2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Movement Templates for Learning of Hitting and Batting

Jens Kober, Katharina Mülling, Oliver Krömer, Christoph H. Lampert, Bernhard Schölkopf, Jan Peters

Abstract— **Hitting and batting tasks, such as tennis forehands, ping-pong strokes, or baseball batting, depend on predictions where the ball can be intercepted and how it can properly be returned to the opponent. These predictions get more accurate over time, hence the behaviors need to be continuously modified. As a result, movement templates with a learned global shape need to be adapted during the execution so that the racket reaches a target position and velocity that will return the ball over to the other side of the net or court. It requires altering learned movements to hit a varying target with the necessary velocity at a specific instant in time. Such a task cannot be incorporated straightforwardly in most movement representations suitable for learning. For example, the standard formulation of the dynamical system based motor primitives (introduced by Ijspeert et al. [1]) does not satisfy this property despite their flexibility which has allowed learning tasks ranging from locomotion to kendama. In order to fulfill this requirement, we reformulate the Ijspeert framework to incorporate the possibility of specifying a desired hitting point and a desired hitting velocity while maintaining all advantages of the original formulation. We show that the proposed movement template formulation works well in two scenarios, i.e., for hitting a ball on a string with a table tennis racket at a specified velocity and for returning balls launched by a ball gun successfully over the net using forehand movements. All experiments were carried out on a Barrett WAM using a four camera vision system.**

I. Introduction

Learning new skills can frequently be helped significantly by choosing a movement template representation that facilitates the process of acquiring and refining the desired behavior. For example, the work on dynamical systems-based motor primitives [1]–[3] has allowed speeding up both imitation and reinforcement learning while, at the same time, making them more reliable. Resulting successes have shown that it is possible to rapidly learn motor primitives for complex behaviors such as tennis swings [1] with only a final target, constrained reaching [4], drumming [5], biped locomotion [2], [6] and even in tasks with potential industrial application [7]. Although some of the presented examples, e.g., the tennis swing [1] or the T-ball batting [8], are striking movements, these standard motor primitives cannot properly encode a hitting movement. Previous work needed to make simplistic assumptions such as having a static goal [8], a learned complex goal function [9] or a stationary goal that could only be lightly touched at the movement's end [1].

Most racket sports require that we hit a non-stationary target at various positions and with various velocities during the execution of a complete striking movement. For example, in table tennis, a typical movement consists of swinging back from a rest postures, hitting the ball at a desired position with a desired orientation and velocity, continuing the swing a bit further and finally returning to the rest posture. See Figure 1 for an illustration. Sports sciences literature [10], [11] indicates that most striking movements are composed of similar phases that only appear to be modified by location and velocity at the interception point for the ball [12]–[15]. These findings indicate that similar motor primitives are being used that are invariant under these external influences similar to the Ijspeert motor primitives [1]–[3] being invariant under the modification of the final position, movement amplitude and duration. However, the standard formulation by Ijspeert et al. cannot be used properly in this context as there is no possibility to directly incorporate either a via-point or a target velocity (if the duration cannot be adapted as, e.g., for an approaching ball). Hence, a reformulation is needed that can deal with these requirements.

In this paper, we augment the Ijspeert approach [1]–[3] of using dynamical systems as motor primitives in such a way that it includes the possibility to set arbitrary velocities at the hitting point without changing the overall shape of the motion or introducing delays that will prevent a proper strike. This modification allows the generalization of learned striking movements, such as hitting and batting, from demonstrated examples. In Section II, we present the reformulation of the motor primitives as well as the intuition behind the adapted approach. We apply the presented method in Section III where we show two successful examples. First, we test the striking movements primitive in a static scenario of hitting a hanging ball with a table tennis racket and show that

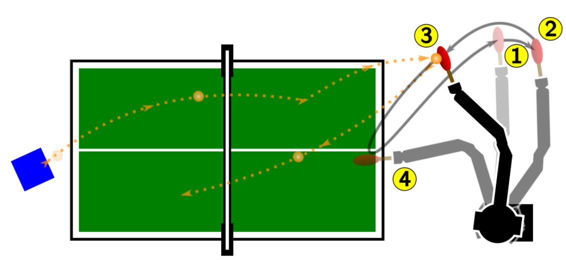
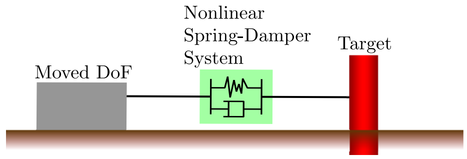
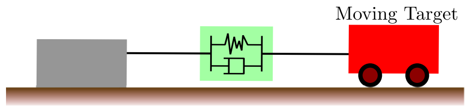


