Workshop on Robot Motion Planning:
Online, Reactive, and in Real-time
2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems, IROS 2012
Vilamoura, Algarve, Portugal, October 7-12, 2012

Dynamic Movement Primitives for Human Robot interaction

Miguel Prada and Anthony Remazeilles*

***Abstract*— A specialization of the generic *Dynamic Movement Primitives* (DMP) framework is proposed in this article to correctly address a key activity for human robot collaboration that is object exchange. As a first step towards implementing this challenging skill, this paper focuses on the arm motion to reach the initially unknown exchange site. Two improvements related with this application are proposed. First of all a better control of the transition in between the two mains components of the DMP –respectively providing a skill *shape-attractor* and a *goal-attractor*– is described, enabling to define when and how the transition in between these two components occur. Then an extension to handle situations where the goal position varies along time is proposed, which improves the convergence of the trajectory towards a moving target (i.e. the human partner's hand). These two improvements are validated by comparing the obtained behavior with human observations realized through motion capture.**

I. INTRODUCTION

The realization of robotic tasks in non completely controlled environment requires to provide the robotic system with a motion control scheme that adapts its behavior to the observed situation. Sensor-based approaches such as visual servoing [3] define the control law as a closed loop minimization of the error observed in between the current and desired visual feature values. Depending on the framework used, the robot motion can be optimal in the configuration space or in the image feature space. However, these approaches, in their basic versions, are strongly goal-driven and do not allow reproducing more complex skills in which the whole motion profile is as important as the convergence towards the goal.

The learning of complex behaviors can be addressed by *programming by demonstration* approaches, in which the robot imitates a task demonstrated either by a human operator observed with a motion capture system, or by manually moving the robot itself.Statistical approaches are frequently used for the learning. In [11], *Hidden Markov Models* are used to recognize and reproduce nine different full body expressions by a simulated humanoid. Calinon *et al.* propose in [2] to combine *Gaussian Mixture Models* and *Gaussian Mixture Regression* to reproduce several grasping tasks taught through kinesthetics. The *Dynamic Movement Primitives* (DMP) method is another approach studied in that field. Initially introduced by Ijspeert *et al.* [10], the DMP approach relies on a non-linear dynamical system forced to follow a desired trajectory by a parametric forcing term. It is proposed in this article to specialize the basic DMP framework to the special case of human-robot interaction during an object transfer.

Physical human-robot interaction, and specifically object exchange, is a key aspect to get a fluid and efficient human robot collaboration. Several recent works are focusing on this specific situation: [6] shows that the human partner can reduce the complexity of this task by adapting to the robot behavior; [8] implements different velocity profiles for the robot, and compares the results with human-human exchange procedures; direct *vs.* indirect (placing the object on a flat surface for the person to grasp) exchange procedures are compared in [4]; and [1] focuses mostly on making the robot transmit the intent of performing an exchange. In [13] the concrete exchange procedure is handled within an off-line planning scheme. The A^* algorithm is used to estimate the best trajectory to exchange the object with the human partner, based on a $3D$ cost map which combines three cost functions focused on safety, visibility and arm convenience criteria. Once the optimal exchange path is obtained, the actual trajectory to follow is computed with the Soft Motion Trajectory planner, allowing active control of maximum jerks, accelerations and velocities [12]. Nevertheless, the obtained trajectory plan is not explicitly driven by the human observation, and neither designed to adapt to the partner behavior, which is something inherent to the DMP approach proposed here. It is furthermore proved here that the initial stage of exchange location can be skipped by adapting accordingly the DMP framework.

This paper is proposing a DMP specialization for realizing human robot object exchanges. As a first step, the focus is set on the definition of the control system to bring the robotic arm towards the exchange site. Two improvements of the basic DMP framework are proposed, in relation with the exchange application. The first one is related to a better control of the transition between the feed-forward and feedback components of the DMP by introducing a custom weighing function. The second one addresses the dynamic nature of the goal position in exchange motions; a velocity based feedback term is appended to the DMP system which improves convergence with the moving goal.

This present paper is organized as follows: next section provides the needed background related to the DMP. Section III describes the two extensions proposed, and the last section compares the resulting scheme's behavior with real human-human exchange data recorded with motion capture equipment.

