

# Dynamic Movement Primitives for Human-Robot interaction: comparison with human behavioral observation\*

Miguel Prada<sup>1</sup>, Anthony Remazeilles<sup>1</sup>, Ansgar Koene<sup>2</sup> and Satoshi Endo<sup>2</sup>

**Abstract**—This article presents the current state of an ongoing work on Human-Robot interaction in which two partners collaborate during an object hand-over interaction. The manipulator control is based on the Dynamic Movement Primitives model, specialized for the object hand-over context. The proposed modifications enable finer control of the dynamic of the DMP to align it to human control strategies, where the contributions of the feedforward and feedback parts of the control are different to the original DMP formulation. Furthermore, the proposed scheme handles moving goals. With these two modifications, the model no longer requires an explicit estimation of the exchange position and it can generate motion purely reactively to the instantaneous position of the human hand. The quality of the control system is evaluated through an extensive comparison with ground truth data related to the object interaction between two humans acquired in the context of the European project CogLaboration which envisages an application in an industrial setting.

## I. INTRODUCTION

Passing an object between people is a commonly performed action that increases productivity by efficiently sharing the work load. Recently, the importance of a robotic assistant which can support an object handover has been advocated [1]. This Human-Robot collaboration is the core of the European project CogLaboration in which we are developing a service robot that is optimized for natural Human-Robot object hand-over interactions, based on observations of Human-Human collaboration.

Whilst the majority of existing studies on object hand-over define the passer or receiver as initiating and leading the task (e.g. [5]; [10]; [13]), object hand-over is more accurately described as a fluid interaction with both passer and receiver taking progressively the lead.

In rare or specific cases, the passer might offer an object at a specific location, with the receiver moving the arm to this position to fetch the object. In other cases, the receiver might request an object at a fixed location, forcing the passer to place the object in the receivers hand. In real object hand-over, pairs of participants usually avoid either extreme, and instead fluidly negotiate the hand-over involving some overlap in the movements of the two partners.

Thus, a mixture of feedforward and feedback control, respectively guided by and leading the person, is required to

coordinate the passer and receiver hands in time and space for successful object hand-over.

The feedforward part of the control system needs some prior information about the type of movement to perform, and this can be effectively handled by using *learning by demonstration* techniques. There is a broad range of such methods for learning robot motions from human demonstration, either through motion capture data or by actually driving the robot by hand. They often involve statistical modeling to extract the essence of a given task from a series of demonstrations, as is the case of [9] where *Hidden Markov Models* are used to learn to recognize and reproduce full body motions by observing a human through a monocular camera. Similarly, [2] presents an approach which uses *Principal Component Analysis*, *Gaussian Mixture Models* and *Gaussian Mixture Regression* to extract the meaningful characteristics of a task amongst several demonstrations, and to be able to reproduce it in different contexts.

The use of dynamical system for synthesizing robot motions has been gaining popularity, with works such as the one presented in [8], where an approach to guarantee obstacle avoidance is presented for controllers based on dynamical systems. The flexibility and robustness provided by the use of dynamical systems to control the robot motions make them a perfect candidate to use in learning motions by demonstration, as exemplified by the widely known *Dynamic Movement Primitives* (DMP) technique. This method, initially introduced in [7], proposes to base the robot movement on a simple second order linear system and learn an arbitrary non-linear *forcing term* which stimulates this system to follow the demonstrated trajectory. Several lines of work have spun from this technique, with some of the most interesting ones working on using the DMP based learning to bootstrap reinforcement learning methods, as shown in [15] and [3].

In this paper, a specialization of the DMP technique for object hand-over, already presented in [12], is benchmarked by using a collection of motion capture sequences of two humans exchanging an object in different configurations related to an industrial context. For each configuration, the performance of the system is repeatedly evaluated by training the desired motion with one movement sequence and testing the system responses to the rest of the samples in the same configuration, using the information of the human motion as the ground truth. Given the high sensitivity of the proposed method with respect to the specific trajectory used for learning, further tests are performed with learning not from a specific sequence but from the mean of all the sequences, which greatly improves the system outcome.

\*This work was supported in part by the European project CogLaboration under contract FP7-ICT-7-2.1-287888, and by the Spanish project Fluent under contract DPI 2009-13653.

<sup>1</sup>M. Prada & A. Remazeilles are with the Health Division of Tecnalia Research and Innovation, Parque Tecnológico de San Sebastián, E-20009 Donostia - San Sebastián, Spain name.surname@tecnalia.com

<sup>2</sup>A. Koene & S. Endo are with the School of Psychology, University of Birmingham, UK {s.endo, a.koene}@bham.ac.uk

The paper starts by describing the base DMP formulation and the modifications proposed by the authors. Section III then describes the details of the proposed benchmark and the results obtained when applying the modified DMP technique to the available dataset. Finally conclusions and potential future lines of work are described in Section IV.

## II. DYNAMIC MOVEMENT PRIMITIVES

### A. Generic Framework

The Dynamic Movement Primitives (DMP) framework can be used to learn a trajectory from a single reference sample. It can then be reproduced and optionally adapted to different configurations. This is achieved by using a second order linear dynamical system (i.e. a damped spring-like model) which is stimulated with a non-linear forcing term.

