

# Learning Generalisable Coupling Terms for Obstacle Avoidance via Low-dimensional Geometric Descriptors

Èric Pairet\*, Paola Ardón, Michael Mistry, and Yvan Petillot

**Abstract**—Unforeseen events are frequent in the real-world environments where robots are expected to assist, raising the need for fast replanning of the policy in execution to guarantee the system and environment safety. Inspired by human behavioural studies of obstacle avoidance and route selection, this paper presents a hierarchical framework which generates reactive yet bounded obstacle avoidance behaviours through a multi-layered analysis. The framework leverages the strengths of learning techniques and the versatility of dynamic movement primitives to efficiently unify perception, decision, and action levels via low-dimensional geometric descriptors of the environment. Experimental evaluation on synthetic environments and a real anthropomorphic manipulator proves that the robustness and generalisation capabilities of the proposed approach regardless of the obstacle avoidance scenario makes it suitable for robotic systems in real-world environments.

## I. INTRODUCTION

Robust reactive behaviours are essential to ensure the safety of robots operating in unstructured environments. For instance, the on-going pick-and-place policy of a robotic system sorting and storing items in a home environment might be interrupted by the sudden appearance of an obstacle in the middle of a pre-planned trajectory. In this scenario, the robot must be able to modulate its behaviour online to succeed in its task while providing some safety guarantees. Given the expertise of humans in dealing with these conditions, it is natural to adopt human behaviour for robotic control.

Human behavioural studies of obstacle avoidance and route selection [1] have shown that the dynamics of perception and action consist of (i) identifying the informational variables useful to guide behaviour and to regulate action, and (ii) interacting with the environment using a particular set of dynamic behaviours. One possible policy descriptor allowing for this hierarchical control are dynamic movement primitives (DMPs) [2]. DMPs are differential equations encoding kinematic control policies towards a goal attractor. Their transient behaviour can be shaped via a non-linear forcing term, which can be initialised via imitation learning and used to reproduce an observed motion while generalising to different start and goal locations, as well as task durations.

A key feature of DMPs is that they allow for online modulation via coupling term functions that create a forcing term. Coupling terms have been exploited for many applications, such as avoidance of joint and workspace limits [3], force control for environment interaction [4], [5], dual-arm manipulation [4], [6] and reactive obstacle avoidance [7]–[11]. This work focuses on the latter challenge, which historically has

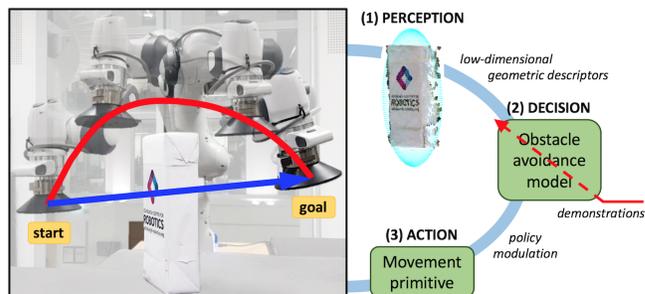


Fig. 1: Proposed hierarchical framework for learning and producing generalisable obstacle avoidance behaviours. Pre-planned start-go-goal (blue) and modulated policy (red).

been approached using potential fields [7], [8], analytical [9] and learning methods [10], [11] (see Section II). As further discussed in Section II-C, analytical formulations become less reactive for imminent collisions (dead-zone problem). Moreover, these approaches do not provide any guidance to the reactive behaviour, thus limiting their applicability to free-floating obstacles. Additionally, analytical formulations uniquely deal with point-mass obstacles and systems. In an attempt to address this latter issue, recent proposals learn coupling terms for a small set of obstacle geometries described by an array of markers on their surface [10], [11], but they fail to generalise actions to novel obstacles. These works are notable in learning the coupling terms from human demonstration. Nonetheless, providing a rich set of demonstrations involving various obstacles geometries can be time-consuming and prone to measurement noise.

This paper presents the hybrid DMP-learning-based obstacle avoidance framework schematised in Figure 1. The proposed approach addresses the limitations of the precedent works with a layered perception-decision-action analysis [1]. The main contributions at the action level (see Section III) are (i) reformulating the coupling terms to provide dead-zone free behaviours, and (ii) guiding the obstacle avoidance reactivity to satisfy task-dependant constraints, while the main contributions at the perception-decision level (see Section IV) are (iii) regulating action according to the extracted unified system-obstacle low-dimensional geometric descriptor, and (iv) learning to regulate the action level via exploration of the parameter space. The experimental evaluation reported in Section V demonstrates that the overall proposed approach generalises obstacle avoidance behaviours to novel scenarios, even when those involve multiple obstacles, or are uniquely described by partial visual-depth observations.

Authors are with the Edinburgh Centre for Robotics at University of Edinburgh and Heriot-Watt University (UK). \*eric.pairet@ed.ac.uk

## II. RELATED WORK

This paper proposes a reactive approach that endows a system with the ability to modulate its policy to avoid unexpected obstacles. The selected strategy uses DMPs for encoding any desired policy and defining an obstacle avoidance behaviour as a coupling term. This section introduces DMPs and coupling terms for obstacle avoidance as they constitute the fundamentals of this work.

### A. Dynamic Movement Primitives

DMPs are a versatile framework that encode primitive motions or policies as nonlinear functions called forcing terms [2]. The DMPs equations define the system's state transition, which can be converted into actuator commands by means of inverse kinematics and inverse dynamics. For a one-degree of freedom (DoF) system, the system's state transition is described by the following set of nonlinear differential equations, known as the transformation system:

$$\tau \dot{z} = \alpha_x (\beta_x (g_x - x) - z) + f(\cdot) + C(\cdot), \quad (1)$$

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

where  $\tau$  is a scaling factor for time,  $x$  is the system's position,  $z$  and  $\dot{z}$  respectively are the scaled velocity and acceleration,  $\alpha_x$  and  $\beta_x$  are constants defining the attraction dynamics towards the model's attractor  $g_x$ , and  $f(\cdot)$  and  $C(\cdot)$  are the forcing and coupling term, respectively.