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

M. Prada and A. Remazeilles are with the Assistive Technologies Department of the Health Division of Tecnalia Research and Innovation, Parque Tecnológico de San Sebastin, E-20009 Donostia - San Sebastián, Spain

II. DYNAMIC MOVEMENT PRIMITIVES

A. Original formulation

The DMP framework learns a trajectory from just one reference sample. It can then reproduce it and optionally adapt it 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 $x(t_0) = x_0$ towards $x(t_f) = g$. In the original DMP framework the following system is introduced [9]:

$$\tau\dot{v} = K(g-x) - Dv + (g-x_0)f(s) \tag{1a}$$

$$\tau\dot{x} = v, \tag{1b}$$

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

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

and:

$$\psi_i(s) = exp(-h_i(s-c_i)^2). \tag{3}$$

The above dynamical system, named *transformation system* by the authors, is composed of two driving components, aside of the global damping term $-Dv$:

- $K(g-x)$ is an attractor towards the goal position.
- $(g-x_0)f(s)$ represents the contribution of the non-linear forcing term scaled by the $g-x_0$ factor.

The variable s on which the forcing term depends is a phase variable and its evolution is determined by the following decoupled linear system, called the *canonical system*:

$$\tau\dot{s} = -\alpha s \tag{4}$$

This variable evolves exponentially from 1 to 0. It is used to remove the direct time dependency of the forcing term $f(s)$, and provides the complete system with a time scalability by adjusting the parameter τ. The phase variable is also used to weigh the forcing term, enabling this way to continuously shift towards a purely goal-attracted system.

When considering multi-dimensional trajectories, either the complete system above needs to be replicated or, as proposed in [9], a common *canonical system* can be used for all dimensions, with specific *transformation systems* for each dimension.

B. Bio-inspired formulation

In [7], Hoffmann highlights that this formulation has scaling issues when the goal position g is close to the trajectory starting point x_0. Furthermore, this model does not adapt correctly to situations where the goal parameter is set to the opposite side of the trajectory origin x_0 with the respect to the original: the complete trajectory is then completely inverted. A slightly different bio-inspired model is thus proposed, based on evidence obtained on *in vivo* studies on frogs. This modified DMP formulation is:

$$\tau\dot{v} = sK(\frac{f(s)}{s} + x_0 - x) + (1-s)K(g-x) - Dv \tag{5a}$$

$$\tau\dot{x} = v \tag{5b}$$

Similarly, this system is mainly composed of two attractor fields:

- The term $K(g-x)$ is an attractor towards the goal position (from now on referred to as the *goal-attractor*).
- The term $K(\frac{f(s)}{s} + x_0 - x)$ represents an attractor towards the moving point $\frac{f(s)}{s} + x_0$ (the *shape-attractor*).

Each of these attractor fields has its influence weighed according to the evolution of the phase variable: the *shape-attractor*, weighed by s, is predominant in the beginning of the movement, when $s \approx 1$; while the *goal-attractor*, weighed by $(1-s)$, is predominant in the end of the movement, as $s \to 0$.

This formulation bypasses the issues arising when the goal is close to the origin of the trajectory, and vastly improves the adaptation to new goals since the *shape-attractor* does not scale anymore with $(g-x_0)$. Also, the addition of the x_0 component on the *shape-attractor* enables the system to behave properly when the initial starting point is changed. These two properties together make the system affine transform-invariant when learning multi-dimensional trajectories.

C. Trajectory learning

The learning procedure is the same in both models. The first step is to give values to the parameters of the system:

- K and D involve the inherent dynamics of the second order linear system, and determine its response to on-line changes in the goal parameter.
- τ is the time constant and should be set to the duration of the sample trajectory $\tau = t_f - t_0$.
- α determines the decay rate of the phase variable. A value $\alpha \approx 4$ will ensure that $s \approx 0.02$ at $t = \tau$.

Once these values are fixed, the next step is to compute the desired values for the forcing term, by isolating it from (5a) (or (1a) for the first formulation), which results in:

$$f_{des}(s) = \frac{1}{K}(\tau\dot{v} - K(g-x) + Dv + K(g-x_0)s) \tag{6}$$

and then inserting the values of the sample trajectory $x = x(t)$, $v = \tau\dot{x}(t)$ and $\dot{v} = \tau\ddot{x}(t)$, by taking into account the nominal evolution of the phase variable $s = exp(-\frac{\alpha}{\tau}t)$.