Let  $x(t)$  denote a one-dimensional trajectory starting at  $x(t_0) = x_0$  and ending at  $x(t_f) = g$ . The DMP model to reproduce such a trajectory can be expressed using the following transformation system:

$$\tau \dot{v} = sK \left( \frac{f(s)}{s} + x_0 - x \right) + (1-s)K(g-x) - Dv \quad (1a)$$

$$\tau \dot{x} = v \quad (1b)$$

with the forcing term  $f$  representing an arbitrary non-linear function as a sum of weighted exponential basis functions:

$$f(s) = \frac{\sum_{i=1}^N \psi_i(s) w_i}{\sum_{i=1}^N \psi_i(s)} s, \quad (2)$$

and:

$$\psi_i(s) = \exp(-h_i(s - c_i)^2). \quad (3)$$

The exponential evolution from 1 to 0 of the phase variable  $s$  is determined by the following decoupled linear system, called the canonical system:

$$\tau \dot{s} = -\alpha s \quad (4)$$

The phase variable is used to remove the direct time dependency of the forcing term  $f(s)$ , thus making the system time-scalable by adjusting the parameter  $\tau$ .

The system can be interpreted as two attractor fields with a damping term:

- $K(g-x)$  is an attractor towards the goal position (from now on referred to as the *goal-attractor*). This term is the one that ensures convergence towards the desired position.
- $K \left( \frac{f(s)}{s} + x_0 - x \right)$  represents an attractor towards the moving point  $\frac{f(s)}{s} + x_0$  (the *shape-attractor*). This term encapsulates the learned trajectory and stimulates the system to mimic this trajectory.

Each of these attractor fields is weighed according to the evolution of the phase variable: the *shape-attractor* is predominant at the beginning of the movement, when  $s \approx 1$ ; while the *goal-attractor* becomes predominant towards the end of the movement, as  $s \rightarrow 0$ .

When considering multi-dimensional trajectories, either the complete system above needs to be replicated, or, as

proposed in [6], a common canonical system can be used for all dimensions, with specific transformation systems for each dimension. In our experiments, we use the latter case.

### B. Limitations of the state of the art model

Despite the fact that the generic framework provides great learning and motion reproduction capabilities, we consider that it presents some drawbacks in the specific application we are considering.

The Human-Robot collaboration through object hand-over requires producing a human-friendly and fluent robot motion. A first step in this direction is to start the robot motion as soon as the human partner has signaled, verbally or non-verbally, the desire for a hand-over, without waiting for the human to reach at the exchange location. In most cases, this requires an estimation of the exchange location, as proposed in [14]. Nevertheless, in spite of such estimation, performing a successful exchange still requires a feedback mechanism to update the motion and ensure that the two partners converge towards the same exchange position.

The DMP model already has such a feedback mechanism, the *goal-attractor*. Furthermore, this feedback should not affect the system at the beginning of the trajectory, so we can avoid the initial estimation step all-together and set the DMP goal directly to the position of the moving hand of the human partner.

However, as we have shown in [12], the use of the phase variable as weight makes the *goal-attractor* active way before the human hand has reached the exchange location, thus greatly disturbing the reproduction of the learned behavior.

In order to fit with human behavior [4], we proposed to change the weights of the two components of the DMP system, to obtain a predominantly feedforward control policy at an earlier phase of the motion (not disturbed by the *goal-attractor*) and a predominantly visual feedback control towards the end of the motion, as the visual cues for the exchange position become more precise.

### C. Specialization for human robot object exchange

To permit a flexible control of the transition between *shape-* and *goal-attractor* dominance, we propose to decouple the weights applied to each of the terms in the transformation system from the phase variable. Instead of weighting each component directly with the phase variable, an arbitrary function of the phase variable can be used to compute the desired weights. This function enables to flexibly set when the transition occurs.

$$\tau \dot{v} = (1 - w_g)(f_w + x_0 - x) + w_g K(g-x) - Dv \quad (5a)$$

$$\tau \dot{x} = v \quad (5b)$$

$$\tau \dot{s} = -\alpha s, \quad (5c)$$

where  $f_w(s)$  is now defined as:

$$f_w(s) = \frac{\sum_{i=1}^N \psi_i(s) w_i}{\sum_{i=1}^N \psi_i(s)}. \quad (6)$$

A sigmoid function, similar to the Cumulative Distribution Function (CDF) of the normal distribution, is chosen as the

weighing function  $w_g(t)$ . This function has the advantage of relying on two parameters which intuitively determine when the shift will occur (the mean  $\mu$  of the normal distribution) and the duration of the transition (the standard deviation  $\sigma$  of the distribution). The expression of this function is, substituting the dependency on  $s$  for dependency on time:

$$w_g(t) = 0.5 \left[ 1 + \operatorname{erf} \left( \frac{t - \mu}{\sigma\sqrt{2}} \right) \right], \quad (7)$$

where  $\operatorname{erf}$  stands for the Gauss error function.

Fig. 1 shows how the parameters of the weighing function influence the transition between the two attractors. The black curve is the reference trajectory learned by the DMP system. This learned trajectory is then reproduced with an altered goal using three different values of the mean parameter  $\mu$  of the weighing function. As can be seen, a larger  $\mu$  delays the response to an altered goal, which is clearly observable by the time difference in the resulting trajectories diverging from the reference trajectory.

By using the Decoupled DMP formulation proposed in (5), the moment where the change of goal affects the output of the DMP algorithm can be adjusted.