The forces generated by the forcing and coupling terms define the transient behaviour of the transformation system. It is common to model the forcing term  $f(\cdot)$  as a weighted linear combination of nonlinear radial basis functions (RBFs). The evaluation of  $f(\cdot)$  at phase  $k \in \mathbf{k}$  is defined as:

$$f(k) = \frac{\sum_{i=1}^N w_i \Psi_i(k)}{\sum_{i=1}^N \Psi_i(k)} k, \quad (3)$$

$$\Psi_i(k) = \exp(-h_i(k - c_i)^2), \quad (4)$$

where  $c_i$  and  $h_i > 0$  are the centres and widths, respectively, of the  $i \in [1, N]$  RBFs, which are weighted by  $w_i$  and distributed along the trajectory. The weights can be initialised via imitation learning and used to reproduce the motion with some generalisation capabilities to changes in start and goal positions. The duration of the motion can be adjusted by the scaling factor  $\tau$ , which modifies the canonical system defining the transient behaviour of the phase variable  $k$  as:

$$\tau \dot{k} = -\alpha_k k, \quad (5)$$

where the initial value of the motion's phase  $\mathbf{k}(0) = 1$  and  $\alpha_k$  is a positive constant.

A common strategy to extend the spatial generalisation capabilities of DMPs is to reference them in a local frame, whose pose in the space is task-dependent [2], [11]. In this work's context, the unit vectors of the local frame are defined as follows: the x-axis points from the start position towards the goal position, the z-axis points upwards and is orthogonal to the local x-axis, and the y-axis is orthogonal to both local x-axis and z-axis following the right-hand convention.

A robot with multiple DoFs uses a transformation system for each DoF, but they all share the same canonical system.

### B. Coupling Terms for Obstacle Avoidance

Early coupling terms for obstacle avoidance were formulated as repulsive potential fields [7], [8]. Potential fields suffer from local minima and can be computationally expensive to calculate on the fly. Alternatively, some coupling terms analytically formalise the influence of an obstacle on the system's behaviour [9]. As depicted in Figure 2a, a point-mass system with position  $\mathbf{x} \in \mathbb{R}^3$  and velocity  $\dot{\mathbf{x}} \in \mathbb{R}^3$  has a heading  $\theta \in \text{SO}(2)$  towards a point-mass obstacle. To avoid a collision, the coupling term generates a repulsive force:

$$C(\cdot) = \mathbf{R} \dot{\mathbf{x}} \dot{\theta}, \quad (6)$$

where  $\mathbf{R} \in \text{SO}(3)$  is a  $\pi/2$  rotation matrix around the vector  $\mathbf{r} = (\mathbf{x}_{obstacle} - \mathbf{x}) \times \dot{\mathbf{x}}$ . The respective obstacle-system position  $\mathbf{x}_{obstacle} - \mathbf{x}$  and the system's velocity  $\dot{\mathbf{x}}$  define the plane  $\mathbf{P} \in \mathbb{R}^2$  where the system is desired to steer away from the obstacle with a turning velocity  $\dot{\theta}$  defined as:

$$\dot{\theta} = \gamma \theta \exp(-\beta |\theta|), \quad (7)$$

where  $\gamma$  and  $\beta$  respectively scale and shape the mapping  $\theta \rightarrow \dot{\theta}$  defined in (7) and represented in Figure 2b.

Building on (6)-(7), human demonstrations were used to retrieve the required parameters to circumvent two non-point obstacles, particularly a sphere and a cylinder [10]. More recently, coupling terms were formulated as independent neural networks (NNs) modelling the desired obstacle avoidance behaviour for a sphere, a cylinder and a cube [11]. These methods do not provide any strategy to avoid obstacles not observed in training time, and they rely on markers identifying an obstacle's boundaries. Their evaluations are conducted either in simulation or in single-obstacle scenarios. Hence, their performance in realistic scenarios is yet to be tested.

### C. Discussion and Contribution

State-of-the-art on coupling terms modelling obstacle avoidance behaviours suffers from four major limitations. First, as illustrated in Figure 3, the analytical term (6)-(7) has a dead-zone where the system becomes less reactive as the heading towards the obstacle narrows, thus compromising the method's reliability. Second, there is no strategy to guide the behaviour's reactivity towards a preferred route to circumnavigate an obstacle. For example, in the scenario

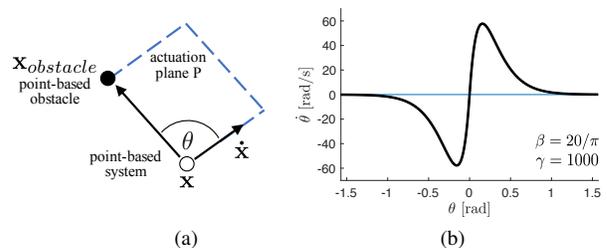


Fig. 2: Original coupling terms for obstacle avoidance [9]. (a) Heading angle  $\theta$  according to velocity vector  $\dot{\mathbf{x}}$  and the relative obstacle-system position in P-plane. (b) Change of steering angle  $\dot{\theta}$  subject to heading angle  $\theta$  as defined by (7).

depicted in Figure 1, there is no constraint on the reactive behaviour preventing the system from hitting the table. Third, when attempting to deal with non-point obstacles, their performance drastically decreases for novel scenarios due to the absence of global features identifying the obstacle geometry during the learning process. Fourth, these works learn the coupling terms from demonstration, which can be time-consuming and prone to measurement noise.

All these issues are jointly addressed within the proposed hierarchical framework, which hybridises the versatility of DMPs and the strengths of learning techniques. Specifically, in Section III, (6)-(7) is reformulated at the action level as a conjunction of coupling terms whose obstacle avoidance behaviour is dead-zone free and can be guided. Then, in Section IV, the formalised action level is exploited to learn via exploration of the parameter space how to regulate the behaviour subject to both the end-effector's and obstacle's geometric properties. This work considers a unified system-obstacle low-dimensional geometric descriptors identifying the relevant features to the action level, thus allowing for enhanced generalisation even in novel real-world scenarios.

### III. COUPLING TERMS FOR DEAD-ZONE FREE AND GUIDED OBSTACLE AVOIDANCE

The proposed hierarchical framework to learn and produce generalisable obstacle avoidance behaviours regardless of the scenario comprises three layers. The DMP-based action level is formalised as a composition of two coupling terms which (i) generate robust obstacle avoidance behaviours, and (ii) guide these in a particular direction of the task space. The parametrisation needs of these terms allow for regulating their actuation scope via reasoning at the decision level.

