

# A novel DMP formulation for global and frame independent spatial scaling in the task space

Leonidas Koutras and Zoe Doulgeri

**Abstract**—In this work we study the DMP spatial scaling in the Cartesian space. The DMP framework is claimed to have the ability to generalize learnt trajectories to new initial and goal positions, maintaining the desired kinematic pattern. However we show that the existing formulations present problems in trajectory spatial scaling when used in the Cartesian space for a wide variety of tasks and examine their cause. We then propose a novel formulation alleviating these problems. Trajectory generalization analysis, is performed by deriving the trajectory tracking dynamics. The proposed formulation is compared with the existing ones through simulations and experiments on a KUKA LWR 4+ robot.

## I. INTRODUCTION

In recent years robotic systems tend to leave their traditional confined and static workstations and are beginning to be integrated to unstructured human environments. In such environments, it is necessary for a robot to move in a more human-like manner, in order to be both safe for the humans around it and predictable. In that direction Imitation Learning and Programming by Demonstration (PbD) are gaining popularity [1]. An essential feature that any PbD framework must possess is the ability to generalize to new environments. Human environments are characterized by perpetual modifications and unpredictable alterations. Any learned movement must be able to be scaled both spatially and temporally, as well as be robust to perturbations.

Many PbD frameworks have been proposed in the literature. Such approaches involve optimization frameworks, probabilistic setups or dynamical systems. In [2] an optimization setup was proposed by minimizing simultaneously an accuracy and a jerk cost term. However as pointed out in [3] this framework fails to appropriately scale the trajectory spatially far from the demonstration workspace. Another optimization approach was proposed in [3], utilizing frame invariant descriptors [4], [5]. The approach is indeed successful in the desired trajectory generalization. However, as it is discussed in the same paper, this approach has large computation times for each control step.

Probabilistic approaches [6], [7], encode a large number of possible trajectory evolutions through multiple demonstrations. Then during execution the framework produces a trajectory according to some similarity measure with the demonstrated ones. These approaches are able to produce

concrete results close to the demonstrated trajectory data, but fail to appropriately generalize the movement far away from the demonstrated trajectories.

Dynamical systems used for PbD are the DS [8] and DMP [9]. DS use a Gaussian Mixture Model setup to encode the trajectory characteristics. Their main disadvantages are the requirement for multiple demonstrations, and again the inability to generalize to new environment setup, outside the demonstration space. DMP offer an intuitive training process, as they require one demonstration to learn a desired trajectory, as well as spatial and temporal scaling characteristics. Due to their formulation as dynamical systems they achieve adaptations to uncertainties and perturbations, by being augmented with appropriate coupling terms [10], [11], or with adaptive equations for their parameters [12]–[14].

Despite their favorable properties, the spatial scaling of the DMP formulation in [9] is performed separately for each coordinate and hence it is frame dependant. This is undesirable in some tasks that demand the preservation of the 3D path pattern. In an attempt to alleviate this problem an alternative Biologically Inspired DMP formulation was proposed in [10], which however loses its global scaling property, as pointed out in [11] and [15].

In this paper we are going to showcase the problems arising from both the original and the Biologically Inspired DMP and propose a novel DMP formulation possessing global scaling abilities. This formulation is inspired by the observation made in [9], that DMP perform different motion generalizations by selecting different coordinate systems. We will demonstrate its effectiveness in various types of tasks encountered in robotic applications with simulations and experiments and compare it with the existing formulations.

## II. DYNAMIC MOVEMENT PRIMITIVES

Dynamic Movement Primitives (DMP) are able to learn arbitrary trajectories in the Cartesian space by augmenting a single point attractor linear dynamical system by a non linear forcing term:

$$\begin{aligned} \tau \dot{\mathbf{z}} &= \alpha_z (\beta_z (\mathbf{g} - \mathbf{y}) - \mathbf{z}) + \text{diag}(\mathbf{g} - \mathbf{y}_0) \mathbf{F}(x) \\ \tau \dot{\mathbf{y}} &= \mathbf{z} \end{aligned} \quad (1)$$

where  $\mathbf{y}, \mathbf{z} \in \mathbb{R}^3$  are the position and the velocity scaled by  $\tau$ ;  $\mathbf{g}$  and  $\mathbf{y}_0$  are the goal and initial position,  $\alpha_z, \beta_z$  are positive gains of the linear part of the system, chosen  $\alpha_z = 4\beta_z$  for the system to be critically damped,  $\tau$  is the temporal scaling parameter and  $\text{diag}(\mathbf{g} - \mathbf{y}_0)$  is a diagonal matrix with its elements being the coordinates of  $\mathbf{g} - \mathbf{y}_0$ . Equation (1) is called the transformation system of the DMP.

Authors are with the Automation & Robotics Lab, Dept. of Electrical & Computer Engineering, Aristotle University of Thessaloniki, Greece. {kleonidas@ece, doulgeri@eng}.auth.gr

The research leading to these results has received funding by the EU Horizon 2020 Research and Innovation Programme under grant agreement No 820767, project CoLLaboratE.