Compared to the regular use of the DMP in which the goal is constant or varies rarely, our application involves a constantly moving goal (the human partner hand). We thus propose a second modification of the model by adding a velocity feedback term to the transformation system:

$$\tau \dot{v} = (1 - w_g)(f_w + x_0 - x) + w_g[K(g - x) + K_v \dot{g}] - Dv \quad (8)$$

The velocity term permits to take the target motion direction into consideration in the feedback controller and thus it can provide a smoother behavior and convergence. The model described in (8) will be the one evaluated in Section III.

#### D. Trajectory learning

The learning procedure requires setting the parameters:

- $K$  and  $D$  involve the inherent dynamics of the second-order linear system, and determine its response to online changes in the goal.

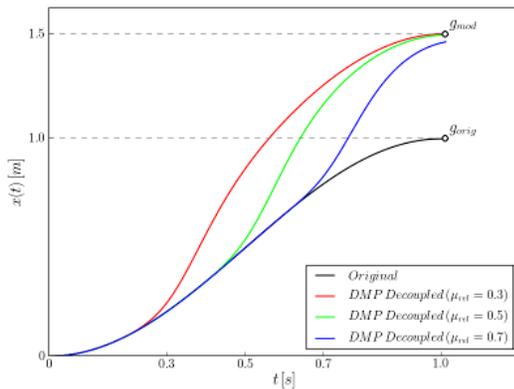


Fig. 1: Decoupled DMP with different values of  $\mu$  and  $\sigma = 0.05$ .

- $t$  is the time constant and should be set to the duration of the sample trajectory  $t = t_f - t_0$ .
- $\alpha$  determines the decay rate of the phase variable. It should be chosen so that  $s \rightarrow 0$  at  $t = \tau$  in order to ensure convergence towards the goal.

Once these values are set (according to the application needs), the next step is to compute the desired values for the forcing term, by isolating it from (8):

$$f_w(s) = \frac{1}{1 - w_g} (\tau \dot{v} - w_g [K(g - x)] + Dv) + x - x_0. \quad (9)$$

and then inserting the values of the sample trajectory  $x = x(t)$ ,  $v = \tau \dot{x}(t)$  and  $\dot{v} = \tau \ddot{x}(t)$ . With these desired values for the forcing term, the appropriate centers and widths of the basis exponentials in (3) can be set, and the weights  $w_i$  can be computed by fitting the right hand terms in (6) and (9). We are currently using a technique based on Sparse Bayesian Models and the Relevance Vector Machine as described in [16]. This method allows to perform a sparse regression, where a function can be fitted using only a few basis functions out of a large pool of basis function candidates.

### III. COMPARISON WITH THE COGLABORATION HUMAN MOTION CAPTURE DATA

#### A. Database description

The previously introduced approach has been compared to the human behavior observed during Human-Human object exchange. This section describes the data acquisition procedure.

The experiment was conducted using pairs of participants replicating the industrial setting presented in [11] with the following three configurations (see Fig. 2):

- working in the engine bay,
- lying under a car,
- working under a hydraulic lift.

In the experiment, one person took the role of a mechanic (P1) while the other person took the role of an assistant (P2). For our simulations the cognitive controller replaces P2 and the simulated trajectory in response to the motion of P1 will be evaluated with respect to the real motion generated by P2 as the *ground-truth*.

A single trial consisted of the following actions:

- P2 picks up a tool from a table and hands it over to P1.
- P1 simulates performing a prescribed tool usage (tapping a load cell) using the given object.
- P1 returns the object to P2.

Each trial in the Human-Human experiment therefore consisted of two object hand-overs exchanges and they were separately analyzed; one for P1 receiving the object and the other for P2 retrieving the object. The first exchange was defined from the time when P2 lifted the object to when the object was fully handed over to P1. The second exchange began when P2 started moving to retrieve the object from P1, following the prescribed tool usage action, and finished when P2 touched the object in the hand of P1. The task was repeated 5 times per configuration, giving a total of 15

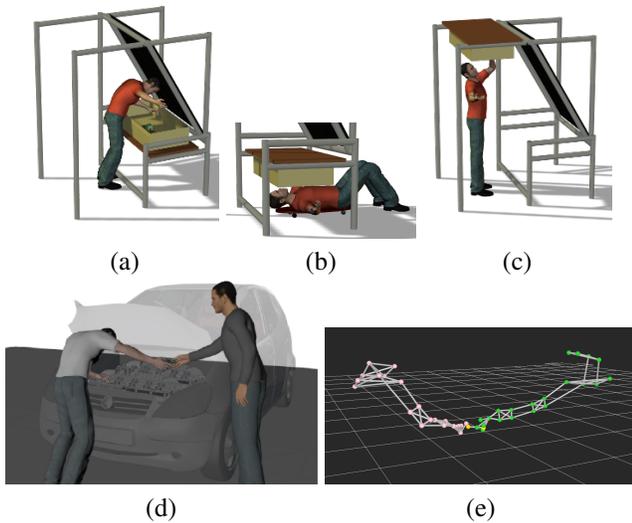


Fig. 2: Postural configurations for the car mechanic scenario; (a) in the engine bay, (b) lying under a car, and (c) under a hydraulic lift. The person shown here illustrates the reference positions in which P1, the *mechanic* is placed. P2, whose position will be taken by the robot, is the assistant standing next to the car for an interaction similar to the one in (d). An illustration of motion tracking data of the object hand-overs are presented in (e).

trials. A 12-camera Oqus motion-tracking system (Qualisys, Sweden) tracked light-weight spherical markers, at 200Hz, placed on the right arms of both participants to record the position and orientation of the arms during the object hand-over. For the purpose of the current study, only the wrist positions of the two participants were analyzed.

