IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 30 May 2024, accepted 14 June 2024, date of publication 24 June 2024, date of current version 12 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3418573

TOPICAL REVIEW

Robots Adapting to the Environment: A Review on the Fusion of Dynamic Movement Primitives and Artificial Potential Fields

IRATI RASINES[1,2], ITZIAR CABANES[1], (Member, IEEE), ANTHONY REMAZEILLES[2], AND JOSEPH MCINTYRE[2,3]

[1]Department of Automatic Control and Systems Engineering, Bilbao School of Engineering, University of the Basque Country (UPV/EHU), 48013 Bilbao, Spain
[2]Tecnalia, Basque Research and Technology Alliance (BRTA), 48160 Derio, Spain
[3]Ikerbasque, Basque Foundation for Science, 48013 Bilbao, Spain

Corresponding author: Irati Rasines (irati.rasines@tecnalia.com)

This work was supported in part by the Basque Government under Grant IT1726-22, in part by Humantech funded by European Union's Horizon Europe Research and Innovation Program under Grant 101058236, in part by Tracebot funded by European Union's H2020-EU.2.1.1. INDUSTRIAL LEADERSHIP Program under Grant 101017089, and in part by Smarthandle funded by HORIZON-CL4-2022-TWIN-TRANSITION-01 Program under Grant 101091792.

ABSTRACT For the development of autonomous robotic systems, Dynamic Movement Primitives (DMP) and Artificial Potential Fields (APF) are two well known techniques. DMPs are a reference algorithm in robotics for one shot learning as they enable learning complex movements and generating smooth trajectories, while APF are outstanding in navigation and obstacle avoidance tasks. By integrating DMPs and APF, the task automation capability can be significantly enhanced, as the precision of DMPs combined with the reactive nature of APF promises, in theory, adaptability and efficiency for the learning algorithm. Despite the numerous papers discussing and reviewing both techniques independently, there is a lack of an objective comparison of the investigations combining both approaches. This paper aims to provide such a comprehensive literature analysis, using a homogenized mathematical formulation. Moreover, a categorization based on their application scope, the robots used and their characteristics is provided. Finally, open challenges in the combination of DMP and APF are discussed, highlighting further works that are worth conducting for improving the integration of both approaches.

INDEX TERMS Artificial potential fields, dynamic movement primitives, learning from demonstration, obstacle avoidance, robot trajectory adaptation.

I. INTRODUCTION

Learning from Demonstration (LfD) is the research field that encompasses all investigations with the common objective of creating intelligent robotic systems capable of learning reliably from human demonstrations. In that field, Dynamic Movement Primitives (DMP) is a well established approach [1]. DMPs are known to be capable of learning and generalizing complex robot behaviors from a single demonstration, in a versatile manner, despite having a relatively simple formulation. Indeed, the "dynamical attractor" properties of DMPs are reminiscent of the spring-like properties of muscles and reflexes that simplify the programming and ensure the stability of biological movements [2], [3]. DMPs were first introduced two decades ago by Ijspeert et al. [4] and Schaal [5] in the context of humanoid robotics and have been used to learn and reproduce a wide range of manipulation tasks, from homelife tasks like ironing [6] up to complex assembly processes [7]. They have also been used for handling human-robot co-manipulation tasks, encoding the synchronized and coordinated motions of the two agents [8], [9]. Human assistance and rehabilitation

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

is yet another application field where DMPs have been used to generate trajectories for passive rehabilitation exercises for knee injuries [10], for ankle rehabilitation [11] or to model the assistance of an exoskeleton to healthy humans during lifting tasks [12], among others.

Generally speaking, the great potential of DMPs can be associated to several characteristics they provide:

- Generalization Capability: they can encapsulate diverse complex movements and allow robots to reproduce them.
- One Shot Learning: a single demonstration is enough to learn a desired behavior.
- Smoothness: they create smooth trajectories which are essential for good robot control.
- Adaptability: the initial and goal positions of the learned task as well as the execution speed can be easily tuned.

Nevertheless, Dynamic Movement Primitives also suffer from some well-known limitations, such as the complexity of tuning well the different parameters required by the model [13], or their limited capability to react to environmental changes in their basic formulation. Indeed, the DMP focuses on the reproduction of a demonstrated motion. It therefore does not provide tools for handling required deviations to avoid obstacles, which is a key capability for reactive motion planning.

There are different ways for achieving obstacle avoidance using DMPs, such as, machine learning, constrained probabilistic movement primitives or potential field-based approaches. Machine learning-based approaches generally use reinforcement learning to optimize the planned trajectories. Reward functions are set to handle obstacles [14], [15], to optimize the DMP parameters [16] or to avoid joint limits [17]. However, such optimization requires an additional learning phase, which eliminates the ''one shot learning'' capability of the DMPs. Probabilistic movement primitives (ProMPs), an extension of DMPs, focus on probabilistic properties and adaptation to new situations through constraints extracted from the environment [18], [19], [20]. Although efficient in obstacle avoidance tasks, constrained probabilistic movement primitives need an adequate formal expression of the constraints of the environment, information that is not always available, and is often computationally expensive to obtain. The potential field-based methods imply adding a coupling term to the DMP formulation. These potentials usually generate a repulsive force from the detected obstacles [3], [21] [22]. DMPs with artificial potential fields typically involve simpler computations compared to constrained ProMPs or reinforcement learning methods. Artificial Potential Fields (APF), first introduced in 1989 [23], are therefore a widely used tool in robotics applications such as path planning [24], collision and singularity avoidance in human-robot collaborative scenarios [25] or obstacle avoidance [26]. They have demonstrated to be good for obstacle avoidance while being relatively easy to implement even in real-time applications.

In light of the above, the combination of DMP and APF has been considered for taking advantage of the learning capability of the former, and the reactiveness of the latter [22]. This paper aims at providing an analysis of the works combining both approaches to enable systems to avoid obstacles. This paper presents several contributions that can be summarized as:

- Literature review on DMP and APF combinations, expressed in a common mathematical notation.
- Discussion of the limitations of the existing investigations and highlight of remaining open issues that could lead to new research directions.

The document is structured as follows. First, the mathematical representations of DMP and APF are introduced in Section II. Section III gathers the main approaches combining these two techniques, with a homogenized mathematical representation. Section IV analyses the existing unresolved issues within this research field. Finally, in Section V the main conclusions are presented.

II. MATHEMATICAL REPRESENTATIONS

In this section, the main concepts and characteristics of Dynamic Movement Primitives and Artificial Potential Fields are reviewed.

A. DYNAMIC MOVEMENT PRIMITIVES (DMPS)

DMPs can be described as a stable, second-order, nonlinear dynamical system [4], [27]. They are composed of a second-order linear dynamical system of type mass-spring damper, in which a point is attracted towards its goal. They also have a non-linear forcing term $f(x)$ that attracts the system towards the demonstration trajectory to be reproduced. Therefore, the forcing term is the element that has to be learned from a demonstration performed by the human.

Let us consider a uni-dimensional trajectory (y), with y_0 as the initial value, and g as final target value. The DMP is defined as the combination of a transformation system and a canonical system [28]. The transformation system represents the trajectory using a simple linear damped-spring system perturbed by a non-linear component as in equations (1-2):

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) - \alpha_z(\beta_z(g - y_0)x) + f(x) \quad (1)$$

$$\tau \dot{y} = z \quad (2)$$

where, α_z and β_z are the gains that define the evolution of the second-order system (the system is critically damped when $\alpha_z = 4\beta_z$) and z is the velocity of the system. The scalar τ is a temporal scaling factor used to tune the velocity of the execution.

The canonical system defines and drives the temporal evolution of the system, and it is expressed as:

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

where x is the phase variable, which combined with the constant α_x allows a re-parametrization of time. Finally, the

forcing term $f(x)$ is literally pushing the system towards the reference trajectory, while it progressively evolves from the initial to the goal position. It is usually defined as a combination of N non-linear Radial Basis Functions (RBF) (equations 4-7):

$$f(x) = \frac{\sum_{i=1}^{N} \omega_i \Psi_i(x)}{\sum_{i=1}^{N} \Psi_i(x)} x, \tag{4}$$

with weights ω_i, and being the basis function $\Psi_i(x)$:

$$\Psi_i(x) = e^{-h_i(x-c_i)^2} \tag{5}$$

where the centers c_i are defined as:

$$c_i = e^{-\alpha_x \frac{i-1}{N-1}} \tag{6}$$

and widths h_i defined as:

$$h_i = \frac{1}{(c_{i+1} - c_i)^2}, i = 0, 1, \ldots, N \tag{7}$$

$$h_N = h_{N-1}$$

Note that the DMP representation presented here follows [28], and permits avoiding the instability observed in the original formulation [4], [27] when initial and goal positions are very similar. Several works use such strategy [21], [22], [29]. This model is scalable to multiple dimensions, to represent robot joint configurations [30] or end-effector Cartesian pose [28]. In such cases, each degree of freedom (DoF) is represented independently with equations (1, 2) but with a common phase (equation (3)) of the canonical system to have them synchronized.

