov models," *Artificial Intelligence*, vol. 174, pp. 215–243, 2010.

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society B*, vol. 39, no. 1, pp. 1–38, 1977.

[6] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, 2013.

[7] Y. Wang and J. Zhu, "DP-space: Bayesian nonparametric subspace clustering with small-variance asymptotics," in *Proc. of the 32nd International Conference on Machine Learning, ICML*, 2015, pp. 862–870.

[8] C. Bouveyron and C. Brunet, "Model-based clustering of high-dimensional data: A review," *Computational Statistics and Data Analysis*, vol. 71, pp. 52–78, 2014.

[9] G. J. McLachlan, D. Peel, and R. W. Bean, "Modelling high-dimensional data by mixtures of factor analyzers," *Computational Statistics and Data Analysis*, vol. 41, no. 3-4, pp. 379–388, 2003.

[10] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[11] S. S. R. Li, T. Tian and M. Yang, "3d human motion tracking with a coordinated mixture of factor analyzers," *International Journal of Computer Vision*, vol. 87, no. 1-2, pp. 170–190, 2010.

[12] M. Field, D. Stirling, Z. Pan, and F. Naghdy, "Learning trajectories for robot programing by demonstration using a coordinated mixture of factor analyzers," *IEEE Trans. on Cybernetics*, 2015.

[13] C. J. Leggetter, "Improved acoustic modelling for HMMs using linear transformations," Ph.D. dissertation, University of Cambridge, 1995.

[14] S. Calinon, D. Bruno, and D. G. Caldwell, "A task-parameterized probabilistic model with minimal intervention control," in *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)*, 2014, pp. 3339–3344.

[15] J. Gancet *et al.*, "DexROV: Dexterous Undersea Inspection and Maintenance in Presence of Communication Latencies," vol. 48, no. 2, 2015, pp. 218 – 223, 4th IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles.

[16] A. d'Avella, P. Saltiel, and E. Bizzi, "Combinations of muscle synergies in the construction of a natural motor behavior," *Nature Neuroscience*, vol. 6, pp. 300–308, 2003.

[17] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[18] P. A. Olsen and R. A. Gopinath, "Modeling inverse covariance matrices by basis expansion," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 1, pp. 37–46, 2004.

[19] N. K. Goel and R. A. Gopinath, "Multiple linear transforms," in *IEEE Intl Conf. on Acoustics, Speech, and Signal Processing, ICASSP*, 2001, pp. 481–484.

[20] A. D. Wilson and A. F. Bobick, "Parametric hidden Markov models for gesture recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, pp. 884–900, 1999.

[21] A. Ureche, K. Umezawa, Y. Nakamura, and A. Billard, "Task parameterization using continuous constraints extracted from human demonstrations," *Robotics, IEEE Transactions on*, vol. 31, no. 6, pp. 1458–1471, 2015.

[22] J. Silverio, L. Rozo, S. Calinon, and D. G. Caldwell, "Learning bimanual end-effector poses from demonstrations using task-parameterized dynamical systems," in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, 2015, pp. 464–470.

[23] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, 2016.

[24] S. Krishnan, A. Garg, S. Patil, C. Lea, G. Hager, P. Abbeel, and K. Goldberg, "Unsupervised surgical task segmentation with milestone learning," in *Proc. Intl Symp. on Robotics Research (ISRR)*, 2015, in Press.

[25] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77:2, pp. 257–285, 1989.

[26] S. Calinon, A. Pistillo, and D. G. Caldwell, "Encoding the time and space constraints of a task in explicit-duration hidden Markov model," in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, September 2011, pp. 3413–3418.

[27] R. Gross and J. Shi, "The CMU motion of body (MoBo) database," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-01-18, June 2001.