#### A. Inherently Robust Obstacle Avoidance

Current coupling terms for obstacle avoidance in the literature suffer from dead-zones, i.e. a heading range towards the obstacle for which the system becomes incoherently less reactive. Ideally, the expected behaviour of those terms would be to become more reactive as (i) the heading of the system is more aligned towards an obstacle, and (ii) the system-obstacle distance is smaller. Bearing these conditions in mind, the coupling term in (6)-(7) is reformulated as:

$$C_{\text{oa}}(\cdot) = \mathbf{R}' \dot{\mathbf{x}} \alpha \text{sign}(\theta) \exp\left(-\frac{\theta^2}{\psi^2}\right) \exp(-\kappa d^2), \quad (8)$$

where  $\alpha \text{sign}(\theta) \exp(-\theta^2/\psi^2)$  addresses the first issue by shaping the absolute change of steering angle as a zero-mean Gaussian-bell function, and  $\exp(-\kappa d^2)$  tackles the second requirement by regulating the coupling term effect according to a parameter  $k$  and the system-obstacle distance  $d$ .

Figure 3 highlights the increase in robustness of the formulated coupling term (8) in contrast to the original term (6)-(7). While the original coupling term (black curves) produces low reactivity for narrow headings towards an obstacle, the dead-zone free proposal (red curves) reacts the most (see Figure 3a). This reformulation has a significant impact in

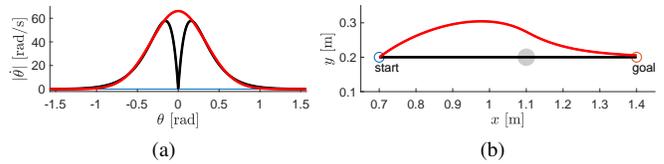


Fig. 3: Dead-zone issue in the original (6)-(7) (black) and proposed (8) (red) coupling terms. (a) (8) reacts for narrow headings towards the obstacle. (b) (6)-(7) fails where (8) smoothly circumvents the point-mass obstacle (grey circle).

the task space, where (8) succeeds on a scenario where (6)-(7) fails to generate an obstacle avoidance behaviour which does not collide with the point-mass obstacle (see Figure 3b).

#### B. Guiding the Obstacle Avoidance Reactivity

The velocity vector  $\dot{\mathbf{x}}$  of a point-mass system also represents the system's orientation. Consequently,  $\dot{\mathbf{x}}$  plays a critical role in determining both the actuation P-plane and the direction of turning  $\dot{\theta}$ . Overall, the behaviour encapsulated in (8) consists of turning to the opposite direction where the obstacle is with respect to the system's heading or velocity vector  $\dot{\mathbf{x}}$ . Although this reactive motion might be the safest behaviour in front of an obstacle, there are many situations where guiding the system towards a particular route might be of interest, such as in constrained environments or when aiming for a trajectory providing a minimum cost.

Given the influence of the system's heading  $\dot{\mathbf{x}}$  on the overall obstacle avoidance reaction, it is natural to modulate  $\dot{\mathbf{x}}$  to guide the reactivity of (8) through a preferred route. Within the DMP motion descriptor, this can be formulated through a coupling term that creates an attractive forcing term to reduce the heading error  $\hat{\theta}$  between the current  $\dot{\mathbf{x}}$  and a desired  $\dot{\mathbf{x}}_d$  system's direction as:

$$C_{\text{HG}}(\cdot) = \mathbf{R}' \dot{\mathbf{x}} \alpha \hat{\theta} \exp(1 + \kappa d^2) \quad (9)$$

where  $\mathbf{R}' \in SO(3)$  is a  $\pi/2$  rotation matrix around the vector  $\mathbf{r}' = \dot{\mathbf{x}} \times \dot{\mathbf{x}}_d$ , and the term  $\alpha \exp(1 + \kappa d^2)$  ensures that (8) and (9) act in counterphase when parameterised for the same  $\alpha$  and  $\kappa$ . This is, (9) uniquely modifies the system's heading when not in proximity to obstacles, where (8) takes over the control to ensure the system's safety.

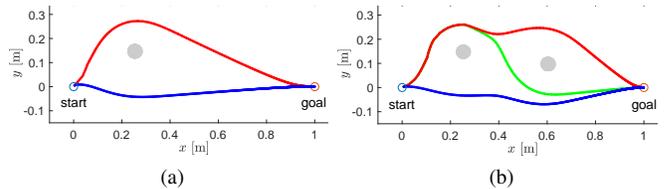


Fig. 4: Route selection for obstacle avoidance in (a) single and (b) multi-object setups. Reactive avoidance (blue), one-time decision (red), two-times decision (green).

### C. Coupling Terms Composition

Figure 4 depicts the significance of using (8) in conjunction with (9) to perform route selection of obstacle avoidance. This is formalised within the DMP in (1)-(2) as the composition of coupling terms  $C(\cdot) = \sum_i C_{\text{OA}}^i(\cdot) + C_{\text{HG}}^i(\cdot)$ , where  $C_{\text{OA}}^i(\cdot)$  and  $C_{\text{HG}}^i(\cdot)$  generate the corresponding forcing terms with respect to the  $i^{\text{th}}$  obstacle in the scenario. This composition allows, in a single-obstacle scenario (see Figure 4a), to guide the reactive behaviour (blue trajectory) in a different direction (red trajectory) by temporarily defining the initial desired heading towards the upper part of the task space. The same applies to multi-obstacle environments (see Figure 4b), where the system's heading can be modified at multiple decision points to obtain a preferred route (green trajectory). In both scenarios, the actuation scope of the coupling term for guiding the system was set manually for illustration purposes. Alternatively, these decision points could be defined by a task-dependant module.

### D. Proof of Lyapunov's Stability

The addition of coupling terms can imperil the inherent stability properties of DMPs [2]. Authors in [10] proved with Lyapunov's theory that the overall dynamical system remains stable when the coupling terms generate a forcing term orthogonal to the system's velocity vector. The coupling terms formulated in (8) and (9) satisfy this condition, therefore proving the global stability of the proposed action level.

