2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Learning Optimized Human Motion via Phase Space Analysis

Paul Gesel[1], Francesco Mikulis-Borsoi[1], Dain LaRoche[2], Sajay Arthanat[3], and Momotaz Begum[1]

2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-7281-6212-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/IROS45743.2020.9340773

***Abstract*— This paper proposes a dynamic system based learning from demonstration approach to teach a robot activities of daily living. The approach takes inspiration from human movement literature to formulate trajectory learning as an optimal control problem. We assume a weighted combination of basis objective functions is the true objective function for a demonstrated motion. We derive basis objective functions analogous to those in human movement literature to optimize the robot's motion. This method aims to naturally adapt the learned motion in different situations. To validate our approach, we learn motions from two categories: 1) commonly prescribed therapeutic exercises and 2) tea making. We show the reproduction accuracy of our method and compare torque requirements to the dynamic motion primitive for each motion, with and without an added load.**

I. Introduction

Our goal is to employ robotic technology for assistive applications, such as robot guided exercise therapy and household service robots. The robot's ability to faithfully reproduce human-like motion from observations is a critical requirement for assistive robots. Trajectory learning is a rich domain in robot learning from demonstration (LfD) research [1], [2], [3], [4]. LfD-powered robots have successfully learned to play T-ball [5], tic-tac-toe [6], chess [7], perform pick-n-place tasks [8], pour drinks [9], do chores [10], and play drums [11].

The vast majority of LfD research is dedicated to manipulation tasks [4]. Success is generally measured by the robot's ability to manipulate objects, while adapting to a dynamic environment. However, object manipulation tasks may involve finely coordinated motion, while satisfying dynamic and joint constraints. For example, therapeutic exercise training must capture both the fine details of human movement and anatomical limitations. Alternatively, tea making is only concerned with reaching a goal configuration. This paper develops a dynamic system based approach, which accurately reproduces trajectories, optimizes for robot dynamics, and adapts to varying end-effector loads.

II. Related Works

State of the art methods for human motion learning include dynamic system learning [12], [13], inverse optimal control (IOC) [14], [15], and statistical learning [16], [17], approaches.

[1] Department of Computer Science, University of New Hampshire, USA paul.gesel@unh.edu, fam1001@wildcats.unh.edu, mbegum@cs.unh.edu
[2] Department of Kinesiology, University of New Hampshire, USA Dain.LaRoche@unh.edu
[3] Department of Occupational Therapy, University of New Hampshire, USA Sajay.Arthanat@unh.edu

Dynamic system based approaches benefit from quick reaction time by mapping the robot's state to actions [18]. State is commonly defined by the joint angles or the end-effector position for manipulation tasks. Another benefit of dynamic system based approaches is robustness to perturbation and local re-planning [12], hence, they can overcome some level of uncertainty. However, these approaches do not necessarily overcome problems with limited training data. Methods that can handle multiple demonstrations, such as Stable Estimator of Dynamical Systems (SEDS), require training examples throughout the entire state space to generate meaningful trajectories. Otherwise, trajectories may not be consistent with the original demonstration due to a lack of information in that state [13].

The dynamic motion primitive (DMP) is a popular dynamic system based approach that learns motion from a single demonstration [12]. It is robust to perturbations and small goal adaptations, but fails to generate meaningful motion in situations that are significantly different than the original demonstration. This is partially because each dimension is learned separately, thus relationships between dimensions are not encoded [13]. Since then, DMP has been extended to include multiple demonstrations for motion learning [19]. Similar to SEDS, a lack of training data can produce inconsistent motion, hence, more demonstrations are needed. Since DMP is implicitly time dependant, a heuristic is required to re-index time in the presence of large temporal perturbations.

The goal of inverse optimal control (IOC) approaches is to find a cost function that explains the underlying structure of the demonstrations [20]. By utilizing the discovered cost function, trajectories that are optimal with respect to the demonstrations can be generated. These approaches benefit from the ability to adapt to situations never seen in the training set. For example, if obstacles are in the path, constraints can be added to avoid penetration. Additionally, penalties can be added for getting too close to obstacles. IOC has demonstrated success in generating control policies for high dimensional problems, such as those seen with humanoids [14]. The major difficulty of IOC is the complexity in learning the true cost function and efficiently optimizing trajectories for the learned cost function. These methods often require large amounts of training data.

The probabilistic approach of trajectory learning in [7] creates a statistical representation of the demonstrated motion via a Gaussian mixture model (GMM). This approach learns a time-dependent trajectory distribution from multiple demonstrations. The criticisms of this approach are the difficulty of online trajectory adaptation, such as avoiding

obstacles and changing the goal configuration. Furthermore, since GMM-based trajectory learning is also time dependent, in the presence of temporal perturbations, a heuristic is again required to re-index time [21].

