IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 December 2023, accepted 26 December 2023, date of publication 1 January 2024,
date of current version 18 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3349093

RESEARCH ARTICLE

A New DMP Scaling Method for Robot Learning by Demonstration and Application to the Agricultural Domain

C. LAURETTI, (Member, IEEE), C. TAMANTINI, (Member, IEEE), AND L. ZOLLO, (Senior Member, IEEE)
Unit of Advanced Robotics and Human-Centred Technologies (CREO Lab), Università Campus Bio-Medico di Roma, 00128 Rome, Italy

Corresponding author: C. Lauretti (c.lauretti@unicampus.it)

This work was supported by the Italian Ministry of Education, Universities and Research (MIUR) through the Project Future Artificial Intelligence Research (FAIR) under Grant CUP: C53C22000800006. C. Lauretti is funded by Piano Operativo Nazionale (PON) "Ricerca e Innovazione" 2014–2020.

ABSTRACT Dynamic Motion Primitives (DMPs) only address the generalization problem for target positions that are close to the demonstration position and in order to enhance the generalization capability, by making the learned movements executable in the entire reachable robot workspace, multiple demonstrations are needed, resulting in an increase in the time required to teach new tasks to the robot. This work aims to propose a novel approach to scale the DMP parameters through two demonstrations in order to enhance the DMP's generalization capability in the robot reachable workspace while guaranteeing a fast and easy learning phase. The proposed method to scale the DMP parameters relies on a linear interpolation performed on the DMP parameters extracted by two demonstrations and is applied to the agricultural field, where the adoption of DMPs can provide a promising solution to meet the demand for a wide range of tasks in an ever-changing and mutable agricultural environment. Four agricultural activities, namely digging, seeding, irrigation, and harvesting, have been learned by the robot using DMPs. The experimental validation was carried out on the Tiago robot and the proposed approach, based on two demonstrations, was compared to two literature methods based on a single demonstration and multiple demonstrations in terms of accuracy of the motion reconstruction, success rate of task execution and speed of the learning process. The obtained results demonstrated that the proposed approach, based on two demonstrations, guarantees a successful execution of the four tasks without exceeding the robot reachable workspace and with acceptable accuracy (mean success rate of task execution is about 95.6%) and a fast training phase (about 70 minutes less than the database approach that is built on multiple demonstrations).

INDEX TERMS Learning by demonstration, teaching by demonstration, robot learning, dynamic motion primitives, motion planning, agricultural robotics.

I. INTRODUCTION

The introduction of robotics in agriculture has a great potential to enhance yield productivity, reduce labour-intensive work, and create a safer working environment for farmers who have to typically face occupational hazards associated with exposure to pesticides (including ocular, dermal, and inhalation toxicity risks), as well as work-related disorders due to biomechanical overloading [1].

Robots can be used to automate repetitive tasks, such as seeding, harvesting, weeding or pesticide spraying, and, equipped with advanced sensing technologies, can collect real-time data about crop health, soil conditions, and environmental factors, enabling farmers to make data-driven decisions for optimized resource management.

The agricultural environment poses unique challenges for robotics, particularly when it comes to determining optimal robot trajectories for efficient and safe task execution. Agricultural operations involve intricate and ever-changing settings, with factors like terrain, vegetation, and crop growth patterns constantly fluctuating. As a result, planning

The associate editor coordinating the review of this manuscript and approving it for publication was Aysegul Ucar.

robot movement in such environments demands easily adaptable strategies that can effectively respond to these temporal variations and to the need for learning new tasks. Additionally, ensuring a high level of safety is crucial in the interaction between farmers and robots to ensure the successful completion of agricultural operations.

Traditional approaches to motion planning in robotics often require expert programmers to manually code rules and algorithms for accomplishing specific new tasks. Among these, conventional techniques adopted for agricultural robot motion planning, rely on point-to-point motion planning [2], [3]. Alternatively, other approaches include i) grid-based [4] or interval-based [5] search methods, which identify optimal obstacle-free paths for both the manipulator and mobile base, ii) reward-based algorithms that involve the robot attempting different paths and receiving positive or negative rewards based on its success or failure [6], [7] iii) artificial potential fields algorithms, which generate attractive or repulsive paths for the manipulator joints and mobile base [8], [9]) and iv) sampling-based algorithms that find optimal paths using roadmaps [10] or probabilistic methods [11], [12].

All of these approaches are time-consuming and complex, and strongly rely on the expertise of skilled individuals. Consequently, to meet the demand for a wide range of tasks in ever-changing scenarios, an intuitive motion planning interface that allows non-expert users, such as farmers, to plan and customize tasks, continuously, is essential for agriculture. Moreover, ensuring the predictability of robot behaviour is crucial for promoting safe human-robot cooperation. Indeed, it is well-known that humans perceive and interpret human-like robot motion more easily, resulting in a sense of safety during interactions [13]. When robots move in a manner resembling humans, farmers can anticipate their motion and adjust their own activities to avoid potential injuries.

To address these challenges, demonstration-based motion planning methods offer a promising solution [14].

Teaching the robot motion by human demonstration is a flexible framework that reduces the complexity of programming robot tasks and allows end-users to operate the robot in a natural and easy way without the need for explicit coding [15], [16]. Indeed, this approach empowers farmers to intuitively teach robots a new task by simply showing them how to perform it, facilitating human-robot collaboration and reducing the barriers to introducing robotic technologies in agriculture. The typical approach used to plan the robot movement by demonstration, in a single shot, is the one based on Dynamic Movement Primitives (DMPs), i.e. a set of nonlinear differential equations with a well-defined landscape attractor [17]. The attractor landscape allows replicating the recorded trajectory by means of a weighted sum of equally spaced Gaussian Kernels. A generic modelling approach to learning the landscape attractor is proposed in [18] and consists of extracting the weight parameters (DMP parameters) from demonstrated movements by means of linear regression algorithms. Learning by Demonstration (LbD) based on DMPs allows to address the generalization problem for target positions close to the demonstration position, since DMPs are formulated so that convergence to different target positions is guaranteed [19].

However, to address the generalization and robustness problem in the entire reachable robot workspace (i.e. also for targets that are far from the one of the initial demonstration), DMP parameters shall be opportunely scaled depending on the distance between the new target position and the one reached during the demonstration. The original formulation of the DMP scaling factor proposed in [17], works very well for movements of monotonic shapes, such as point-to-point reaching tasks. However, for executing more complex movements, such as the ones to be accomplished in the agricultural field, and improving robustness and generalization capability in the entire reachable robot workspace, an ad-hoc scaling procedure for the DMP parameters needs to be found.

Therefore, the objective of this work is to address DMP generalization capability in the reachable robot workspace by proposing a way to scale the DMP parameters using only two demonstrations. This approach, which is based on a linear interpolation performed on the learning parameters extracted by two demonstrations, is integrated into a DMP-based motion planner which always guarantees that the computed trajectories do not exceed the robot reachable workspace, differently from the original formulation proposed in [17].

The proposed method to scale the DMP parameters is applied to the agricultural field, where the adoption of DMPs can provide a promising solution to meet the demand for a wide range of tasks in an ever-changing and mutable agricultural environment. DMPs could significantly speed up and simplify the motion planning of agricultural robots by proposing a tool that does not require an expert programmer to plan new tasks for the robot. Moreover, this approach is highly efficient, as it minimizes the time required for task planning: it just entails the farmer demonstrating the desired task to the robot through two demonstrations.

The proposed DMP motion planner was tested on the Tiago robot [20] (developed by Pal Robotics) during the fulfilment of four agricultural activities, such as digging, seeding, irrigation, and harvesting, and the obtained results were compared to the one achieved by using the original formulation of the DMPs [17]. The comparative analysis was performed by means of quantitative indices aimed at evaluating i) the accuracy of the motion reconstruction (assessed in terms of position error and human-likeness of the movement) ii) the success rate of the task execution and iii) the time needed to train the motion planner.

To sum up the main contributions of this paper are:

- to develop a motion planning system based on Learning by Demonstration that integrates a novel approach to scale the DMP parameters enhancing the robot generalization capability and reducing the learning time
- to validate the proposed approach in the agricultural field where the adoption of DMPs is limited and could

provide a promising solution to simplify the robot motion planning and meet the demand for a wide range of tasks in an ever-changing and mutable agricultural environment.

The paper is structured as follows: the literature analysis is in-deep revised in Section II. The proposed Learning by demonstration framework with a new formulation of the DMP scaling factor is presented in Section III. The experimental setup and experimental protocol are described and the results are discussed in Sect. IV. Lastly, conclusions and future works are reported in Section V.