Fig. 1. This figure illustrates the different phases of a table tennis stroke. The blue box on the left represents a ping-pong ball launcher, the table is shown in green and the different states of the robot are superposed. A typical racket trajectory is indicated by the dark gray solid line while the orange dashed line represents the ball's trajectory. The robot goes through four stages: it swings back (1 → 2), it strikes the ball at a virtual hitting point with a goal-oriented velocity at posture 3, it follows the strike through (3 → 4) and finally returns to the rest posture 1. While the complete arm movement can be modeled with the Ijspeert approach, the reformulation in this paper is required to be able to properly strike the ball.

All authors are with the Department of Empirical Inference, Max Planck Institute for Biological Cybernetics, Spemannstr. 38, 72076 Tübingen, Germany, email: {firstname.lastname}@tuebingen.mpg.de



(a) Ijspeert Motor Primitive



(b) Hitting Motor Primitive

Fig. 2. In this figure, we convey the intuition of the presented reactive templates for learning striking movements. The Ijspeert formulation can be seen as a nonlinear spring damper system that pulls a degree of freedom to a stationary goal while exhibiting a learned movement shape. The presented approach allows hitting a target with a specified velocity without replanning if the target is adapted and, in contrast to the Ijspeert formulation, can be seen as a degree of freedom pulled towards a moving goal.

the movement generalizes well to new ball locations. After this proof of concept, we take the same movement template representation in order to learn an adaptable forehand for table tennis. The later is tested in the setup indicated by Figure 1 where we use a seven degrees of freedom Barrett WAM in order to return balls launched by a ball cannon.

II. Movement Templates for Learning to Strike

Ijspeert et al. [1] suggested to use a dynamical system approach in order to represent both discrete point-to-point movements as well as rhythmic motion with motor primitives. This framework allows the representation of arbitrarily shaped movements through the primitive's policy parameters, and these parameters can be estimated straightforwardly by locally weighted regression. In the discrete case, these primitives can be modified through their hyperparameters in order to adapt to the final goal position, the movement amplitude or the duration of the movement. The resulting movement can start from arbitrary positions and velocities and go to arbitrary final positions while maintaining the overall shape of the trajectory. In Section II-A, we review the most current version of this approach based on [2], [3].

However, as outlined in Section I, this formulation of the motor primitives cannot be used straightforwardly in racket sports as incorporating a desired virtual hitting point [10], [11] (consisting of a desired target position and velocity) cannot be achieved straightforwardly. For example, in table tennis, fast forehand movements need to hit a ball at a pre-specified speed, hitting time and a continuously adapted location. In the original Ijspeert formulation, the goal needs to be determined at the start of the movement and at approximately zero velocity as in the experiments in [1]; a via-point target can only be hit properly by modifying either the policy parameters [8] or, indirectly, by modifying the goal parameters [9]. Hence, such changes of the target can only be achieved by drastically changing the shape of the trajectory and duration.

As an alternative, we propose a modified version of Ijspeert's original framework that overcomes this limitation and is particularly well-suited for striking movements. This modification allows setting both a hitting point and a striking velocity while maintaining the desired duration and the learned shape of the movement. Online adaptation of these hyperparameters is possible and, hence, it is well-suited for learning racket sports such as table tennis. The basic intuition behind the modified version is similar to the one of Ijspeert's primitives, i.e., both assume that the controlled degree of freedom is connected to a specific spring damper system; however, the approach presented here allows overcoming previous limitations by assuming a connected moving target, see Figure 2. The resulting approach is explained in Section II-B.

A further drawback of the Ijspeert motor primitives [1]–[3] is that, when generalizing to new targets, they tend to produce large accelerations early in the movement. Such an acceleration peak may not be well-suited for fast movements and can lead to execution problems due to physical limitations; Figure 4 illustrates this drawback. In Section II-C, we propose a modification that alleviates this shortcoming.