## IV. LEARNING OBSTACLE AVOIDANCE FOR NON-POINT OBJECTS

The set of coupling terms formalised in the previous section efficiently generates guided collision-free trajectories for point-mass objects, i.e. obstacles and systems. Nonetheless, objects in real-world scenarios present different shapes and sizes. This section details the encoding of objects as low-dimensional geometric descriptors, which allows for (i) the design of a learning module that regulates the action level to generalise over different obstacle geometries while considering the system's geometry, and (ii) the use of heuristics to rapidly perform route selection in constrained environments.

### A. Superquadrics as Geometric Approximates

Objects obstructing the execution of a policy might present different shapes and dimensions. This geometric diversity complicates the design of an intelligent module able to generalise obstacle avoidance behaviours across geometries [11]. This work considers global features to approximate the geometric properties of an object. One possible encoding strategy are superquadrics [12], which have been used, among others, to ease the computation of system-obstacle distances [13], and to generate repulsive potential fields [14]. Alternatively to these task space applications, this work is interested in the low-dimensional parametric encoding of such geometric approximate, which is defined as:

$$F(x, y, z, \lambda): \left( \left( \frac{x}{\lambda_1} \right)^{\frac{2}{\lambda_5}} + \left( \frac{y}{\lambda_2} \right)^{\frac{2}{\lambda_5}} \right)^{\frac{\lambda_5}{\lambda_4}} + \left( \frac{z}{\lambda_3} \right)^{\frac{2}{\lambda_4}}, \quad (10)$$

where  $F(\cdot)$  defines whether a given 3D point  $(x, y, z)$  lies inside ( $F < 1$ ), outside ( $F > 1$ ), or on the surface ( $F = 1$ ) of a superquadric described by  $\lambda = [\lambda_1, \dots, \lambda_5]$ . In particular,  $(\lambda_1, \lambda_2, \lambda_3)$  set the superquadric semi-axes lengths, and  $(\lambda_4, \lambda_5)$  parameters define the superquadric shape.

The parameter vector  $\lambda$  can be estimated from a discrete representation of the obstacle's surface by minimisation of:

$$\min_{\lambda} \sum_{i=1}^N \left( \sqrt{\lambda_1 \lambda_2 \lambda_3} (F(x_i, y_i, z_i, \lambda) - 1) \right), \quad (11)$$

where  $\sqrt{\lambda_1 \lambda_2 \lambda_3}$  penalises the fitting of large superquadrics.

### B. Unified Low-dimensional Geometric Descriptors

The process in (10)-(11) provides a geometrical descriptor  $\lambda$  from a discrete representation of an object. However, it is of interest to obtain a descriptor accounting for both the system's and obstacle's geometry. Figure 5 schematises the extraction of a unified obstacle-system low-dimensional geometric descriptor. An approximate of the system's geometry (see blue prism in Figure 5a) is used to dilate [15], [16] the obstacle's discrete representation (see Figure 5b). The dilated obstacle representation is then encoded using (11) while imposing  $\lambda_4 = \lambda_5 = 1$ , i.e. restricting the superquadric to shape as an ellipsoid. Figure 5c portrays the significance on the descriptor's difference when considering the raw obstacle representation (blue ellipsoid) and its dilated version (rose ellipsoid). Interestingly, ellipsoids hold the property that any random projection or section of these results in an ellipse, providing a strategy to extract the unified obstacle-system's geometric features relevant to the obstacle avoidance coupling term. This is, the P-plane defined by the respective obstacle-system position  $\mathbf{x}_{\text{obstacle}} - \mathbf{x}$  and the system's heading  $\dot{\mathbf{x}}$ , intersects the unified geometric approximation. Thus, the descriptor  $\lambda$  can be further reduced to  $\lambda' = (\lambda'_1, \lambda'_2) \in \mathbb{R}^2$  such that  $\lambda' = g(\lambda)$  where  $g(\cdot): \mathbb{R}^5 \rightarrow \mathbb{R}^2$  maps an arbitrary vector onto the P-plane. The resulting low-dimensional descriptor is an ellipse laying on the P-plane with semi-axis lengths  $\lambda'$  (see Figure 5d).

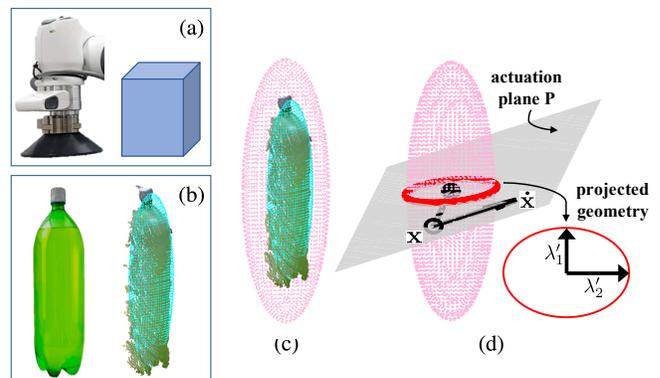


Fig. 5: Extraction of unified low-dimensional descriptors accounting for the (a) end-effector's and (b) obstacle's geometry. (c) Ellipsoid (rose) fitting the dilated obstacle cloud. (d) Relevant descriptor  $\lambda'$  along the P-plane (red ellipse).

### C. Geometry-conditioned Parameter Regressor

Leveraging the unified low-dimensional descriptor  $\lambda'$  from Section IV-B, this section proposes a method to learn the correspondence between  $\lambda'$  and the non-independent parameters  $(\alpha, \psi, \kappa)$  of the coupling term, subject to a user-defined clearance  $\Delta$ , i.e. the minimum distance between the end-effector and the obstacle. This multiple target regression problem is formulated as a regressor chain (RC) [17], which defines an ordered chain  $U = (Y_1, Y_2, Y_3)$  of single target regressions. This is, given an input vector  $\mathbf{h} = \{\lambda', \Delta\}$ , the proposed RC-based learning module is composed of three models:  $Y_1: \mathbf{h} \rightarrow \kappa$  adjusts the actuation span of the coupling term,  $Y_2: (\mathbf{h}, \kappa) \rightarrow \psi$  regulates the relevance of the relative system-obstacle heading, and finally  $Y_3: (\mathbf{h}, \kappa, \psi) \rightarrow \alpha$  tunes the strength of the behaviour. Each regressor  $Y_i$  is modelled as a NN which provides a powerful strategy to learn and represent approximations to non-linear mappings, and is suitable for reactive decisions due to its rapid response. Considering the relevance of the input features, each NN regressor is arranged with four layers; the hidden layers are hyperbolic tangent sigmoid units, and the output layer is a log-sigmoid to avoid negative settings of the targets.