II. RELATED WORKS

In the literature, to enhance the robustness of the DMPs to external perturbations, e.g. an obstacle on the robot path, a stylistic parameter, that is learned via multiple demonstrations, is introduced in the DMP equations [21], [22]. In the context of DMPs, a stylistic parameter refers to a parameter that influences the execution style or characteristics of a movement. It allows for fine-tuning the desired behaviour of the DMP, such as the speed, amplitude, or emphasis on certain aspects of the motion. By adjusting the stylistic parameter, one can modify the execution of the movement while still preserving the underlying shape and dynamics represented by the DMP. These parameters provide a flexible way to customize the style and execution nuances of robotic movements based on specific requirements or preferences. However, this approach does not deal with generalization problems since the information about the final target is not considered while learning the stylistic parameter [23].

Reinforcement Learning (RL) methods are promising solutions to enhance LbD generalization capabilities for different targets. Through RL the robot can go beyond imitation and actively explore the action space in real-time to learn a more generalized policy. RL algorithms, such as Q-learning [24] or Policy Gradient methods [25], allow the robot to interact with the environment, receive feedback (rewards or penalties), and adjust its actions accordingly. In [26] a method for learning tasks by observing an expert's demonstration without explicitly knowing the reward function is presented. The algorithm aims at recovering the unknown reward function and generating a policy that performs close to the expert's level. While RL has shown promise in enhancing the generalization capabilities of LbD approaches, it is important to note that RL comes with the main drawback that it requires a significant number of trials, both in simulation and real-world scenarios, before converging to an optimal policy. This iterative process of exploration and refinement can be time-consuming and resource-intensive, particularly when dealing with complex tasks or high-dimensional action spaces. A feasible approach to improving DMPs generalization capability offline is to resort to databases of parameters that are previously built via dozens of demonstrations [27], [28], [29], [30]. To better generalize with respect to the different targets, parameters that best suit the specific movement to be performed are selected from these databases and DMPs are subsequently computed, on the basis of the selected parameters, to plan the robot motion.

Other approaches based on the Gaussian Mixture Model (GMM) and the Gaussian Mixture Regression (GMR) resort to probabilistic methods for merging multiple demonstrations and building a representative trajectory [31], [32], [33]. References [34] and [35]. During the merging process, GMR takes into account the probabilistic nature of the GMM to compute the expected output given a specific input. Given a query input, GMR performs a weighted combination of the outputs from all the Gaussian components in the GMM, where the weights are determined by the probabilities associated with each component. This weighted combination allows GMR to merge the information from multiple demonstrations and generate a representative output that captures the overall trend and characteristics of the demonstrated data. These multiple demonstrations can be statically recorded offline or on-demand, throughout active incremental learning which requires the operator to record additional demonstrations as needed, i.e. whenever the trajectory uncertainty becomes too high [36].

Moreover, fuzzy logic [37] and artificial neural networks [38], [39], [40] are combined with GMM and GMR approaches to improve the capability of fitting the demonstrated samples and dealing with environmental perturbations [41], [42]. In [43] a novel methodology, that integrates GMM-based DMPs and Dynamic Time Warping (DTW) techniques, is also presented. The method is used to 3-D model manipulation skills after multiple demonstrations that could be recorded throughout multimodal or multisensor wearable systems [44], [45]. In [46], Long Short-Term Memory (LSTM) networks are used to generalize with respect to different target positions by leveraging the sequential nature of the demonstrations. LSTM networks can learn from a set of demonstrations that exhibit variations in target positions and extract the underlying patterns of the demonstrated motions to infer the desired actions or trajectories even when the target positions vary.

Nevertheless, providing the robot with many demonstrations could be time-consuming and sometimes impractical, even though they are recorded on-demand during incremental learning.

The ambition of this work is to move beyond the current state-of-the-art by proposing a method that, compared to other approaches that make use of multiple demonstrations or huge databases of parameters to generalize with respect to different target positions and tasks, requires only two sets of parameters for each task, with a significant reduction of time in the learning process.

III. THE PROPOSED SCALING METHOD FOR THE DMP MOTION PLANNER

The block scheme of the proposed DMP-based motion planner is shown in Fig. 1. In the first phase, i.e. Offline Task Learning, the motion performed by a demonstrator is

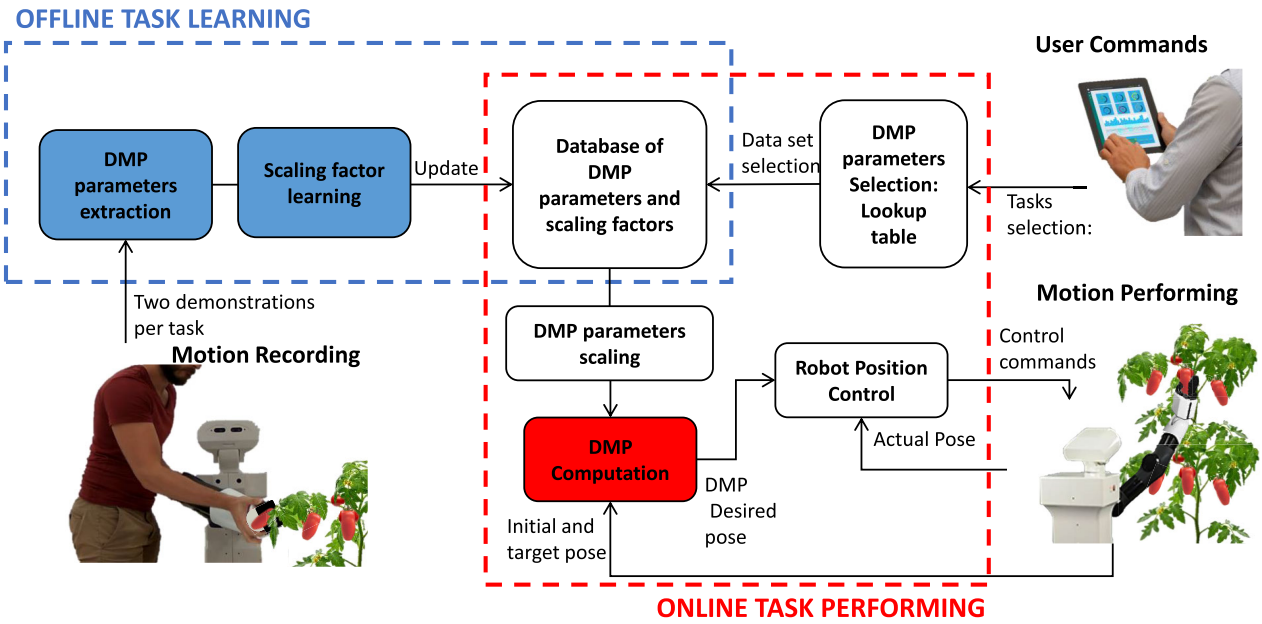

FIGURE 1. Block scheme of the proposed DMP-based Motion Planner.

firstly recorded (Motion recording in Fig. 1). Two trajectories for each task to be performed are demonstrated to the robot through a hands-on approach, i.e. a human demonstrator is required to passively move the robot arm and position sensors embedded into its joints are used to measure the performed motion. These trajectories are preferably executed at the boundary of the robot workspace (which could be experimentally retrieved by the user). By choosing target positions at the workspace boundary, the demonstrated trajectories will cover a wider range of the robot reachable space. This helps capture the full extent of the robot motion capabilities and increases the likelihood of the robot being able to generalize its movements to target positions within that workspace.

Considering the proposed demonstration method, where the task is demonstrated by passively moving the same platform that needs to learn the task, it becomes evident that the proposed approach is inherently adaptable to different platforms. Subsequently, Forward Kinematics (FK) is adopted on recorded joint trajectories to retrieve the robot Cartesian motion; afterwards, distinctive features (called DMP parameters) are extracted from the two demonstrations using the Locally Weighted Regression algorithm [17] and a DMP scaling factor that modulates the extracted parameters, depending on the target to be reached, is learned (Scaling factor learning in Fig. 1).

Afterwards, when a human subject wants the robot to perform one of the recorded tasks (Online Task Performing in Fig. 1), an online selection of the DMP parameters from the dataset is performed depending on the task to be performed and on the target position to be reached, which is generally retrieved through cameras combined with machine learning techniques that recognize the objects to be manipulated in the scene and evaluate their location [47], [48] (DMP parameters selection in Fig. 1). The task to be performed can be selected by the operator, through a tablet or a remote computer, from a list of previously learnt ones.

Selected parameters are then scaled on the basis of the target to be reached (DMP parameters scaling in Fig. 1) and used to compute DMPs that execute the desired task (DMP computation in Fig. 1).

In the following, theoretical details about the DMP computation and DMP parameters extraction are provided. Moreover, the literature method used to scale the DMP parameters and the proposed one based on two demonstrations are described. The method presented in this work tries to address generalization problems that may arise from the adoption of the original formulation of the DMP scaling factor. The proposed motion planning could be also provided with an obstacle avoidance module to guarantee computation of collision-free trajectories proposed in [49] and with an optimized spatial allocation of the Gaussian kernels that minimize the number of parameters to be learnt while keeping the same level of accuracy in the motion reconstruction proposed in [27]. For the sake of brevity, results obtained from the adoption of these modules are not shown in this work.