The desired trajectory is encoded with the non linear function  $F(x)$ , called forcing term:

$$\mathbf{F}(x) = \frac{\sum_{i=1}^N \mathbf{w}_i \Psi_i(x)}{\sum_{i=1}^N \Psi_i(x)} x \quad (2)$$

where  $\Psi_i(x) = \exp(-h_i(c_i - x)^2)$  are  $N$  Gaussian kernel functions, with  $h_i, c_i$  their inverse widths and centers, and  $\mathbf{w}_i \in \mathbb{R}^3$  their weights to be learned in order to encode the desired trajectory. The weights are usually trained using Locally Weighted Regression. To avoid explicit time dependency the canonical system of the phase variable  $x$  is used:

$$\tau \dot{x} = -\alpha_x x \quad (3)$$

with its initial value  $x_0 = 1$  and  $\alpha_x$  a positive gain.

DMP where used for orientation trajectories using unit quaternions in [11] and extended in [16]. However in [17] we showed that those orientation formulations are problematic and proposed a different DMP orientation formulation:

$$\begin{aligned} \tau \dot{z} &= -\alpha_z (\beta_z e_Q + z) + \text{diag}(\mathbf{Q}_g * \bar{\mathbf{Q}}_0) \mathbf{F}(x) \\ \tau \dot{e}_Q &= z \end{aligned} \quad (4)$$

with  $e_Q, z$  we denote the orientation error and its scaled by  $\tau$  velocity,  $\mathbf{Q} = [\eta \ \epsilon] = \exp(-\frac{1}{2} e_Q) * \mathbf{Q}_g$  the orientation as a unit quaternion and  $\mathbf{Q}_0, \mathbf{Q}_g$  the initial and goal orientation.

The Biologically Inspired DMP framework proposed in [10], [18], [19] has the same structure with a modified transformation system:

$$\begin{aligned} \tau \dot{z} &= \alpha_z (\beta_z (\mathbf{g} - \mathbf{y}) - z) + \alpha_z \beta_z (-\mathbf{g} - \mathbf{y}_0) x + \mathbf{F}(x) \\ \tau \dot{\mathbf{y}} &= z \end{aligned} \quad (5)$$

which can be generalized to the orientation space using the formulation of [17]:

$$\begin{aligned} \tau \dot{z} &= -\alpha_z (\beta_z e_Q + z) + \alpha_z \beta_z (-2 \log(\mathbf{Q}_g * \bar{\mathbf{Q}}_0)) x + \mathbf{F}(x) \\ \tau \dot{e}_Q &= z \end{aligned} \quad (6)$$

The original DMP framework (1), (4) scales the forcing term by multiplying it with the matrix  $\text{diag}(\mathbf{g} - \mathbf{y}_0)$ . Conversely, the Biologically Inspired framework scales the forcing term by using the additive component  $-(\mathbf{g} - \mathbf{y}_0)x$ .

### III. DMP SPATIAL SCALING

To find out how both DMP formulations scale the trajectory, we will derive the trajectory tracking dynamics for each formulation. We will perform our analysis to the position DMP, however the same results apply to the orientation formulation. To train a DMP, the trajectory data  $\mathbf{y}_d(t), \dot{\mathbf{y}}_d(t), \ddot{\mathbf{y}}_d(t)$  are collected for  $t \in [0 \dots T_d]$ , with  $\mathbf{y}_{0,d} = \mathbf{y}_d(0)$  and  $\mathbf{g}_d = \mathbf{y}_d(T_d)$ . Then the values of  $\alpha_z, \beta_z$  and  $\tau$  are chosen, with the most common choices for the temporal scaling parameter being  $\tau = 1$  and  $\tau = T_d$ . Utilizing the trajectory the desired forcing term is calculated. The resulting trained forcing term will be equal to the desired plus an error term  $\epsilon_t$  for each time step. Thus, for the original DMP:

$$\mathbf{F} = (\text{diag}(\mathbf{g}_d - \mathbf{y}_{0,d}))^{-1} (\tau^2 \ddot{\mathbf{y}}_d + \alpha_z \tau \dot{\mathbf{y}}_d + \alpha_z \beta_z (\mathbf{y}_d - \mathbf{g}_d)) + \epsilon_t \quad (7)$$

and for the Biologically Inspired:

$$\mathbf{F} = \frac{1}{\alpha_z \beta_z} (\tau^2 \ddot{\mathbf{y}}_d + \alpha_z \tau \dot{\mathbf{y}}_d + \alpha_z \beta_z (\mathbf{y}_d - \mathbf{g}_d)) + (\mathbf{g}_d - \mathbf{y}_{0,d}) x + \epsilon_t \quad (8)$$

where indexes  $(t), (x)$  were omitted for better presentation.

Substituting (7) to (1) we get the trajectory tracking dynamics of the original DMP:

$$\begin{aligned} \tau^2 (\ddot{\mathbf{y}} - \mathbf{S}_g \ddot{\mathbf{y}}_d) + \alpha_z \tau (\dot{\mathbf{y}} - \mathbf{S}_g \dot{\mathbf{y}}_d) + \alpha_z \beta_z (\mathbf{y} - \mathbf{S}_g \mathbf{y}_d) = \\ \alpha_z \beta_z (\mathbf{y}_0 - \mathbf{S}_g \mathbf{y}_{0,d}) + \text{diag}(\mathbf{g} - \mathbf{y}_0) \epsilon_t \end{aligned} \quad (9)$$

where matrix  $\mathbf{S}_g$  is defined as:

$$\mathbf{S}_g = \text{diag}(\mathbf{g} - \mathbf{y}_0) (\text{diag}(\mathbf{g}_d - \mathbf{y}_{0,d}))^{-1} \quad (10)$$

and we also substituted  $\mathbf{g} = \mathbf{S}_g (\mathbf{g}_d - \mathbf{y}_{0,d}) + \mathbf{y}_0$ .