It should be noted that the regulation of the action level formalised in Section III is conducted along the P-plane. As explained previously in Section IV-A, this sub-space contains all essential information to circumnavigate an obstacle and is efficiently defined using the relative system-obstacle state. Namely, changes in the obstacle avoidance scene such as different start and goal positions, obstacle location and geometries do not alter the encoding of the problem in the P-plane. Therefore, the prediction capabilities of the designed RC-based learning module extend to a wide range of setups, including in the presence of multiple obstacles in the scene.

### D. Route Selection via Heuristic Cost Rings

Real-world environments and physical systems constrain the amount of feasible reactive behaviours. Exhaustively evaluating all possible directions in  $SO(3)$  which satisfy these additional constraints can slow the decision response. To ease the reasoning complexity of the route selection problem, this work proposes a twofold heuristic analysis called cost rings which (i) considers an orthographic projection of the obstacle onto the YZ-plane  $\in \mathbb{R}^2$  of the local frame, i.e. confining the direction space  $\omega \in SO(2)$ , to then efficiently (ii) find the obstacle avoidance direction  $\omega_d$  minimising a metric  $\eta(\omega)$ . The resulting direction  $\omega_d$  is used with the coupling terms composition formulated in Section III-C to guide the obstacle avoidance behaviour towards  $\omega_d$ .

The advantage of route selection via heuristic cost rings is exemplified in Figure 6, where the path cost  $\eta(\omega)$  is determined according to three metrics: (i) the physical constraints imposed by the table  $\eta_{table}(\omega)$ , (ii) the length of the trajectory  $\eta_{length}(\omega)$ , and (iii) the robot's workspace limit  $\eta_{limits}(\omega)$ , such that  $\omega_d$  can be found by minimisation of:

$$\min_{\omega} \eta_{table}(\omega) + \eta_{length}(\omega) + \eta_{limits}(\omega), \quad (12)$$

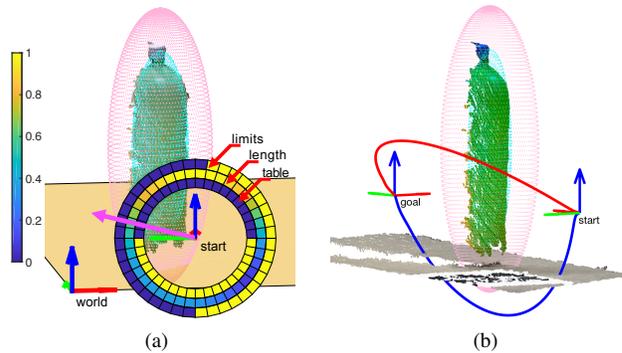


Fig. 6: Route selection via heuristic rings. (a) Cost evaluation on the local YZ-plane to penalise workspace limits, long trajectories, and collisions with the table. Overall best direction is marked in magenta. (b) A reactive behaviour (blue) would lead the system colliding with the table, whereas the guided behaviour (red) generates the route with lowest cost.

where  $\eta_{table}(\omega) = 1$  if the end-effector would collide with the table and 0 otherwise,  $\eta_{length}(\omega) \in [0, 1]$  is the normalised trajectory length, and  $\eta_{limits}(\omega) = 1$  if the end-effector would move outside of its workspace and 0 otherwise. Figure 6a illustrates these estimated costs rings and the resulting direction  $\omega_d \in \omega$  (magenta) with minimum cost. As depicted in Figure 6b, using this reasoning to initially guide the behaviour enables the system to avoid the obstacle in the direction with lowest cost (red), whereas the non-guided reactive behaviour leads with collision with the table (blue).

### E. Convergence to Goal

The required path  $\hat{\pi}$  to avoid obstacles may be longer than the pre-planned trajectory  $\pi$ , thus needing more time to finalise the encoded task. This fact is especially critical when dealing with non-point objects as failing to account for this can imperil convergence to the desired goal [11]. To address this issue, this work regulates the DMP duration by scaling  $\tau = length(\hat{\pi})/length(\pi)$ , i.e. an approximate of the increase of trajectory length. Here,  $length(\hat{\pi})$  is estimated with linear interpolation of the finite sequence of  $\mathbb{R}^3$  points  $\{\mathbf{x}_s, \mathbf{x}_p^1, \dots, \mathbf{x}_p^N, \mathbf{g}_x\}$ , where  $\mathbf{x}_s$  and  $\mathbf{g}_x$  are the start and goal positions, and  $\mathbf{x}_p^i$  is the extreme point of the ellipse encoding the  $i \in [1, N]$  dilated obstacle's geometry along its P-plane.

## V. EXPERIMENTAL EVALUATION

The proposed framework has been evaluated in simulated environments and on a physical system. This section first explains the training of the RC model via exploration of the parameter space. Thereafter, it reports the performance and generalisation capabilities of the proposed approach in familiar and novel obstacle avoidance settings. Finally, this section details the deployment of the proposed framework on an anthropomorphic Franka Emika Panda arm engaged in a start-to-goal policy in the presence of unplanned obstacles.

An extended illustration of the experimental evaluation is documented in: <https://youtu.be/1ym5cCbjI3k>,

and the corresponding source code can be found in: [https://github.com/ericpairet/ral\\_2019](https://github.com/ericpairet/ral_2019).

### A. Training the RC-based Learning Module

This work has designed a RC-based learning module to regulate the action level according to a unified obstacle-system descriptor  $\lambda'$  and a possible clearance constraint  $\Delta$ . The unconstrained model is denoted as  $RC(\lambda')$ , while the constrained model is referred to as  $RC(\lambda', \Delta)$ . The training of these models is conducted leveraging the knowledge of the action level to create a synthetic dataset via exploration of the parameter space. This is, given different obstacle avoidance scenarios, training explores the parameters  $\{\alpha, \psi, \kappa\}$  of the coupling term (8) generating a collision-free trajectory.