### B. Database preparation

The current evaluation is focused on the reaching motion towards the exchange location. The human wrist motion is used to define the reference trajectory for the robotic wrist in Cartesian space. The orientation of the end effector as well as the robot's redundant DoF are to be handled independently based on other constraints yet to be determined (e.g. specific orientation constraints may apply for some types of container objects). The end effector orientation constraints may affect the feasibility of the Cartesian trajectory, but this will only happen in the vicinity of the robot's workspace boundaries or at kinematic singularities, where some position/orientation configurations are not reachable. As long as the trajectory stays well within these limits it should be possible to set the Cartesian position and orientation of the end effector independently without the two sets of constraints interfering.

For each task configuration, five motion capture traces similar to the ones presented on Fig. 3 have been considered. Each trace provides two exchange interactions:

- *Give*: P2 gives the object to P1 (first exchange).
- *Take*: P2 takes the object back from P1 (second exchange).

As shown in Fig. 3, the three Cartesian dimensions are considered separately. The orientation of the coordinate system in this case was fixed to the workspace with the Z-axis pointing opposite to gravity, the X- and Y-axis pointing along the length and width of the imaginary car, roughly to the front and towards the left of P1, respectively. The position displayed is always related to the partners wrist 3D position.

As mentioned in section II-A, if each dimension has its own transformation system, the canonical system is common for all dimensions (the transition time between the feed forward and feedback control is the same in the three dimensions).

From Fig. 3, we can notice that in some cases the motion of P1 (red lines) is really limited (in particular in the *Give* exchange, first gray zone), as his hand was already at the desired exchange location when P2 started moving. This is due to the fact that in some cases, the passer P2 starts by picking up the object from a remote site and thus the receiver arrives at the exchange location earlier. This is not the case in the *Take* exchange (Fig. 3, second gray zone), however.

The evaluation procedure is systematic: for the *Give* and *Take* cases, each sequence of P2 (green curves) was consecutively used as reference for the trajectory learning through the mechanism presented in section II-D. The learned behavior was then used to simulate a response behavior to each P1 movement as defined by the five trials of P1 data traces (red curves) of a single task configuration. The resulting simulated movements were compared with the observed motion capture trace that P2 performed in the corresponding experiment trial (green curves). In this comparison, for each configuration, five object hand-over trials were considered for the performance evaluation (we also compared the behavior of the cognitive controller against the same sequence used for learning).

### C. Selected Samples

Fig. 4, 5 and 6 illustrate the type of data obtained during this analysis. These figures are related to the gray shaded areas in Fig. 3, showing the X,Y and Z wrist traces as a function of time. The red trajectory (labeled *observed*) corresponds to the motion capture trace for P1. The motion generated by our *decoupled DMP* algorithm, labeled *DMP* is represented with the blue colored line. This trajectory is to be compared with the observed motion capture trace for P2, shown in green line and labeled *compare*.

In all these three figures, the left column of panels corresponds to the replication of the training data trace that was learned by the *decoupled DMP*. The right column presents how this learned behavior responds to a P1 movement trace from another trial in our data set.

On the left column, where the cognitive controller responded to the same stimulation that produced the reference trajectory that has been learned, the two sequences (*compare* and *DMP*) do not overlap precisely. On the initial part of the motion, where the *shape-attractor* is predominant, the difference is due to the number of Gaussian functions used in (6) to learn the trajectory. The deviation observed in the

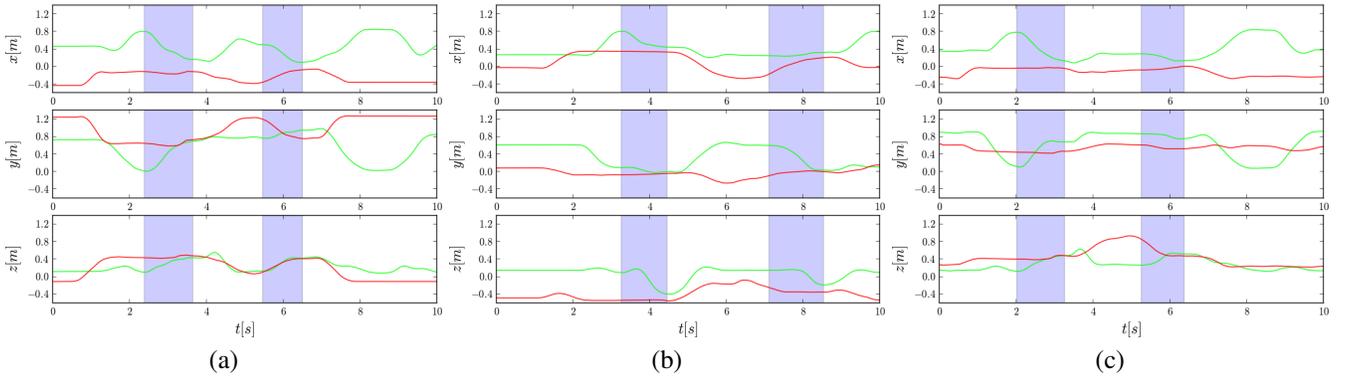


Fig. 3: A sequence of the wrist trajectories in the three configurations across time: (a) *Engine bay*; (b) *Lying under car*; (c) *Hydraulic lift*. The red and green lines refer to the movements of P1 and P2, respectively. The highlighted areas are periods of the *Give* and *Take* exchanges.