As in every other learning from demonstration approach, two phases compose a learning system based on DMP: the learning phase and the execution phase (also called rollout). In the learning phase, the desired forcing term is calculated using equation (1), and afterwards, with equation (4) and Locally Weighted Regression (LWR), the weights ω_i are calculated. In the execution phase, after setting a new goal position g, the forcing term is computed using equation (4) with the weights calculated in the previous phase. Once the forcing term is calculated, equation (1) is applied to obtain a new trajectory with the same dynamics as the original one. For this process, a single human demonstration is enough, making DMP a one-shot learning technique.

DMPs offer several options to adapt the learned trajectory to changes in the environment:

- Spatial Scaling: small changes of the starting or goal positions can be handled during the execution phase. Updated values just have to be injected into the transformation system (equation 1) during rollout, and the trajectory is automatically scaled accordingly.
- Temporal Scaling: the variable τ in the above formulas can be tuned to obtain an execution that is faster ($\tau > 1$) or slower ($\tau < 1$) than the original one.
- Phase Stopping: during the execution phase, DMPs have the ability to modulate the movement until a specific condition is met by holding the system's phase. This phase stopping can be used, for instance, to pause or slow down the rollout, until measured interaction forces get closer to the ones observed during the learning stage [7].
- Dynamic Adjustments: the trajectory generated in the execution phase can be modified by adding coupling terms to the transformation system (equation (1)). This can be used, for example, for reacting to and avoiding obstacles in the scene [22], [28].

This last option is particularly interesting to provide DMPs with reactivity during the execution, without requiring an extra learning. The coupling terms are frequently expressed as forces. The combination of DMPs and forces are described in Section III.

B. ARTIFICIAL POTENTIAL FIELDS (APF)

In general, Artificial Potential Field approaches involve leveraging attractive and repulsive forces to guide the robot safely trough the environment, pulling the robot towards goal positions while pushing it away from obstacles in the scene.

There are three types of artificial potential fields generating forces to the robot [31]: attractive, repulsive and hybrid potentials. An attractive potential creates an attractive force towards the goal. A repulsive potential tends to keep the system away from specific locations (typically obstacles). Finally, a hybrid potential is the superposition of the attractive potential with the repulsive potential.

Khatib [23] proposed the following static repulsive potential field that has been used over the years:

$$U(\mathbf{y}) = \begin{cases} \frac{\eta}{2}(\frac{1}{p(\mathbf{y})} - \frac{1}{p_0})^2 & if\ p(\mathbf{y}) \leq p_0 \\ 0 & otherwise \end{cases} \tag{8}$$

where p_0 is the radius of influence of the obstacle, $p(\mathbf{y})$ is the distance between the current position and the obstacle and η is a constant gain. The variable $\mathbf{y}$ ($\mathbb{R}^3$) encodes the robot's end-effector position in space. The force φ is deduced from the potential U with the following equation:

$$\varphi(\mathbf{y}, \mathbf{v}) = -\frac{\partial U(\mathbf{y}, \mathbf{v})}{\partial \mathbf{y}} \tag{9}$$

Such a potential field has several limitations. When working alone, it can get trapped in local minima and therefore be not capable of making the robot reach the goal. It can also suffer uncontrollable oscillations in the presence of obstacles and narrow passages [32]. In recent years alternative representation, including superquadratic static potentials [21], dynamic potentials [22] or steering-angle-based potentials [33] have been proposed in the literature to overcome the different limitations.

APFs have emerged as a well-established approach for providing reactivity to control schemes [21], [22], [23], [33]. They have been incorporated into various robotic paradigms such as subsumption architectures and motor schema for robot navigation [34], behavior trees (encapsulated as

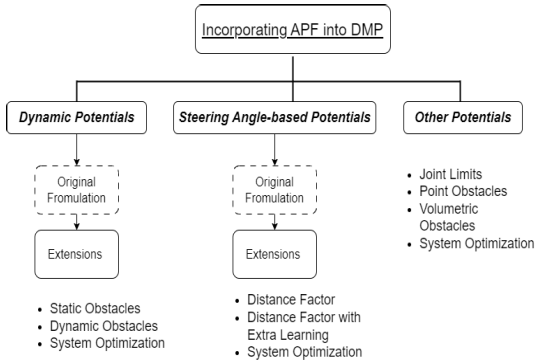

FIGURE 1. Structure of the section III, introducing Artificial Potential Fields into DMP formulation.

nodes representing specific behaviors or decision-making processes) [35], reinforcement learning (used to define the reward structure) [36], soft computing techniques (combined with fuzzy logic rules) [37], [38], [39], or in learning from demonstration scenarios (to handle obstacles in scene without a need to gather more demonstrations or retrain the model) [22], [28]. In this review, we focus on this last option, with the objective of allowing Dynamic Movement Primitives to react to environmental changes. The next section describes how potential fields can be incorporated into DMPs, and categorizes articles proposing such approach.

III. INCORPORATING APFS INTO DMPS

The combination of APFs with DMPs takes place during the execution phase; the learning stage is not impacted. The transformation system is extended with a repulsive force φ deduced from the potential U. The transformation system is now:

$$\tau\dot{z} = \alpha_z(\beta_z(\mathbf{g}-\mathbf{y}) - z) - \alpha_z(\beta_z(\mathbf{g}-\mathbf{y_0})x) + f(x) + \varphi \quad (10)$$

Depending on the implementation, the repulsive term depends only on the robot's position $\mathbf{y}$, or also contemplates the system's velocity $\mathbf{v}$. The usage of $\varphi(\mathbf{y}, \mathbf{v})$ or $\varphi(\mathbf{y})$ in the following sections will indicate whether the potential depends on the velocity (dynamic potential) or not (static potential), respectively. This section gives a comprehensive overview of the research done combining both approaches.

The combination of APF and DMP was introduced by Park et al. [22] in 2008 (dynamic potentials) and Hoffman et al. in 2009 [28] (steering angle-based potentials). They pioneered the two primary research streams on this field which have been adopted and extended by other researchers over time. The following subsections present these two methodologies, with an effort to harmonize the formulation for ease of comparison. In addition, a third subsection will present additional works that do not rely on these two initial approaches. This classification leads to the creation of the taxonomy shown in Figure 1, which also describes the structure of this section. The main characteristics of the works reviewed are summarized in Table 1.

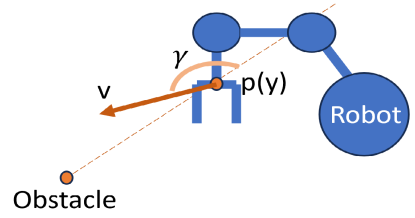

FIGURE 2. Illustration of the angle representation between a point obstacle and robot.

A. DMP COMBINED WITH DYNAMIC POTENTIALS

1) ORIGINAL FORMULATION

Park et al. [22] proposed using a potential field based on both system position and velocity. That potential field is represented as:

$$U(\mathbf{y}, \mathbf{v}) = \begin{cases} \lambda(-cos\gamma)^{\beta \frac{\|\mathbf{v}\|}{p(\mathbf{y})}} & if \ \ \frac{\pi}{2} \leq \gamma \leq \pi \\ 0 & if \ \ 0 \leq \gamma \leq \frac{\pi}{2} \end{cases} \quad (11)$$

where λ and β are constant gains. γ is the angle between the current velocity $\mathbf{v}$ and the shortest line between the robot and the obstacle $p(\mathbf{y})$ (see Figure 2).

In this modeling, the consideration of the velocity enables improving the smoothness of the reaction in comparison with the standard potential (eq. 8) that only relies on the system position.

This potential was tested both in simulation and real environment by placing a spherical obstacle in the middle of a pick-and-place trajectory. The trajectories obtained are smooth even nearby the obstacle. Nevertheless, local minima could still appear [22]. Such a potential has also been used in combination with DMP in [40] to control a 5 DoF robotic arm while avoiding a ball in the scene. Several extensions of this formulation have been proposed in the literature: adding an isopotential function to obtain smoother trajectories near obstacles, with both static [21], [41] and dynamic obstacles [42], and adding an extra optimization layer to improve overall results [16], [42], [43]. These approaches are detailed below.

2) STATIC OBSTACLES

The use of an isopotential function in the potential definition aims at smoothing the reaction to static volumetric obstacles [21], [41]. The isopotential function C is incorporated in the potential U as follows:

$$U(\mathbf{y}, \mathbf{v}) = \begin{cases} \lambda(-cos\gamma)^{\beta \frac{\|\mathbf{v}\|}{C^{\eta}(\mathbf{y})}} & if \ \ \frac{\pi}{2} \leq \gamma \leq \pi \\ 0 & if \ \ 0 \leq \gamma \leq \frac{\pi}{2} \end{cases} \quad (12)$$

where the exponent η is an additional gain. This isopotential was first proposed for elliptic obstacles in [21]:

$$C(\mathbf{y}) = \sum_{j=1}^{d} \left(\frac{y_j - \hat{y}_j}{l_j} \right)^{2n_j} - 1 \quad (13)$$

with $\hat{y}_j$ being the centers of the ellipsoid, l_j the axis of the ellipsoid, d the dimension number, and n_j the exponential parameters of the isopotential. By replacing $p(\mathbf{y})$ in the original formulation (equation 11) with $C(\mathbf{y})$, the new formulation is able to handle volumetric obstacles instead of only point-like obstacles.