From (9) it is clear that the trajectory is scaled according to the matrix  $\mathbf{S}_g$ . The term  $\alpha_z \beta_z (\mathbf{y}_0 - \mathbf{S}_g \mathbf{y}_{0,d})$  serves the purpose of transferring the trajectory accordingly when beginning from a different initial position than the demonstrated. The training error is considered negligible, therefore we assume that the DMP generated trajectory follows the demonstrated one, scaled by  $\mathbf{S}_g$  and translated by  $\alpha_z \beta_z (\mathbf{y}_0 - \mathbf{S}_g \mathbf{y}_{0,d})$ . Notice that the scaling matrix is diagonal, which implies that scaling is performed separately for each coordinate and hence it is frame dependant. This behavior is in general problematic with a few exceptions. In fact, one cannot encode or execute trajectories when one coordinate has the same initial and goal value. This case often arises in tasks involving planar paths since then, the same coordinate value is involved along the axis normal to the plane. Moreover, when the initial and goal value are close to each other along one coordinate, the scaling is amplified and may demand positions and velocities that exceed the robot's limits or go through rigid working surfaces when e.g. a Cartesian task is executed over a workbench.

Regarding the trajectory tracking dynamics of the Biologically Inspired DMP we can substitute (8) to (5) to get :

$$\begin{aligned} \tau^2 (\ddot{\mathbf{y}} - \ddot{\mathbf{y}}_d) + \alpha_z \tau (\dot{\mathbf{y}} - \dot{\mathbf{y}}_d) + \alpha_z \beta_z (\mathbf{y} - \mathbf{y}_d) = \\ \alpha_z \beta_z (\mathbf{g} - \mathbf{g}_d) - \alpha_z \beta_z x (\mathbf{g} - \mathbf{y}_0 - \mathbf{g}_d + \mathbf{y}_{0,d}) + \alpha_z \beta_z \epsilon_t \end{aligned} \quad (11)$$

Notice that the term  $\alpha_z \beta_z (\mathbf{g} - \mathbf{g}_d)$  has the effect of the appropriate trajectory displacement. From (11), it is also clear that the scaling is performed by the exponentially decaying term  $-\alpha_z \beta_z x (\mathbf{g} - \mathbf{y}_0 - \mathbf{g}_d + \mathbf{y}_{0,d})$  which affects the scaled trajectory additively. Thus problems of the original formulation are alleviated and for small initial position or goal displacements, the Biologically Inspired framework offers good results. However, when larger displacements are required, the learned trajectory pattern is ruined since the scaling term dominates the forcing term  $\mathbf{F}(x)$ . As the scaling term is dependant on the gains  $\alpha_z$  and  $\beta_z$ , large values yield large initial trajectory displacements and velocities (where  $x = 1$ ). Small gain values cannot remedy this problem as they deteriorate trajectory tracking (11).

We claim that the spatial scaling problems present in the current formulations stem from the frame coordinate dependence of the scaling. Our objective is a DMP formulation achieving global scaling, by preserving the motion pattern and its desired characteristics. We will consider free motion tasks in unconstrained spaces, planar tasks and tasks with orientation constraints. Planar tasks refer to demonstrated planar paths that should be generalizable to different planes. A lot of robotic tasks like spraying on tilted surfaces require such type of generalization. In such tasks scaling with the existing formulations, may generate non-planar paths. Furthermore the user cannot explicitly define the desired plane of task execution beforehand. When using the original DMP formulation in particular, many planar tasks cannot even be encoded. Tasks with orientation constraints refer to a great number of industrial (e.g. pick and place, assembly etc) and domestic (e.g. cutlery welding, liquid pouring etc) tasks. Such tasks usually have targets that are constrained with regards to the orientation that is associated with the direction of the workbench normal vector. Such constraints cannot be handled by existing formulations and spatial scaling may result to trajectories that may even penetrate the workbench plane.