later part of the motion may result from two reasons. On one hand, during the learning procedure, the dynamics of the observation were not considered. Therefore, only the trajectory of P2 was used, and the goal was set to the ending point of this reference trajectory. During the online reproduction, the goal was set to the current observed position of P1 (offset according to the two reference end points to fill the gap). Furthermore, as proposed in Section II-C, during the feedback phase the dynamics of the observation were explicitly compensated by the velocity-dependent control term. This is visibly demonstrated on the Z-axis in Fig. 4 around time 1.0. The simulated DMP movement trace (blue line) follows the shape of the P1 trace (red lines) since the *goal-attractor* is predominant. This suggests that during the feedback phase, the human adaptation to the perceived motion is not as strong as expected, at least when the perception does not diverge too far from the scheduled behavior.

Fig. 4, 5 and 6 clearly show that the simulated motion (blue curves) presents a similar shape to the ground-truth data presented in green. The DMP model nicely adapts to motion variations of the observation (red curve), as it can be seen in particular on the X and Z dimensions of Fig. 5 and 6. The analysis of the Y dimension of Fig. 6 demonstrates the great variability of the human behavior during hand-over interactions: both P1 and P2 motions differ in a significant manner in the two presented sequences. Despite of these differences, the DMP model succeeds in converging towards the exchange location, with a trajectory consistent with the learned model.

#### D. Statistical analysis

To evaluate the proposed model, a statistical analysis was carried out to compare similarities between the real and simulated trajectories. In order to focus on the spatial relationship between the paired trajectories, the data were first normalized in time.

Each one of the five trials per configuration was separated into the constituent *Give* and *Take* exchanges and used to train a *decoupled DMP* controller which was then used to respond to five recorded P1 trajectories, giving a total of 25

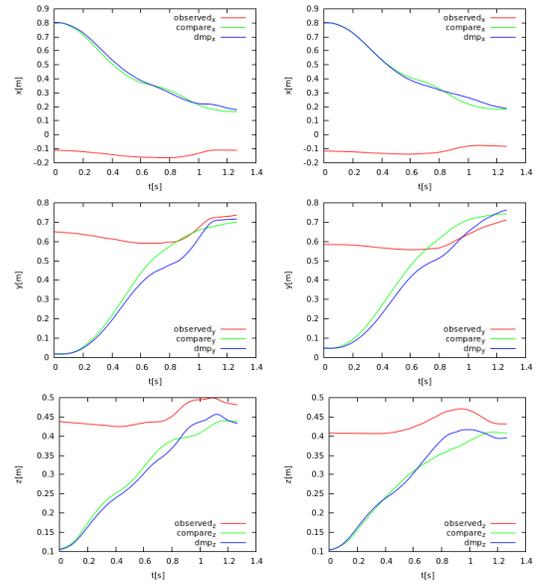


Fig. 4: *In engine bay* configuration, P2 gives to P1. Sequence #1 learned (left), compared with sequence #2 (right)

combinations per configuration.

For each configuration, we computed the average and standard deviation ( $\sigma$ ) of the trajectories to test how the simulated trajectories fell within the natural variability of the recorded P2 trajectories.

Fig. 7 to 9 present the average and standard deviations of the time-normalized trajectories in each configuration. The left column shows trajectories during the *Give* exchange, and the right column shows those during the *Take* exchange. The blue lines represent the mean trajectory of the respective configuration. The dark shaded regions show the  $1\sigma$  distance to the mean at each data point, and the lighter shaded regions indicate the  $3\sigma$  boundary. The green lines correspond to the generated trajectories that remain inside the  $3\sigma$  boundary the whole time, while the red curves are the generated motions that go out of this  $3\sigma$  boundary for at least one iteration.

The simulation of the *Give* exchange in the *Engine bay*

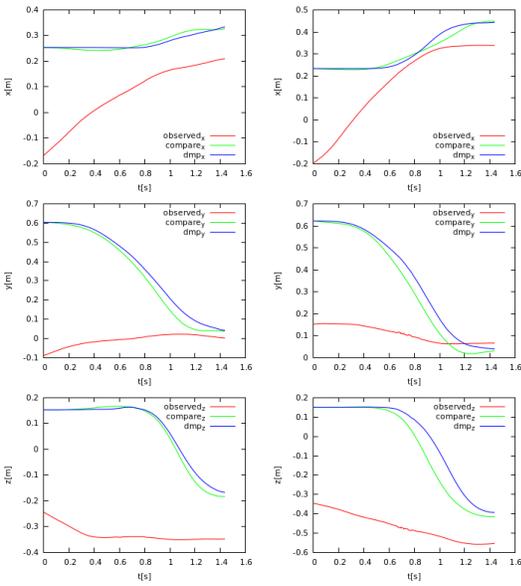


Fig. 5: *Lying under the car* configuration, P2 takes from P1. Sequence #1 learned (left), compared with seq. #3 (right)