In [21], a static superquadratic potential function was also proposed to compare the obstacle avoidance capabilities of static and dynamic superquadratic potentials. It does not depend on the system velocity and takes the form:

$$U(\mathbf{y}) = \frac{\alpha e^{-\eta C(\mathbf{y})}}{C(\mathbf{y})} \tag{14}$$

being α, η gain parameters.

Both approaches were validated with synthetic data using different types of obstacles (ellipsoid, circular and non-convex obstacles). For both the static and dynamic potentials, the generated trajectories were smooth movements around obstacles. Compared to the trajectories created with the original formulation [22], the isopotential-based method was able to generate trajectories closer to the human demonstrations. These potential fields were also tested on three real robots: a Panda and a DaVinci for placing rings into pegs, and a Youbot robot for corridor navigation. The real experimentation demonstrated deviations significantly higher than in simulation when using dynamic potentials, and very dependent on the obstacle's position.

Another application shows how these potentials can create safe and human-like trajectory planning for self-driving cars [41]. For the obstacle avoidance term, the potential field presented in equation (12) is used, being $C(\mathbf{y})$ in this case the isopotential function of an elliptic obstacle in 2D, which represents a particular case of equation (13):

$$C(\mathbf{y}) = \left(\frac{y_1 - o_1}{l_{1y}}\right)^2 + \left(\frac{y_2 - o_2}{l_{2y}}\right)^2 - 1 \tag{15}$$

where (y_1, y_2) are the position of the car, (o_1, o_2) the position of the obstacle vehicle and (l_{1y}, l_{2y}) are the lengths of the axis of the ellipse, i.e., the size of the car to avoid.

In order to prevent the car from veering off the road, a dynamic coupling term is added, which is represented as:

$$\varphi(\mathbf{y}, \mathbf{v}) = \frac{\mathbf{v}}{(p(\mathbf{y}) - W_{car})^{\eta}} \tag{16}$$

where $\mathbf{v}$ is the velocity of the vehicle, $p(\mathbf{y})$ is the distance to the road border, W_{car} represents the width of the car and η is a gain parameter. This repulsive force is always normal to the road boundaries. It increases both with the increase of the relative velocity towards the wall and the decrease of the distance to it.

As validation tests, three scenarios were proposed, where the trajectories were generated for a vehicle circulating in a road with other cars, without colliding or going off the road. In all the experiments the obtained trajectory was successful. As there was not a concrete positional goal, the system did not suffer from oscillatory behaviors, and convergence issues did not appear. In fact, the obstacles were always positioned far from the goal so that the system could converge.

3) DYNAMIC OBSTACLES

The consideration of dynamic obstacles requires particular care to avoid possible excessive avoidance behaviors due to abrupt object motions. An approach to alleviate this consists of incorporating an additional layer of model predictive control using Kalman filters, as in [42]. The Kalman filters are used to predict the future positions of the obstacles, and the model predictive control layer adjusts the repulsive forces accordingly.

In order to incorporate the velocity of the obstacle, equation (12) is modified to:

$$U(\mathbf{y}, \mathbf{v}) = \begin{cases} \lambda(-cos\gamma)^{\beta \frac{\|\mathbf{v}-\dot{\mathbf{o}}\|}{C^{\eta}(\mathbf{y})}} & if \quad \frac{\pi}{2} \leq \gamma \leq \pi \\ 0 & if \quad 0 \leq \gamma \leq \frac{\pi}{2} \end{cases} \tag{17}$$

where $\dot{o}$ is the velocity of the obstacle, and γ is the angle between the relative velocity $\|\mathbf{v} - \dot{\mathbf{o}}\|$ and the direction between the end-effector's position and the obstacle. $C(\mathbf{y})$ is the isopotential of equation (13) for a sphere and η a positive constant. The authors first evaluated their proposal in simulation and then with a Kinova MOVO robotic arm, using a moving sphere as an obstacle. In both cases, a fast convergence to the demonstration was obtained.

4) SYSTEM OPTIMIZATION

In addition, several works have proposed involving different techniques to optimize the system's behavior instead of modifying the original potential formulation. In [42], cited in the previous subsection, the incorporation the model predictive control layer enables optimizing the repulsive force generated by the potential. Similarly, the original potential has also been combined with policy improvement through black-box optimization with an adaptive covariance matrix (PIBB-CMA) method to ensure that the robot arm passes the points generated by the path planner [43]. Moreover, in order to enhance the formulation's adaptability and efficiency in generating the desired movements, Li et al. [16] used reinforcement learning to tune the DMP parameters and the gain λ in equation (11). It was tested on a real robot for a simple point obstacle avoidance.

B. DMP COMBINED WITH STEERING ANGLE-BASED POTENTIALS

In the previous collections of approaches, the new coupling term for the DMP was derived from a dynamic potential to take into account system's velocity. Another category of DMP-APF combination relies on coupling terms derived from steering angle-based potentials.

1) ORIGINAL FORMULATION

The definition of the steering angle-based potentials approach can be attributed to Hoffmann et al. in 2009, who introduced

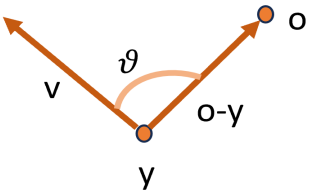

FIGURE 3. Illustration of the angle representation of equation (19).

a novel perturbation inspired by biological dynamical systems [28]:

$$\varphi(\mathbf{y}, \mathbf{v}) = \alpha \mathbf{R}\mathbf{v}\vartheta e^{-\beta\vartheta} \tag{18}$$

where α and β are constant gains of the potential, $\mathbf{R}$ is the rotation matrix between $\frac{\pi}{2}$ and the axis $\mathbf{r} = (\mathbf{o} - \mathbf{y}) \times \mathbf{v}$ being $\mathbf{o}$ the position of the obstacle and $\mathbf{v}$ the robot's velocity. The steering angle ϑ is defined as (see also figure 3):

$$\vartheta = arcos(\frac{(\mathbf{o} - \mathbf{y})^T \dot{\mathbf{v}}}{\|\mathbf{o} - \mathbf{y}\| \, \|\dot{\mathbf{v}}\|}) \tag{19}$$

This term is inspired by the steering angle introduced by Fajen and Warren [33] which proposed a similar equation to describe the way humans avoid obstacles.

A remarkable advantage of the steering angle approach, compared to the ones presented in previous section, is that it guarantees the convergence to the goal, as it is proven in [28]. Nevertheless, as equation (18) does not depend on the distance to the obstacles, the same relevance is given to close and far obstacles which may lead to oscillatory behaviors in the execution phase [44].

Several applications of this potential can be found in the literature, from pouring tasks [29], [45], writing [46], pushing and sliding [47], to mobile navigation [48]. Chi et al. [49] used the same perturbation term to handle cup placement on a table using a robotic arm mounted on a wheelchair for a service-robot application. Besides, this potential has been referenced in Pastor et al. [50] as a reference potential inside a review of how DMPs can contribute to learning motor skills while adapting to the environment and in [27] as an example of how to modulate the dynamical system of the DMPs.

In addition, several works have proposed extensions of the basic model. On one hand, there are approaches that introduce the distance to the obstacle inside the coupling term equation to avoid giving the same relevance to far and close objects [51]. Some of these approaches add an extra optimization layer to the previous distance dependency to enhance the results [52], [53] while others use a human as an online teacher for such approaches [54]. On the other hand, there are the approaches that purely optimize the process via parameter tuning [55] or via fuzzy logic [38].

2) DISTANCE FACTOR

The original coupling term can be modified to hold information about the distance to the obstacle, in order to adapt the potential's response to far or close obstacles. Zhai et al. [51] presented an extension of the coupling term proposed in [28] for static objects adding the distance to the object as expressed in the following equation:

$$\begin{aligned}\varphi(\mathbf{y}, \mathbf{v}) = \mathbf{R}\mathbf{v}\alpha \frac{\vartheta}{|\mathbf{d}|} e^{(-\mu|\mathbf{d}| - \beta|\vartheta|)} \\ - \mathbf{R}\mathbf{v}\alpha\Delta\theta|\Delta\mathbf{y}|e^{(\mu'|\mathbf{d}| + \rho|\Delta\theta|)}\end{aligned} \tag{20}$$

where $|\mathbf{d}|$ is the real time distance between the end-effector and the obstacle and μ, μ', β and ρ are positive constants.