Bearing in mind that the learning module uniquely regulates the action level along its plane of actuation, 100 synthetic scenarios were created to simulate possible intersections between a unified system-obstacle ellipsoid approximation and the actuation plane  $P \in \mathbb{R}^2$ . This resulted in 100 ellipses parameterised with semi-axis values  $\lambda' = (\lambda'_1, \lambda'_2)$  uniformly sampled in the range 2.5 to 25cm. Each of these sections was placed in the middle of a one-metre length start-goal baseline. For each scenario, a set of trajectories were generated using (8) with a  $50 \times 50 \times 50$  grid of the parameters  $\{\alpha, \psi, \kappa\}$ . Only those input-target  $\{(\lambda'_1, \lambda'_2), (\alpha, \psi, \kappa)\}$  pairs involving a collision-free trajectory were integrated into the dataset along with the resulting clearance.

The RC architectures were trained using a 70% of the synthetic dataset. Each NN was trained independently using the Levenberg-Marquardt algorithm with a random initialisation of the weights and biases. The remaining 30% of the dataset was used to test the performance of the trained RC models. Since the aim of a RC model is to reduce the prediction error on every single target [17], each model  $Y_i$  was validated by computing the normalised mean squared error (NMSE) on the training and testing sets. As shown in Table I, the parameter prediction error of the models reduces significantly when considering the clearance in the input vector  $\mathbf{h}$ . This is because the clearance allows differentiating the influence of the targets among all possible collision-free trajectories. It is worth noting that the performance of the RC does not deteriorate when being evaluated on the test set.

### B. Experiments on Familiar Scenarios

The performance of both  $RC(\lambda')$  and  $RC(\lambda', \Delta)$  models in the P-plane space was evaluated for the same obstacle geometries as in the training dataset, i.e. 100 ellipses. For the  $RC(\lambda', \Delta)$  model, the considered constraints on the clearance

	NMSE( $Y_1$ )		NMSE( $Y_2$ )		NMSE( $Y_3$ )	
	train	test	train	test	train	test
$RC(\lambda')$	0.539	0.543	0.802	0.802	0.893	0.897
$RC(\lambda', \Delta)$	0.251	0.253	0.244	0.243	1.6e-4	1.7e-4

TABLE I: Prediction error on every single target of the two modelled RC architectures for the training and test datasets.

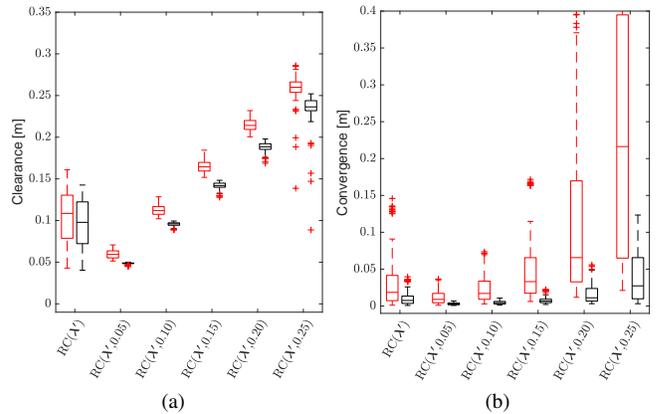


Fig. 7: (a) Clearance and (b) convergence of the avoidance behaviours generated in familiar scenarios, when scaling the trajectory duration (black plots) and when not (red plots).

were  $\Delta = \{0.05, 0.1, 0.15, 0.2, 0.25\}$  metres. All six models were evaluated with and without scaling the trajectory duration  $\tau$  according to its estimated length as explained in Section IV-E. Overall, this led to the testing of the RC architecture under 12 different settings. Performance in the P-plane space was evaluated for the metrics (i) number of collisions, (ii) minimum distance to an obstacle (clearance), and (iii) distance to goal (convergence). The obtained results over the 1,200 scenarios are illustrated in Figure 7.

Figure 7a and Figure 7b respectively represent the clearance to the obstacle and convergence to the goal for the 1,200 scenarios evaluated across the 12 settings of the RC architecture. Overall, constraining the model with a desired clearance leads to more bounded behaviours. However, as the clearance constraint increases, the convergence rapidly deteriorates for those models not scaling the trajectory duration (red boxes). Instead, when scaling the time (black boxes), the convergence is at most of 3cm for the most constrained model  $RC(\lambda', 0.25)$ . This fact highlights the importance of scaling the time when larger trajectories are required. Indifferently from the model setup, none of the 1,200 conducted tests resulted with a trajectory colliding with an obstacle. The remainder of the experimental evaluation is conducted with the  $RC(\lambda', 0.15)$  model and scaling the trajectory duration according to its estimated length.

### C. Experiments on Novel Scenarios

Given the variety of obstacle avoidance scenarios that a system may face in the real-world, the proposed  $RC(\lambda', 0.15)$  model was evaluated for its performance and generalisation capabilities on scenarios not seen during the training process. Notably, the approach was tested for its suitability to deal with three-dimensional (3D) obstacles via the extraction of relevant unified low-dimensional geometrical features laying on the P-plane as described in Section IV-A.

Novel 3D scenarios were created by sampling the location and dilated geometry of the obstacle randomly. The obstacle was arbitrarily located along the x-axis between the start and goal configurations preserving 5cm of margin, and around the

	Clearance to obstacle [m]		Convergence to goal [m]		$n^\circ$ of collisions
	mean	min	mean	max	
Goal at 0.5m	0.144	-5.01e-4	4.41e-4	0.017	2
Goal at 1.0m	0.184	0.060	4.23e-4	0.017	0
Goal at 1.5m	0.196	0.068	5.22e-4	0.023	0
Goal at 2.0m	0.202	0.076	6.49e-4	0.027	0

TABLE II: Clearance, convergence and number of collisions of the trained  $RC(\lambda', 0.15)$  model for 4,000 novel settings.

baseline between -0.4 and 0.4m along both the y-axis and z-axis. The unified system-obstacle ellipsoid approximation had random width, height and length within the spectrum 5 to 50cm, leading to representative candidates of possible object geometries in real-world environments. This spectrum corresponds to semi-axis values  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  laying in the range 2.5 to 25cm. These boundaries also ensured that none of the extracted low-dimensional features  $\lambda'$  would result beyond the limits for which the RC model was trained for.