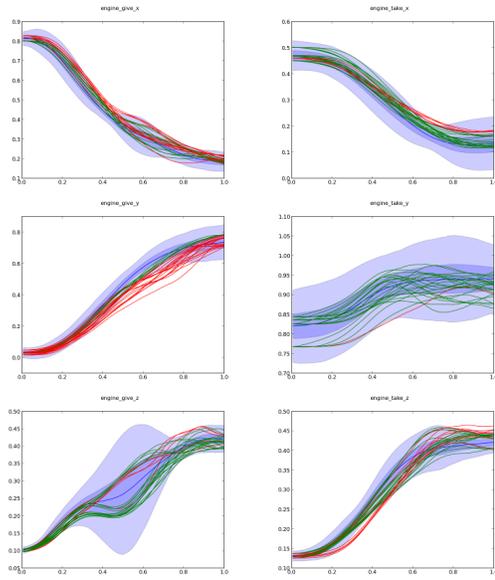


Fig. 7: In *engine bay* configuration, *Give & Take* exchanges

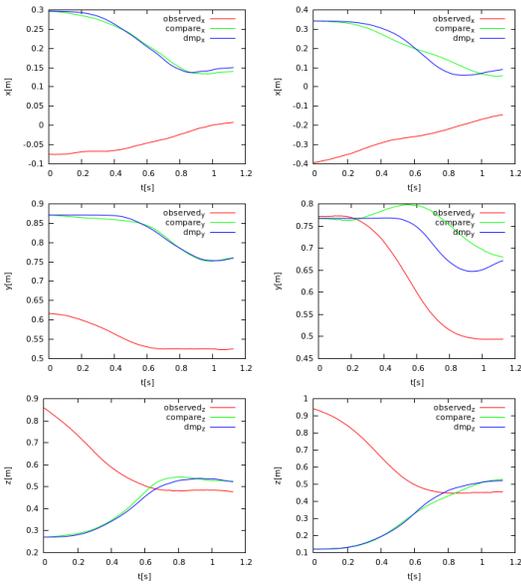


Fig. 6: *Hydraulic lift* configuration, P2 takes from P1. Sequence #1 learned (left), compared with sequence #3 (right)

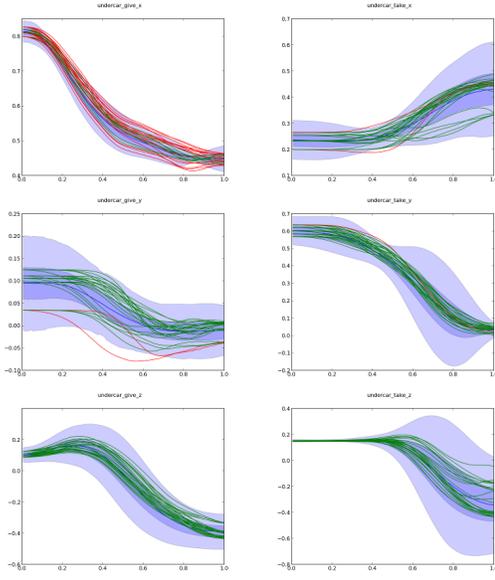


Fig. 8: *Lying under car* configuration, *Give & Take* exchanges

configuration (see Fig. 7) performed worst in this evaluation which is likely due to the small variability in the observed human movement sequences; around time 0.4, the  $3\sigma$  envelope is narrower than 4 cm on the X axis. These "bottlenecks" in the envelopes are assumed to be probabilistic artifacts due to the small number of available test trajectories. Similarly, on the Y dimension, the "authorized distance" to the mean is also quite narrow between  $t=0.4$  and  $0.7$ . At that time, even if the rejected curves are at most 20 cm away from the mean, one can see that the dynamic of the generated trajectory is similar to the sample trajectories. These cases will need to be evaluated based on qualitative feedback from users as it

is difficult to judge how such a deviation affects the user's perception of exchange fluency.

The *Take* exchange of the *Engine bay* configuration presents statistically much better results on all three dimensions. Most of the rejected trajectories were obtained using sequence #5 as training sequence. A closer inspection indicated that the Z-axis trajectory of sample #5, very similar to a heavy-side transition observed in the other samples, presented a transition slope that was much higher than the others, and it was not a representative trajectory.

On the *lying under car Give* exchange, almost all the rejected curves (16 of 18) were seen in the X dimension. In this configuration, a narrow boundary around time 0.85 is the