The first term of eq. (20) introduces with $|\mathbf{d}|$ a dependency to the obstacle distance, which was not present in eq. (18). Nevertheless, it still generates a high deviation away from the demonstrated trajectory. The second component of the equation (20) is thus introduced to mitigate this behavior by balancing the repulsive force generated by the first term. In this second term, $\Delta\theta$ is the angle between the current velocity direction and the desired velocity direction. $\Delta\mathbf{y}$ represents the distance between the current trajectory and the desired trajectory. Thus, the second term of equation (20) is proportional to $\Delta\mathbf{y}$, and to $|\mathbf{d}|$. This coupling term avoided in their experiments the jitters and oscillations that would appear otherwise in the generated trajectory.

This potential can also be used to avoid dynamic obstacles using a similar formulation. Instead of using the robot's velocity $\mathbf{v}$, the relative velocity between the robot and the obstacle is used ($\mathbf{v_{relative}} = \mathbf{v} - \dot{\mathbf{o}}$) in equation (20) [51]. This approach was shown to generate curves very similar to the one obtained without obstacles, to guarantee the minimum loss of free space. In their work, the authors compared their outcomes and the ones with the proposal in [21] in terms of distance to the desired curve. The acceleration of the obtained curves was also calculated, to visually compare the smoothness in a laboratory environment with a real Baxter robot and obstacles in different positions. They demonstrated that the obtained trajectories do not collide with the obstacles and are smooth.

3) DISTANCE FACTOR WITH EXTRA LEARNING

The outcomes of distance factor based approaches can be improved by integrating additional learning layers to the system [52], [53]. For instance, to address the management of larger obstacles, in [52], the repulsion was defined as a weighted combination of three components:

$$\varphi(\mathbf{y}, \mathbf{v}) = \alpha_1 \mathbf{R}\mathbf{v}\Phi_1 + \alpha_2 \mathbf{R}\mathbf{v}\Phi_2 + \alpha_3 \mathbf{R}\mathbf{v}\Phi_3 \tag{21}$$

where Φ_1, Φ_2 and Φ_3 encode the steering angle potential generated respectively by the obstacle's center, the closest surface point, and a uniform repulsion to the obstacle. They were defined as follows:

$$\begin{aligned}\Phi_1 &= \vartheta e^{-\beta\vartheta} e^{-kd} \\ \Phi_2 &= \vartheta e^{-\beta\vartheta_p} e^{-kd_p} \\ \Phi_3 &= e^{-kd}\end{aligned} \tag{22}$$

where d is the distance from the robot end-effector to the obstacle center, d_p is the distance from the end-effector to the nearest point in the surface and k a constant gain. Φ_3 was

added to avoid null forces when heading directly towards the obstacle, i.e. when $\vartheta = 0$. When this situation happens, the system is considered to be in a so-called dead zone. In order to optimize the system's behavior, the authors also propose to learn the weights of equation (21) by deducing the target coupling term $\varphi(\mathbf{y}, \mathbf{v})$ from human demonstration performing the obstacle avoidance task. The proposed approach was validated in simulation and with a real KUKA arm [52]. The extra learning required a significant number of trials to be recorded (more than a thousand in their application).

Pairet et al. [53] proposed a different solution to handle the so-called dead zones for point obstacles, composed by a combination of a repulsive and attractive coupling terms:

$$\varphi(\mathbf{y}, \mathbf{v}) = \sum_i C^i_{rep} + C^i_{attr} \tag{23}$$

The repulsive term is a reformulation of equation (18):

$$C_{rep} = \mathbf{R}\mathbf{v}\alpha sign(\vartheta) e^{(-\frac{\vartheta^2}{\psi^2})} e^{(-\kappa d^2)} \tag{24}$$

where *sign* function returns the sign of ϑ, $\alpha sign(\vartheta) e^{-\frac{\vartheta^2}{\psi^2}}$ shapes the absolute change of steering angle as a zero-mean Gaussian-bell function to be more reactive when the system becomes aligned with the obstacle and $e^{-\kappa d^2}$ regulates the coupling term effect when the distance to the obstacle d increases or decreases. ψ is the standard deviation of the distribution.

The additional attractive coupling term is proposed to bring the mobile system towards a preferred direction. It is defined as:

$$C_{attr} = \mathbf{R}'\mathbf{v}\alpha\hat{\vartheta} e^{1+\kappa d^2} \tag{25}$$

where R' is a $\pi/2$ rotation matrix around the vector $(\mathbf{v} \times \mathbf{v_d})$, being $\mathbf{v_d}$ the desired velocity, and α and κ the same constants as in C_{rep}.

In this case, the extra learning layer is added to learn the parameters of the coupling term α, ψ and κ as a multiple target regression problem from demonstrations of the avoidance task. The proposed framework is evaluated in simulation and with a real Franka Emika robot [53] where the test bed consists of obstacles interrupting a straight trajectory towards the goal.

4) SYSTEM OPTIMIZATION

In the pursuit of optimizing system performance, researchers have explored diverse methodologies, each offering unique solutions to different problems. In a conventional optimization scenario, Duan et al. [55] use reinforcement learning techniques to learn the hyperparamenters of the potential fields to determine the strength of the APF. In this case, the potential proposed in [28] is combined with constrained dynamic movement primitives to avoid robot's joints limits and evaluates in a iCub real robot. The task in hand is to transport a sponge from one point to another mimicking the trajectory that a human demonstrator had made, but with an obstacle in the middle.

Fuzzy logic is a valuable tool for system optimization too, offering a framework to model and reason with imprecise or uncertain information. With the objective of better handling the adaptation to changing goals and asymmetric obstacle management, Sharma et al. incorporate fuzzy logic to the potential representation [38]. The proposed approach handles both static and moving objects, and it was validated using a nonholonomic mobile robot.

Furthermore, acknowledging the exceptional optimization capacities inherent in humans, an alternative strategy incorporates human-inclusive optimization. Based on this assumption, in [54] the learned trajectory is adjusted according to the modifications suggested by the human. This change is performed introducing in the transformation system the following coupling term:

$$\varphi(\mathbf{y}, \mathbf{v}) = \alpha s(\|\mathbf{m} - \mathbf{y}\|) e^{-\beta\vartheta} \mathbf{di} \tag{26}$$

where $\mathbf{y}$ is the end-effector position, $\mathbf{m}$ is the human's hand position, $\mathbf{di}$ is the pointing gesture direction (or the direction of the perturbation), α and β are constant gains. ϑ is defined as in equation (19). The term (s) is linked to the position in Cartesian space when the potential starts pushing:

$$s(\mathbf{y}) = \frac{1}{1 + e^{\eta(\mathbf{y} - \mathbf{y_m})}} \tag{27}$$

being η an scaling factor and $\mathbf{y_m}$ the distance where the potential field starts pushing. This approach was tested on a real robot for a wiping task, where the human coach instructed the robot using gestures to adapt its trajectory with satisfactory results [54].

C. ALTERNATIVE APPROACHES

In this section, works that use APF combined with DMP that do not belong to the two previous categories (DMP combined with dynamic potentials III.A, and combined with steering angle-based potentials III.B) are presented. The applications of these alternative approaches can be divided in the ones centered in avoiding joint limits [17], point obstacles [56], [57], volumetric obstacles [44] or walls [58]. Finally, as in previous sections, there is also a system optimization proposal [59].

1) AVOID JOINT LIMITS

The combination of DMP and APF is not limited to the avoidance of obstacles and it can also be used for avoiding joint limits as in [17]. Here, an additional force is designed to push the trajectory away from the angular limits. They model the force to avoid robot's joint limits as:

$$\varphi(\theta) = \begin{cases} 0 : |\theta| < \theta_t \\ -10|\omega|tan(\frac{\theta_t}{\theta_{max}}) : \theta_t \leq |\theta| \ \wedge \ |tan(\frac{\theta_t}{\theta_{max}})| < 10 \\ -10|\omega sign(\theta)| : |tan(\frac{\theta_t}{\theta_{max}})| \geq 10 \ \wedge \ |\theta| < \theta_{max} \\ -|\omega_{\mathbf{max}}|\theta : |\theta| > \theta_{max} \end{cases} \tag{28}$$

being θ_{max} the angular limits of the robot, θ_t the threshold angle at which the potential field start working and ω the weighting factors of the potential. The formulation ensures that within a small amount of time-steps, the trajectory that exceeds the set bounds comes back quickly to the desired range to minimize the loss of free space. In this way, the learning also is extended to include the mechanical constraints of the robot.