A. Discrete Movement Primitives

While the original formulation in [1], [2] for discrete dynamical systems motor primitives used a second-order system to represent the phase z of the movement, this formulation has proven to be unnecessarily complicated in practice. Since then, it has been simplified and in [3] it was shown that a single first order system suffices

$$\dot{z} = -\tau \alpha_z z. \tag{1}$$

This canonical system has the time constant $\tau = 1/T$ where T is the duration of the motor primitive, a parameter α_z which is chosen such that $z \approx 0$ at T to ensure that the influence of the transformation function, shown in Eq. (3), vanishes. Subsequently, the internal state $\mathbf{x}$ of a second system is chosen such that positions $\mathbf{q}$ of all degrees of freedom are given by $\mathbf{q} = \mathbf{x}_1$, the velocities $\dot{\mathbf{q}}$ by $\dot{\mathbf{q}} = \tau \mathbf{x}_2 = \dot{\mathbf{x}}_1$ and the accelerations $\ddot{\mathbf{q}}$ by $\ddot{\mathbf{q}} = \tau \dot{\mathbf{x}}_2$. Under these assumptions, the learned dynamics of Ijspeert motor primitives can be expressed in the following form

$$\begin{aligned} \dot{\mathbf{x}}_2 &= \tau \alpha_x \left(\beta_x \left(\mathbf{t} - \mathbf{x}_1 \right) - \mathbf{x}_2 \right) + \tau \mathbf{A} \mathbf{f}\left(z\right), \\ \dot{\mathbf{x}}_1 &= \tau \mathbf{x}_2. \end{aligned} \tag{2}$$

This set of differential equations has the same time constant τ as the canonical system, parameters α_x, β_x set such that the system is critically damped, a goal parameter $\mathbf{t}$, a transformation function $\mathbf{f}$ and an amplitude matrix $\mathbf{A} = \mathrm{diag}(a_1, a_2, \ldots, a_n)$, with the amplitude modifier $\mathbf{a} = [a_1, a_2, \ldots, a_n]$. In [3], they use $\mathbf{a} = \mathbf{t} - \mathbf{x}_1^0$ with the initial position $\mathbf{x}_1^0$, which ensures linear scaling. Other choices are possibly better suited for specific tasks, see e.g., [16]. The transformation function $\mathbf{f}\left(z\right)$ alters the output of the first system, in Eq. (1), so that the second system, in Eq. (2), can represent complex nonlinear patterns and it is given by

$$\mathbf{f}\left(z\right) = \textstyle\sum_{i=1}^{N} \psi_i\left(z\right) \mathbf{w}_i z. \tag{3}$$

Here $\mathbf{w}_i$ contains the i^{th} adjustable parameter of all degrees of freedom, N is the number of parameters per degree of

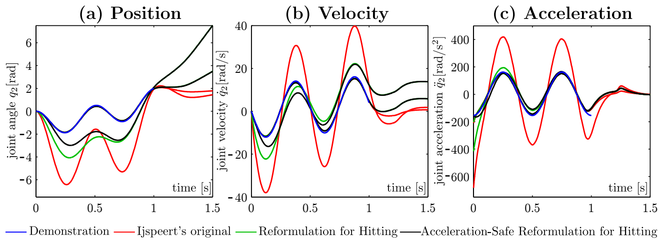


Fig. 3. Target velocity adaptation is essential for striking movements. This figure illustrates how different versions of the dynamical system based motor primitives are affected by a change of the target velocity. Here, an artificial training example (i.e., $q = 2t^2 + \cos(4t\pi) - 1$) is generated. After learning, all motor primitive formulations manage to reproduce the movements accurately from the training example for the same target velocity and cannot be distinguished. When the target velocity is tripled, this picture changes drastically. For Ijspeert's original model the amplitude modifier $\mathbf{a}$ had to be increased to yield the desired velocity. The increased amplitude of the trajectory is clearly visible for the positions and even more drastic for the velocities and accelerations. The reformulations presented in this paper, stay closer to the movement shape and amplitude. Particularly the velocities and accelerations exhibit that the new approach allows much better generalizing of the learned behavior. This figure furthermore demonstrates how a large initial step in acceleration appears for Ijspeert's original model (and the reformulation for hitting) even if a transformation function is used to partially suppress it for the training example.

freedom, and $\psi_i(z)$ are the corresponding weighting functions [3]. Normalized Gaussian kernels are used as weighting functions given by

$$\psi_i(z) = \frac{\exp\left(-h_i\,(z - c_i)^2\right)}{\sum_{j=1}^{N} \exp\left(-h_j\,(z - c_j)^2\right)}. \tag{4}$$

These weighting functions localize the interaction in phase space using the centers c_i and widths h_i. Note that the degrees of freedom (DoF) are usually all modeled independent in the second system in Eq. (2). All DoFs are synchronous as the dynamical systems for all DoFs start at the same time, have the same duration, and the shape of the movement is generated using the transformation $\mathbf{f}(z)$ in Eq. (3) that is learned as a function of the shared canonical system in Eq. (1).