A 1,000 novel 3D scenarios were created for the different start-to-goal baselines of 0.5, 1.0, 1.5 and 2.0m along the x-axis of the local frame, adding up to a total of 4,000 evaluations. The semi-axis  $\lambda_1$  was limited to a maximum of 20cm for the baseline of 0.5m to be consistent with the 5cm margin across experiments. All environments required the action level to modulate a start-to-goal policy to avoid collision and preserve the desired clearance. Out of the 4,000 tests, 1,296 environments already had the baseline in collision with the obstacle. The performance of  $RC(\lambda', 0.15)$  on the unseen settings was evaluated for the metrics (i) number of collisions, (ii) clearance to an obstacle, and (iii) convergence to goal. Table II summarises the extracted metrics across the evaluation, and Figure 8 depicts the performance of the proposal on some novel single and multi-obstacle settings.

Results in Table II reflect the performance of the designed  $RC(\lambda', 0.15)$  model when dealing with 3D scenarios via their section on the P-plane. The overall success rate is of 99.95% on novel scenarios while providing, in average, a clearance similar to the requested one of 0.15m and a close convergence to the goal. This implies an enhancement of 31.75 times over the success rate reported on known objects in [11]. However, the performance of the approach is slightly compromised in some scenarios, obtaining clearances of 6cm

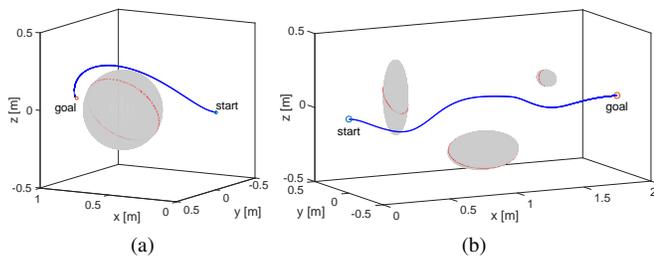


Fig. 8: Generalisation capabilities of the trained  $RC(\lambda', 0.15)$  model in novel settings. Parameters  $\lambda'$  are extracted from the relevant section (red ellipse) where the coupling term acts.

and convergences up to 2.7cm. The proposed approach could not cope uniquely with two scenarios out of 4,000, where the generated trajectory penetrated 0.501mm an obstacle of 40cm along the x-axis and 50cm along the y-axis and z-axis placed in the middle of a 0.5m long baseline. Albeit these extreme scenarios for which more data could be provided at training time, the proposed approach has proved to generalise not only to different object sizes and locations, but also to different start-to-goal baselines. Further experimentation also showed the suitability of the framework to deal with multi-obstacle scenarios (see Figure 8b). Since the action level is referenced in a local frame (see Section II-A), the performance of the framework does not deteriorate regardless of the local frame's pose in the task space. Within the local frame, the outstanding generalisation capabilities are mainly due to regulating action according to the relative system-obstacle state defining the P-plane, and extracting relevant system-obstacle low-dimensional geometrical descriptors.

#### D. Experiments on a Robotic Platform

The proposed hierarchical framework for obstacle avoidance has been deployed on an anthropomorphic 7-DoF Franka Emika Panda arm operated with OROCOS [18]. The DMP-encoded system's transient behaviour is converted to joint configurations using a Cartesian inverse dynamic controller with null space optimisation. The environment is partially observed with a depth camera ASUS Xtion previously calibrated with Aruco markers [19]. The acquired point cloud is processed applying standard filtering techniques to segment the clusters describing obstacles and the table. The partial observation of each obstacle is dilated to also account for the system's geometry (see Section IV-B). The location of the table is used to constrain the reactive behaviour along the upper part of the task space (see Section IV-D).

As in [10], [11], the test-bed consisted of obstacles interrupting a straight trajectory underlying a start-to-goal policy. However, differently than [10], [11], the assortment of considered obstacles had not been seen before. This included, but was not limited to, regular objects, such as the cardboard box in Figure 1, irregular objects, such as the pile of plastic bottles in Figure 9c, and also aleatory combinations of them, such as the cluttered environment with six obstacles in Figure 9a and Figure 9b. As summarised in Table III, the robot engaged in the pre-planned policy (blue trajectories)

		Clearance to obstacle [m]	Convergence to goal [m]	$n^\circ$ collisions & $\Delta < 0.15$ m
regular (Fig. 1)	pp.	fail	fail	1 & 1
	mod.	0.168	9.58e-3	0 & 0
irregular (Fig. 9c)	pp.	fail	fail	1 & 1
	mod.	0.163	4.17e-3	0 & 0
multi-obs (Fig. 9b)	pp.	fail	fail	2 & 5
	mod.	0.172	1.29e-2	0 & 0

TABLE III: Clearance, convergence, number of collisions and unsatisfied clearances for the pre-planned (pp.) and modulated (mod.) start-to-goal policies on the real robot.

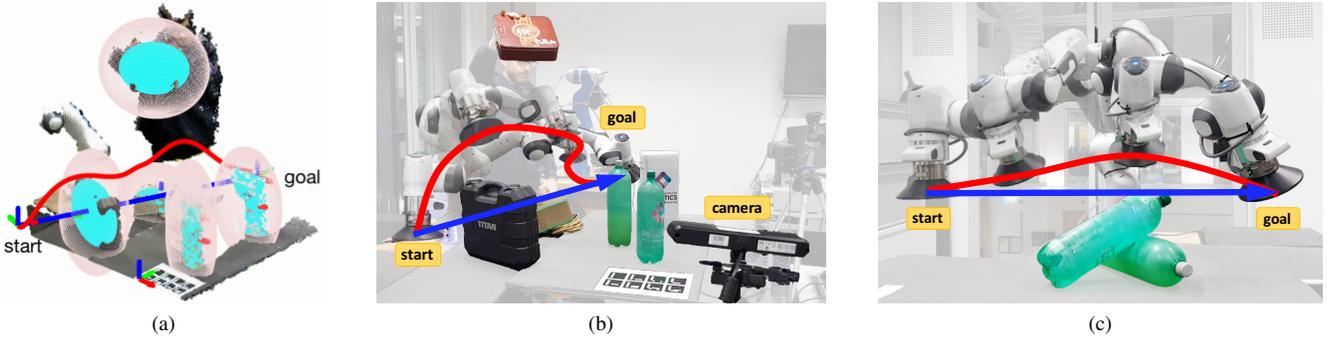


Fig. 9: Panda arm engaged in a start-to-goal policy (blue trajectories) while modulating its behaviour (red trajectories). (a) Environment perception with unified low-dimensional encoding of the system’s and obstacle’s geometry (rose ellipsoids). Proposed hierarchical framework dealing with (b) multiple obstacles in a cluttered environment, and (c) an irregular obstacle.