#### IV. PROPOSED DMP

To alleviate the original DMP framework's problems, and maintain its global scaling properties we propose the following DMP structure:

$$\begin{aligned} \tau \dot{\mathbf{z}} &= \alpha_z(\beta_z(\mathbf{g} - \mathbf{y}) - \mathbf{z}) + s_g \mathbf{R}_g \mathbf{F}(x) \\ \tau \dot{\mathbf{y}} &= \mathbf{z} \end{aligned} \quad (12)$$

where matrix  $\mathbf{R}_g$  is a rotation matrix that rotates a unit vector in the direction of  $\mathbf{g}_d - \mathbf{y}_{0,d}$  to a unit vector in the direction of  $\mathbf{g} - \mathbf{y}_0$ , and  $s_g$  is a magnitude scaling constant, calculated as:

$$s_g = \frac{\|\mathbf{g} - \mathbf{y}_0\|}{\|\mathbf{g}_d - \mathbf{y}_{0,d}\|} \quad (13)$$

The forcing term  $\mathbf{F}(x)$  is given by (2) and the desired values of the forcing term are calculated from the demonstrated trajectory data as:

$$\mathbf{F}_d(t) = \tau^2 \ddot{\mathbf{y}}_d(t) + \tau \alpha_z \dot{\mathbf{y}}_d(t) + \alpha_z \beta_z (\mathbf{y}_d(t) - \mathbf{g}_d) \quad (14)$$

The proposed formulation alleviates the problems of the original DMP as it does not scale each trajectory coordinate separately, but performs a three-dimensional scaling, involving both a rotation and a magnitude scaling. To study the spatial scaling of the proposed DMP we will again derive the trajectory tracking dynamics, by substituting  $\mathbf{F}(x) = \mathbf{F}_d(t) + \epsilon_t$  to (12), using (14), assuming training error  $\epsilon_t$ :

$$\begin{aligned} \tau^2 (\ddot{\mathbf{y}} - s_g \mathbf{R}_g \ddot{\mathbf{y}}_d) + \alpha_z \tau (\dot{\mathbf{y}} - s_g \mathbf{R}_g \dot{\mathbf{y}}_d) + \\ \alpha_z \beta_z (\mathbf{y} - s_g \mathbf{R}_g \mathbf{y}_d) = \alpha_z \beta_z (\mathbf{y}_0 - s_g \mathbf{R}_g \mathbf{y}_{0,d}) + s_g \mathbf{R}_g \epsilon_t \end{aligned} \quad (15)$$

From (15) it is clear that the DMP generated trajectory is scaled by  $s_g \mathbf{R}_g$  and translated by  $\alpha_z \beta_z (\mathbf{y}_0 - s_g \mathbf{R}_g \mathbf{y}_{0,d})$  to accommodate for different initial positions. Therefore, the trajectory will be accordingly rotated, by the matrix  $\mathbf{R}_g$  and

then multiplied by the magnitude scaling constant  $s_g$ . Due to the magnitude scaling being performed using the norms of  $\mathbf{g} - \mathbf{y}_0$  and  $\mathbf{g}_d - \mathbf{y}_{0,d}$ , the only case where the proposed formulation is not able to encode, is when the initial and the goal position are identical, instead of having just one equal coordinate as in the original DMP. However this property is generally not true for discrete motions.

For the orientation space, the proposed formulation can be written as:

$$\begin{aligned} \tau \dot{\mathbf{z}} &= -\alpha_z(\beta_z \mathbf{e}_Q + \mathbf{z}) + s_g \mathbf{R}_g \mathbf{F}(x) \\ \tau \dot{\mathbf{e}}_Q &= \mathbf{z} \end{aligned} \quad (16)$$

where similarly to the position  $\mathbf{R}_g$  is the rotation matrix that rotates a unit vector in the direction of  $2 \log(\mathbf{Q}_{g,d} * \overline{\mathbf{Q}}_{0,d})$  to a unit vector in the direction of  $2 \log(\mathbf{Q}_g * \overline{\mathbf{Q}}_0)$ , and  $s_g$  is a magnitude scaling constant, calculated as:

$$s_g = \frac{\|\log(\mathbf{Q}_g * \overline{\mathbf{Q}}_0)\|}{\|\log(\mathbf{Q}_{g,d} * \overline{\mathbf{Q}}_{0,d})\|} \quad (17)$$

##### A. Selection of Rotation Matrix

The rotation matrix in (12) and (16) is a rotation matrix that rotates the demonstrated trajectory during the execution. For our analysis we denote  $\mathbf{n}_d = \frac{\mathbf{g}_d - \mathbf{y}_{0,d}}{\|\mathbf{g}_d - \mathbf{y}_{0,d}\|}$  and  $\mathbf{n} = \frac{\mathbf{g} - \mathbf{y}_0}{\|\mathbf{g} - \mathbf{y}_0\|}$ . The same analysis yields for the orientation DMP of (16). Therefore  $\mathbf{R}_g$  is computed to rotate  $\mathbf{n}_d$  to  $\mathbf{n}$ . The rotation matrix is not however unique by construct and its selection is highly dependant on the task to be executed. A rotation around any equivalent axis that lies on the plane normal to the plane defined by  $\mathbf{n}_d$  and  $\mathbf{n}$ , passing from the line dissecting the angle between them when put in the same origin can be used. The dissecting plane's normal vector can be computed as:

$$\hat{\mathbf{p}} = \frac{\mathbf{n} - \mathbf{n}_d}{\|\mathbf{n} - \mathbf{n}_d\|} \quad (18)$$

Thus, a set of candidate rotation axes are defined as follows:

$$S_k = \{\mathbf{k} \in \mathbb{R}^3 \mid \mathbf{k}^T \hat{\mathbf{p}} = 0, \|\mathbf{k}\| = 1\} \quad (19)$$