As suggested in [1], [2] locally-weighted linear regression can be used for imitation learning. The duration of discrete movements is extracted using motion detection and the time-constants are set accordingly. Additional feedback terms can be added as shown in [3], [16], [17].

B. Adapting the Motor Primitives for Striking Movements

The regular formulation [1]–[3] which was reviewed in Section II-A, allows to change the initial position $\mathbf{x}_1^0$ and target position $\mathbf{t}$ (which corresponds to the position at the end of the movement at time T) of the motor primitive while maintaining the overall shape of the movement determined by the parameters $\mathbf{w}_i$. For disturbed initial conditions, the attractor dynamics that pull the motor primitive to the trained behavior and it is guaranteed to finally reach to the target position $\mathbf{t}$, see [1]. However, the formulation above only considers the case of a final goal with a favored velocity of $\dot{\mathbf{x}}_1(T) = 0$ at the target $\mathbf{t}$ and final time T. However, using the transformation function $\mathbf{f}(z)$ in Eq. (3), it can be forced to arbitrary final velocities by changing the shape parameters of the movement. As the last basis function in the transformation function $\mathbf{f}(z)$ decays almost to zero at time T the active parameters, the last parameter $\mathbf{w}_N$ needs to be over-proportionally large. If the motor primitive is trained with $\dot{\mathbf{x}}_1(T) = 0$ it simply rests at $\mathbf{x}_1 = \mathbf{t}$ if it runs for longer than T. However, large $\mathbf{w}_N$ often cause overshooting in $\mathbf{x}_1$ and the trajectory is subsequently pulled back to the final position $\mathbf{t}$ only using the linear attractor dynamics in Eq. (2) which may not be suitable for a given task. The target velocity $\dot{\mathbf{x}}_1(T)$ can only be changed either by scaling the duration of the movement T or with the amplitude modifier $\mathbf{a}$; however a mapping of $\mathbf{t}$ and $\dot{\mathbf{x}}_1(T)$ to $\mathbf{a}$ has to be established first. The main downsides of these approaches, respectively, are that either the total duration is changed (which makes the interception of a table tennis ball hard) or that $\mathbf{a}$ modifies the whole motion including shape and amplitude (which causes undesired movements and often requires overly strenuous movements in table tennis). These effects are illustrated in Figure 3. Note, if the target is constantly adapted as in table tennis (where the ball trajectory is not certain until the ball has bounced on the table the last time), these effects will produce significantly stronger undesired effects and, possibly, unstable behavior.

As an alternative for striking movements, we propose a modification of the dynamical system based motor primitives that allows us to directly specify the desired $\dot{\mathbf{x}}_1(T)$ while maintaining the duration of the movement and having the possibility to change the amplitude of the motion independently. For doing so, we introduce a moving target and include the desired final velocity in Eq. (2). We use a linearly moving target but other choices may be better suited for different tasks. This reformulation results in the following equations for the learned dynamics

$$\dot{\mathbf{x}}_2 = \tau\alpha_g \left(\beta_g\,(\mathbf{t}_m - \mathbf{x}_1) + \frac{(\dot{\mathbf{t}} - \dot{\mathbf{x}}_1)}{\tau}\right) + \tau\mathbf{A}\mathbf{f}, \tag{5}$$

$$\dot{\mathbf{x}}_1 = \tau\mathbf{x}_2,$$

$$\mathbf{t}_m = \mathbf{t}_m^0 - \dot{\mathbf{t}}\frac{\ln(z)}{\tau\alpha_h}, \tag{6}$$

where $\dot{\mathbf{t}}$ is the desired final velocity, $\mathbf{t}_m$ is the moving target and the initial position of the moving target $\mathbf{t}_m^0 = \mathbf{t} - \tau\dot{\mathbf{t}}$ ensures that $\mathbf{t}_m(T) = \mathbf{t}$. The term $-\ln(z)/(\tau\alpha_h)$ is proportional to the time if the canonical system in Eq. (1) runs unaltered; however, adaptation of z allows the straightforward adaptation of the hitting time. If $\dot{\mathbf{t}} = \mathbf{0}$, this formulation is exactly the same as the original formulation. The imitation learning approach mentioned in Section II-A can be adapted straightforwardly to this formulation. Figure 3 illustrates how the different approaches behave when forced to achieve a specified desired final velocity.