With these desired values for the forcing term, the appropriate centers and widths of the basis exponentials in (2) can be set, and the weights w_i can be computed by fitting (2) to (6) by least squares.

D. Limitations with respect to the intended application

Both the above formulations are quite sensitive to variations in the goal from the very beginning, as illustrated on Fig. 1, where a sample trajectory $x(t)$ (black solid line) is learnt and reproduced with the goal changed from 1 to 1.5 from the beginning. In the case of the original formulation (red curve), the contribution of g in both the *shape-attractor* and *goal-attractor* (see (1a)) makes both components scale when the goal is changed. In the case of the bio-inspired formulation, as it can be seen on (5a), the *shape-attractor* is not affected by the goal parameter. Nevertheless, by studying the evolution of the phase variable (Fig. 2)one can observe that more weight is given to the *goal-attractor* for $t > 0.173\tau$ (i.e. starting at less than 20% of the trajectory duration). Thus, from this early moment, any variation of the goal with respect to the reference one has a strong effect which overrides the influence of the *shape-attractor* term.

As previously mentioned, the application we are considering is the arm control during an object exchange with a human partner. The involvement of the human in the loop requires the robotic system to deal with the exchange location uncertainty. It also naturally constraints the robot motions to be human-friendly or fluent.

One of the means to improve the fluency of object exchange is to overlap the motion of the robot with the motion of the human partner, without waiting for the human to reach a stable position to start moving. A solution to achieve this is to launch the robot motion using an estimation of the exchange site, as proposed in [13]. Nevertheless, this initial guess would still need to be adjusted on-line to adjust the robot motions to the human behavior.

To avoid this initial estimation, we are proposing to set the DMP goal to the current position of the hand of the human partner from the beginning of the movement generation. This enables to ensure the convergence towards the exchange site (which is currently assumed to be the human's final hand location). Nevertheless, from this perspective, the fact that the DMP generator is too sensitive to alterations in the goal parameter is considered as a shortcoming, since the initial goal fed to the system can be quite different to the position reached by the non predictable human partner.

In addition, the analysis of the human behavior suggests that reaching motions performed by humans contain two successive components [5]:

- The onset of the movement is performed based on imperfect target information and mostly determined by an internal dynamical model and feed-forward control.
- The final part is dominated by visual feedback control, once the target position information gets more precise.

This evidence supports the objective of initiating the movement with a dominantly feed-forward control policy,

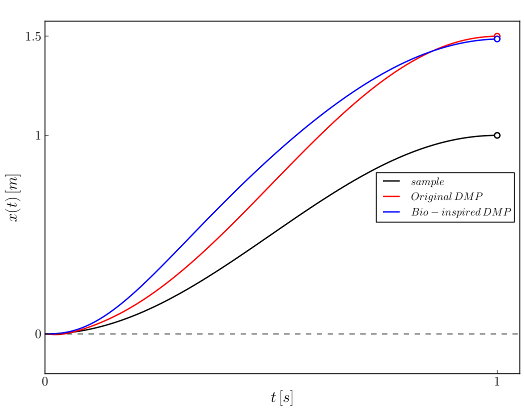

Fig. 1. Sensitivity of both the original (red curve) and the bio-inspired DMP (blue curve) generated motions with respect to a change in the goal. The black curve represents the learn trajectory.

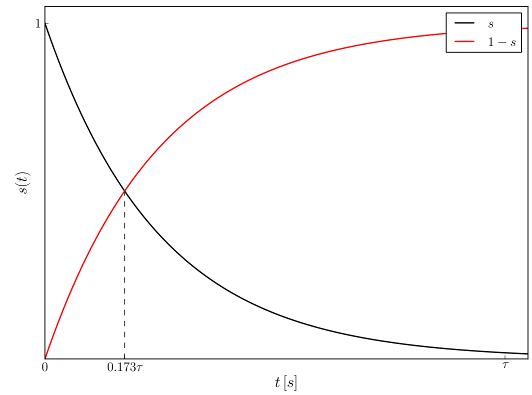

Fig. 2. Evolution of the weights of the *shape-attractor* (in black) and the *goal-attractor* (in red) within the original and bio-inspired DMP models.

and delaying the shift of weights towards the feedback component of the DMP *transformation system* to later in the trajectory. This way the first part of motion is mainly shape-driven, and less dependent on the goal variation, while the second part takes care of the convergence towards the goal. Next section presents the proposed modifications to the DMP method to achieve this desired behavior.