None of these approaches combine efficient optimization of trajectories with respect to the robot's dynamics, exhibit time-invariant behavior, and are robust to perturbations. It is often assumed that there exist a low-level controller, which can track the trajectory generated by these approaches. If the demonstration is teleoperated or kinesthetically delivered, then this is a realistic assumption because the robot's dynamics are implicitly captured. However, this becomes less likely if the robot's dynamics change due to added end-effector load. Additionally, if the motion is demonstrated by a human and then mapped to the robot, the robot's dynamics are not implicitly captured. Therefore, we propose a dynamic system based approach that includes the robot's dynamics in its formulation, enabling it to balance execution accuracy with dynamic limitations.

We take inspiration from human movement literature to formulate trajectory learning as an optimal control problem. Instead of optimizing a single trajectory, we aim to optimize a parametric control policy. Additionally, we formulate our approach to include constraints on the robot's dynamics and joint angles. The main benefits of this approach are that:

- small perturbations during execution are handled by the dynamic system based controller
- robot dynamic constraints are not violated
- trajectories naturally adapt to varying end-effector loads

The model proposed in this paper combines the strengths of dynamic system LfD and optimal control into one framework. To the extent of the authors' knowledge, there are no other time-invariant dynamic system based approaches, which can efficiently account for varying robot dynamics.

III. The Proposed Approach

We employ a dynamic system based LfD controller, namely the Phase Space Model (PSM) [22], to learn generic manipulation tasks. In section III-A, we present an algorithm to generate robot joint trajectories corresponding to human demonstrations. In section III-B, we present a control strategy derived from the PSM. We then develop basis objective functions, linear constraints, and propose an optimization algorithm in sections III-C, III-D, and III-E, respectively.

$n \in \mathbb{N}$: number of robot joints
$m \in \mathbb{N}$: number of frames in the demonstration
$Q \in \mathbb{R}^{n \times m}$: correspondence matrix
$\dot{Q} \in \mathbb{R}^{n \times m}$: time derivative of the correspondence matrix
$V \in \mathbb{R}^{n \times n}$: eigenvector matrix of QQ^T
$Q' \in \mathbb{R}^{n \times m}$: Q expressed in transformed coordinates
$\dot{Q}^{'} \in \mathbb{R}^{n \times m}$: time derivative of $Q^{'}$
$q(t) \in \mathbb{R}^n$: joint position state
$\dot{q}(t) \in \mathbb{R}^n$: joint velocity state
$\ddot{q}(t) \in \mathbb{R}^n$: joint acceleration action

$D \in \mathbb{R}^{6 \times m}$: human demonstration matrix
$D^e \in \mathbb{R}^{3 \times m}$: vectors from human shoulder to elbow
$D^w \in \mathbb{R}^{3 \times m}$: vectors from human elbow to wrist
$J^{e\omega} \in \mathbb{R}^{3 \times n}$: jacobian's rotational part of robot's elbow
$J^{w\omega} \in \mathbb{R}^{3 \times n}$: jacobian's rotational part of robot's wrist
$R^e(q) \in SO(3)$: rotation matrix from base to elbow
$R^w(q) \in SO(3)$: rotation matrix from base to wrist
$s \in \mathbb{N}$: number of segmentation points along q_1'
$P' \in \mathbb{R}^{s \times n}$: cut point matrix in transformed coordinates
$\hat{q} \in \mathbb{R}^n$: piece-wise spline approximation of the 2nd through nth elements of q' as a function of q_1'
$p^o \in \mathbb{N}$: order of the piece-wise polynomial
$A \in \mathbb{R}^{(p^o+1) \times (s-1) \times (n-1)}$: tensor of spline coefficients
$z = [0, 0, 1]^T$
$l \in 1...m$: frame index
$i \in 1...n$: joint index
$r \in 1...s$: segment index

We will use the above notation throughout the paper. Additionally, elements of vectors are indexed with a single subscript and elements of matrices are indexed with two subscripts. A single subscript on a matrix indicates a column of the matrix. Some symbols are written with superscript notation. This is for designation and does not represent a mathematical operation, e.g the prime superscript ($'$) designates transformed coordinates. Furthermore, the hat ($\hat{}$) and dot ($\dot{}$) notations represent an approximate function and time derivative, respectively.

A. Human to Robot Correspondence

The robot's anthropomorphism plays a critical role in finding an meaningful correspondence between the robot and the human demonstration. Ideally, the robot should have a human-like range of motion and form factor. The Yumi robot is perhaps the best suited commercially available robot, given its human-like structure. This helps simplify the correspondence problem.

We take inspiration from the therapeutic exercise setting to develop a correspondence mapping. The clinical standard for measuring joint angles is a goniometric measurement [23]. The goniometer approximates joint angles by measuring the angle between limb segments. For example, the elbow joint angle is approximated by measuring the angle between the upper arm and the forearm. To this end, we define the objective function in eq. (1), which penalizes misalignment between the robot's and therapist's elbow and wrist vectors.

$$\min_q \left|\left| \frac{D_l^e}{\|D_l^e\|} - R(q)^e z \right|\right| + \left|\left| \frac{D_l^w}{\|D_l^w\|} - R(q)^w z \right|\right| \quad (1)$$

Here, D^e is a matrix of vectors from the shoulder to the elbow and D^w contains the vectors from the elbow to the wrist for each frame of the demonstration. These vectors are illustrated in Fig. 1.