A DMP is a second-order non-linear system that incorporates the desired kinematic state of a robot, including position, velocity, and acceleration. It utilizes an attractor landscape to replicate a recorded trajectory by combining weighted Gaussian kernels evenly spaced throughout the landscape.

A. COMPUTATION OF DMPS

A theoretical formulation for DMPs used in robot Cartesian motion planning is described by the following equation:

$$\tau \ddot{y} = \alpha_y \left(\beta_y \left(g - y \right) - \dot{y} \right) + f_y \tag{1}$$

In this equation, τ represents a time constant, α_y and β_y are positive constants, g denotes the final point of the trajectory and f_y is a forcing term that implements the attractor landscape. The positive constants α_y and β_y in Equation (1) play a significant role in shaping the behaviour of the DMP. The constant α_y controls the speed of convergence towards the goal position g during the execution of the DMP. A higher value of α_y leads to a faster convergence, meaning that the DMP will reach the goal position more quickly. Conversely, a lower value of α_y results in a slower convergence, allowing for smoother and more gradual movements towards the goal. The constant β_y controls the spring-like behaviour of the DMP, determining the strength of the force pulling the system towards the goal position. A higher value of β_y increases the attraction force, resulting in stronger and more pronounced movements towards the goal. On the other hand, a lower value of β_y reduces the attraction force, leading to more relaxed and less forceful movements.

Solving Equation (1) yields the DMP trajectory for each degree of freedom (DoF) of the robot Cartesian space.

The forcing term f_y is expressed as:

$$f_y(x) = \frac{\sum_{i=1}^{N} \Psi_i(x)\, \psi_i}{\sum_{i=1}^{N} \Psi_i(x)} x\,(g - y_0) \tag{2}$$

In Equation (2), y_0 and g denote the initial and final points of the trajectory, respectively, which shall not be equal, for a successful adoption of discrete DMPs (see [17] for more details). Conversely, $\Psi_i(x)$ represents a fixed basis function defined as Gaussian function:

$$\Psi_i(x) = \exp\left(-\frac{1}{2\sigma^2}(x - c_i)^2\right) \tag{3}$$

Here, σ_i, c_i, and N denote the width, centres, and the number of Gaussian functions, respectively. ψ_i are weight parameters (DMP parameters) used to fit the recorded trajectory, and x is a state variable introduced to remove the time dependency from the system. It is important to note that the time dependency of Equation (1) is expressed as:

$$\tau \dot{x} = -\alpha_x x \tag{4}$$

This equation establishes a relationship between the time and the state variable x of the entire system. In [17], the range of variation for x and c_i is defined as [0, 1].

B. EXTRACTION OF DMP PARAMETERS

The extraction of DMP parameters ψ_i in Equation (2) is accomplished by using a locally weighted regression (LWR) algorithm [18]. The position, velocity, and acceleration of the recorded trajectory (y_d, $\dot{y}_d$, and $\ddot{y}_d$) are incorporated into the forcing term of Equation (1) as follows:

$$f_t = \tau^2 \ddot{y}_d - \alpha_y\left(\beta_y(g - y_d) - \dot{y}_d\right). \tag{5}$$

This formulation transforms the problem into a function approximation task, aiming to find the ψ_i parameters that minimize the discrepancy between f_t and f_y. For each kernel function $\Psi_i(t)$, the LWR algorithm searches for the corresponding ψ_i that minimizes the locally weighted quadratic error using the following cost function [17]

$$J_i = \sum_{t=1}^{P} \Psi_i(t)\,(f_t(t) - \psi_i \epsilon(t))^2 \tag{6}$$

Here, $\epsilon(t) = x\,(g - y_0)$, and P represents the number of data points in the trajectory.

C. APPROACH IN THE LITERATURE FOR SCALING DMP PARAMETERS

The scaling factor for DMP parameters proposed in [17], denoted as $(g - y_0)$ in Equation (1), has noticeable drawbacks that can be illustrated with a practical example.

Consider a scenario where a robot learns how to pick an object from an initial position A and place it at a target position B (see Fig. 2) through LbD. In this case, the robot is manually guided by a human demonstrator from A to B, and the DMP parameters $\psi_{\mathbf{AB}}$ are extracted from the recorded motion during the task. An example of this motion for the robot Cartesian coordinates is shown in green (Demo 0) in Fig. 2.

Now, suppose the robot is required to perform the same task for different target positions (C and D) shown in Fig. 2 (green lines, i.e. demo 1 and demo 2, represent the correct trajectory to be executed by the robot). If we apply the original formulation of the scaling factor, $(g - y_0)$, to scale the initial DMP parameters $\psi_{\mathbf{AB}}$, the planned motion appears as the blue trajectories in Fig. 2 (i.e. dmp1 and dmp2).

The first major drawback of this approach to scaling the initial DMP parameters, $\psi_{\mathbf{AB}}$, is evident from the Z coordinate, which is not a monotonic function. When the recorded motion has minimum or maximum points, scaling them by the factor $(g - y_0)/(g - y_0)_{AB}$ can lead to an amplification of the motion, sometimes extending beyond the robot reachable workspace if the factor is greater than 1.

In particular, a DMP can generalize to different target positions if the following relation holds:

$$a = a_{AB}\,\frac{(g - y_0)}{(g - y_0)_{AB}} \in [y_{min}, y_{max}] \tag{7}$$

Here, a represents the amplitude of the executed DMP, a_{AB} is the amplitude of the recorded motion from A to B, and $[y_{min}, y_{max}]$ denotes the acceptable Range of Motion (RoM) for the corresponding DoF of the robot.

From Equation (7), we observe that $a = (g - y_0)$ when the motion is monotonic. Therefore, if the initial and target positions are chosen within the reachable workspace of the robot, all the intermediate points of the planned DMP will also fall within that workspace.

However, if $a_{AB} > (g - y_0)_{AB}$, even if the initial and target positions are within the robot workspace, there is no guarantee that all the DMP points will remain within that workspace. In general, the capability of generalization decreases proportionally to $a_{max_{AB}}/(g - y_0)_{AB}$, as it happens to the motion along the Z axis in Fig. 2.

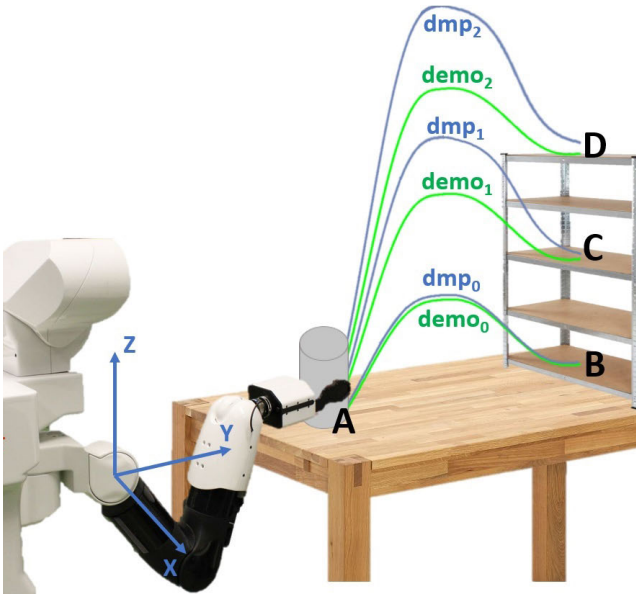

FIGURE 2. Graphical illustration of conventional DMP scaling factor adoption.

Another drawback of the approach proposed in [17] is that when $g_{AB} = y_{0_{AB}}$, the extraction of DMP parameters becomes infeasible since Equation (7) becomes indeterminate.

Therefore, in order to address these issues, a customized scaling factor for the DMP parameters needs to be developed.

D. THE PROPOSED DMP SCALING METHOD

To address the limitations of the existing DMP scaling approach, as the main contribution of this work, we introduce a different scaling factor in lieu of the traditional one adopted in [17] (i.e. $(g - y_0)$ in Equation (1)). In the following, the experimental method used to determine this scaling factor for the DMP parameters using two demonstrations is reported.

Consider the demonstrations demo_0 and demo_2 shown in Fig. 2, representing the boundaries of the workspace. We compute the DMP weights for target positions within demo_0 (target g_B) and demo_2 (target g_D) as follows:

$$\tilde{\psi}(g) = \psi_0 + \frac{\psi_2 - \psi_0}{g_2 - g_0} \cdot (g - g_0) \tag{8}$$

Here, ψ_0 and ψ_2 are the DMP weights extracted from demo_0 and demo_2, respectively, and g represents the target position within the range of g_B and g_D.