2) POINT OBSTACLES

Regarding point obstacle avoidance, an original distance based coupling term is presented in Tan et al. [57]. The equation of the forcing term is presented as:

$$\varphi = \frac{\beta}{(d-r)^2} \tag{29}$$

where β is constant, d is the distance between the obstacle and the end-effector and r is the radius of the obstacle. The originality of this approach is that the goal g of the transformation system is replaced by a temporal goal $g_{temp} = g + g_{impedance}$, being $g_{impedance} = \alpha\varphi$ when the current state in the execution phase of the DMP is under the influence of the obstacle (α is a constant gain). When a position generated by the DMP enters into the area of action of the obstacle, the goal state g is moved to a virtual state g_{temp} to calculate the position in the next time-step. When the robot goes out of the influence of the object, the original goal g is used again. Their method is capable of handling better than other techniques high speed control. This implementation was only tested in simulation, generating different trajectories with artificial point obstacles in the middle of the trajectory.

In real life scenarios, not only the contacts between end-effector and obstacles should be avoided, but the whole system must be collision free. In [56], the presented system was composed by a quadrucopter that carries a robotic arm, so not only the aerial robot should avoid the obstacle but also the robot arm, meaning that the link collisions should also be avoided. In this research work, the original artificial potential field proposed by Khatib [23] was combined with DMPs. The learning phase consisted of learning a DMP for the trajectory without obstacles of each link of the robot arm, plus another one for the quadrucopter. In the execution phase, only tested in simulation, they obtained satisfactory results.

3) VOLUMETRIC OBSTACLES

Transitioning now to the domain of volumetric obstacles, the general superquadratic static potential function of equation (14) was used in different research works [44], [60]. In [44] equation (14) was proposed for the first time and compared to the steering angle approach of [28]. Although the superquadratic potential approach offers computational efficiency over the steering angle method, the latter ensures convergence to the goal, a guarantee not provided by superquadratic potentials. The proposed potential was tested in a laboratory environment where the robot should avoid known obstacles (cylindrical pegs) in scene, obtaining satisfactory results [44]. The same superquadratic potential was used in [60] for obstacle avoidance in pick and place tasks in simulated environments using a UR5 robot.

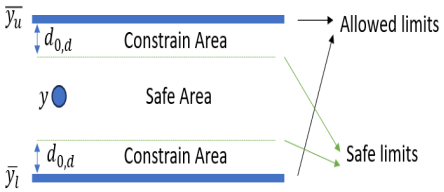

FIGURE 4. Graphical representation of the wall obstacle and constrained area in [58].

4) WALL OBSTACLES

Moving forward, the recent research now centers on wall-type obstacles, with a focus on how they can be handled with new coupling terms, as in [58]. The proposed coupling term in [58] is expressed mathematically as:

$$\varphi(y) = \begin{cases} k\left(\frac{1}{(y-\bar{y_l})} - \frac{1}{d_{0,d}}\right)\frac{1}{(y-\bar{y_l})^2} e^{\left(\frac{1}{(y-\bar{y_l})} - \frac{1}{d_{0,d}}\right)^2} \\ y - \bar{y_l} < d_{0,d} \\ k\left(\frac{1}{d_{0,d}} - \frac{1}{(\bar{y_u}-y)}\right)\frac{1}{(\bar{y_u}-y)^2} e^{\left(\frac{1}{d_{0,d}} - \frac{1}{(\bar{y_u}-y)}\right)^2} \\ \bar{y_u} - y < d_{0,d} \\ 0\text{otherwise} \end{cases} \tag{30}$$

Figure 4 shows a graphical representation of their physical paradigm linked with equation (30). y is the position of the system, $d_{0,d}$ the width of the constrain area from the allowed limits where the potential should get activated, $\bar{y_l}$ the lower allowed limit, $\bar{y_u}$ the upper allowed limit and k a gain constant. Therefore, there is safe area to move on, which is to a given distance from the allowed limits ($d_{0,d}$), and a constrained area where the coupling term gets activated. The approach was tested it in four different situations of a drawing task, where a real robot was able to adapt its movements and avoid satisfactorily going outside the safe margins.

5) SYSTEM OPTIMIZATION

Unlike in the previously presented approaches where a potential field is mathematically described to generate a coupling term, the coupling terms can also be gathered from human demonstrations. In [59], a neural network is trained using human demonstrations to create a coupling term model instead of mathematically calculating it. For this, demonstrations without obstacles are used to learn the forcing term as in the classic DMP approach. The demonstrations with obstacles are then used to calculate the coupling term value, by computing the difference between the original forcing term and the forcing term in the presence of obstacles. Once the coupling terms are calculated for all the demonstrations with obstacles, a neural network is trained. In real applications, the output of the network is

TABLE 1. Classification of the bibliography according their characteristics, application and robot used. A category represents dynamic potentials, B steering angle based potentials and C alternative approaches. Y stands for yes and N stands for no.

Category	Ref	Year	Obstacle Type	Potential Type	Distance Dependent	Oscillation Mitigation	Convergence Guaranteed	Smooth Trajectory	Application	Robot
A	[22]	2008	Point	Dynamic	Y	N	N	Y	Pick & place	Sarcos Follower
A	[40]	2011	Point	Dynamic	Y	N	N	Y	Avoid static obstacle	5DOFs robot
A	[21]	2021	Volume	Static Dynamic	Y	Improved	N	Y	Pick & place	Youbot, Davinci, Panda
A	[16]	2021	Point	Dynamic	Y	N	N	Y	Avoid static obstacle	7-DOF robot
A	[41]	2023	Point & Wall	Dynamic	Y	Improved	–	Y	Autonomous driving avoiding cars	None
A	[42]	2023	Volume	Dynamic	Y	N	N	Y	Dynamic obstacle avoidance	Kinova MOVO.
A	[43]	2023	Volume	Dynamic	Y	N	N	Y	Pick, scan and place	Unnamed
B	[28]	2009	Point	Dynamic	N	N	Y	Y	Pick & place	Sarcos Follower
B	[29]	2009	Point	Dynamic	N	N	Y	Y	Pouring task	Sarcos Leader and Follower
B	[50]	2013	Point	Dynamic	N	N	Y	Y	-	-
B	[52]	2014	Volume	Dynamic	Y	N	Y	Y	Avoid static obstacle	Simulated Kuka LWR
B	[54]	2016	Point	Dynamic	Y	N	Y	Y	Coaching the robot	Kuka LWR4
B	[48]	2017	Point	Dynamic	N	N	Y	Y	Obstacle in path	Mobile robot
B	[45]	2019	Point	Dynamic	N	N	Y	Y	Pouring task	Kuka LWR4
B	[49]	2019	Point	Dynamic	N	N	Y	Y	Place cup on table with a robot on a wheelchair	JACO robot
B	[38]	2019	Point	Dynamic	N	N	Y	Y	Pick & place	Mobile robot
B	[53]	2019	Point	Dynamic	Y	N	Y	Y	Avoid static obstacle	Franka Emika robot
B	[55]	2020	Point	Dynamic	N	N	Y	Y	Pick & place	iCub robot
B	[46]	2021	Point	Dynamic	N	N	Y	Y	Writing task	UR10 robot
B	[51]	2022	Point	Dynamic	Y	Improved	Y	Y	Avoid static & dynamic obstacle	Baxter robot
B	[47]	2024	Point	Dynamic	N	N	Y	Y	Pusher-Slider with obstacle	Joystick
C	[57]	2011	Point	Static	Y	N	Y	Y	Avoid obstacle in simulated scene	Simulation
C	[17]	2017	–	Static	N	-	-	Y	Limit robot's joint motion	ABB IRB120 simulator
C	[56]	2017	Point	Static	Y	-	-	Y	Link collision and obstacle avoidance	Quadrotor with robotic arm
C	[59]	2017	Volume	-	-	-	Y	Y	Obstacle avoidance	Barrett robot
C	[44]	2019	Volume	Static	Y	Improved	N	Y	Obstacle in path	Panda robot
C	[60]	2020	Volume	Static	Y	Improved	N	Y	Pick& place	Simulated UR5
C	[58]	2024	Wall	Static	Y	-	Y	Y	Drawing task	Arm prototype

stabilized using filtering rules to to ensure an adequate robot behavior. In contrast to traditional coupling term calculation methods that need manual hyperparameter tuning, this approach alleviates the need for manual intervention. However, this approach requires relying more heavily on gathering human demonstrations to ensure effective training (they used 21 recordings with no obstacle and 600 with obstacles in different configurations).

D. APPLICATIONS COMBINING DMP AND APF

Table 1 serves as a comprehensive compilation of all the research works cited in section III. Organized by category, it offers a structured overview of the diverse studies discussed. Each entry in the table outlines the different characteristics of the research, facilitating the comparison between methods. Inside each category, works are chronologically ordered and categorized based on their potential type, characteristics, application and robot.

The table provides the following information:

- *Category*: states the group the presented potential belongs to: *A*, dynamic potentials, *B* steering angle and *C* alternative approaches.
- *Year*: year of publication.
- *Obstacle Type*: *point* for unique point obstacles or *volume* for volumetric obstacles, *walls* for continuous wall shape obstacles.