Applying algorithm 1 will generate the correspondence matrix Q from the human demonstration using an iterative least squares approach. For each frame of the demonstration, a joint configuration q is found that minimizes eq. (1) and added

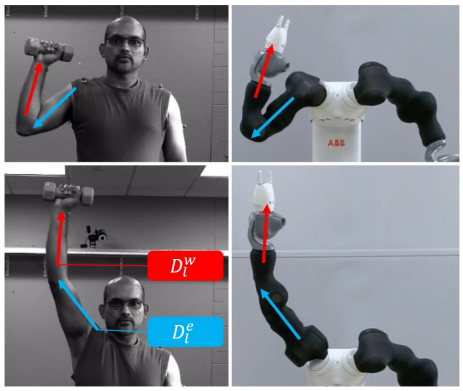

Fig. 1: robot therapist vector based mapping. The vector D_l^e is from the shoulder to the elbow and the vector D_l^w is from the elbow to the wrist at frame l.

Algorithm 1 Generate Correspondence Matrix

Input: D^e, D^w, x_0
Output: Q
$\theta \leftarrow 0.001$
$\lambda \leftarrow 0.0001$
$u \leftarrow 0.005$
$l \leftarrow 0$
$Q_1 = x_0$
while $l < m$ **do**
  $l \leftarrow l + 1$
  do
    $\epsilon_e \leftarrow \frac{D_l^e}{\|D_l^e\|} - R^e z$
    $Q_l \leftarrow Q_l + \theta \left((J_e^\omega)^T J_e^\omega + \lambda I\right)^{-1} (J_e^\omega)^T \epsilon_e$
    $\epsilon_w \leftarrow \frac{D_l^w}{\|D_l^w\|} - R^w z$
    $Q_l \leftarrow Q_l + \theta \left((J_w^\omega)^T J_w^\omega + \lambda I\right)^{-1} (J_w^\omega)^T \epsilon_w$
  while $\|\epsilon_e\| + \|\epsilon_w\| \geq u$
end while

as a column in the correspondence matrix. This procedure aligns the robot's 3rd joint axis with the therapist's upper arm vector and the robot's 5th joint axis with the therapist's lower arm vector. It is important that a reasonable initial value (x_0) is used; otherwise, the algorithm could have poor convergence and correspondence. For all tasks considered in this paper, initializing x_0 to the zero vector was sufficient. It should be noted that θ, λ, and the convergence threshold u may need fine-tuning.

B. Control Strategy

Given a correspondence matrix Q, we define a new matrix $Q^{'}$ in transformed coordinates with its origin at Qm.

$$Q^{'} = V^{-1}(Q - Q_m \mathbb{1}^T) \tag{2}$$

Here, V is the eigenvector matrix of QQ^T, sorted by eigenvalues. We then build the cut point matrix $P^{'}$ by sampling s points from $Q^{'}$ uniformly in time. We assume that $q_1^{'}$ is never decreasing, more specifically, $\dot{q}_1^{'} \geq 0$ for the entire movement. To ensure this, if $P_{11}^{'}$ is positive, then the eigenvector matrix is negated. Our approach is to implement the PSM in the transformed coordinates. The PSM is a dynamic system based LfD method that models motion with a set of piece-wise linear time invariant systems. We denote these systems as Phase Space Transition Functions (PSTFs), which are expressed in eq. (3),

$$\ddot{q}_1^{'}(q^{'}) = \begin{cases} k_{j(q_1^{'})} q_1^{'} + c_{j(q_1^{'})} & \dot{q}_1^{'} \geq 0 \\ b & \dot{q}_1^{'} < 0 \\ \frac{-\dot{q}_1^{'2}}{2(q_{ob}^{'} - q_1^{'})} & CD = 1 \end{cases} \tag{3}$$

where k and c are parameter vectors for the PSTFs, j is the segment index function (eq. (4)), and $q_1^{'}$ is the first element of the robot joint angles in the transformed coordinates. The terms b, CD, and $q_{ob}^{'}$ are discussed in our previous work.

$$j(x) = \underset{y \in \{1 \ldots s\}}{\text{argmax}} \{y,\ x - P_{y1}^{'} > 0\} \tag{4}$$

Since $q_1^{'}$ is always increasing, we can define $\hat{q}_i$ as a generic function $f_i(q^{'})$, where $i = 2, 3, 4 \ldots, n$, to approximate the demonstration.