Concerning the selection of the rotation axis, we study the three groups of tasks discussed in Section III. For the free category of tasks, the trivial choice  $\mathbf{k} = \mathbf{n}_d \times \mathbf{n}$  suffices. This selection gives the minimal angle rotation between the two vectors, namely  $\theta = \cos^{-1}(\mathbf{n}_d^T \mathbf{n})$ . Then the rotation matrix can be computed by the Rodrigues' rotation formula [20]:

$$\mathbf{R}_g = \mathbf{I}_3 + \mathbf{S}(\mathbf{k}) \sin \theta + \mathbf{S}^2(\mathbf{k})(1 - \cos \theta) \quad (20)$$

In the case of the planar task, we consider the normal vector of the task plane during the demonstration  $\hat{\mathbf{p}}_d$  and during the execution  $\hat{\mathbf{p}}_e$ . Then desired rotation matrix has to rotate both  $\mathbf{n}_d$  to  $\mathbf{n}$  and  $\hat{\mathbf{p}}_d$  to  $\hat{\mathbf{p}}_e$ . As  $\mathbf{n}_d$  and  $\hat{\mathbf{p}}_d$ , as well as  $\mathbf{n}$  and  $\hat{\mathbf{p}}_e$  are orthogonal, the resulting rotation matrix is indeed unique and computed as:

$$\mathbf{R}_g = [\mathbf{n} \quad \hat{\mathbf{p}}_e \quad \mathbf{n} \times \hat{\mathbf{p}}_e] [\mathbf{n}_d \quad \hat{\mathbf{p}}_d \quad \mathbf{n}_d \times \hat{\mathbf{p}}_d]^T \quad (21)$$

For the case of tasks executed over a workbench the rotated trajectory is expected to be least modified in the direction of the normal vector to the workbench plane,

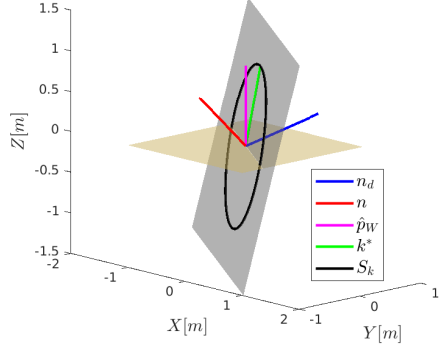


Fig. 1: Rotation axis for workbench tasks

denoted  $\hat{p}_W$ . This requirement prevents the scaled trajectory from penetrating the workbench and violating orientation constraints. Thus, the rotation axis is chosen as the projection of the workbench plane normal vector to the dissecting plane:

$$\mathbf{k} = \frac{(\mathbf{I}_3 - \hat{p}\hat{p}^T)\hat{p}_W}{\|(\mathbf{I}_3 - \hat{p}\hat{p}^T)\hat{p}_W\|} \quad (22)$$

In Figure 1 an example of the rotation axis computation is depicted. The yellow plane is the workbench plane, and the gray the dissecting plane where  $S_k$  lies. To calculate the rotation angle we use:

$$\theta = \cos^{-1} \left( \frac{\mathbf{n}_d^T (\mathbf{I}_3 - \mathbf{k}\mathbf{k}^T) \mathbf{n}}{\|(\mathbf{I}_3 - \mathbf{k}\mathbf{k}^T) \mathbf{n}_d\| \|(\mathbf{I}_3 - \mathbf{k}\mathbf{k}^T) \mathbf{n}\|} \right) \quad (23)$$

Matrix  $R_g$  is then computed from (20).

## V. SIMULATION RESULTS

To demonstrate the proposed DMP formulation's spatial scaling capabilities we simulated (3), (12), (16) and compared the results with the original DMP formulation (3), (1), (4) and the Biologically Inspired DMP formulation (3), (5), (6). As desired trajectory data we used trajectories demonstrated by a user to a KUKA LWR 4+ robotic manipulator in gravity compensation. The desired trajectory begun from initial position and orientation  $\mathbf{y}_{0,d} = [-0.32 \ 0.54 \ 0.18]^T$ ,  $\mathbf{Q}_{0,d} = [0.37 \ 0.6263 \ -0.67 \ -0.14]^T$  and reached  $\mathbf{g}_d = [0.11 \ 0.75 \ 0.4]^T$ ,  $\mathbf{Q}_g = [0.49 \ -0.73 \ 0.39 \ -0.27]^T$  in  $T_d = 7.7s$ . All DMP formulations were trained using the same parameters  $\alpha_z = 80$ ,  $\beta_z = 20$ ,  $\alpha_x = 4.61$ ,  $\tau_d = T_d$ ,  $N = 100$ . Then all formulations were executed from new initial position and orientation  $\mathbf{y}_0 = [-0.1 \ 0 \ 0.05]^T$ ,  $\mathbf{Q}_0 = [0.74 \ 0.51 \ -0.42 \ 0.13]^T$  to new goal  $\mathbf{g} = [0.4 \ 0.05 \ 0.7]^T$ ,  $\mathbf{Q}_g = [0.72 \ -0.39 \ 0.48 \ -0.31]^T$ . For the selection of the rotation axis, we used (22), assuming a task executed over a workbench. The workbench plane's normal vector was  $\hat{p}_W = [0 \ 0 \ 1]^T$ . Thus the computed rotation axis and angle were  $\mathbf{k}_P = [0.33 \ 0.58 \ 0.74]^T$ ,  $\theta_P = -0.99rad$  for position and  $\mathbf{k}_Q = [0.3 \ 0.14 \ 0.94]$ ,  $\theta_Q = 0.25rad$  for orientation. The trajectory was scaled at  $s_{g,P} = 1.57$  and  $s_{g,Q} = 1.58$ . The results are shown in Figures 3 - 5.