- *Potential Type*: *dynamic* potential makes reference to potentials that depend on the velocity, and *static* potentials are those which only relay on position.
- *Distance Dependent*: specifies when the formulation directly depends on the distance to the obstacle with yes (*Y*) or no (*N*).
- *Oscillation Mitigation*: states whether the oscillatory behavior is explicitly avoided. This can be yes (*Y*), no (*N*) or Improved if they compare themselves to other methods and improve the other's results.
- *Convergence Guaranteed*: makes reference to whether the goal reaching is guaranteed (*Y*) or not (*N*).
- *Smooth Trajectory*: will only be yes (*Y*) if the trajectories generated with the DMP & APF are smooth in the performed experiments.
- *Application*: the chosen task to validate the strategy.
- *Robot*: in case a real robot has been used, which one has it been.

In cases where the use case does not allow directly addressing any column, a '–' symbol is displayed.

1) DISCUSSION

Following the table's presentation, this discussion explores its classification from different viewpoints, aiming to offer a nuanced understanding of its implications and complexities, providing insights into its multifaceted nature.

According to the data presented in the table, three distinct categories have emerged: those founded on dynamic potentials [22], those rooted in steering angle [28], and alternative approaches that deviate from these established categories. Upon closer examination of the distribution of papers across these categories, it becomes apparent that the category based on steering angle [28] possesses the highest frequency of occurrence, indicating its prevalence within the field. However, an analysis of temporal trends reveals that the three categories have been utilized with relative uniformity across the last five years. One of the main reasons of the predominance of steering angle-based approaches is that the convergence to the goal is guaranteed when generating obstacle avoidance trajectories.

Regarding the obstacles discussed in the literature analysis, three distinct types are commonly addressed: point obstacles, volumetric obstacles, and walls. Point obstacles are frequently employed to simplify computational processes in the proposed research, offering a concise scenario focused on the exerted forces more than in the design of the object descriptors. Volumetric obstacle avoidance, on the other hand, often involves using superquadratic equations to describe objects or utilizing points on the surface of obstacles instead of modeling the entire obstacle volume. Generally these type of obstacles are handled in category A. Despite the prevalence of point and volumetric obstacles, there is comparatively less investigation into methods aimed at maintaining robots or vehicles within a desired motion zone. This disparity underscores a potential area for further exploration and innovation within the field.

Focusing on the application or use case, very similar scenarios are presented across all categories. This makes it challenging to determine which approach is most suitable for particular tasks or contexts.

Concerning the aspect of smoothness of the generated curve, none of the presented category can be said to outperform the others, as there is a lack of an objective analysis to compare the various methods proposed in the literature. Typically, all presented methods generate stylish smooth trajectory without offering a quantifiable benchmark for comparison between methods. But only a very few compare themselves with other state of the art techniques and this comparison is just done with graphical representations. In the analyzed literature there is no objective quantification of the smoothness of the generated curves.

A similar situation arises when analyzing the computational cost of each approach. While optimization approaches may offer superior effectiveness compared to manual parameter tuning for example, they require more computational cost and more human resources to gather all the additional demonstrations of the task in hand. However, in highly constrained environments where changes are minimal, the efforts invested in recording extra demonstrations and training models to optimize the coupling terms could be deemed worthwhile. In any case, any modification or improvement made to the original potentials should not only demonstrate the enhanced performance in terms of effectiveness in the obstacle avoidance task but also should evaluate computational trade-offs.

IV. OPEN CHALLENGES FOR COMBINING DMPS AND APF

Combining both DMP and APF can enhance the power of both methodologies, bringing together the generalization capability of the DMP with the reactivity to obstacles of the APF. But as in any motion primitive representation, there are also inherent limitations and unresolved problems. Based on the literature review analysis, this section highlights persisting open challenges, suggesting possible ways for future advancements in the field.

A. CONVERGENCE TO THE GOAL

As of today, there is not an optimal way to guarantee goal convergence while avoiding local minima and oscillatory behaviors. On the one hand, the works based on the first dynamic potential are not able to guarantee convergence. On the other hand, the original steering-angle formulation ensures convergence but as the steering angle does not depend on the distance to the obstacle, the same importance is given to close or far obstacles, which leads to oscillatory behaviors. Adding a dependency to the obstacle distance in the steering angle formulation is an interesting proposed solution but cannot be considered as a generalized solution due to the limited complexity of use-cases in which it has been used. It would be interesting to test them in a more complex scenario with more than one obstacle in the scene, and some of them close to each other to ensure the viability of the method in real world applications.

B. OBSTACLE TYPE

Regarding the obstacle type, most of the works focus on either point-like or volumetric obstacles, static or in movement. Only a few research works have dealt with continuous obstacles. These works do not have convergence problems since they do not have a goal close to any obstacle, so it would be interesting to push the algorithms and test them in such scenarios. Besides, the scope of the described works for wall-type obstacles is limited to two dimensional scenarios. It will be interesting, and challenging, to transfer their proposals to a robotic 3D scenario.

C. ORIENTATION DMPS

Another remarkable challenge is the necessity of incorporating orientation into the DMP implementation to be able to automate realistic tasks where certain orientations are ensured. Imagine a use case where the task to automate is to pour some liquid samples or enter a cavity to disassemble two pieces. Here, a correct management of the orientation constraints and obstacle- or wall-avoidance is essential for a successful execution. All the presented implementations focus on the positional versions of DMP even though orientation is critical to handle many real life applications.

D. SMOOTHNESS

Up to now, there is no quantification regarding the smoothness of the generated trajectories. All these works present how the generated curves are valid due to their apparent reduction in oscillations, how they avoid the obstacles or in the best case scenario how close they are to the original curve without obstacles. But there is a lack of an objective analysis of the smoothness of the resulting motion. Therefore, it would be interesting in future publications, to add an objective evaluation and quantification of the smoothness of the generated curves and the comparison of the results generated with different methods.

E. SYSTEM OPTIMIZATION

An efficient hyperparameter selection in the combination of DMP and APF or obtaining the most suitable coupling term is considered another challenge. Generally, the tuning of DMP and APF parameters relies on experts, who draw on their intuitive insights and practical experience instead of a deterministic methodology. As mentioned in this overview, some approaches propose using optimization techniques to select the optimal parameters for a given task. Other approaches, directly learn the desired coupling terms from human demonstrations using different machine learning techniques. However, although this might enhance the results for a particular task, this extra learning requirement compromises the intrinsic one-shot learning capability of the DMP and APF combination, which is essential in some scenarios (e.g. in human teaching robots in hazardous or distant environments). Here, the capability to teach a robot in a single demonstration can minimize the mental burden of the human teacher as it allows rapid training and testing without the need of hours of multiple demonstration recordings or several waiting hours to obtain the optimum parameters to test. Moreover, these optimized parameters and coupling terms might be excellent for static obstacles but any change in the configuration of the environment may make them not suitable anymore. Therefore, developing parameter optimization techniques that require few demonstrations to optimize the hyperparameters, would significantly reduce human's involvement while improving overall system's efficiency.

F. COMPUTATIONAL COST

Generally, there is also a lack in all the works concerning the quantification of how the addition of a potential slows down the execution phase of the DMP. The incorporation of APF supposes a computational burden for the DMP that could limit their viability in real-life, particularly with the steering angle, where equal importance is given to obstacles regardless their position respect to the robot. Besides, machine learning-based approaches require extra time for demonstration recording and model training. For a fair comparison between methods, not only the smoothness of the generated trajectory and the capability of obstacle avoidance should be compared, but also the time taken by the algorithm to obtain the trajectory. Besides, proving that the proposed combination achieves high-speed response during the execution phase will lead to consider the proposed method to be usable in real time, which is a valuable contribution for task automation research.

G. SCENARIO COMPLEXITY

Finally, ensuring the generalization of learned movements while incorporating obstacle avoidance remains a challenge outside the laboratory. A remarkable point is that all the research of table 1 lacks real-life experimentation. Indeed, all relied on laboratory experimentation, with varying complexity depending on the work. Even in the case of [21] where more complex scenarios were designed and three robots tested, there is not a real task that aims to be automatized supporting the experiments. Indeed, this investigation shows how conclusions drawn on simulated data do not always generalize to real use cases. It would be interesting to have the methods tested to attempt to automate real-world processes as an ultimate challenge to really ensure that the proposed combination is robust in different real life scenarios.

V. CONCLUSION

This work provides an analysis of different approaches combining DMP and APF, with the objective of extending the motion learning capability of the DMP, with the possibility to react to undesired situation through the APF.

Two main families of approaches have been identified, the ones based on the first dynamic potential and the ones based on the steering angle. The original dynamic potential and its derivatives are capable of successfully avoiding obstacles

with smoother movements, compared to static potentials, but they do not guarantee convergence to the goal nor do they avoid oscillatory behaviors. The approaches based on the steering angle family, on the contrary, are capable of guaranteeing the convergence to the goal but still suffer from oscillations when the formulation does not depend on the distance to the obstacles. There is only a single investigation that aimed to address both challenges, but the simplicity of the proposed use case does not ensure its applicability in real life.