$$\hat{q}_i(q^{'}) = \begin{cases} q_1^{'} & i = 1 \\ f_i(q^{'}) & i > 1 \end{cases} \tag{5}$$

The first and second derivatives of $\hat{q}_i$ are the following equations:

$$\dot{\hat{q}}_i(q^{'}) = \frac{d}{dq_1^{'}} f_i(q^{'}) \dot{q}_1^{'} \tag{6}$$

$$\ddot{\hat{q}}_i(q^{'}) = \frac{d^2}{dq_1^{'2}} f_i(q^{'}) \dot{q}_1^{'2} + \frac{d}{dq_1^{'}} f_i(q^{'}) \ddot{q}_1^{'} \tag{7}$$

We define each function f_i to be a piece-wise polynomial.

$$f_i(q^{'}) = \sum_{d=0}^{p^o} A_{(d+1)j(q_1^{'})(i-1)} q_1^{'d} \tag{8}$$

Here, A is a tensor containing coefficients of the splines and p^o is the order of the polynomial. Combining eq. (3), eq. (7), and a position feedback term, results in the final controller as shown in eq. (9).

$$\ddot{q}_i^{'}(q^{'}) = \begin{cases} \ddot{q}_1^{'}(q^{'}) & i = 1 \\ \ddot{\hat{q}}_i(q^{'}) + p(\hat{q}_i(q^{'}) - q_i^{'}) & i > 1 \end{cases} \tag{9}$$

Here, p is a proportional gain parameter. Fig. 2 is a visual representation of the control setup.

C. Basis Objective Functions

In our previous work, we designed the PSM for trajectory learning without considering optimality. The PSTFs' parameters were computed with regression, while constraining velocity. In this section, we derive meaningful basis objective functions to accommodate optimality. Human movement literature suggests that the central nervous system optimizes the basis objective functions shown in Tab. I with task-specific constraints to generate motion [24], [25]. We approximate three of these basis objective functions as linear functions of the PSTFs' parameters. The kinetic energy, torque, and jerk,

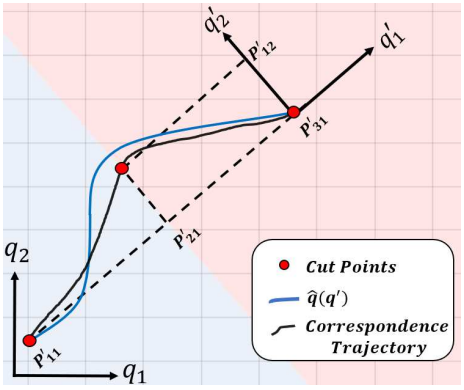

Fig. 2: 2D illustration with three cut points. The blue and red shaded regions indicate where $j(q_1^{'})$ is 1 and 2, respectively.

TABLE I human movement basis objective functions [24], where θ represents joint angles, τ represents joint torques, and T is the final time.

Energy	$\int_0^T \lvert\dot{\theta}\tau\rvert dt$
Torque	$\int_0^T \tau^2 dt$
Torque change	$\int_0^T \dot{\tau}^2 dt$

are shown in equations (10), (11), and (12), respectively.

$$h(q^{'}) = \frac{1}{2}k_{j(q_1^{'})}(q_1^{'2} - P'^2_{j(q_1^{'})1}) + c_{j(q_1^{'})}(q_1^{'} - P'_{j(q_1^{'})1})$$

$$E_i(q^{'}) = \begin{cases} h(q^{'}) + \frac{1}{2}\dot{Q}'^2_{1i} & j(q_1^{'}) = 1, i = 1 \\ h(q^{'}) + E_1(P'_{j(q_1^{'})1}) & j(q_1^{'}) > 1, i = 1 \\ E_1(q^{'})(\frac{d}{dq_1^{'}} f_i(q^{'}))^2 & i > 1 \end{cases} \quad (10)$$

$$\tau(q^{'}) = M(V\hat{q}(q^{'}) + Q_m)V\ddot{q}^{'}(q^{'}) + g(V\hat{q}(q^{'}) + Q_m) \quad (11)$$

$$\dot{\tau}(x, y) = \tau(y) - \tau(x) \quad (12)$$

Here, τ represents the robot's joint torques, M is the mass inertia matrix function, and g is the gravity torque function. It should be noted that eq. (11) is an approximation of the joint torques because the Coriolis matrix is not included. The Coriolis terms would introduce non-linearity in terms of the PSTFs' parameters, preventing the formulation of a quadratic basis objective function. We found that Coriolis and centrifugal forces are relatively small compared to the gravitational and inertial forces, hence, it is a reasonable simplification. Quadratic basis objective functions analogous to those in Tab. I are summarized in Tab. II. The main difference between the functions in Tab. I and Tab. II, is that the former are integrated over time, while the latter are evaluated at discrete points along the trajectory. Furthermore, the basis objective function C_1 minimizes the kinetic energy difference between the demonstration and the trajectory generated by the PSTFs, instead of minimizing total energy. Finally, C_4 is is a quadratic basis objective function that optimizes the spline coefficients A to minimize the distance between the demonstration and $\hat{q}$.