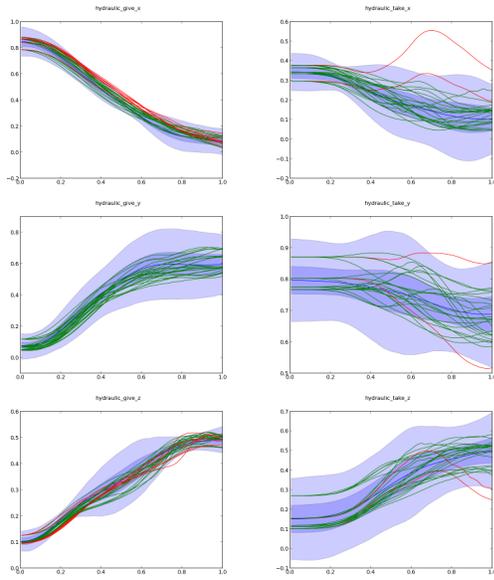


Fig. 9: Hydraulic lift configuration, Give & Take exchanges

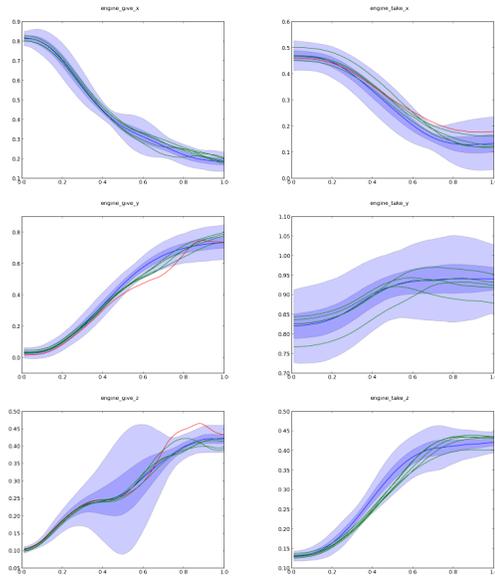


Fig. 10: Engine bay configuration, Give & Take exchanges, mean behavior learned

reason behind the statistical rejection of those curves, despite the fact that they were less deviated than 5 cm from the mean trajectory. On the other hand, the simulated trajectories were much more satisfactory for the other dimensions.

On the *hydraulic lift Give* exchange, the reference sequences #1 and #5 produced 14 of the 16 rejections, mainly on the X and Z axes (in a quite narrow boundary, with a  $3\sigma$  of respectively less than 10 and 5 cm). The rest of the trajectories were quite close to the observed mean trajectory.

As already stated, this systematic study considered the behavior of the controller irrespective of the quality of the training trajectory. As such, the result could indeed be improved by selectively choosing the most representative reference trajectory for the training. This is demonstrated in the next section.

#### E. Learning a mean behavior

Since the result of the DMP controller is very dependent on the trajectory that is used for learning, an additional set of simulations was carried out using the mean trajectories of each case as an input to the learning system. The information provided on Fig. 10, 11 and 12 follows the same procedure as in the previous section, considering that now only one trajectory is learned per case.

As expected, these drawings show an improved performance of the *decoupled DMP* controller when using a mean trajectory as reference. Similarly to the previous observation, trajectory rejection is mainly due to the narrow band-width of the  $\sigma$  envelope (see X dimension of *Engine-Give* on Fig. 10, X dimensions on Fig. 11 and X dimension of *Hydraulic-Take* on Fig. 12), as well as the too strong dependency on the varying goal in the feedback mode that is produced in some situations (like on the Y and Z dimension of *Engine-Give* on Fig. 10, the X dimension of *UnderCar-Give* on Fig. 11 and the Z dimension of *Hydraulic-Give* on Fig. 12).

Table I summarizes the performance of the *decoupled DMP* controller for the three configurations and the two exchange directions by listing the percentage of trajectories respectively inside the  $1\sigma$ ,  $2\sigma$  and  $3\sigma$  envelopes. It can be noted that some of the configurations in the *Learn mean* case give slightly worse results than the ones obtained when learning individual trajectories. However, the difference is so small that it cannot be considered statistically significant, given the scarcity of the available samples in the *Learn mean* with respect to the *Learn single* set. Furthermore, the last row demonstrates that with higher confidence the overall quality is increased when the model used the mean trajectory as reference.

## IV. CONCLUSIONS

We have presented an analysis of a modified DMP model, specialized for the Human-Robot collaboration through object hand-over. Our model enables explicit control of the transition between the feedforward and feedback parts of the system, and takes moving goals into consideration. Its quality has been studied by comparing its behavior with the ground truth data of object hand-over between two humans.

The model studied in this paper is currently being implemented on a robotic platform consisting of a Kuka Lightweight Robot and a Kinect sensor. The DMP model with the proposed variations is used to generate on-line Cartesian motions for the robot's wrist, based on the feedback provided by a fusion of different tracking algorithms which provide the human hand position. Additional orientation constraints are added to the Cartesian position command by using priority-based inverse kinematics, and the resulting joint velocity command is then fed to the robot controller through an on-line interpolation system. The current implementation is focused on the reaching motion, and preliminary

trials with the project team show promising results. Extensive testing with naive participants will be carried out in the context of the CogLlaboration project to perform a qualitative and quantitative evaluation.