Regarding the identification of futures lines of work, there is a lack of more realistic scenarios where the algorithms can be tested. The DMP-APF combination is ultimately intended to work in real environments, when a completely automated task may not be optimum due to environmental changes. Full 3D implementations, including control of end-effector orientation and the potential for collisions along the entire robotic arm significantly increase the complexity, yet here is where they have to prove their value.

In robotics, it is important to take into account two more aspects: the computational cost necessary to generate the trajectory and the smoothness of the generated trajectory. Regarding the computational cost, the incorporation of the APF does not affect the learning phase but does influence the execution phase when the trajectory to avoid obstacles is generated. This phase should not last more than minutes to ensure that the whole demonstration, learning and execution phase do not burden the human. Finally, the generated curves should be smooth in order to avoid robot's undesired variations in joint velocities. For a fair comparison between algorithms, not only their capacity (the task accomplishment ratio) should be evaluated, but also the quality in terms of smoothness of the generated curves.

In summary, the innovative ideas behind DMPs and APF have provided a fruitful framework for implementing efficient learning from demonstration that has attracted significant attention in recent years. While there is still much work to be done, as outlined in this targeted overview, the combination of DMPs and APF is a promising solution to the challenge of simplifying the programming of robotic trajectories in the real world.

REFERENCES

[1] M. Saveriano, F. J. Abu-Dakka, A. Kramberger, and L. Peternel, "Dynamic movement primitives in robotics: A tutorial survey," *Int. J. Robot. Res.*, vol. 42, no. 13, pp. 1133–1184, Nov. 2023.

[2] F. Mussa-Ivaldi, N. Hogan, and E. Bizzi, "Neural, mechanical, and geometric factors subserving arm posture in humans," *J. Neurosci.*, vol. 5, no. 10, pp. 2732–2743, Oct. 1985.

[3] N. Hogan, "The mechanics of multi-joint posture and movement control," *Biol. Cybern.*, vol. 52, no. 5, pp. 315–331, Sep. 1985.

[4] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2, Jun. 2002, pp. 1398–1403.

[5] S. Schaal, "Dynamic movement primitives—A framework for motor control in humans and humanoid robotics," in *Adaptive Motion of Animals and Machines*. Cham, Switzerland: Springer, 2006, pp. 261–280.

[6] P. Kormushev, S. Calinon, and D. G. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Adv. Robot.*, vol. 25, no. 5, pp. 581–603, Jan. 2011.

[7] F. J. Abu-Dakka, B. Nemec, J. A. Jørgensen, T. R. Savarimuthu, N. Krüger, and A. Ude, "Adaptation of manipulation skills in physical contact with the environment to reference force profiles," *Auto. Robots*, vol. 39, no. 2, pp. 199–217, Aug. 2015.

[8] M. Wu, B. Taetz, Y. He, G. Bleser, and S. Liu, "An adaptive learning and control framework based on dynamic movement primitives with application to human–robot handovers," *Robot. Auto. Syst.*, vol. 148, Feb. 2022, Art. no. 103935.

[9] L. Peternel, N. Tsagarakis, D. Caldwell, and A. Ajoudani, "Robot adaptation to human physical fatigue in human–robot co-manipulation," *Auto. Robots*, vol. 42, no. 5, pp. 1011–1021, Jun. 2018.

[10] R. J. Escarabajal, J. L. Pulloquinga, P. Zamora-Ortiz, Á. Valera, V. Mata, and M. Vallés, "Imitation learning-based system for the execution of self-paced robotic-assisted passive rehabilitation exercises," *IEEE Robot. Autom. Lett.*, vol. 8, no. 7, pp. 4283–4290, Jul. 2023.

[11] F. J. Abu-Dakka, A. Valera, J. A. Escalera, M. Abderrahim, A. Page, and V. Mata, "Passive exercise adaptation for ankle rehabilitation based on learning control framework," *Sensors*, vol. 20, no. 21, p. 6215, Oct. 2020.

[12] F. Lanotte, Z. McKinney, L. Grazi, B. Chen, S. Crea, and N. Vitiello, "Adaptive control method for dynamic synchronization of wearable robotic assistance to discrete movements: Validation for use case of lifting tasks," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 2193–2209, Dec. 2021.

[13] M. Ginesi, N. Sansonetto, and P. Fiorini, "Overcoming some drawbacks of dynamic movement primitives," *Robot. Auto. Syst.*, vol. 144, Oct. 2021, Art. no. 103844.

[14] F. Stulp and S. Schaal, "Hierarchical reinforcement learning with movement primitives," in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, Oct. 2011, pp. 231–238.

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

[16] A. Li, Z. Liu, W. Wang, M. Zhu, Y. Li, Q. Huo, and M. Dai, "Reinforcement learning with dynamic movement primitives for obstacle avoidance," *Appl. Sci.*, vol. 11, no. 23, p. 11184, Nov. 2021.

[17] M. Ossenkopf, P. Ennen, R. Vossen, and S. Jeschke, "Reinforcement learning for manipulators without direct obstacle perception in physically constrained environments," *Proc. Manuf.*, vol. 11, pp. 329–337, Jan. 2017.

[18] R. A. Shyam, P. Lightbody, G. Das, P. Liu, S. Gomez-Gonzalez, and G. Neumann, "Improving local trajectory optimisation using probabilistic movement primitives," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 2666–2671.

[19] D. Koert, J. Pajarinen, A. Schotschneider, S. Trick, C. Rothkopf, and J. Peters, "Learning intention aware online adaptation of movement primitives," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3719–3726, 2019.

[20] A. Colomé and C. Torras, "Demonstration-free contextualized probabilistic movement primitives, further enhanced with obstacle avoidance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 3190–3195.

[21] M. Ginesi, D. Meli, A. Roberti, N. Sansonetto, and P. Fiorini, "Dynamic movement primitives: Volumetric obstacle avoidance using dynamic potential functions," *J. Intell. Robot. Syst.*, vol. 101, no. 4, pp. 1–20, Apr. 2021.

[22] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *Proc. 8th IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2008, pp. 91–98.

[23] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.*, vol. 5, no. 1, pp. 90–98, Mar. 1986.

[24] J. Luo, Z.-X. Wang, and K.-L. Pan, "Reliable path planning algorithm based on improved artificial potential field method," *IEEE Access*, vol. 10, pp. 108276–108284, 2022.

[25] D. Rodríguez-Guerra, A. Mosca, A. Valente, I. Cabanes, and E. Carpanzano, "An advanced dual APF-based controller for efficient simultaneous collision and singularity avoidance for human–robot collaborative assembly processes," *CIRP Ann.*, vol. 72, no. 1, pp. 5–8, 2023.

[26] A. Ma'Arif, W. Rahmaniar, M. A. M. Vera, A. A. Nuryono, R. Majdoubi, and A. Çakan, "Artificial potential field algorithm for obstacle avoidance in UAV quadrotor for dynamic environment," in *Proc. IEEE Int. Conf. Commun., Netw. Satell. (COMNETSAT)*, Jul. 2021, pp. 184–189.