TABLE II human movement basis objective functions expressed in terms of the PSTFs' parameters. C_1, C_2, C_3, and C_4 penalize energy difference, torque, jerk, and spline approximation error, respectively.

C_1	$\sum_{l=1}^{m} \lVert 2E_1(Q_l^{'}) - \dot{Q}'^2_{l1} \rVert$
C_2	$\sum_{l=1}^{m} \lVert \tau(Q_l^{'}) \rVert$
C_3	$\sum_{l=1}^{m-1} \lVert \dot{\tau}(Q_l^{'}, Q_{l+1}^{'}) \rVert$
C_4	$\sum_{i=2}^{n} \sum_{l=1}^{m} \lVert f_i(Q_l^{'}) - Q_{li}^{'} \rVert$

D. Constraints

We consider two types of constraints for trajectory learning: (1) dynamic and joint constraints and (2) task-specific constraints. Dynamic and joint constraints are generally inequality constraints, e.g. joint angles and torques are upper and lower bounded. Task-specific constraints are often equality constraints, to ensure goal configurations are reached. We express the dynamic constraints as follows,

$$L_imin < f_i(Q_l^{'}) < L_imax \quad l = 1...m, i = 2...n \quad (13)$$

$$\tau_{min} < \tau(Q_l^{'}) < \tau_{max} \quad l = 1...m \quad (14)$$

where τ_{max} is the maximum and τ_{min} is the minimum allowable torque, and L_imin and L_imax correspond to the joint limits of element i in transformed coordinates. To minimize unwanted jumps in torque between segments and preserve continuity of position, velocity, and acceleration, we impose the following constraints for $r = 2...s-1, i = 1...n-1$.

$$\sum_{d=0}^{p^o} A_{(d+1)(r-1)i} P'^d_{r1} = \sum_{d=0}^{p^o} A_{(d+1)ri} P'^d_{r1} \quad (15)$$

$$\sum_{d=0}^{p^o} dA_{(d+1)(r-1)i} P_{r1}^{'(d-1)} = \sum_{d=0}^{p^o} dA_{(d+1)ri} P_{r1}^{'(d-1)} \quad (16)$$

$$\sum_{d=0}^{p^o} d(d-1)A_{(d+1)(r-1)i} P_{r1}^{'(d-2)} = \sum_{d=0}^{p^o} d(d-1)A_{(d+1)ri} P_{r1}^{'(d-2)} \quad (17)$$

The task specific constraints we consider in this paper are initial position and path direction, final position and path direction, and final velocity. These constraints are expressed in the following equations, for $i = 2...n$.

$$f_i(Q_1^{'}) = Q_{i1}^{'} \quad (18)$$

$$\sum_{d=0}^{p^o} dA_{(d+1)1(i-1)} Q_{11}^{'(d-1)} = \frac{\dot{Q}_{i1}^{'}}{\dot{Q}_{11}^{'}} \tag{19}$$

$$f_i(Q_m^{'}) = Q_{im}^{'} \tag{20}$$

$$\sum_{d=1}^{p^o} dA_{d(s-1)(i-1)} Q_{1m}^{'(d-1)} = \frac{\dot{Q}_{im}^{'}}{\dot{Q}_{1m}^{'}} \tag{21}$$

$$E_1(Q_m) = \epsilon \tag{22}$$

In eq. (22), ϵ is a small non-zero value. For our implementation, we used $\epsilon = 0.00001$. A small, but non-zero final velocity ensures that an unstable equilibrium does not occur at the goal position, which would require infinite time to converge. One additional inequality constraint is imposed to ensure convergence.

$$E_1(Q_l^{'}) > 0 \quad l = 1...m \tag{23}$$

The inequality above ensures that the kinetic energy never falls below zero, guaranteeing $q_1^{'}$ is always increasing.

E. Optimization

We optimize torque, jerk, and the trajectories of the robot's joints with an iterative constrained linear least squares method. Assuming the true objective function is expressed as a weighted sum of the basis objective functions, we define eq. (24) and eq. (25).

$$C^p = w_1^p C_1 + w_2^p C_2 + w_3^p C_3 \tag{24}$$

$$C^s = w_2^s C_2 + w_3^s C_3 + w_4^s C_4 \tag{25}$$

Here, C^p and C^s are the objective functions for the parameters of the PSTFs and splines, respectively. Since C^p is a quadratic function of the PSTFs' parameters, the optimal parameters can be calculated with constrained linear least squares. If C_4 is the only non-zero weighted basis objective function, then C^s is a quadratic function of A. In general, the objective functions C_2 and C_3 introduce nonlinear behavior. However, if we assume that the inertial matrix M and the gravitational torque g are constant, then the problem is solvable with linear least squares. The previous assumption is often violated, so we use a trust parameter θ to limit the step in the solution direction. Our iterative optimization method is described in algorithm 2. We alternate between optimizing the PSTFs' and splines' parameters for several iterations.