III. Extension of the DMP model

A. Decoupled weighing function

Two approaches are considered to modify the evolution of importance of each term driving the motion generation in the *transformation system*:

- A change in the evolution of the phase variable can change the weight balance between the two components. This can be used to delay the shift of importance from the *shape-attractor* towards the *goal-attractor*.
- A *decoupling* of the weights applied to each of the terms in the transformation system from the phase variable. Instead of weighing the attractors directly with the phase variable, an arbitrary function of the phase variable can be used to compute the desired weights.

The first approach proposed requires to find an appropriate substitute for the *canonical system* with the desired evolution, and in some cases this system might be difficult or even impossible to find without recurring to piecewise or unstable systems. Also, changing the evolution of the phase variable by means of altering the *canonical system* affects all the dimensions of the trajectory being reproduced by the DMP method.

The second approach is interesting in the sense that the *canonical system* can be kept in its original form. Furthermore, each of the *transformation systems* depending on the same phase variable can use a different weighing function if needed. Therefore it is decided to stick with this second approach which is considered more versatile.

The new system equations which use the *decoupling* approach proposed are ($f_w(s)$ and $w_g(s)$ are respectively noted f_w and w_g for notational compactness):

$$\tau\dot{v} = (1-w_g)(f_w + x_0 - x) + w_g K(g-x) - Dv \tag{7a}$$

$$\tau\dot{x} = v \tag{7b}$$

$$\tau\dot{s} = -\alpha s, \tag{7c}$$

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

$$f_w(s) = \frac{\sum_{i=1}^{N} \psi_i(s) w_i}{\sum_{i=1}^{N} \psi_i(s)}. \tag{8}$$

In comparison with (2) the phase variable s is not included in the (8) anymore, since it's not longer required for the forcing term to fade away.

Note that the gain K multiplying the *shape-attractor* in (5a) has been dropped as well, since it does not have any effect at all on the system response; this is obvious by observing how the desired values of the forcing term are computed with (6).

It is proposed to use a weighing function in the shape of a sigmoid similar to the Cumulative Distribution Function (CDF) of the Normal distribution. This function has the advantage of relying on two parameters which easily allow determining when the shift will occur (the mean μ of the Normal distribution) and the duration of the shift (the standard deviation σ of the distribution). Fig. 3 shows several variations obtained by changing these two parameters. The expression for the function is, substituting the dependency on s for dependency on time:

$$w_g(t) = 0.5\left[1 + erf\left(\frac{t-\mu}{\sigma\sqrt{2}}\right)\right], \tag{9}$$

where erf stands for the Gauss error function.

This weighing function has one problem, which becomes evident when the formula for the desired shape of the forcing term $f_w(s)$ is computed. To obtain the desired $f_w(s)$ one needs to isolate it from (7a), resulting in:

$$f_w = \frac{1}{(1-w_g)}(\tau\dot{v} - w_g K(g-x) + Dv) - x_0 + x \tag{10}$$

(where the dependence on s has been dropped again for compactness). It is easy to see that this expression tends to infinity as $w_g \to 1$, thus causing numerical issues. A simple product of the sigmoid function with a linear term (e.g. starting at 0.9 for $t=0$ and tending towards 1 as $t \to \tau$) solves this problem, while still ensuring that the *shape-attractor* influence is fading away when $s \to 0$ (i.e. $t \to \tau$). This results on the functions shown in Fig. 4.

By using the Decoupled DMP formulation proposed in (7a), (7b) and (7c), the moment where the change of goal affects the output of the DMP algorithm can be adjusted at will. Fig. 5 illustrates the behavior of the new formulation proposed. The original trajectory learnt as well as the value of the goal set during execution are the same as the one used in Fig. 1. Three trajectories are generated with different values of μ, showing how the system output is affected. In the three cases, the g parameter is set to its final value $g = 1.5$ from the beginning of the trajectory, but this only affects the trajectory at the chosen point in time. Notice that the rightmost case, with $\mu = 0.7$, switches to the *goal-attractor* too late for the trajectory to reach the goal at $t = \tau$, although it will reach it shortly after, since by that time the system is almost purely a stable linear second order system.