[27] 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.
[28] H. Hoffmann, P. Pastor, D.-H. Park, and S. Schaal, "Biologically-inspired dynamical systems for movement generation: Automatic real-time goal adaptation and obstacle avoidance," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 2587–2592.
[29] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 763–768.
[30] S. Bitzer and S. Vijayakumar, "Latent spaces for dynamic movement primitives," in *Proc. 9th IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2009, pp. 574–581.
[31] M. J. Hong and M. R. Arshad, "A balance-artificial potential field method for autonomous surface vessel navigation in unstructured riverine environment," *Proc. Comput. Sci.*, vol. 76, pp. 198–202, Jan. 2015.
[32] J. Sun, G. Liu, G. Tian, and J. Zhang, "Smart obstacle avoidance using a danger index for a dynamic environment," *Appl. Sci.*, vol. 9, no. 8, p. 1589, Apr. 2019.
[33] B. R. Fajen and W. H. Warren, "Behavioral dynamics of steering, obstable avoidance, and route selection," *J. Experim. Psychol., Hum. Perception Perform.*, vol. 29, no. 2, pp. 343–362, 2003.
[34] R. C. Arkin, "Integrating behavioral, perceptual, and world knowledge in reactive navigation," *Robot. Auto. Syst.*, vol. 6, nos. 1–2, pp. 105–122, Jun. 1990.
[35] X. Lin, Z.-Q. Wang, and X.-Y. Chen, "Path planning with improved artificial potential field method based on decision tree," in *Proc. 27th Saint Petersburg Int. Conf. Integr. Navigat. Syst. (ICINS)*, May 2020, pp. 1–5.
[36] Q. Yao, Z. Zheng, L. Qi, H. Yuan, X. Guo, M. Zhao, Z. Liu, and T. Yang, "Path planning method with improved artificial potential field—A reinforcement learning perspective," *IEEE Access*, vol. 8, pp. 135513–135523, 2020.
[37] V. Stoian, M. Ivanescu, E. Stoian, and C. Pana, "Using artificial potential field methods and fuzzy logic for mobile robot control," in *Proc. 12th Int. Power Electron. Motion Control Conf.*, Aug. 2006, pp. 385–389.
[38] R. S. Sharma, S. Shukla, H. Karki, A. Shukla, and L. Behera, "DMP based trajectory tracking for a nonholonomic mobile robot with automatic goal adaptation and obstacle avoidance," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 8613–8619.
[39] P. Nattharith and M. S. Güzel, "Machine vision and fuzzy logic-based navigation control of a goal-oriented mobile robot," *Adapt. Behav.*, vol. 24, no. 3, pp. 168–180, Jun. 2016.
[40] A. Phung, J. Malzahn, F. Hoffmann, and T. Bertram, "Get out of the way–obstacle avoidance and learning by demonstration for manipulation," *IFAC Proc. Volumes*, vol. 44, no. 1, pp. 11514–11519, 2011.
[41] M. Cui, Y. Hu, S. Xu, J. Wang, Z. Bing, B. Li, and A. Knoll, "Safe and human-like trajectory planning of self-driving cars: A constraint imitative method," *Adv. Intell. Syst.*, vol. 5, no. 10, Oct. 2023, Art. no. 2300269.
[42] Q. Li, Z. Wang, W. Wang, Z. Liu, Y. Chen, X. Ng, and M. H. Ang, "A model predictive obstacle avoidance method based on dynamic motion primitives and a Kalman filter," *Asian J. Control*, vol. 25, no. 2, pp. 1510–1525, Mar. 2023.
[43] X. Liu, Z. Wang, J. Li, A. Cangelosi, and C. Yang, "Demonstration learning and generalization of robotic motor skills based on wearable motion tracking sensors," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023.
[44] M. Ginesi, D. Meli, A. Calanca, D. Dall'Alba, N. Sansonetto, and P. Fiorini, "Dynamic movement primitives: Volumetric obstacle avoidance," in *Proc. 19th Int. Conf. Adv. Robot. (ICAR)*, Dec. 2019, pp. 234–239.
[45] 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.
[46] B. Ti, Y. Gao, M. Shi, L. Fu, and J. Zhao, "Movement generalization of variable initial task state based on Euclidean transformation dynamical movement primitives," *Int. J. Adv. Robot. Syst.*, vol. 18, no. 6, Nov. 2021, Art. no. 172988142110655.
[47] Z. Lu, N. Wang, and C. Yang, "A dynamic movement primitives-based tool use skill learning and transfer framework for robot manipulation," *IEEE Trans. Autom. Sci. Eng.*, early access, 2004. [Online]. Available: https://ieeexplore.ieee.org/document/10458061
[48] Z. Mei, Y. Chen, M. Jiang, H. Wu, and L. Cheng, "Mobile robots path planning based on dynamic movement primitives library," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 6906–6911.
[49] M. Chi, Y. Yao, Y. Liu, and M. Zhong, "Learning, generalization, and obstacle avoidance with dynamic movement primitives and dynamic potential fields," *Appl. Sci.*, vol. 9, no. 8, p. 1535, Apr. 2019.
[50] P. Pastor, M. Kalakrishnan, F. Meier, F. Stulp, J. Buchli, E. Theodorou, and S. Schaal, "From dynamic movement primitives to associative skill memories," *Robot. Auto. Syst.*, vol. 61, no. 4, pp. 351–361, Apr. 2013.
[51] D.-H. Zhai, Z. Xia, H. Wu, and Y. Xia, "A motion planning method for robots based on DMPs and modified obstacle-avoiding algorithm," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 4, pp. 2678–2688, 2022.
[52] A. Rai, F. Meier, A. Ijspeert, and S. Schaal, "Learning coupling terms for obstacle avoidance," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Nov. 2014, pp. 512–518.
[53] È. Pairet, P. Ardón, M. Mistry, and Y. Petillot, "Learning generalizable coupling terms for obstacle avoidance via low-dimensional geometric descriptors," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3979–3986, Oct. 2019.
[54] A. Gams, T. Petrič, M. Do, B. Nemec, J. Morimoto, T. Asfour, and A. Ude, "Adaptation and coaching of periodic motion primitives through physical and visual interaction," *Robot. Auto. Syst.*, vol. 75, pp. 340–351, Jan. 2016.
[55] A. Duan, R. Camoriano, D. Ferigo, Y. Huang, D. Calandriello, L. Rosasco, and D. Pucci, "Learning to avoid obstacles with minimal intervention control," *Frontiers Robot. AI*, vol. 7, p. 60, Jan. 2020.
[56] B. Jeon, H. Kim, and H. J. Kim, "Collision avoidance of robotic arm of aerial manipulator," in *Proc. 11th Asian Control Conf. (ASCC)*, Dec. 2017, pp. 1859–1864.
[57] H. Tan, E. Erdemir, K. Kawamura, and Q. Du, "A potential field method-based extension of the dynamic movement primitive algorithm for imitation learning with obstacle avoidance," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2011, pp. 525–530.
[58] Y. Huang, T. Li, Y. Ning, and Y. Zhang, "A robot motion skills method with explicit environmental constraints," *Ind. Robot, Int. J. Robot. Res. Appl.*, vol. 51, no. 3, pp. 387–399, May 2024.
[59] A. Rai, G. Sutanto, S. Schaal, and F. Meier, "Learning feedback terms for reactive planning and control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 2184–2191.
[60] J. Jia, Y. Wang, G. Zuo, Y. Cao, and N. Yu, "Research and implementation of complex task based on DMP," in *Proc. IEEE 9th Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC)*, vol. 9, Dec. 2020, pp. 730–735.

IRATI RASINES received the bachelor's degree in physics science, the master's degree in control engineering, automation, and robotics, and the master's degree in biomedical engineering from the University of the Basque Country, in 2009, 2011, and 2015, respectively, obtaining the prize for the best student in her class in both master degrees. She has been working as a Researcher with the Health Division, Tecnalia, since 2009, with a focus on artificial intelligence applied to different fields. She has also contributed to various projects sponsored by European Commission in the last years, such as CogLaboration, SARAFun, RobotUnion, TraceBot, Humantech, or Smarthandle. Her actual research interest includes automatic robot learning based on human demonstrations in human–robot collaborative scenarios.

ITZIAR CABANES (Member, IEEE) received the Ph.D. degree in engineering from the University of the Basque Country (UPV/EHU), in 2001. She is currently a Full Professor with the Department of Automatic Control and Systems Engineering, Bilbao School of Engineering, UPV/EHU. She is the Head of the Virtual Sensorization Research Group granted by the Basque Government. Her research interests include industrial robotics, virtual sensors, and artificial intelligence to improve manufacturing processes and healthcare applications. She is the author of more than 150 papers in indexed scientific journals, chapters of books, and conferences, where she has been awarded with eight prizes for the best scientific contribution. During the Ph.D. degree, she received the Extraordinary Doctorate Award.

ANTHONY REMAZEILLES received the M.Sc. degree in computer science and artificial intelligence and the Ph.D. degree in computer science from the University of Rennes, in 2001 and 2004, respectively. From 2001 to 2006, his research concerned the vision-based control of robotics systems within large navigation spaces. From 2006 to 2008, he was a Postdoctoral Researcher with the Commissariat à l'Énergie Atomique (CEA-LIST), Fontenay-aux-Roses, France. He was involved in the development of a semi-autonomous mobile arm providing vision-based grasping assistance to disabled people within the framework of the ITEA ANSO project that received an ITEA Appreciation Award, in 2008. Since 2008, he has been working as a Researcher and the Project Leader of the Health Division, Tecnalia, designing intelligent robotics systems for human-centered applications. He has contributed to 24 peer-reviewed papers (conferences and journals) and holds a patent on vision-based object grasping. He has been actively contributing to various projects sponsored by European Commission, such as Florence, CogLaboration, STIFF-FLOP, SARAFun, ROSIN, RobotUnion, EUROBENCH, and TraceBot. In 2004, he received the Best Paper Award in Pattern Recognition at the French Congress on Pattern Recognition and Artificial Intelligence (RFIA).

JOSEPH MCINTYRE received the bachelor's degree in engineering and the bachelor's degree in biology from California Institute of Technology, in 1982 and 1983, respectively, and the Ph.D. degree in computational neuroscience from MIT, in 1990. He has carried out fundamental research on multisensory integration and the control of movement, first as a Postdoctoral Researcher with the College de France, Paris, and the Fondazione Santa Lucia, Rome, and then as a Research Scientist with the Centre National de la Recherche Scientifique (CNRS), France. In January 2014, he accepted an Ikerbasque Research Professorship to work with the Tecnalia's Health Division on the development of robotic technologies for health, while at the same time continuing his efforts in fundamental research into human multi-sensory integration and motor control through projects with CNRS and the French (CNES), European (ESA) and American (NASA) space agencies. He currently directs the Department of Medical Technologies, Health Unit, Tecnalia Research and Innovation.