In Figure 2 the proposed formulation is depicted, where the trajectory has been appropriately rotated and scaled,

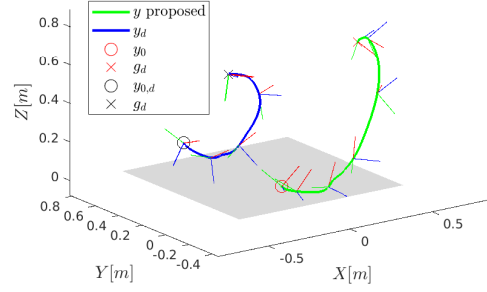


Fig. 2: Demonstrated trajectory and proposed DMP trajectory

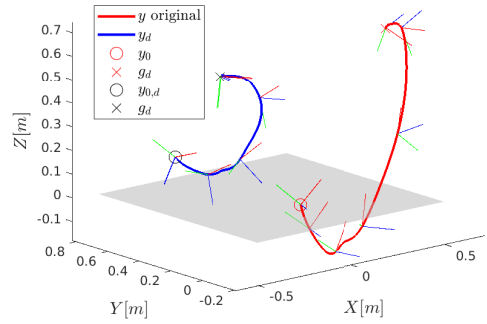


Fig. 3: Demonstrated trajectory and original DMP trajectory

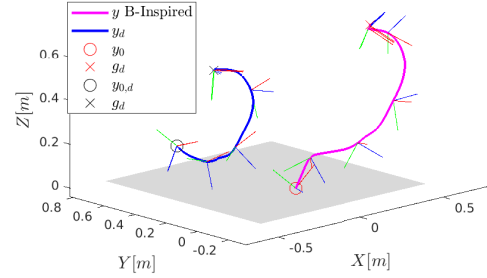


Fig. 4: Demonstrated trajectory and Biologically Inspired DMP trajectory

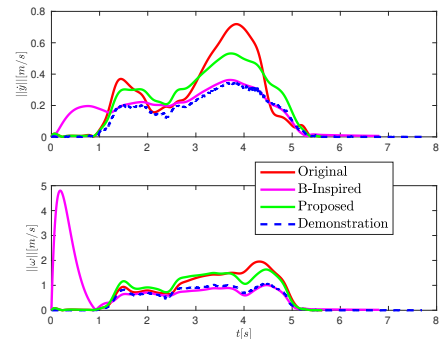


Fig. 5: Norms of velocity and angular velocity of demonstration and all DMP executions

maintaining the learnt pattern. In Figure 3 the original DMP formulation is shown. The trajectory is scaled using matrix (10) yielding  $\mathbf{S}_{g,P} = \text{diag}[1.19 \ 0.23 \ 2.88]^T$  and  $\mathbf{S}_{g,Q} = \text{diag}[2.41 \ 1.41 \ 1.19]^T$ . As the figure depicts, the scaling performed by the original formulation does not maintain the trajectory shape in three-dimensional space and leads to large scalings. Thus the robot's trajectory exceeds its workspace and penetrates the workbench plane. The Biologically Inspired DMP formulation is depicted in Figure 4. As explained in Section III this formulation generalizes the motion by an additive term. The effect of this term is clear at the beginning of the trajectory, where effect of the learnt pattern is minimal, and the generalization term is dominating. Thus the demonstrated motion profile is again not preserved.

Figure 5 depicts the velocity and angular velocity norms of all formulations. The original DMP scales the velocity with the same matrix as the trajectory, leading to large velocity values. Due to the Biologically Inspired formulation's additive term, which is dependant of the phase variable there is a velocity peak in the beginning of motion, until the effect of the additive term diminishes. This peak however leads to non human-like behavior and can exceed the robot's velocity capabilities. The proposed formulation's velocity profile is higher than the demonstrated, in order to reach the new goal, but is kept to low enough levels to be able to be executed by a robotic system maintaining human-like characteristics.

## VI. EXPERIMENTAL RESULTS

For our experiments we demonstrated trajectories to a 7 dof KUKA LWR 4+ robotic manipulator, in gravity compensation, with control frequency  $500Hz$ . We then used the demonstrated data to train all DMP formulations and executed the systems with new initial and goal positions. During the execution the robot was under position control where the DMP generated velocity was fed to a Closed Loop Inverse Kinematics scheme [21], and mapped to the joint velocity with the Jacobian pseudoinverse. The resulting joint velocity was integrated to produce the commanded joint position.

We performed two experiments. In the first one, the robot was taught a planar task, and then was required to execute it on a different plane. In the second experiment a liquid pouring task was demonstrated, and then the robot was required to execute it with new initial and goal positions. A video of both tasks can be found in <https://youtu.be/VgjZB3wVQ1c>.

