Velocity-Dependent Volumetric Obstacle Avoidance for Dynamic Movement Primitives

Michele Ginesi¹, Daniele Meli¹, Andrea Roberti¹, Nicola Sansonetto¹, and Paolo Fiorini¹

¹Department of Computer Science, University of Verona, Verona, Italy

Email:{ michele.ginesi, daniele.meli, andrea.roberti, nicola.sansonetto, paolo.fiorini }@univr.it

Abstract—We propose a novel method to implement obstacle avoidance within the Dynamic Movement Primitives (DMPs) framework. The proposed method is able to deal with volumetric obstacles. Moreover, being dependent on the velocity of the system (and not only on its position), this new method is able to achieve smooth behaviors.

Index Terms—Obstacle Avoidance, Dynamic Movement Primitives, Learning from Demonstration

I. INTRODUCTION

In last years, *Learning from Demonstrations* (LfD) approaches (such as *Gaussian Mixture Models* [1] and *Extreme Learning Machines* [2]) have been developed in order to replicate human gestures in robotics. A widely used LfD technique is the *Dynamic Movement Primitives* (DMPs) framework [3], [4]. DMPs permit to learn a trajectory from just one demonstration by encoding the trajectory in a system of second-order linear Ordinary Differential Equation (ODE), where a forcing term is learned as a linear combination of predefined time-dependent functions. Obstacle avoidance for DMPs has been successfully treated for point-like obstacles (e.g. [4] and [5]). On the other hand, volumetric obstacle avoidance has been treated in our previous work [6] using potential functions.

Here, we propose a new potential function that improves our previous method [6] by extending it so to take into account also the velocity of the system (and not only the position). This allow to achieve a smoother obstacle avoidance behavior.

In Section II we recall the DMP's theory; in Section III we present our proposed method for obstacle avoidance; in Section IV we test it; and in Section V we present the conclusions.

II. DMP THEORY

DMPs is a framework for trajectory learning. It is based on an Ordinary Differential Equation (ODE) of spring-massdamper type with a forcing term. They consist of the following system of Ordinary Differential Equations:

$$\begin{cases} \tau \dot{\mathbf{v}} = \mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{D}\mathbf{v} - \mathbf{K}(\mathbf{g} - \mathbf{x}_0)s + \mathbf{K}\mathbf{f}(s) + \boldsymbol{\varphi}(\mathbf{x}, \mathbf{v}) \\ \tau \dot{\mathbf{x}} = \mathbf{v} \end{cases}$$

This research has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, ARS (Autonomous Robotic Surgery) project, grant agreement No. 742671.

This research is currently in consideration for publication by the Journal of *Intelligent and Robotic Systems*.



Fig. 1: Depiction of the angle θ in (2).

Vectors $\mathbf{x}, \mathbf{v} \in \mathbb{R}^d$ are, respectively, the *position* and *velocity* of the system; and $\mathbf{x}_0, \mathbf{g} \in \mathbb{R}^d$ are, respectively, the *starting* and *goal positions*. Matrices $\mathbf{K}, \mathbf{D} \in \mathbb{R}^{d \times d}_+$ are, respectively, the *elastic* and *damping terms* of the system. Both are diagonal matrices, $\mathbf{K} = \text{diag}(K_1, K_2, \ldots, K_d)$, $\mathbf{D} = \text{diag}(D_1, D_2, \ldots, D_d)$, and satisfy the critical damping relation $D_i = 2\sqrt{K_i}$, so that the un-perturbed system, i.e. when $\mathbf{f} \equiv \mathbf{0}$, converges as fast as possible to the unique equilibrium $(\mathbf{x}, \mathbf{v}) = (\mathbf{g}, \mathbf{0})$. Scalar $\tau \in \mathbb{R}_+$ is a *temporal scaling factor* which can be used to make the execution of the trajectory faster or slower. Function $\mathbf{f} : \mathbb{R} \to \mathbb{R}^d$ is the *forcing term*, expressed in term of basis functions. Scalar $s \in (0, 1]$ is a re-parametrization of time $t \in [0, T]$ governed by the so called *canonical system* $\tau \dot{s} = -\alpha s$, where $\alpha \in \mathbb{R}_+$ and the initial state is s(0) = 1.

Function $\varphi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ is the perturbation term used to deal with obstacle avoidance. Intuitively, it can be seen as a term that 'pushes' the system away from the obstacle. In the next Section, we present our new formulation of function φ .