Equation (8) is formulated as a linear interpolation between the DMP weights extracted from two demonstrations, demo_0 and demo_2, to compute the weights $\tilde{\psi}$ for a target position g within the range of g_B and g_D.

The equation (8) reports a linear relationship in the form of $y = mx + c$, where y corresponds to $\tilde{\psi}(g)$, x corresponds to g, m represents the slope, and c is the y-intercept. In this case, m is computed as $\frac{\psi_2 - \psi_0}{g_2 - g_0}$, which represents the rate of change of the DMP weights with respect to the target position.

By multiplying the slope m with the difference between the target position g and the initial position g_0, and adding the y-intercept ψ_0, we obtain the interpolated DMP weights $\tilde{\psi}(g)$. This interpolation allows us to estimate the DMP weights for any target position within the range defined by demo_0 and demo_2, providing a smooth and continuous scaling of the DMP parameters.

It is important to note that the proposed approach eliminates the dependence on the initial position y_0 of the DMP. This simplifies the fitting of DMP parameters and ensures more accurate reproduction of the DMP. In this approach, DMP parameters ψ_0 and ψ_2 are extracted based on the relative motion $\tilde{y}_d = y_d - y_0$.

Therefore, when computing a DMP for a new target position g within the range of g_B and g_D, with an initial position y_0, we utilize Equation (8) to obtain new weights $\tilde{\psi}$ that are then used in Equation (1). Finally, the robot reference position $\tilde{y}$ is calculated as $\tilde{y} = y_0 + y$, where y represents the DMP obtained by integrating Equation (1).

1) ADVANTAGIES OVER EXISTING METHODS

The proposed method provides two key advantages compared to the literature approach used for scaling the DMP parameters: enhanced generalization capabilities and reduced amplification issues. Indeed, by capturing the relationship between target positions and DMP weights, the method enables a better generalization to new target positions within the robot workspace. This means that the robot can effectively adapt its motion to different target positions without the need for explicit parameter tuning. Moreover, the method mitigates the risk of amplifying the motion beyond the robot reachable workspace. This ensures that planned trajectories remain within the acceptable RoM, avoiding potential undesired movements.

Additionally, in comparison to existing methods based on multiple demonstrations, the proposed approach significantly simplifies and expedites the learning process, making it particularly well-suited for dynamic environments such as agriculture, where continuous adaptation of robot activities is required.

IV. APPLICATION TO AGRICULTURAL ACTIVITIES AND EXPERIMENTAL VALIDATION

DMPs could be effectively used to plan complex activities in robotics, such as the ones to be performed in the agricultural domain, and could significantly speed up and simplify the motion planning of agricultural robots. Indeed it could provide the end-user with a tool that does not require an expert programmer to plan new tasks for the robot.

In this section, benefits from the application of DMPs to the agricultural domain are pointed out in terms of the efficacy of the motion planning (assessed in terms of motion reconstruction accuracy and success rate of the task execution) and simplicity/speediness of the learning process.

The experimental validation of the proposed DMP-based motion planner was carried out on four agricultural activities, i.e. digging, seeding, irrigation, and harvesting.

In the following, the i) experimental robotic platform, ii) experimental protocol, and iii) results and discussions are reported.

A. EXPERIMENTAL ROBOTIC PLATFORM

The robotic platform used to test the proposed DMP-based motion planner is the Tiago robot [20] developed by PAL Robotics S.L. The main robot components used to carry out the experiments are shown in Fig. 1. They are: i) the RGB-D camera (Astra S manufactured by Orbbec [50]) with a resolution of 640x480 and depth range of $0.6 - 8m$, ii) the lifting torso (able to move at 50 mm/s from a minimum to a maximum height of 1.10 m and 1.45 m, respectively), iii) the 7 Dof arm (composed of four M90 modules and one 3 DoF wrist) with a payload of 3 kg iv) the Pal gripper (including two independent fingers with a linear range of 4 cm) and v) a workstation Dell G5 15 5500 with Ubuntu 16.04 Operating System equipped with an Intel® Core™ i7-10870H processor at 8 × 5 GHz and 16 GB of RAM.

B. EXPERIMENTAL PROTOCOL

The experimental validation consisted of two phases, named in the following a) *Offline task learning* and b) *Online task performing*. The 1^{st} phase was aimed to record the motion from a demonstrator during the execution of four working activities, i.e. digging, seeding, irrigation, and harvesting, and to subsequently extract from this motion the set of DMP parameters to be stored in the database. The 2^{nd} phase was intended to validate the proposed approach and to assess its generalization capability with respect to different target positions. For this purpose, the proposed method based on 2 demonstrations was compared to two methods adopted in the literature, i.e. the conventional approach based on a single demonstration [17] and the conventional approach based on multiple demonstrations [27].

In the following, performance indices used to carry out the comparative analysis are defined. Moreover, the two phases of the experimental validation, i.e. *offline task learning* and *online task performing*, are described.

1) PERFORMANCE INDICES

Four performance indices [27], namely i) Normalized Position Error (NPE), ii) Human-Likeness Index (HLI), iii) Database Building Time (DBT) and iv) Success Rate (SR), were computed to perform the comparative analysis among the different approaches.

- The NPE assesses the capability of the proposed approach to accurately replicate the demonstrated motions. It is normalized with respect to the overall displacement of the recorded motion and is computed as follows

$$NPE = \frac{1}{N} \cdot \frac{1}{\|g - y_0\|} \sum_{i=1}^{N} \|p(i) - p_m(i)\| \tag{9}$$

  where N is the number of collected samples, $p(i)$ is the position of the computed DMP and $p_m(i)$ is the recorded position at the i-th sample. The lower the NPE the higher the trajectory reconstruction accuracy.
- the HLI evaluates the capability of the proposed DMP-based Motion Planner to imitate the human style motion and is assessed as

$$HLI = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(\|a^r(i)\| - \|a^c(i)\| \right)^2} \tag{10}$$

  where N is the number of time instants and $a^r(i)$ and $a^c(i)$ are the accelerations of the recorded and computed DMP trajectories, evaluated at the sample i. Lower HLI values are expected when a high degree of similarity between the DMP trajectory style and the human one is reached.
- the DBT returns a measure of the time required to build a database and is calculated as

$$DBT = \sum_{j=1}^{\bar{N}_t} \bar{N}_d * \sum_{i=1}^{N_d} \sum_{j=1}^{N_t} \frac{T_{ij}}{N_d * N_t} \tag{11}$$

  where $\bar{N}_d$ and $\bar{N}_t$ are the number of demonstrations and tasks to be stored in the database, respectively, N_d and N_t are the total number of demonstrations and tasks performed during the experimental session (i.e. 25 demonstrations and 4 tasks), respectively, and T_{ij} is the time required to record the i-th demonstration of the j-th task.
- the SR of the task execution is used to evaluate the capability of a given approach to accomplish the task and is evaluated as

$$Success\ rate = \frac{N_{succ}}{N_{tot}} \times 100 \tag{12}$$

  where N_{succ} is the number of trials successfully accomplished and N_{tot} is the number of all the performed trials. A task is considered successfully accomplished if the trajectory is within the robot reachable workspace for the given task (see Fig. 4).

2) STATISTICAL ANALYSIS

The mean value and Standard Deviation (SD) of the all previously described indices were computed on the 25 target positions for each task and DMP method. Since the data were not normally distributed, a statistical analysis based on Wilcoxon paired-sample test was performed in order to carry out the comparative analysis among the different approaches, and Bonferroni correction was applied on multiple comparisons ($p - value < 0.05/N_c$, where N_c is the number of comparisons).

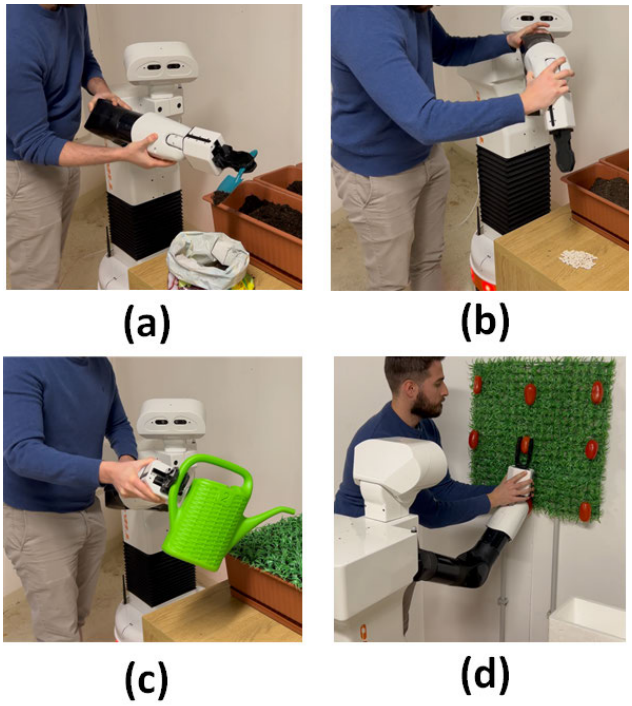

FIGURE 3. A picture of the offline task learning for the (a) digging, (b) seeding, (c) irrigation and (d) harvesting.