In both experiments all DMP formulations were trained using the data collected from the demonstrations. The parameters used for all DMP were  $\alpha_z = 40$ ,  $\beta_z = 10$ ,  $N = 80$ ,  $\alpha_x = 4.61$ ,  $\tau = \tau_d = T_d$ .

For the first experiment, the robot was taught to write the letter "R", on the plane  $z = 0$  and then was tasked to execute the learnt motion pattern on the plane  $y = 0.67$ . The demonstrated trajectory begun from initial position  $\mathbf{y}_{0,d} = [0 \ 0.59 \ 0]^T$  to goal  $\mathbf{g}_d = [0.1 \ 0.6 \ 0]^T$  in  $T_d = 9.55s$  and was executed from  $\mathbf{y}_0 = [-0.02 \ 0.67 \ 0.21]^T$  to  $\mathbf{g} = [0.13 \ 0.67 \ 0.21]^T$ . Thus the resulting trajectory is scaled by  $s_g = 1.4$  and rotated with a rotation matrix for a planar task,

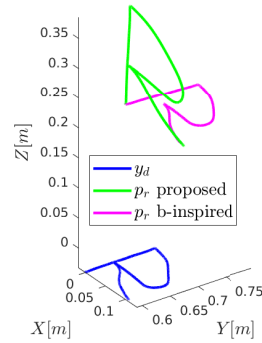


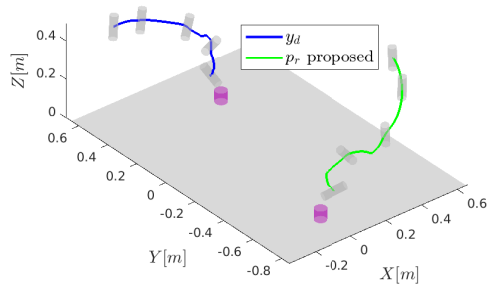
Fig. 6: Demonstration trajectory of planar task and trajectory of Biologically Inspired and proposed DMP

calculated from equation (21). The manipulator's orientation in both experiments is kept fixed, orthogonal to the task plane. The result of the first experiment is shown in Figure 6, where the trajectory of the robot, denoted  $\mathbf{p}_r$  is shown for the proposed and Biologically Inspired DMP formulations. The original DMP formulation fails to encode this task, as the initial and the goal position have the same  $Z$  coordinate in training and  $Y$  coordinate in execution. The proposed formulation is indeed successful in reproducing the pattern on the new plane. The Biologically Inspired formulation however fails to rotate the trajectory, and corrupts the pattern. Notice the unconnected line in the middle of the letter in the Biologically Inspired execution.

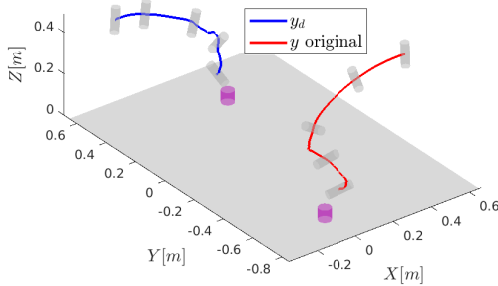
For the second experiment a liquid pouring task was demonstrated from initial position and orientation  $\mathbf{y}_{0,d} = [-0.1 \ 0.66 \ 0.38]^T$ ,  $\mathbf{Q}_{0,d} = [0.06 \ 0.01 \ -0.71 \ -0.7]^T$  to  $\mathbf{g}_d = [0.26 \ 0.43 \ 0.1]^T$ ,  $\mathbf{Q}_{g,d} = [0.1 \ -0.73 \ 0.13 \ -0.66]^T$  in  $T_d = 9.63s$ . During the execution the robot was initialized at  $\mathbf{y}_0 = [0.64 \ -0.36 \ 0.43]^T$ ,  $\mathbf{Q}_0 = [0.62 \ 0.61 \ 0.34 \ 0.36]^T$  and the goal was set at  $\mathbf{g} = [0.05 \ -0.67 \ 0.12]^T$ ,  $\mathbf{Q}_g = [0.45 \ 0.45 \ -0.6 \ 0.49]^T$ . The task can be characterized as a task performed on a workbench, thus the calculated rotation axes, angles and scalings for the proposed formulation are  $\mathbf{k}_P = [0.09 \ 0 \ 0.99]^T$ ,  $\theta_P = -2.08rad$ ,  $s_{g,P} = 1.42$ ,  $\mathbf{k}_Q = [0 \ 0.06 \ 0.99]^T$ ,  $\theta_Q = -1.91rad$  and  $s_{g,Q} = 0.86$ . The results are presented in Figures 7 - 9. In Figure 7 the proposed formulation has successfully generalized the trajectory, staying inside the robot's workspace and pouring the liquid inside the cup successfully. Conversely, in Figures 8 and 9 the trajectories produced by the original and the Biologically inspired DMP formulations are shown to fail to appropriately generalize the trajectory. In both cases the resulting motion would result in spilling the liquid. Furthermore, both trajectories are not able to be executed by the robot, as in the case of the original DMP the robot's workspace is violated and in the case of the Biologically Inspired DMP the robot's velocity limits are exceeded.