B. Adaptation for dynamic goals

As previously mentioned, and as a first simplification, the DMP goal is set to the position of the human partner's hand. If [13] proposes to realize an off-line estimation of the best exchange location, our approach presents the advantage of avoiding such estimation, while maintaining a reactive process so that the robot adapts to the human behavior and not the contrary.

However, in some cases, and even if the modification explained in the previous section is in place, the fact of using the human's current hand position as goal at each instant in the motion generation may introduce some undesired oscillations in the resulting trajectory. The example on Fig. 6 shows this effect with a set of data from real human motion. In this figure the black line shows the original trajectory used to learn the robot motion; the blue line shows the observed motion of the human partner, with whom the robot is performing the exchange operation; the red line represents the generated trajectory (with the DMP modifications

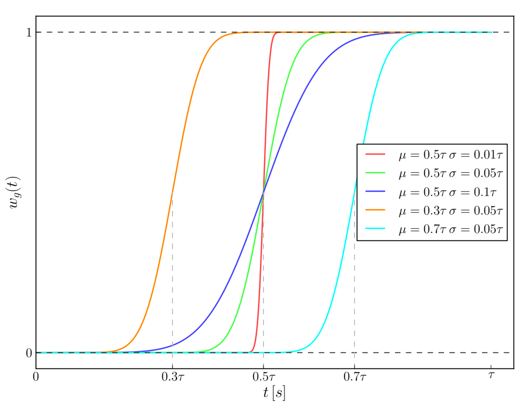

Fig. 3. Weighing function $w_g(t)$ with different sets of parameters

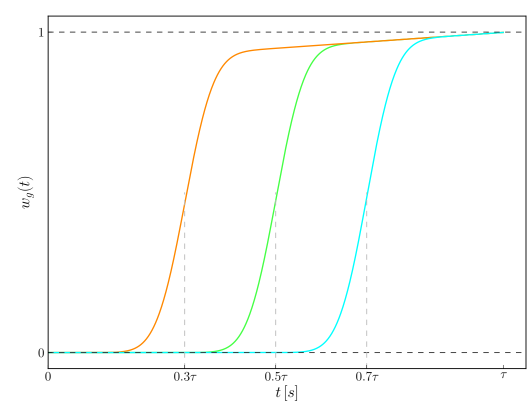

Fig. 4. Weighing function modified to avoid divide-by-zero numerical errors using the same color code as in Fig. 3

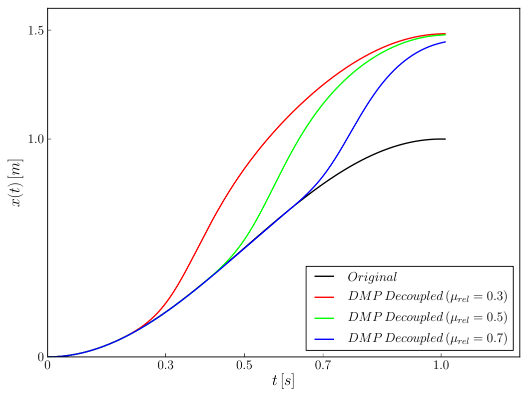

Fig. 5. Decoupled DMP with different values of μ and $\sigma = 0.05$.

presented) as response to the observed movement. It can be seen that, given that the partner's position is lagging with respect to the robot's one when the shift of weights is done in favor of the *goal-attractor*, the robot motion reverses for a certain time lapse. This oscillation is not desired, and a gentle deceleration would be much more convenient.

To alleviate this issue, a modification of the model is proposed which improves the smoothness of the convergence towards a moving goal. This modification consists in adding a velocity feedback term to the *transformation system*, resulting in:

$$\tau\dot{v} = (1 - w_g)(f_w + x_0 - x) + w_g[K(g - x) + K_v\dot{g}] - Dv \quad (11)$$

Fig. 9 on the following section shows the response trajectory generated to the same observed human motion, with the velocity feedback term in place.