3) OFFLINE TASK LEARNING

In the 1st phase of the experimental validation, a human subject was asked to teach the robot how to perform the working activities, namely digging, seeding, irrigation, and harvesting, by means of a hands-on approach (please see Fig. 3).

In other words, the subject was required to passively move the robot arm in order to accomplish the task. During the task execution, the robot was driven by a zero-torque control[1], [51] and the sensors embedded in the robot, i.e. the encoders, were used to record the joints' motion. The tasks are divided into several elementary actions, as follows:

- the digging task was divided into 5 subtasks, i.e. i) tool reaching, ii) digging, iii) soil placing into the bucket, iv) tool placing and v) homing
- the seeding task was divided into 3 subtasks, i.e. Seed reaching, Seed placing into the hall and Homing
- the irrigation task was divided into 4 subtasks, i.e. i) seeds irrigation with a watering can, ii) watering can placing and iii) homing
- the harvesting task was divided into 4 subtasks, i.e. i) vegetable reaching, ii) vegetable detaching from the plant, iii) vegetable placing into the crate and iv) homing

The four tasks were performed for 25 different positions of the targets. These positions are reported in Fig. 5 for the four tasks.

[1]A zero-torque control is a control modality in which the robot reference torque is set to zero so that it can be moved by a user who exerts controlled forces on the robot kinematic chain).

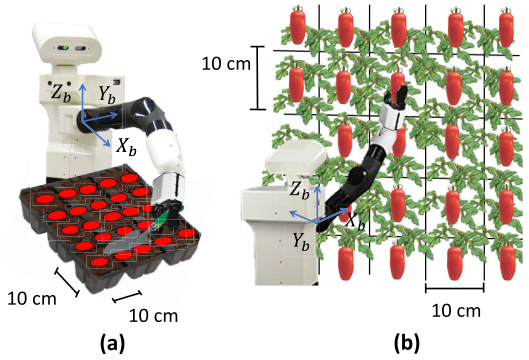

FIGURE 4. A graphical illustration of the target positions for the (a) digging, seeding, irrigation and (b) harvesting.

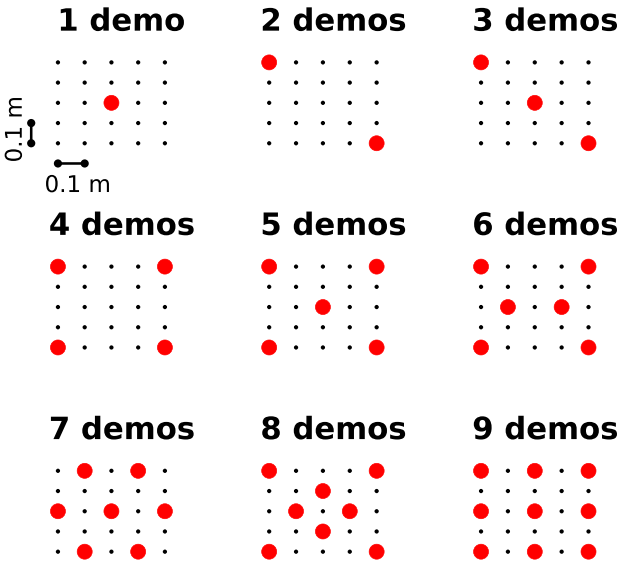

FIGURE 5. Target positions used to build up the multiple demos database.

Subsequently, Cartesian trajectories were computed by means of the robot FK and DMP parameters were extracted from these trajectories.

4) ONLINE TASK PERFORMING

The 2nd phase of the experimental validation was aimed at assessing the generalization capability of the proposed approach based on a new formulation of the scaling factor and comparing it to the ones based on the conventional formulation of the scaling factor [17], [27] (see Fig. 6).

For this purpose, three experimental sessions were carried out.

The 1st experimental session was aimed at evaluating the number of demonstrations per task needed to achieve an acceptable generalization capability with the conventional approach based on multiple demonstrations (i.e. SR of the task execution $>$ 90). 25 databases with an increasing number of demonstrations per task were built. They are named in the following *1-demo-database*, *2-demos-database* ...and *25-demos-database*. The demonstrations included in these

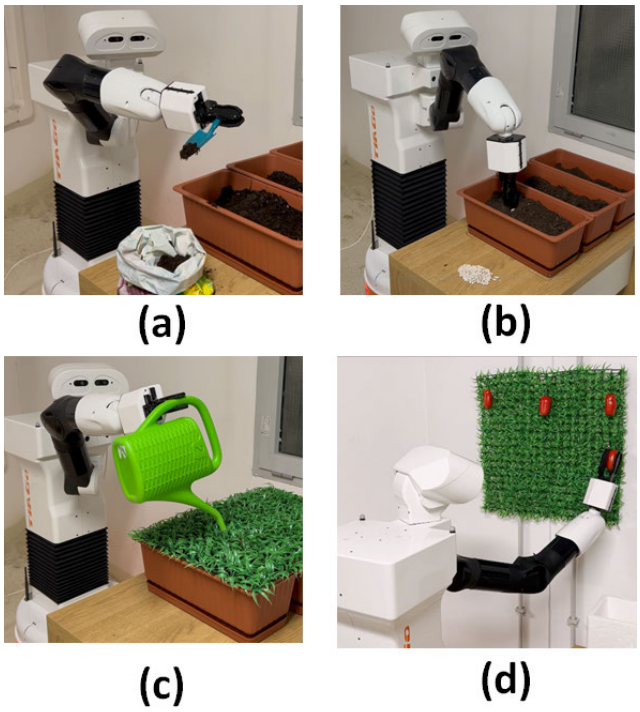

FIGURE 6. A picture of the online task performing for the (a) digging, (b) seeding, (c) irrigation and (d) harvesting.

databases have been chosen to be equally spaced in the robot workspace in terms of target positions. Target positions of the demonstrations included in the 1-demo-database, 2-demos-database ... and 9-demos-databases are shown in Fig. 5 as an example.

Subsequently, the robot arm was operated to perform the four tasks, i.e. digging, seeding, irrigation, and harvesting, for the 25 different target positions shown in black in Fig. 5 using the previously built databases, i.e. *1-demo-database*, *2-demos-database* ... and *25-demos-database*. Hence, the previously mentioned performance indicators were calculated on the obtained DMPs and the number of demonstrations per task needed to achieve a success rate of the task execution > 90 was calculated.

The 2^{nd} experimental session was aimed at comparing the proposed approach based on 2 demonstrations and the conventional ones based on single and multiple demonstrations. For this purpose, three databases were built.

For the proposed approach, two sets of DMP parameters (i.e. the ones computed for upper and lower boundary trajectories in the robot workspace shown in Fig. 5) were used to calculate DMP scaling parameters for each task and robot Cartesian DoF. These parameters were then stored in a database named *proposed-2-demos-database*. In other words, among the 25 recorded trajectories performed by the demonstrator, only two trajectories (i.e. the ones at the boundary of the workspace) were used to build the database for the proposed motion planner training. The other ones were used as benchmark trajectories to evaluate the motion planner performance.

FIGURE 7. A picture of the online performing of the harvesting task in a realistic scenario.

Differently, for the conventional approach based on a single demonstration [17], one set of DMP parameters (i.e. the ones computed for the middle target point shown in Fig. 5) was stored for each task and robot Cartesian DoF and a database named *conventional-1-demo-database* was built.

Finally, for the conventional approach based on multiple demonstrations, M_i sets of DMP parameters per task were stored in a database named *conventional-multi-demos-database*. M_i is the number of demonstrations, of the $i-th$ task, that is required to achieve a success rate of the task execution > 90. It was computed during the 1^{st} experimental session.

Afterwards, the robot arm was operated to perform the four tasks, i.e. digging, seeding, irrigation, and harvesting, for the 25 different target positions shown in black in Fig. 5 using the previously built databases, i.e. *conventional-1-demo-database*, *conventional-multiple-demos-database*, and *proposed-2-demos-database*. Hence, the previously mentioned performance indicators were calculated based on the obtained DMPs.

The 3^{nd} experimental session was aimed at evaluating the performance of the proposed approach while executing the harvesting task in a more realistic scenario made of real plants (see Fig. 7).

This specific task is indeed particularly challenging, due to the presence of leaves and stems around the fruits and to the 3-dimensional disposition of the fruits that may impact the algorithm performance.