## VII. CONCLUSIONS

In this paper we proposed a novel DMP for encoding a trajectory in the Cartesian space, which produces reasonably



**Fig. 7:** Demonstrated trajectory of liquid pouring task and trajectory of proposed DMP

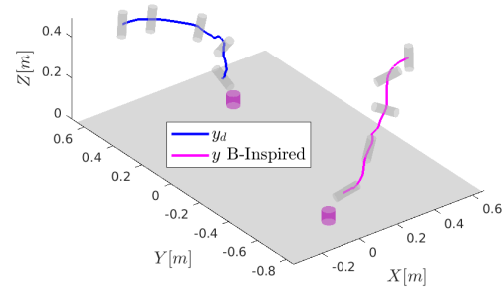


**Fig. 8:** Demonstrated trajectory of liquid pouring task and trajectory of original DMP

scaled trajectories globally, maintaining the learnt characteristics. We showed that spatial scaling of the original DMP formulation is problematic in a variety of tasks owing to the frame dependency of the trajectory generalization method while the subsequent Biologically Inspired formulation performs satisfactorily only locally. Our approach remedies this problem by appropriately rotating the executed trajectory and scaling its magnitude to alleviate frame dependency. The proposed formulation is shown to achieve the desired behavior through theoretical analysis, and validated through simulations and experiments.

## REFERENCES

- [1] A. G. Billard, S. Calinon, and D. Rüdiger, *Springer Handbook of Robotics*. Springer International Publishing, 2016, ch. Learning from Humans.
- [2] Y. Meirovitch, D. Bennequin, and T. Flash, “Geometrical invariance and smoothness maximization for task-space movement generation,” *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 837–853, Aug 2016.
- [3] M. Vochten, T. D. Laet, and J. D. Schutter, “Generalizing demonstrated motion trajectories using coordinate-free shape descriptors,” *Robotics and Autonomous Systems*, vol. 122, p. 103291, 2019.
- [4] J. De Schutter, “Invariant Description of Rigid Body Motion Trajectories,” *Journal of Mechanisms and Robotics*, vol. 2, no. 1, 11 2009.
- [5] R. Soloperto, M. Saveriano, and D. Lee, “A bidirectional invariant representation of motion for gesture recognition and reproduction,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 6146–6152.



**Fig. 9:** Demonstrated trajectory of liquid pouring and trajectory of Biologically Inspired DMP

- [6] A. Paraschos, C. Daniel, J. Peters, and G. Neumann, “Probabilistic movement primitives,” in *Advances in Neural Information Processing Systems*, 2013, p. 2616–2624.
- [7] S. Kim, R. Haschke, and H. Ritter, “Gaussian mixture model for 3-dof orientations,” *Robotics and Autonomous Systems*, vol. 87, pp. 28 – 37, 2017.
- [8] E. Gribovskaya and A. Billard, “Learning nonlinear multi-variate motion dynamics for real-time position and orientation control of robotic manipulators,” in *2009 9th IEEE-RAS International Conference on Humanoid Robots*, Dec 2009, pp. 472–477.
- [9] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, “Dynamical movement primitives: Learning attractor models for motor behaviors,” *Neural Computation*, vol. 25, no. 2, pp. 328–373, Feb 2013.
- [10] H. Hoffmann, P. Pastor, D. Park, and S. Schaal, “Biologically-inspired dynamical systems for movement generation: Automatic real-time goal adaptation and obstacle avoidance,” in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 2587–2592.
- [11] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, “Online movement adaptation based on previous sensor experiences,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2011, pp. 365–371.
- [12] L. Koutras and Z. Doulgeri, “Dynamic movement primitives for moving goals with temporal scaling adaptation,” in *2020 IEEE International Conference on Robotics and Automation*, 2020.
- [13] J. Kober, K. Mülling, O. Krömer, C. H. Lampert, B. Schölkopf, and J. Peters, “Movement templates for learning of hitting and batting,” in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 853–858.
- [14] M. Karlsson, A. Robertsson, and R. Johansson, “Convergence of dynamical movement primitives with temporal coupling,” in *2018 IEEE European Control Conference (ECC)*. IEEE, 6 2018.
- [15] P. Pastor, M. Kalakrishnan, S. Chitta, E. Theodorou, and S. Schaal, “Skill learning and task outcome prediction for manipulation,” in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 3828–3834.
- [16] A. Ude, B. Nemeč, T. Petrić, and J. Morimoto, “Orientation in cartesian space dynamic movement primitives,” in *2014 IEEE International Conference on Robotics and Automation*, May 2014, pp. 2997–3004.
- [17] L. Koutras and Z. Doulgeri, “A correct formulation for the orientation dynamic movement primitives for robot control in the cartesian space,” in *Proceedings of the Conference on Robot Learning*, Oct–Nov 2019.
- [18] D. Park, H. Hoffmann, P. Pastor, and S. Schaal, “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields,” in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*, Dec 2008, pp. 91–98.
- [19] 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*, May 2009, pp. 763–768.
- [20] E. Piña, “Rotations with Rodrigues’ vector,” *European Journal of Physics*, vol. 32, no. 5, pp. 1171–1178, Sep 2011.
- [21] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer-Verlag, 2007, ch. 11.