IV. EXPERIMENTAL VALIDATION

To validate the proposed technique before implementing it onto a real robotic system, some tests have been performed on real data involving two persons exchanging different objects from different locations, as shown in Fig. 7. Markers were installed on the human bodies, mainly on the right arm of each partner (on the shoulder, elbow and hand), although in the present study only the hand markers are effectively used. Markers were tracked using a Vicon motion capture system. The DMP version presented in this paper was used to learn the three Cartesian dimensions of the right hand motion data from a selected sequence. Then data from different sequences have been used as observed human motions, and the resulting generated trajectories have been compared to the recorded response of the partner.

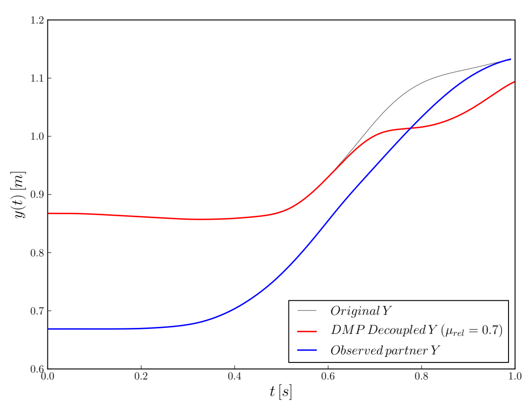

Fig. 6. Oscillation with the DMP proposed in section III-A.

The resulting behavior for one specific data set is shown in Figs. 8, 9 and 10. In each of these figures the black solid line represents the sample trajectory used for learning, the blue solid line represents the data used as "observed" Human hand position, the red solid line represents the output of the proposed DMP method, the dotted blue line represents the real recorded response of the other Human partner to the movement in the solid blue line, and the solid green line shows the response of the bio-inspired DMP formulation under the same conditions. The measured positions are in millimeters, and the reference used for the data capture is located on the floor between the two users, oriented as shown in Fig. 7, where the XYZ axis are colored in RGB order.

Also, for every motion dimension being learnt the same set of parameters has been used for the weighing function: $\mu = 0.7$ and $\sigma = 0.05$.

It can be seen that the generated trajectories adjust to the observed partner trajectory without loosing the inherent dynamics of the sample trajectory from which they were learnt.

It is also evident that the trajectory generated resembles much more closely the real recorded response of the human partner than the response of the bio-inspired DMP method. This supports the idea that the previous versions of the DMP do actually require an initial estimation of the exchange location, since using the current hand position of the partner as goal creates some unpredictable and undesired effects in the motion generated, especially on Figs 8 and 9. As illustrated on these examples, The extended model we are proposing does not require such initial estimation to provide a satisfactory behavior.

V. CONCLUSIONS

This article has proposed an extension of the DMP framework to correctly learn and reproduce the human arm approach during an object transfer procedure. By changing the phase variable behavior, we obtain a better control of

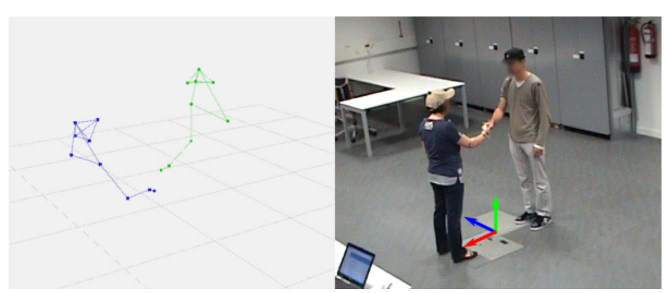

Fig. 7. Motion capture data aquired (left) and a picture of the capture sessions (right).

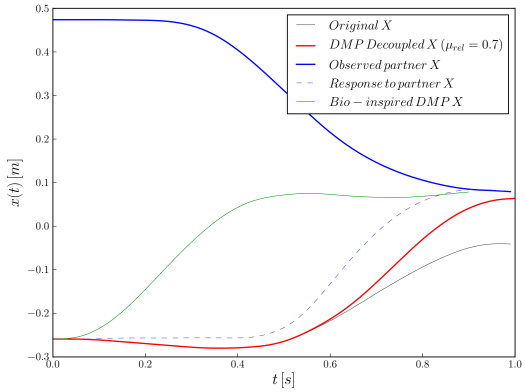

Fig. 8. Evolution of the generated trajectory in the X axis.