C. Safer Dynamics for Generalization

Generalizing fast movements such as a forehand in table tennis can become highly dangerous if the primitive requires exceedingly high accelerations or has large jumps in the acceleration (e.g., the fastest table tennis moves that we

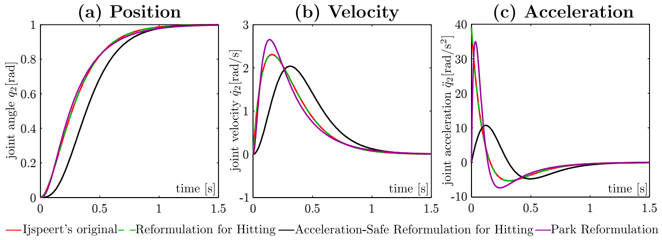


Fig. 4. An important aspect of the Ijspeert framework is that such primitives are guaranteed to be stable and, hence, safe for learning. A problem of the regular formulation highly unevenly distributed acceleration with a jump at the beginning of the movement of its unaltered dynamics. These high accelerations affect the movement when the behavior is either generalized to new goals or when during trial-and-error learning where the initial parameters are small. Some of these problem have previously been noted by Park [16], and are particularly bad in the context of fast striking movements. Here, we compare the different formulations with respect to their acceleration in the unaltered dynamics case (i.e., $\mathbf{w} = \mathbf{0}$). For a better comparison, we set the target velocity to zero ($\dot{\mathbf{t}} = 0$). The Ijspeert formulation clearly shows the problem with the large acceleration, as does the reformulation for hitting (with a hitting speed of $\dot{\mathbf{t}} = 0$ both are identical). While the Park modification starts without the jump in acceleration, it requires almost as high accelerations shortly afterwards. The acceleration-safe reformulation for hitting also starts out without a step in acceleration and does not require huge accelerations.

have executed on our WAM had a peak velocity of $7m/s$ and $10g$ maximum acceleration). Hence, the initial jump in acceleration often observed during the execution of the Ijspeert primitives may lead to desired accelerations that a physical robot cannot properly execute, and may even cause damage to the robot system. In the following, we will discuss several sources of these acceleration jumps and how to overcome them. If the dynamics are not altered by the transformation function, i.e., $\mathbf{w} = \mathbf{0}$, the highest acceleration during the original Ijspeert motor primitive occurs at the very first time-step and then decays rapidly. If the motor primitives are properly initialized by imitation learning, the transformation function will cancel this initial acceleration, and, thus, this usually does not pose a problem in the absence of generalization. However, when changing the amplitude $\mathbf{a}$ of the motion (e.g., in order to achieve a specific target velocity) the transformation function will over- or undercompensate for this initial acceleration jump. The adaptation proposed in Section II-B does not require a change in amplitude, but suffers from a related shortcoming, i.e., changing the target velocity also changes the initial position of the target, thus results in a similar jump in acceleration that needs to be compensated. Using the motor primitives with an initial velocity that differs from the one used during imitation learning has the same effect. Figures 3 and 4 illustrate these initial steps in acceleration for various motor primitive formulations. As an alternative, we propose to gradually activate the attractor dynamics of the motor primitive (e.g., by reweighting them using the output of the canonical system). When combined, these two adaptations result in

$$\dot{\mathbf{x}}_2 = (1-z)\tau\alpha_g\left(\beta_g(\mathbf{t}_m - \mathbf{x}_1) + \frac{(\dot{\mathbf{t}} - \dot{\mathbf{x}}_1)}{\tau}\right) + \tau\mathbf{A}\mathbf{f}. \quad (7)$$

Surprisingly, after this modification the unaltered dynamics (i.e., where $\mathbf{w} = \mathbf{0}$ and, hence, $\tau\mathbf{A}\mathbf{f}(z) = \mathbf{0}$) result in trajectories that roughly resemble a minimum jerk movements and, hence, look very similar to human movements. Exactly as for the Ijspeert formulation, we can arbitrarily shape the behavior by learning the weights of the transformation function. Note that [16] also introduced a similar modification canceling the initial acceleration caused by the offset between initial and target position; however, their approach cannot deal with a deviating initial velocity.

The proposed acceleration jump compensation also yields smoother movements during the adaptation of the hitting point as well as smoother transitions if motor primitives are sequenced. The later becomes particularly important when the preceding motor primitive has a significantly different velocity than during training (by imitation learning) or if it is terminated early due to external events. All presented modifications are compatible with the imitation learning approach discussed in Section II-A and the adaptation is straightforward. Figures 3 and 4 show how the presented modifications overcome the problems with the initial jumps in acceleration.