III. NEW METHOD FOR OBSTACLE AVOIDANCE

To define our perturbation term, we start by defining the concept of *isopotential* as any function $C : \mathbb{R}^d \to \mathbb{R}$ that vanishes on the surface of the obstacle and increases as the distance from the obstacle increases. An example in \mathbb{R}^3 of isopotential is the function

$$C(\mathbf{x}) = \left(\frac{x_1 - \hat{x}_1}{\ell_1}\right)^{2n_1} + \left(\frac{x_2 - \hat{x}_2}{\ell_2}\right)^{2n_2} + \left(\frac{x_3 - \hat{x}_3}{\ell_3}\right)^{2n_3} - 1,$$
(1)

that vanishes on a *pseudo-ellipsoid* of center $\hat{\mathbf{x}} = [\hat{x}_1, \hat{x}_2, \hat{x}_3]^{\mathsf{T}}$ and semi-axis (ℓ_1, ℓ_2, ℓ_3) .



Fig. 2: Example of the dynamic potential $U_D(\mathbf{x}, \mathbf{v})$ given in (3) for an ellipse in \mathbb{R}^2 . In both figures, the potential has been cropped at the value 1 for display purposes.



Fig. 3: Pick-and-place task with the Panda robot.

Next, we define the angle θ , depicted in Figure 1, as

$$\cos \theta = \frac{\langle \nabla_{\mathbf{x}} C(\mathbf{x}), \mathbf{v} \rangle}{\|\nabla_{\mathbf{x}} C(\mathbf{x})\| \|\mathbf{v}\|}.$$
(2)

At this point, we define the dynamic potential

$$U_d(\mathbf{x}, \mathbf{v}) = \begin{cases} \lambda(-\cos\theta)^{\beta} \frac{\|\mathbf{v}\|}{C^{\eta}(\mathbf{x})} & \text{if } \theta \in (\frac{\pi}{2}, \pi] \\ 0 & \text{if } \theta \in [0, \frac{\pi}{2}] \end{cases}, \quad (3)$$

where $\lambda, \beta, \eta \in \mathbb{R}_+$ are constant gains. Figure 2 shows an example the potential. This potential has three major advantages: 1. The magnitude of the potential decreases with the distance of the system from the obstacle; 2. The magnitude of the potential increases with the velocity of the system $\|\mathbf{v}\|$; 3. The magnitude of the potential decreases with the angle between current velocity direction and the direction towards the obstacle, and is null when the system is going away from the obstacle.

Finally, we define the perturbation term as the gradient, w.r.t. the system's position \mathbf{x} of U_D :

$$\varphi(\mathbf{x}, \mathbf{v}) = -\nabla_{\mathbf{x}} U_D(\mathbf{x}, \mathbf{v}).$$

IV. RESULTS

We test our obstacle avoidance framework with a Panda industrial manipulator. The task consists in grasping a ring and releasing it onto the same colored peg, while avoiding the other pegs and the base. The steps of the task are shown in Figure 3.

The task consists of two movements, namely move_to_ring in which the Panda has to reach the ring, and move_to_peg in which the robot carries the ring over the peg. Both gestures are modeled with DMPs with



Fig. 4: Obstacle avoidance with the Panda robot.

null forcing term $\mathbf{f} = \mathbf{0}$ so that the system converges toward the goal in a straight-line movement.

The pegs and the base are modeled as obstacles using pseudoellipsoid setting $n_1 = n_2 = 1$, and $n_3 = 2$ in (1). When the ring is grasped, the radius of the pegs is enlarged (as can be seen in Figure 4b) so that neither the end-effector of the robot nor the ring collides with the pegs.

Figure 4 shows the execution. For the move_to_ring gesture (Figure 4a), the DMP is not perturbed by the potential, since there is no risk of collision. On the other hand, the trajectory for the move_to_peg movement is perturbed by the presence of the obstacles.

V. CONCLUSIONS

In this work, we presented a novel method for obstacle avoidance within the DMPs framework. The proposed method models the obstacle with a potential field that depends both on the distance from the obstacle and the angle between the system's velocity and the direction towards the obstacle. This allows the potential to be null when the system has no risk of collision with the obstacle. Thus, when there is no risk of collisions, the trajectory is not influenced by the presence of obstacle.

REFERENCES

- S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, 2011.
- [2] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.
- [3] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," in *Advances in neural information processing systems*, 2003, pp. 1547–1554.
- [4] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *Humanoid Robots*, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on. IEEE, 2008, pp. 91–98.
- [5] 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 *Robotics and Automation*, 2009. *ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 2587– 2592.
- [6] M. Ginesi, D. Meli, A. Calanca, D. Dall'Alba, N. Sansonetto, and P. Fiorini, "Dynamic movement primitives: Volumetric obstacle avoidance," in 2019 19th International Conference on Advanced Robotics (ICAR), Dec 2019, pp. 234–239.