Algorithm 2 Optimize Trajectory

Input: $Q^{'}$, $\dot{Q}^{'}$, w^p, w^s
Output: c, k, A
$\theta \leftarrow 0.2$
$x \leftarrow 0$
$A \leftarrow$ initialize to zeros
while $x < 10$ **do**
  $[k, c] =$ lsqlin$(w_1^p C_1 + w_2^p C_2 + w_3^p C_3)$
  $A = (1-\theta)A + \theta$ lsqlin$(w_2^s C_2 + w_3^s C_3 + w_4^s C_4)$
  $x \leftarrow x + 1$
end while

IV. Experiments

We conducted two human subject studies, involving exercise therapy (experiment 1) and tea making (experiment 2). For each experiment, the participants' movements were recorded with a Qualisys motion capture system. Demonstration matrices (D^e and D^w), were constructed and then converted into correspondence matrices using algorithm 1. Reference trajectories for DMP were extracted from the rows of the correspondence matrix. The piece-wise polynomial order was 6 for all results. Additionally, 12 cut points were used for the scaption exercise and 6 for the milk pouring task. In the following sections, we compare our method to DMP for three different experiments.

A. Experiment 1: Scaption Exercise Movement

Six commonly prescribed therapeutic exercises, namely shoulder press, lateral raise, forward raise, external rotation, internal rotation, and scaption were performed by a therapist (the third author of this paper). Fig. 3 illustrates correspondence values calculated for several of these exercises. In this experiment, we wanted to show how our approach can closely mimic the human demonstration, like DMP, while still generating a smooth motion. To achieve this, all basis objective function weights, except w_1^p and w_4^s were set to zero. This minimizes the velocity and position difference between the demonstration and the learned motion. The results for the scaption movement are shown in Fig. 4 and Tab. III. We achieve similar accuracy, torque, and execution time compared to DMP.

B. Experiment 2: Milk Pouring Task

The participants performed a tea making sequence composed of seven movements, including: turning an oven on, pouring water, adding sugar, pouring milk, adding a tea bag, stirring, and turning an oven off. We are interested in the milk pouring task because it involves moving a heavy object with variable weight. In this experiment, we wanted to show how our approach can optimize the splines' and PSTFs' parameters to minimize torque and jerk. All basis objective function weights were non-zero, except w_4^s. The optimization generated a significantly different path than the human demonstration in order to minimize torque. The results are shown in Fig. 5 and Tab. III. We require substantially less torque than DMP's trajectories, to achieve the task in the same amount of time.

C. Experiment 3: Adaptation to External Load

In this experiment, we wanted to show how our method adapts its motion in the presence of an added external load, hence, 1 kg was added to the end-effector for the previous two movements. The basis objective function weights were left unchanged, but a 16 (Nm) torque limit was imposed on the scaption movement. The adapted trajectories for both

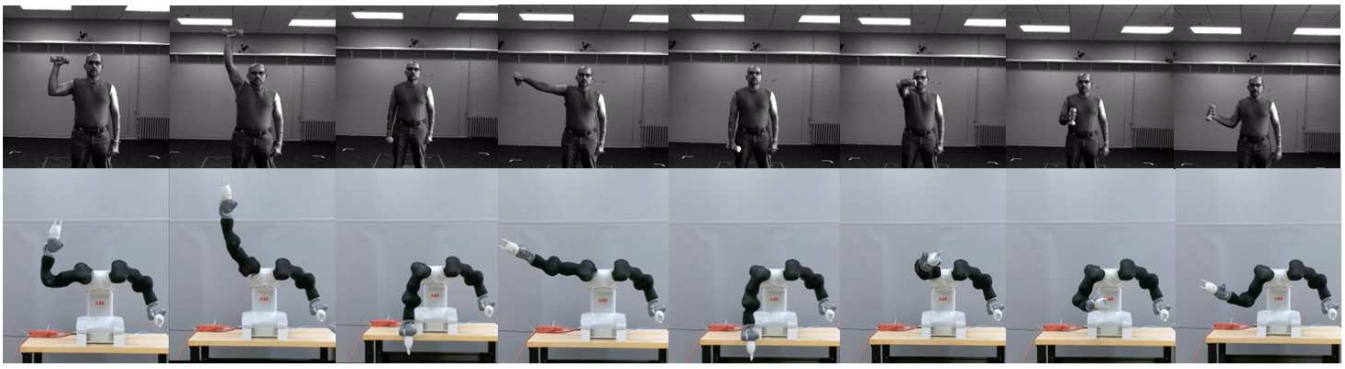

Fig. 3: robot configurations corresponding to the therapist's demonstrations for shoulder press, lateral raise, forward raise, and external rotation.