III. Robot Evaluation

In this section, we evaluate the presented reactive templates for representing, learning and executing forehands in the setting of table tennis. For doing so, we evaluate our representation for striking movements first on hitting a hanging ball in Section III-A and, subsequently, in the task of returning a ball served by a ball launcher presented in Section III-B.

When hitting a ping-pong ball that is hanging from the ceiling, the task consists of hitting the ball with an appropriate desired Cartesian velocity and orientation of the paddle. Hitting a ping-pong ball shot by a ball launcher requires predicting the ball's future positions and velocities in order to choose an interception point. The latter is only sufficiently accurate after the ball has hit the table for the last time. This short reaction time underlines that the movement templates can be adapted during the trajectory under strict time limitations when there is no recovery from a bad generalization, long replanning or inaccurate movements.

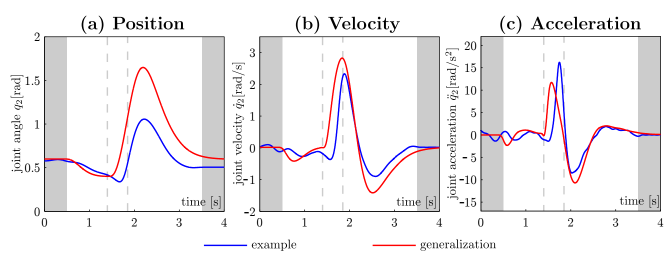
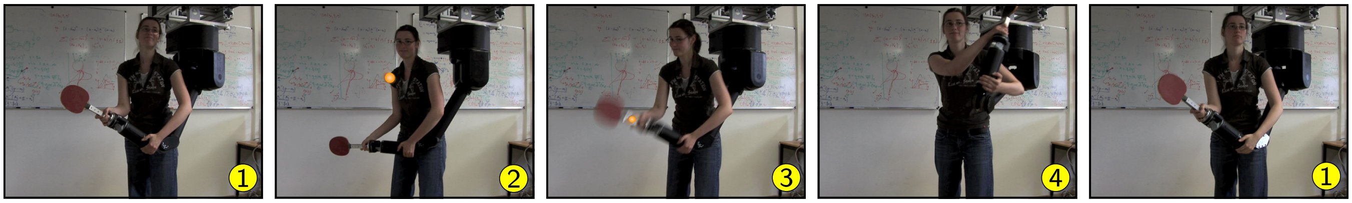
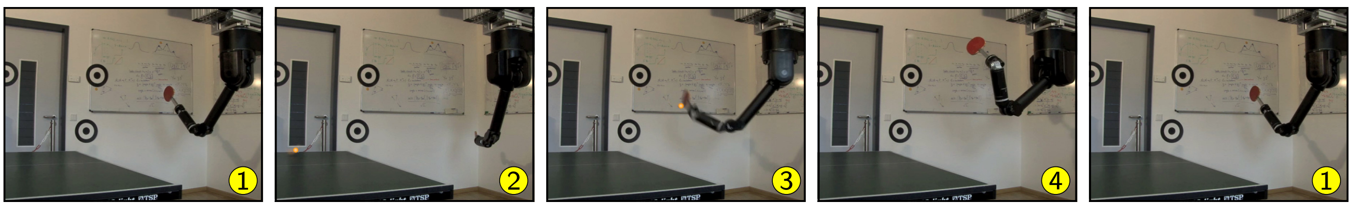


Fig. 5. This figure demonstrates the generalization of an imitated behavior to a different target that is $15cm$ away from the original target. Note that this trajectory is for a static target, hence the slow motion. The depicted degree of freedom (DoF) is shoulder adduction-abduction (i.e., the second DoF). The solid gray bars indicate the time before and after the main movement, the gray dashed lines indicate the phase borders also depicted in Figure 1 and the target is hit at the second border.



(a) Demonstration by a Human Instructor

(b) Example: Reproduction for Hitting a Stationary Ball



(c) Application: Returning Balls launched by a Ball Gun

Fig. 6. This figure presents a hitting sequence from the demonstration, a generalization on the robot with a ball attached by a string as well as a generalization hitting a ball shot by a ping-pong ball launcher. The demonstration and the flying ball generalization are captured by a $25Hz$ video camera, the generalization with the attached ball is captured with $200Hz$ through our vision system. From left to right the stills represent: rest posture, swing-back posture, hitting point, swing-through and rest posture. The postures ($1 - 4$) are the same as in Figure 2.