## REFERENCES

- [1] R. Bischoff and T. Guhl. The strategic research agenda for robotics in europe [industrial activities]. *Robotics Automation Magazine, IEEE*, 17(1):15–16, 2010.
- [2] S. Calinon, F. Guenter, and A. Billard. On learning, representing, and generalizing a task in a humanoid robot. *IEEE transactions on systems, man, and cybernetics*, 37(2):286–298, April 2007.
- [3] Christian Daniel, Gerhard Neumann, and Jan Peters. Learning concurrent motor skills in versatile solution spaces. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3591–3597, October 2012.
- [4] M. Desmurget and S. Grafton. Forward modeling allows feedback control for fast reaching movements. *Trends in cognitive sciences*, 4(11):423–431, November 2000.
- [5] M. Huber, M. Rickert, A. Knoll, T. Brandt, and S. Glasauer. Human-Robot Interaction in Handing-Over Tasks. In *IEEE International Symposium on Robot and Human Interactive Communication*, pages 107–112, 2008.
- [6] A. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal. Learning Nonlinear Dynamical Systems Models. *Neural Computation*, pages 1–33, 2010.
- [7] A. Ijspeert, J. Nakanishi, T. Shibata, and S. Schaal. Nonlinear dynamical systems for imitation with humanoid robots. *IEEE-RAS International Conference on Humanoid Robots*, 2001.
- [8] Seyed Mohammad Khansari-Zadeh and Aude Billard. A dynamical system approach to realtime obstacle avoidance. *Autonomous Robots*, 32(4):433–454, March 2012.
- [9] D. Lee and Y. Nakamura. Mimesis scheme using a monocular vision system on a humanoid robot. In *IEEE International Conference on Robotics and Automation*, pages 10–14, April 2007.
- [10] A. H. Mason and C. L. MacKenzie. Grip forces when passing an object to a partner. *Experimental Brain Research*, 163:173–187, 2005.
- [11] G. Pegman. Report on scenarios, tasks and goals. Technical Report Deliverable D2.10, European Project Cogllaboration, FP7-ICT-7-2.1 (No. 287888), 2012.
- [12] M. Prada and A. Remazeilles. Dynamic Movement Primitives for Human Robot Interaction. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, workshop on Robot Motion Planning: online, reactive and in Real-time*, Algarve, Portugal, 2012.
- [13] S. Shibata, K. Tanaka, and A. Shimizu. Experimental analysis of handing over. In *IEEE International Workshop on Robot and Human Communication*, pages 53–58, Tokyo, Japan, 1995.

Scenario	Distance	Learn single		Learn mean	
		Give	Take	Give	Take
Engine bay	$\leq 1\sigma$	1.3%	9.3%	0.0%	20.0%
	$\leq 2\sigma$	10.7%	41.3%	33.3%	40.0%
	$\leq 3\sigma$	52.0%	80.0%	86.6%	93.3%
Under car	$\leq 1\sigma$	13.3%	24.0%	33.3%	33.3%
	$\leq 2\sigma$	53.3%	68.0%	66.7%	66.7%
	$\leq 3\sigma$	76.0%	96.0%	86.7%	93.3%
Hydraulic lift	$\leq 1\sigma$	8.0%	24.0%	33.3%	33.3%
	$\leq 2\sigma$	52.0%	77.3%	73.3%	93.3%
	$\leq 3\sigma$	78.7%	92.0%	86.7%	93.3%
Summary	$\leq 1\sigma$	13.3%		25.6%	
	$\leq 2\sigma$	50.4%		62.2%	
	$\leq 3\sigma$	79.1%		90.0%	

TABLE I: Distances to the mean for the generated curves

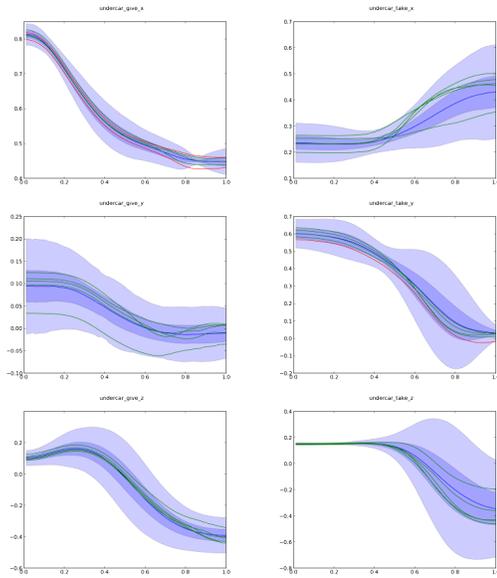


Fig. 11: Lying under car configuration, Give & Take exchanges, mean behavior learned

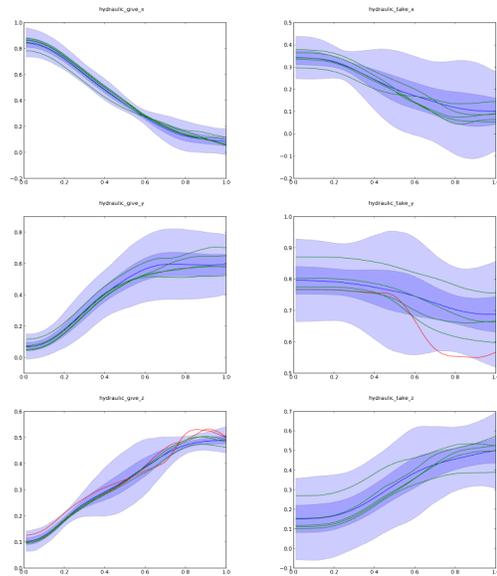


Fig. 12: Hydraulic lift configuration, Give & Take exchanges, mean behavior learned

- [14] E. Sisbot, L. Marin-Urias, X. Broquere, D. Sidobre, and R. Alami. Synthesizing robot motions adapted to human presence. *International Journal of Social Robotics*, 2(3), 2010.
- [15] F Stulp. Adaptive exploration for continual reinforcement learning. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.
- [16] M. Tipping and A. Faul. Fast marginal likelihood maximisation for sparse Bayesian models. In *International Workshop on Artificial Intelligence and Statistics*, Key West, USA, 2003.