TABLE III mean norm of joint torques from experiments 1, 2, and 3 in (N^2m^2).

weight	Scaption		Pour Milk	
0 kg	107.9	109.9	31.4	38.2
1 kg	188.9	190.6	84.6	110.6
	PSM	DMP	PSM	DMP

movements are shown in Fig. 4 and 5 and torque results are summarized in Tab. III. For the scaption movement, the maximum torque experienced during the movement was 16 (Nm), as expected, and 19.5 (Nm) for DMP.

For our experiments, we chose to minimize the deviation between the robot and therapist for the exercise movement and to minimize torque and jerk for the milk pouring task. While the true objective functions for these tasks may be more complicated, we believe the flexibility of our approach is applicable to multiple domains. Additionally, the robot needs to be aware of its dynamic limitations to ensure safe operating conditions, hence, torque limits are required. This is especially true when load is added to the end-effector or when motion is mapped directly from a human demonstration to the robot.

V. Conclusions and Future Work

We took inspiration from the therapeutic exercise domain to develop a correspondence mapping from a human demonstration to the robot. Basis objective functions, analogous to those seen in human movement literature, were formulated in terms of the PSTFs' and splines' parameters. Linear constraints on torque, jerk, and joint angles were also developed in terms of these parameters. Our method successfully generated trajectories for therapeutic exercises and tea making, with and without added end-effector loads. This approach further generalizes the PSM to adapt its trajectories in a more contextually meaningful way. In the future, we will explore the possibility of learning the basis objective function weights from multiple demonstrations.

Acknowledgment

This work was supported in part by the National Science Foundation (IIS 1830597).

References

[1] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 358, no. 1431, pp. 537–547, 2003.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[3] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Springer handbook of robotics*. Springer, 2008, pp. 1371–1394.

[4] S. Chernova and A. L. Thomaz, "Robot learning from human teachers," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 8, no. 3, pp. 1–121, 2014.

[5] J. Peters and S. Schaal, "Policy gradient methods for robotics," in *Int. Conf. Intelligent Robots and Systems*. IEEE, 2006.

[6] S. Niekum, S. Osentoski, G. Konidaris, S. Chitta, B. Marthi, and A. G. Barto, "Learning grounded finite-state representations from unstructured demonstrations," *The International Journal of Robotics Research*, vol. 34, no. 2, pp. 131–157, 2015.

[7] A. B. S. Calinon, F. Guenter, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 2, pp. 286–298, 2007.

[8] F. Stulp, E. A. Theodorou, and S. Schaal, "Reinforcement learning with sequences of motion primitives for robust manipulation," *IEEE Transactions on robotics*, vol. 28, no. 6, pp. 1360–1370, 2012.

[9] B. Akgun, M. Cakmak, J. W. Yoo, and A. L. Thomaz, "Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. IEEE/ACM, 2012, pp. 391–398.

[10] N. Figueroa, A. L. Pais Ureche, and A. Billard, "Learning complex sequential tasks from demonstration: A pizza dough rolling case study," in *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, ser. HRI '16. IEEE Press, 2016, pp. 611–612. [Online]. Available: http://dl.acm.org/citation.cfm?id=2906831.2907006

[11] D. Pongas, A. Billard, and S. Schaal, "Rapid synchronization and accurate phase-locking of rhythmic motor primitives," in *Int. Conf. Intelligent Robots and Systems*. IEEE, 2005.

[12] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 763–768.

[13] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, Oct 2011.

[14] A. Byravan, M. Monfort, B. D. Ziebart, B. Boots, and D. Fox, "Graph-based inverse optimal control for robot manipulation," in *IJCAI*, 2015.

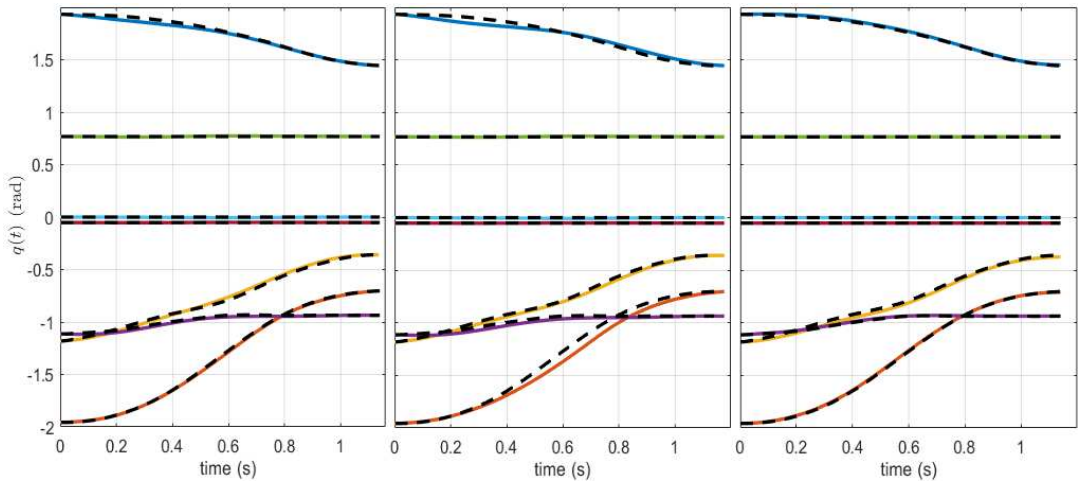

Fig. 4: Reproduced shoulder scaption movement for the exercise experiment. From left to right is our approach with no added load, 1 kg added, and DMP. The correspondence matrix values are shown in black.