A. Generalizing Forehands on Static Targets

As a first experiment, we evaluated how well this new formulation of hitting primitives generalizes forehand movements learned from imitation as shown in Figure 6 (a). First, we collected arm, racket and ball trajectories for imitation learning using the 7 DoF Barrett WAM robot as an haptic input device for kinesthetic teach-in where all inertial forces and gravity were compensated. In the second step, we employ this data to automatically extract the duration of the striking movement, the duration of the individual phases as well as the Cartesian target velocity and orientation of the racket when hitting the ball. We employ a model (as shown in Section II) that has phases for swinging back, hitting and going to a rest posture. Both the phase for swing-back and return-to-home phases will go into intermediary still phases while the hitting phase goes through a target point with a pre-specified target velocity. All phases can only be safely executed due to the "safer dynamics" which we introduced in Section II-C.

In this experiment, the ball is a stationary target and detected by a stereo camera setup. Subsequently, the supervisory level proposed in [11] determines the hitting point and the striking velocity in configuration space. The motor primitives are adjusted accordingly and executed on the robot in joint-space using an inverse dynamics control law. The robot successfully hits the ball at different positions within a diameter of approximately $1.2m$ if kinematically feasible. The adaptation for striking movements achieves the desired velocities and the safer dynamics allow generalization to a much larger area while successfully removing the possibly large accelerations at the transitions between motor primitives. See Figure 5 for a comparison of the training example and the generalized motion for one degree of freedom and Figure 6 (b) for a few frames from a hit of a static ball. Please see the video accompanying this submission.

B. Playing against a Ball Launcher

This evaluation adds an additional layer of complexity as the hitting point and the hitting time has to be estimated from the trajectory of the ball and continuously adapted as the hitting point cannot be reliably determined until the ball has bounced off the table for the last time. In this setting, the ball is tracked by two overlapping high speed stereo vision setups with $200Hz$ cameras. In order to obtain better estimates of the current position and to calculate the velocities, the raw 3D positions are filtered by a specialized Kalman filter [18] that takes contacts of the ball with the table and the racket into account [11]. When used as a Kalman predictor, we can again determine the target point for the primitive with a pre-specified target velocity with the method described in [11]. The results obtained for the still ball generalize well from the static ball to the one launched by a ball launcher at $3m/s$ which are returned at speeds up to $8m/s$. A sequence of frames from the attached video is shown in Figure 6. The plane of possible virtual hitting points again has a diameter of roughly $1m$ as

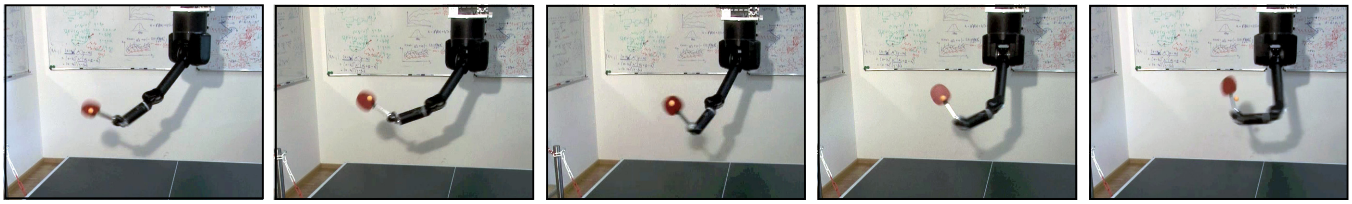


Fig. 7. Generalization to various targets (five different forehands at posture ③) are shown approximately when hitting the ball.

shown in Figure 7. The modified motor primitives generated movements with the desired hitting position and velocity. The robot hit the ball in the air in approx. 95% of the trials. However, due to a simplistic ball model and execution inaccuracies the ball was often not properly returned on the table. Please see the videos accompanying this submission `http://www.robot-learning.de/Research/HittingMPs` .

Note that our results differ significantly from previous approaches as we use a framework that allows us to learn striking movements from human demonstrations unlike previous work in batting [19] and table tennis [20]. Unlike baseball which only requires four degrees of freedom (as, e.g., in Senoo et al. [19] who used a 4 DoF WAM arm in a manually coded high speed setting), and previous work in table tennis (which had only low-inertia, was overpowered and had mostly prismatic joints [20]–[22]), we use a full seven degrees of freedom revolutionary joint robot and, thus, have to deal with larger inertia as the wrist adds roughly $2.5kg$ weight at the elbow. Hence, it was essential to train trajectories by imitation learning that distribute the torques well over the redundant joints as the human teacher was suffering from the same constraints.