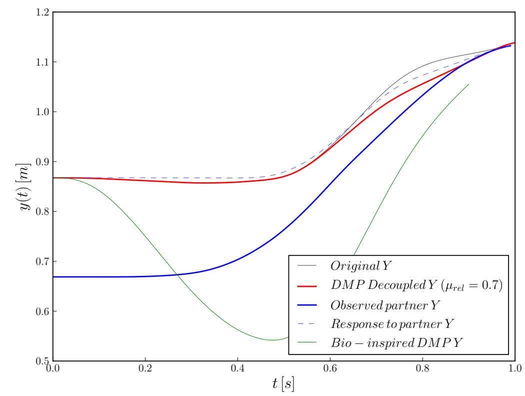

Fig. 9. Evolution of the generated trajectory in the Y axis.

the transition in between the *shape-attractor* and the *goal-attractor*, thus avoiding the need for an exchange location estimation. Furthermore, by adding in the *transformation system* a compensation for the goal velocity, the model obtained improves its convergence towards moving targets. It would be interesting to investigate how these improvements could benefit other applications of the DMP framework.

These experiments do not take yet into account the response of the human partner to the robot motion; indeed, the behavior of the human might not be equivalent when interacting with a person or with a robot. In order to complete the validation of our approach and to analyze the perception and reaction of the human when interacting with such system, at the time of writing this article, this method is being implemented onto a real robotic setup. The equipment used is a Kuka LWR robot, mounted onto a vertical structure to resemble the configuration of a human shoulder and arm; and a Kinect device to capture the motion of the human partner in front of the robot.

Finally, one of the main issues that will need to be tackled regarding such application is the triggering of the robotic motion start to get a perfect timing with the human partner. The proper implementation of such a triggering method will indeed highly influence the real time behavior of the presented technique.

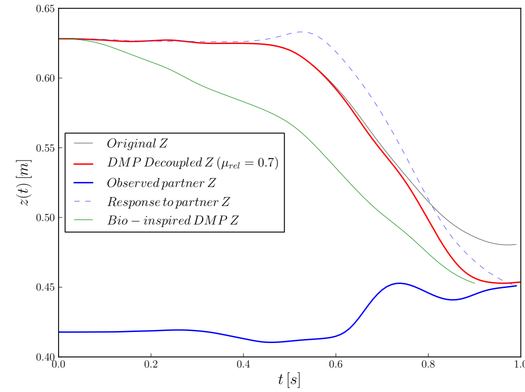

Fig. 10. Evolution of the generated trajectory in the Z axis.

REFERENCES

[1] M. Cakmak, S. Srinivasa, M.K. Lee, S. Kiesler, and J. Forlizzi. Using Spatial and Temporal Contrast for Fluent Robot-Human Hand-overs. In *ACM/IEEE International Conference on Human-Robot Interaction*, 2011.

[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] F. Chaumette and S. Hutchinson. Visual servo control, part 2: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, March 2007.

[4] Y.S. Choi, T. Chen, A. Jain, C. Anderson, J. Glass, and C. Kemp. Hand It Over or Set It Down: A User Study of Object Delivery with an Assistive Mobile Manipulator. In *IEEE International Symposium on Robot and Human Interactive Communication*, 2009.

[5] M. Desmurget and S. Grafton. Forward modeling allows feedback control for fast reaching movements. *Trends in cognitive sciences*, 4(11):423–431, November 2000.

[6] A. Edsinger and C. Kemp. Human-Robot Interaction for Cooperative Manipulation : Handing Objects to One Another. In *IEEE International Symposium on Robot and Human Interactive Communication*, 2007.

[7] 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. *IEEE International Conference on Robotics and Automation*, pages 2587–2592, May 2009.

[8] 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.

[9] A. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal. Learning Nonlinear Dynamical Systems Models. *Neural Computation*, pages 1–33, 2010.

[10] 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.

[11] 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.

[12] E. Sisbot, L. Marin-Urias, R. Alami, and T. Simeon. A human aware mobile robot motion planner. *IEEE Transactions on Robotics*, 23(5), 2007.

[13] 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.