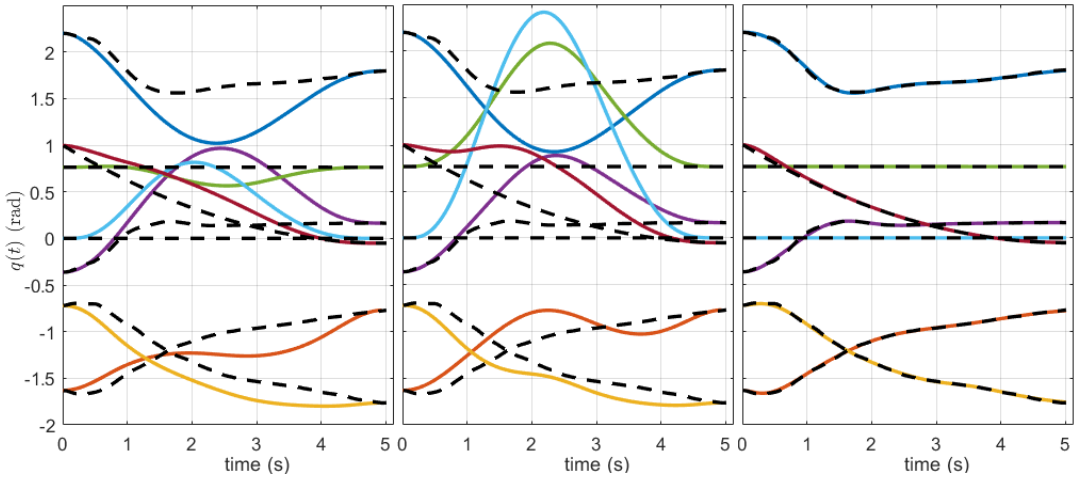

Fig. 5: Reproduced milk pouring movement for the tea making experiment. From left to right is our approach with no added load, 1 kg added, and DMP. The correspondence matrix values are shown in black.

[15] E. Todorov, W. Li, and X. Pan, "From task parameters to motor synergies: A hierarchical framework for approximately optimal control of redundant manipulators," *J. Robot. Syst.*, vol. 22, no. 11, pp. 691–710, Nov. 2005. [Online]. Available: http://dx.doi.org/10.1002/rob.v22:11

[16] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4674–4683, 2017.

[17] Y. Cheng, W. Zhao, C. Liu, and M. Tomizuka, "Human motion prediction using adaptable neural networks," *ArXiv*, vol. abs/1810.00781, 2018.

[18] S. Kim, A. Shukla, and A. Billard, "Catching objects in flight," *IEEE Transactions on Robotics*, vol. 30, pp. 1049–1065, 2014.

[19] C. Chen, C. Yang, C. Zeng, N. Wang, and Z. Li, "Robot learning from multiple demonstrations with dynamic movement primitive," in *2017 2nd International Conference on Advanced Robotics and Mechatronics (ICARM)*, Aug 2017, pp. 523–528.

[20] M. Johnson, N. Aghasadeghi, and T. Bretl, "Inverse optimal control for deterministic continuous-time nonlinear systems," in *52nd IEEE Conference on Decision and Control*, Dec 2013, pp. 2906–2913.

[22] P. Gesel, M. Begum, and D. L. Roche, "Learning motion trajectories from phase space analysis of the demonstration," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7055–7061.

[21] S. M. Khansari-Zadeh and A. Billard, "Imitation learning of globally stable non-linear point-to-point robot motions using nonlinear programming," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2010, pp. 2676–2683.

[23] J. P. Vox and F. Wallhoff, "Evaluation of motion tracking methods for therapeutic assistance in everyday living environments," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Sep. 2016, pp. 96–101.

[24] B. Berret, E. Chiovetto, F. Nori, and T. Pozzo, "Evidence for composite cost functions in arm movement planning: An inverse optimal control approach," *PLOS Computational Biology*, vol. 7, no. 10, pp. 1–18, 10 2011. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1002183

[25] J. F. Lin, V. Bonnet, A. M. Panchea, N. Ramdani, G. Venture, and D. Kulić, "Human motion segmentation using cost weights recovered from inverse optimal control," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, Nov 2016, pp. 1107–1113.