IV. CONCLUSION

In this paper, we rethink previous work on dynamic systems motor primitive [1]–[3] in order to obtain movement templates that can be used reactively in batting and hitting sports. This reformulation allows to change the target velocity of the movement while maintaining the overall duration and shape. Furthermore, we present a modification that overcomes the problem of an initial acceleration step which is particularly important for safe generalization of learned movements. Our adaptations retain the advantages of the original formulation and perform well in practice. We evaluate this novel motor primitive formulation first in hitting a stationary table tennis ball and, subsequently, in returning ball served by a ping pong ball launcher. In both cases, the novel motor primitives manage to generalize well while maintaining the features of the demonstration. This new formulation of the motor primitives can hopefully be used together with *goal learning* [9] in a *mixture of motor primitives* [9] in order to create a complete framework for learning tasks like table tennis autonomously.

REFERENCES

[1] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2002.

[2] S. Schaal, J. Peters, J. Nakanishi, and A. J. Ijspeert, "Learning Motor Primitives," in *Proceedings of the International Symposium on Robotics Research*, 2003.

[3] S. Schaal, P. Mohajerian, and A. J. Ijspeert, "Dynamics systems vs. optimal control — a unifying view," *Progress in Brain Research*, vol. 165, no. 1, pp. 425–445, 2007.

[4] A. Gams and A. Ude, "Generalization of example movements with dynamic systems," in *Proceedings of the International Conference on Humanoid Robots (HUMANOIDS)*, 2009.

[5] D. Pongas, A. Billard, and S. Schaal, "Rapid synchronization and accurate phase-locking of rhythmic motor primitives," in *Proceedings of the International Conference on Intelligent RObots and Systems (IROS)*, 2005.

[6] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato, "Learning from demonstration and adaptation of biped locomotion," *Robotics and Autonomous Systems (RAS)*, vol. 47, no. 2-3, pp. 79–91, 2004.

[7] H. Urbanek, A. Albu-Schäffer, and P. van der Smagt, "Learning from demonstration repetitive movements for autonomous service robotics," in *Proceedings of the International Conference on Intelligent RObots and Systems (IROS)*, 2004.

[8] J. Peters and S. Schaal, "Policy gradient methods for robotics," in *Proceedings of the International Conference on Intelligent RObots and Systems (IROS)*, 2006.

[9] J. Peters, J. Kober, K. Mülling, D. Nguyen-Tuong, and O. Krömer, "Towards motor skill learning for robotics," in *Proceedings of the International Symposium on Robotics Research (ISRR)*, 2009.

[10] M. Ramanantsoa and A. Durey, "Towards a stroke construction model," *International Journal of Table Tennis Science*, vol. 2, pp. 97–114, 1994.

[11] K. Mülling, "A computational model of human table tennis for robot application," Unpublished M.Sc. Thesis, University of Tübingen, 2009.

[12] R. Schmidt and C. Wrisberg, *Motor Learning and Performance*, 2nd ed. Human Kinetics, 2000.

[13] R. Bootsma and P. van Wieringen, "Timing an attacking forehand drive in table tennis," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 16, pp. 21–29, 1990.

[14] A. Hubbard and C. Seng, "Visual movements of batters," *Research Quaterly*, vol. 25, pp. 42–57, 1954.

[15] D. Tyldesley and H. Whiting, "Operational timing," *Journal of Human Movement Studies*, vol. 1, pp. 172–177, 1975.

[16] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," *Proceedings of the International Conference on Humanoid Robots (HUMANOIDS)*, 2008.

[17] J. Kober, B. Mohler, and J. Peters, "Learning perceptual coupling for motor primitives," in *Proceedings of the International Conference on Intelligent RObots and Systems (IROS)*, 2008.

[18] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[19] T. Senoo, A. Namiki, and M. Ishikawa, "Ball control in high-speed batting motion using hybrid trajectory generator." in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[20] R. Andersson, *A robot ping-pong player: experiment in real-time intelligent control*. Cambridge, MA, USA: MIT Press, 1988.

[21] H. Fässler, H. A. Beyer, and J. T. Wen, "A robot ping pong player: optimized mechanics, high performance 3d vision, and intelligent sensor control," *Robotersysteme*, vol. 6, no. 3, pp. 161–170, 1990.

[22] M. Matsushima, T. Hashimoto, M. Takeuchi, and F. Miyazaki, "A learning approach to robotic table tennis," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 767–771, 2005.