C. RESULTS AND DISCUSSIONS

Fig. 8 reports the results obtained for the 1^{st} experimental session, which was aimed at evaluating the number of demonstrations per task needed to achieve an acceptable

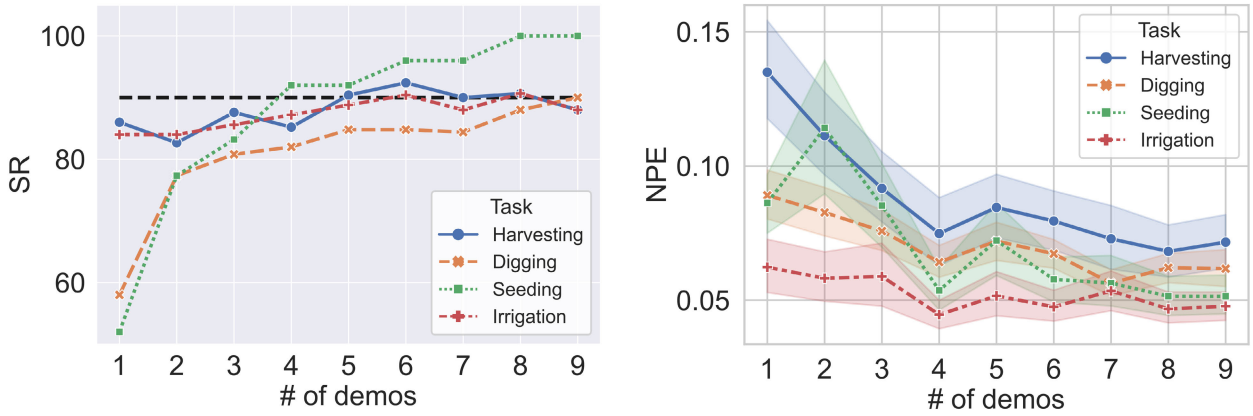

FIGURE 8. Experimental results obtained for the 1st experimental session. The black dashed line represents the acceptability threshold for SR, i.e. 90%.

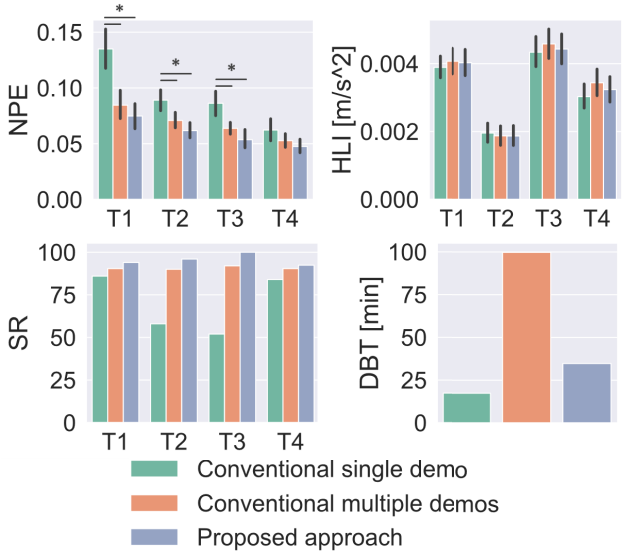

FIGURE 9. Experimental results obtained for the 2nd the experimental session (T1=Harvesting, T2=Digging, T3=Seeding and T4=Irrigation). The star symbol highlights statistically significant differences ($p - value < 0.01$).

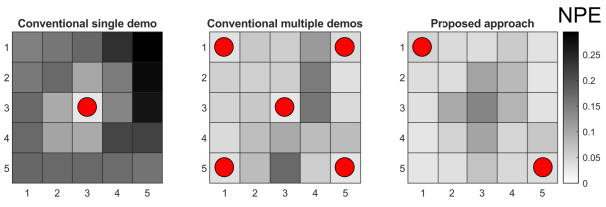

FIGURE 10. Heat map of NPE for the three approaches during the fulfilment of the harvesting task. Red dots are the demonstration target positions.

generalization capability with the conventional approach ($SR > 90$). In this figure, the mean value and SD of the NPE and SR computed on the 25 target positions for each task using the conventional approach based on multiple demonstrations are reported.

It is worth noticing from the figure that the higher the number of demonstrations stored in the database the better the generalization capability of the conventional approach based on multiple demonstrations. Indeed, the SR of the task execution shows an increasing trend as the number of demonstrations grows. Likewise, the NPE as a function of the number of demonstrations reports a decreasing trend. In general, results obtained from the 1st experimental session highlighted that at least a total of 24 demonstrations, namely 5 demonstrations for the harvesting, 9 demonstrations for the digging, 4 demonstrations for the seeding and 6 demonstrations for the irrigation, are needed for the conventional approach to achieve a $SR > 90\%$.

Fig. 9 reports the results obtained for the 2nd experimental session, i.e. the mean value and SD of the NPE, HLI and DBT, computed on the 25 target positions for each task and DMP method, namely *proposed 2 demos DMP*, *conventional single demo DMP* and *conventional multiple demos DMP*.

As expected, the *conventional single demo DMP*, compared to the other methods, had the worst performance in terms of NPE (statically significant differences among the approaches with $p - value < 0.01$ are highlighted by the star symbol in the figure). As shown in the NPE heatmap of Fig. 10, the best performance of the 1-Demo DMP is obtained for target positions that are very close to one of the demonstrations (Maximum NPE = 0.04) and drastically decreases when the new target positions, where the DMPs are tested, are far from it (Maximum NPE = 0.32). In contrast, when the *conventional multiple demos DMP* is adopted, the new target positions to be reached are often close to one of the demonstrations (maximum distance < 20 cm); hence, higher performance in terms of NPE is obtained (Maximum NPE = 0.09) and this performance is comparable to the one achieved with the proposed method ($p - value > 0.01$). The *conventional multiple demos DMP*, despite its high performance in terms of NPE, has a long-lasting training phase, as confirmed by the BDT in Fig. 9. Indeed, it requires the user to collect multiple demonstrations for each task:

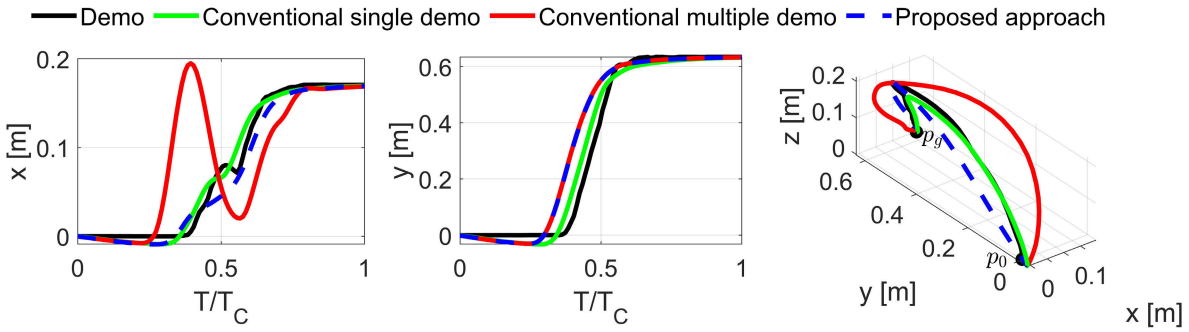

FIGURE 11. End-effector Cartesian position during the fulfilment of the seeding task.

a total of 24 demonstrations were collected to build the multiple-demos database, which is very time-consuming if one considers that about 5 minutes, on average, were required to record one demonstration.

On the other hand, the proposed approach requires only two demonstrations to be recorded for each task, with consequent time savings during the training phase. These time savings could be also appreciated during the execution phase. In particular, the computational burden of the look-up table approach, which strongly depends on the dimensions of the DMP database to be queried, is significantly lower for the proposed approach compared *conventional multiple demos DMP* (approximately N times lower, where N is the number of DMP parameter sets) [27]. Moreover, as evident from Fig 9, the ability of the proposed approach to replicate trajectories with a high degree of similarity with respect to the one performed by the human demonstrator, is outlined by the HLI that is always comparable to the one obtained with the *conventional multiple demos DMP* and *conventional single demo DMP* ($p - value > 0.01$).

The achieved results are also confirmed by the Success Rate of the task execution which was higher than 90% for all the tasks when the *proposed 2 demos DMP* and *conventional multiple demos DMP* were adopted.

Conversely, the *conventional single demo DMP* achieved for some tasks (i.e. digging and seeding) very low performance (about 52%) highlighting that it is not able to generalize with respect to the entire reachable robot workspace for the given task to be performed. Indeed, the original formulation of the DMP scaling factor proposed in [17], which works well for movements with monotonic shapes, may encounter difficulties when applied to more complex tasks such as digging and seeding in the agricultural field. These types of tasks may potentially involve movements with minimum and maximum points, which may be inadvertently amplified beyond the robot reachable workspace when the traditional approach to scaling the DMP parameters is adopted, resulting in poor task execution and a low success rate.