would impact with the obstacles. Instead, endowing the robot with the ability to modulate such policy, allows the system to successfully circumvent all obstacles with the desired 0.15m clearance while converging to the goal (red trajectories).

The presented results demonstrate that the proposed hierarchical framework which (i) extracts relevant geometric descriptors, (ii) uses them in the designed RC-based learning module to (iii) regulate the DMP-based action level, endows a system with the ability to modulate its behaviour in settings never seen before, while stably converging to the goal.

## VI. FINAL REMARKS

This paper has presented a biologically-inspired hierarchical framework which safely modulates an on-going policy to avoid obstacles. The proposed approach follows a multi-layered perception-decision-action analysis which (i) extracts unified system-obstacle low-dimensional geometric descriptors, then (ii) exploits them to rapidly reason about the environment with a combination of heuristics and learning techniques, and finally (iii) guides and regulates the obstacle avoidance behaviour with a conjunction of coupling terms modulating a DMP-encoded policy. Experimentation conducted in synthetic environments highlights this method’s generalisation capabilities to confront novel scenarios at the same time of ensuring the convergence of the system to the goal. Additionally, real-world trials on an anthropomorphic manipulator demonstrated the framework’s suitability to successfully modulate a policy in the presence of multiple novel obstacles described by partial visual-depth observations, while satisfying a user-defined clearance constraint.

The proposed framework is not restricted to the presented experimental evaluation nor platform. Any robotic system following a DMP-encoded policy can benefit from this work to safely modulate its behaviour in the presence of unexpected obstacles. Similarly to [7], collisions of the links can also be considered by finding the closest geometric section on the robot to the obstacle, and then modulating the kinematic null-space movement with the proposed approach. An interesting venue for future work is to modulate the system’s orientation policy to overcome an obstacle, which,

for instance, might have a significant impact on a manipulator carrying a large bulk. Another interesting extension of this work is learning route selection priorities in cluttered environments, so systems can autonomously reason about the most convenient direction to avoid an obstacle.

## REFERENCES

- [1] B. R. Fajen and W. H. Warren, “Behavioral dynamics of steering, obstacle avoidance, and route selection,” *Journal of Experimental Psychology: Human Perception and Performance*, vol. 29, no. 2, p. 343, 2003.
- [2] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, “Dynamical movement primitives: learning attractor models for motor behaviors,” *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013.
- [3] A. Gams, A. J. Ijspeert, S. Schaal, and J. Lenarčič, “On-line learning and modulation of periodic movements with nonlinear dynamical systems,” *Autonomous robots*, vol. 27, no. 1, pp. 3–23, 2009.
- [4] A. Gams, B. Nemeč, A. J. Ijspeert, and A. Ude, “Coupling movement primitives: Interaction with the environment and bimanual tasks,” *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 816–830, 2014.
- [5] G. Sutanto, Z. Su, S. Schaal, and F. Meier, “Learning sensor feedback models from demonstrations via phase-modulated neural networks,” in *IEEE International Conference on Robotics and Automation*, pp. 1142–1149, 2018.
- [6] É. Pairet, P. Ardón, F. Broz, M. Mistry, and Y. Petillot, “Learning and generalisation of primitives skills towards robust dual-arm manipulation,” in *AAAI Fall Symposium on Reasoning and Learning in Real-World Systems for Long-Term Autonomy*, pp. 62–69, 2018.
- [7] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, “Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields,” in *IEEE-RAS International Conference on Humanoid Robots*, pp. 91–98, 2008.
- [8] S. M. Khansari-Zadeh and A. Billard, “A dynamical system approach to realtime obstacle avoidance,” *Autonomous Robots*, vol. 32, no. 4, pp. 433–454, 2012.
- [9] 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 *IEEE International Conference on Robotics and Automation*, pp. 2587–2592, 2009.
- [10] A. Rai, F. Meier, A. Ijspeert, and S. Schaal, “Learning coupling terms for obstacle avoidance,” in *IEEE-RAS International Conference on Humanoid Robots*, pp. 512–518, 2014.
- [11] A. Rai, G. Sutanto, S. Schaal, and F. Meier, “Learning feedback terms for reactive planning and control,” in *IEEE International Conference on Robotics and Automation*, pp. 2184–2191, 2017.
- [12] A. Barr, “Superquadrics and angle-preserving transformations,” *IEEE Computer graphics and Applications*, vol. 1, no. 1, pp. 11–23, 1981.
- [13] V. Perdureau, C. Passi, and M. Drouin, “Real-time control of redundant robotic manipulators for mobile obstacle avoidance,” *Robotics and Autonomous Systems*, vol. 41, no. 1, pp. 41–59, 2002.

- [14] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous robot vehicles*, pp. 396–404, Springer, 1986.
- [15] T. Lozano-Perez, "Spatial planning: A configuration space approach," in *Autonomous robot vehicles*, pp. 259–271, Springer, 1990.
- [16] L. Huber, A. Billard, and J.-J. Slotine, "Avoidance of convex and concave obstacles with convergence ensured through contraction," *Robotics and Automation Letters*, vol. 4, no. 2, pp. 1462–1469, 2019.
- [17] E. Spyromitros-Xioufis, G. Tsoumakas, W. Groves, and I. Vlahavas, "Multi-target regression via input space expansion: treating targets as inputs," *Machine Learning*, vol. 104, no. 1, pp. 55–98, 2016.
- [18] H. Bruyninckx, "Open robot control software: the orocos project," in *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2523–2528, 2001.
- [19] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.