Fig. 11 shows the Cartesian position of the robot end-effector during the fulfilment of a representative task (i.e. seeding task), when the three approaches are adopted.

It is worth observing from the figure that, when the recorded trajectory is monotonic, all the approaches are somewhat able to reproduce the desired motion, as it happens for the Y axis. Contrariwise, when the recorded trajectory is non-monotonic and the new target position to be reached by the DMP is far from the one of the demonstration (as for X axis), the *conventional 1 demo DMP* results in an undesired amplification of the motion shape that leads to an unsuccessful execution of the task. This issue is well addressed by the proposed method that proposes a new formulation of the scaling factor based on two demonstrations, as evident from the figure.

The results obtained in laboratory settings for the 1^{st} and 2^{nd} experimental sessions are consistent with those obtained for the 3^{rd} experimental session carried out in the realistic scenario. Specifically, when performing the harvesting task on a real tomato plant across five target positions, the performance of the proposed approach does not deteriorate compared to the outcomes achieved in laboratory settings: the mean value and SD computed on the position and orientation error calculated for the five targets are 8 ± 2 *mm* and 0.09 ± 0.02 *rad*, respectively, achieving a success rate of the task execution of 100 %.

1) LIMITATIONS AND POSSIBLE SOLUTIONS

While the proposed DMP scaling method offers some advantages compared to existing literature approaches, it is important to consider potential limitations and explore possible solutions to enhance its robustness and adaptability in various situations.

One potential limitation of the proposed method is its sensitivity to a limited demonstration range. As it relies on linear interpolation between two demonstrations, its effectiveness may decrease when applied to target positions outside the demonstrated range. Extrapolation beyond this range can lead to less accurate trajectory reproduction.

To address this limitation, it is recommended to select the target positions of the two demonstrations as close to the boundaries of the robot workspace as possible. By expanding the range of the demonstrations, the method can better handle target positions beyond the initially demonstrated range.

Another possible limitation of the proposed method is the inability to handle nonlinear relationships between the DMP weights extracted from the two demonstrations. Since the method employs linear interpolation, it assumes a linear relationship between target positions and DMP weights. This limitation restricts the ability to capture complex nonlinear relationships, which may be necessary for certain tasks.

To overcome this issue, an extension of Equation (8) can be made to enable non-linear interpolation. This can be achieved by utilizing a different interpolation function, such as a spline or polynomial function, thus enhancing the flexibility of the approach to capture more intricate relationships between target positions and DMP weights.

V. CONCLUSION

In this work, a novel approach to scaling the DMP parameters via two demonstrations has been presented, with the purpose of increasing the DMPs generalization with respect to the different target positions and reducingsing the time required to train the motion planner during the learning phase.

The feasibility of such an approach to planning complex activities in robotics, such as the ones to be fulfilled in the Agricultural field, has been assessed; four agricultural activities, namely digging, seeding, irrigation, and harvesting, have been successfully taught to the Tiago robot taking, therefore, a step forward with respect to the literature in this application domain. The achieved results demonstrated the high generalization capability of the proposed method that was able to successfully perform all the agricultural tasks in the robot reachable workspace with an improvement in the Success Rate of 25.6% and 4.9%, in comparison to the *conventional single demo DMP* and *conventional multiple demo DMP*, respectively. On average, a mean SR of 95.60% was achieved with the proposed approach. This capability has been also demonstrated through the accuracy of the motion reconstruction (a maximum NPE of 0.16 was achieved when the proposed method was adopted). The obtained result could be considered acceptable for the tested application, considering that a few unsuccessful trials generally occur when the target position is at the edge of the reachable robot workspace. Indeed, for robotic platforms such as the one adopted in the experimentation (i.e. mobile manipulators), unsuccessful tasks at the edges of the workspace may be even overcome by slightly moving the robot base as needed.

Moreover, the achieved results in terms of DBT demonstrated that the proposed approach can significantly speed up and simplify the motion planning of agricultural robots by proposing a tool that does not require an expert programmer to plan new tasks for the robot.

Future works will be mainly addressed to i) test the proposed approach on other application domains, such as robotic surgery, rehabilitation, remote inspection and maintenance etc. ii) increase the number of tasks of the database and iii) integrate into the framework an obstacle avoidance module that enhances the robustness of the motion planner to external perturbations.

REFERENCES

[1] C. W. Bac, E. J. van Henten, J. Hemming, and Y. Edan, "Harvesting robots for high-value crops: State-of-the-art review and challenges ahead," *J. Field Robot.*, vol. 31, no. 6, pp. 888–911, Nov. 2014.
[2] T. T. Nguyen, E. Kayacan, J. De Baedemaeker, and W. Saeys, "Task and motion planning for apple harvesting robot," *IFAC Proc. Volumes*, vol. 46, no. 18, pp. 247–252, Aug. 2013.
[3] I. A. Sucan, M. Moll, and L. E. Kavraki, "The open motion planning library," *IEEE Robot. Autom. Mag.*, vol. 19, no. 4, pp. 72–82, Dec. 2012.
[4] L. Jaulin, "Path planning using intervals and graphs," *Reliable Comput.*, vol. 7, no. 1, pp. 1–15, 2001.
[5] M. A. F. Jensen, D. Bochtis, C. G. Sørensen, M. R. Blas, and K. L. Lykkegaard, "In-field and inter-field path planning for agricultural transport units," *Comput. Ind. Eng.*, vol. 63, no. 4, pp. 1054–1061, Dec. 2012.
[6] J. Zeng, R. Ju, L. Qin, Y. Hu, Q. Yin, and C. Hu, "Navigation in unknown dynamic environments based on deep reinforcement learning," *Sensors*, vol. 19, no. 18, p. 3837, Sep. 2019.
[7] G. G. R. D. Castro, G. S. Berger, A. Cantieri, M. Teixeira, J. Lima, A. I. Pereira, and M. F. Pinto, "Adaptive path planning for fusing rapidly exploring random trees and deep reinforcement learning in an agriculture dynamic environment UAVs," *Agriculture*, vol. 13, no. 2, p. 354, Jan. 2023.
[8] Z. Fang and X. Liang, "Intelligent obstacle avoidance path planning method for picking manipulator combined with artificial potential field method," *Ind. Robot, Int. J. Robot. Res. Appl.*, vol. 49, no. 5, pp. 835–850, Jun. 2022.
[9] M. Zhao and X. Lv, "Improved manipulator obstacle avoidance path planning based on potential field method," *J. Robot.*, vol. 2020, pp. 1–12, Jan. 2020.
[10] T. T. Nguyen, E. Kayacan, J. De Baerdemaeker, and W. Saeys, "Motion planning algorithm and its real-time implementation in apples harvesting robot," in *Proc. Int. Conf. Agricult. Eng.*, 2014, pp. 6–10.
[11] C. Liu, Q. Feng, Z. Tang, X. Wang, J. Geng, and L. Xu, "Motion planning of the citrus-picking manipulator based on the TO-RRT algorithm," *Agriculture*, vol. 12, no. 5, p. 581, Apr. 2022.
[12] Y. Chen, Y. Fu, B. Zhang, W. Fu, and C. Shen, "Path planning of the fruit tree pruning manipulator based on improved RRT-connect algorithm," *Int. J. Agricult. Biol. Eng.*, vol. 15, no. 2, pp. 177–188, 2022.
[13] B. R. Duffy, "Anthropomorphism and the social robot," *Robot. Auto. Syst.*, vol. 42, nos. 3–4, pp. 177–190, Mar. 2003.
[14] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Auto. Syst.*, vol. 57, no. 5, pp. 469–483, May 2009.
[15] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annu. Rev. Control, Robot., Auto. Syst.*, vol. 3, no. 1, pp. 297–330, May 2020.
[16] P. A. Lasota and J. A. Shah, "Analyzing the effects of human-aware motion planning on close-proximity human–robot collaboration," *Hum. Factors, J. Hum. Factors Ergonom. Soc.*, vol. 57, no. 1, pp. 21–33, Feb. 2015.
[17] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Comput.*, vol. 25, no. 2, pp. 328–373, Feb. 2013.
[18] S. Schaal and C. G. Atkeson, "Constructive incremental learning from only local information," *Neural Comput.*, vol. 10, no. 8, pp. 2047–2084, Nov. 1998.
[19] Z. Xie, Q. Zhang, Z. Jiang, and H. Liu, "Robot learning from demonstration for path planning: A review," *Sci. China Technological Sci.*, vol. 63, no. 8, pp. 1325–1334, Aug. 2020.
[20] J. Pages, L. Marchionni, and F. Ferro, "TIAGo: The modular robot that adapts to different research needs," in *Proc. Int. Workshop Robot Modularity*, vol. 290, 2016, pp. 1–4.
[21] L. Koutras and Z. Doulgeri, "A novel DMP formulation for global and frame independent spatial scaling in the task space," in *Proc. 29th IEEE Int. Conf. Robot Human Interact. Commun. (RO-MAN)*, Aug. 2020, pp. 727–732.
[22] T. Matsubara, S.-H. Hyon, and J. Morimoto, "Learning stylistic dynamic movement primitives from multiple demonstrations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 1277–1283.
[23] M. Ginesi, N. Sansonetto, and P. Fiorini, "Overcoming some drawbacks of dynamic movement primitives," *Robot. Auto. Syst.*, vol. 144, Oct. 2021, Art. no. 103844.
[24] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, and I. Osband, "Deep Q-learning from demonstrations," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 1–8.

[25] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Netw.*, vol. 21, no. 4, pp. 682–697, May 2008.
[26] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, p. 1, doi: 10.1145/1015330.1015430.
[27] C. Lauretti, F. Cordella, E. Guglielmelli, and L. Zollo, "Learning by demonstration for planning activities of daily living in rehabilitation and assistive robotics," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1375–1382, Jul. 2017.
[28] C. Tamantini, F. Cordella, C. Lauretti, and L. Zollo, "The WGD—A dataset of assembly line working gestures for ergonomic analysis and work-related injuries prevention," *Sensors*, vol. 21, no. 22, p. 7600, Nov. 2021.
[29] C. Tamantini, M. Lapresa, F. Cordella, F. S. di Luzio, C. Lauretti, and L. Zollo, "A robot-aided rehabilitation platform for occupational therapy with real objects," in *Proc. Int. Conf. NeuroRehabilitation*. Cham, Switzerland: Springer, 2020, pp. 851–855.
[30] C. Lauretti, F. Cordella, A. L. Ciancio, E. Trigili, J. M. Catalan, F. J. Badesa, S. Crea, S. M. Pagliara, S. Sterzi, and N. Vitiello, "Learning by demonstration for motion planning of upper-limb exoskeletons," *Frontiers neurorobotics*, vol. 12, p. 5, Jan. 2018.
[31] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with Gaussian mixture models," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 943–957, Oct. 2011.
[32] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intell. Service Robot.*, vol. 9, no. 1, pp. 1–29, Jan. 2016.
[33] A. Hewitt, C. Yang, Y. Li, and R. Cui, "DMP and GMR based teaching by demonstration for a KUKA LBR robot," in *Proc. 23rd Int. Conf. Autom. Comput. (ICAC)*, Sep. 2017, pp. 1–6.
[34] S. Chernova and M. Veloso, "Confidence-based policy learning from demonstration using Gaussian mixture models," in *Proc. 6th Int. Joint Conf. Auto. Agents Multiagent Syst.*, May 2007, pp. 1–8.
[35] E. Pignat and S. Calinon, "Bayesian Gaussian mixture model for robotic policy imitation," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4452–4458, Oct. 2019.
[36] G. Maeda, M. Ewerton, T. Osa, B. Busch, and J. Peters, "Active incremental learning of robot movement primitives," in *Proc. Conf. Robot Learn.*, 2017, pp. 37–46.
[37] F. Hoffmann, "Fuzzy behavior coordination for robot learning from demonstration," in *Proc. IEEE Annu. Meeting Fuzzy Inf.*, Jun. 2004, pp. 157–162.
[38] J. Kim, N. Cauli, P. Vicente, B. Damas, F. Cavallo, and J. Santos-Victor, "'iCub, clean the table!' A robot learning from demonstration approach using deep neural networks," in *Proc. IEEE Int. Conf. Auto. Robot Syst. Competitions (ICARSC)*, Apr. 2018, pp. 3–9.
[39] S. Thakur, H. van Hoof, J. C. G. Higuera, D. Precup, and D. Meger, "Uncertainty aware learning from demonstrations in multiple contexts using Bayesian neural networks," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 768–774.
[40] M. Umar Suleman and M. M. Awais, "Learning from demonstration in robots: Experimental comparison of neural architectures," *Robot. Comput.-Integr. Manuf.*, vol. 27, no. 4, pp. 794–801, Aug. 2011.
[41] C. Yang, C. Chen, N. Wang, Z. Ju, J. Fu, and M. Wang, "Biologically inspired motion modeling and neural control for robot learning from demonstrations," *IEEE Trans. Cognit. Develop. Syst.*, vol. 11, no. 2, pp. 281–291, Jun. 2019.
[42] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 777–787, Mar. 2019.
[43] H. Su, A. Mariani, S. E. Ovur, A. Menciassi, G. Ferrigno, and E. De Momi, "Toward teaching by demonstration for robot-assisted minimally invasive surgery," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 484–494, Apr. 2021.
[44] W. Qi and A. Aliverti, "A multimodal wearable system for continuous and real-time breathing pattern monitoring during daily activity," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 8, pp. 2199–2207, Aug. 2020.
[45] W. Qi, S. E. Ovur, Z. Li, A. Marzullo, and R. Song, "Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 6039–6045, Jul. 2021.
[46] S. Pareek and T. Kesavadas, "IART: Learning from demonstration for assisted robotic therapy using LSTM," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 477–484, Apr. 2020.
[47] F. Wu, Z. Yang, X. Mo, Z. Wu, W. Tang, J. Duan, and X. Zou, "Detection and counting of banana bunches by integrating deep learning and classic image-processing algorithms," *Comput. Electron. Agricult.*, vol. 209, Jun. 2023, Art. no. 107827.
[48] F. Meng, J. Li, Y. Zhang, S. Qi, and Y. Tang, "Transforming unmanned pineapple picking with spatio-temporal convolutional neural networks," *Comput. Electron. Agricult.*, vol. 214, Nov. 2023, Art. no. 108298.
[49] C. Lauretti, F. Cordella, and L. Zollo, "A hybrid joint/Cartesian DMP-based approach for obstacle avoidance of anthropomorphic assistive robots," *Int. J. Social Robot.*, vol. 11, no. 5, pp. 783–796, Dec. 2019.
[50] Orbbec. (2023). *Camera Astra S Technical Specifications*. Accessed: Jun. 17, 2023. [Online]. Available: https://shop.orbbec3d.com/Astra
[51] D. Formica, L. Zollo, and E. Guglielmelli, "Torque-dependent compliance control in the joint space for robot-mediated motor therapy," *J. Dyn. Syst., Meas., Control*, vol. 128, no. 1, pp. 152–158, Mar. 2006, doi: 10.1115/1.2173009.

C. LAURETTI (Member, IEEE) received the M.S. degree in biomedical engineering and the Ph.D. degree in bio-engineering and bio-science from the Campus Bio-Medico University of Rome, Italy, in October 2015 and 2019, respectively. He is currently an Assistant Professor of bio-engineering with the Campus Bio-Medico University of Rome. His main research interests include the fields of rehabilitation and assistive robotics, robotic surgery, agricultural robotics, and robotics for hazardous environments. His activities are focused on the following topics: human–robot interaction, multimodal interfaces for robot control, machine learning techniques for robot path-planning and control, bio-cooperative control systems, and sensor data processing and fusion. In 2017, he was awarded the "First Robotics Made in Italy Video Contest," organized by the IEEE Robotics Automation Society Italian Chapter. In 2021, he received the Best Paper Award from the MED Energy Conference (Ravenna 2021).

C. TAMANTINI (Member, IEEE) received the B.Sc. degree in medical engineering from the Tor Vergata University of Rome, in 2016, and the M.Sc. degree in biomedical engineering and the Ph.D. degree in science and engineering for humans and the environment from Università Campus Bio-Medico di Roma (UCBM), Italy, in 2018 and 2023, respectively. He is currently a Postdoctoral Research Fellow with the Research Unit of Advanced Robotics and Human-Centred Technologies. His current research interests include the development of algorithms and approaches for user state estimation to be used to adapt robot behaviors.

L. ZOLLO (Senior Member, IEEE) received the M.S. degree in electronic engineering from Università degli Studi di Napoli Federico II, in 2000, and the Ph.D. degree in bioengineering from Scuola Superiore Sant'Anna di Pisa, in 2004. She is currently a Full Professor of Bioengineering and the Director of the Master of Science in Biomedical of Università Campus Bio-Medico di Roma (UCBM), where she is also the Director of the CREO Lab-Laboratory of Advanced Robotics and Human-Centred Technologies. She has been involved in more than 40 EU-funded and national projects in her application fields, such as EU H2020/FET-Open/SOMA as a Project Coordinator and EU H2020/ODIN and AIDE as a Scientific Coordinator with UCBM. She has authored/coauthored more than 160 scientific publications and six patents. Her research interests include rehabilitation and assistive robotics, bio robotics and bionics, human–machine interfaces, and collaborative robotics.

Open Access funding provided by 'Università "Campus Bio-Medico" di Roma' within the CRUI CARE Agreement