2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Online approach for altering robot behaviors based on human in the loop coaching gestures

Tadej Petrič[1,2], Andrej Gams[1,3], Leon Žlajpah[1], Aleš Ude[1,2], and Jun Morimoto[2]

Abstract— **The creation and adaptation of motor behaviors is an important capability for autonomous robots. In this paper we propose an approach for altering existing robot behaviors online, where a human coach interactively changes the robot motion to achieve the desired outcome. Using hand gestures, the human coach can specify the desired modifications to the previously acquired behavior. To preserve a natural posture while performing the task, the movement is encoded in the robot's joint space using periodic dynamic movement primitives. The coaching gestures are mapped to the robot joint space via robot Jacobian and used to create a virtual force field affecting the movement. A recursive least squares technique is used to modify the existing movement with respect to the virtual force field. The proposed approach was evaluated on a simulated three degrees of freedom planar robot and on a real humanoid robot, where human coaching gestures were captured by an RGB-D sensor. Although our focus was on rhythmic movements, the developed approach is also applicable to discrete (point-to-point) movements.**

I. INTRODUCTION

The interaction between a pupil and a teacher when learning or improving existing skills usually involves natural communication such as speech or gestures. Based on the instructions of a coach, humans can quickly learn and modify their motion patterns to achieve the desired behavior. The development of an effecting coaching system for humanoid robots is, however, a difficult task. In practice, modifying robot behaviors remains the task of experts in robotics.

Robotics researchers developed various robot coaching methods in the past decade. For example, Nakatani et al. [1] used the coach's qualitative evaluations of the robot performance to improve balancing and walking. In [2], supervised learning was combined with voice commands of a human coach, where the voice commands were used as a reward function in the learning algorithm. In [3], coaching was used on a mobile platform with the emphasis on learning high level task representations rather than motor skills. An approach that uses qualitative, verbal instructions to modify movements obtained by human demonstration was proposed in [4]. The developed system was suitable also for non-expert users. Kinesthetic teaching with iterative updates to modify a humanoid behavior was proposed in [5]. An area closely related to robot coaching is learning by demonstration, where a variety of different methods were proposed [6], [7], [8], [9], [10], [11]. However, most of the learning by demonstration methods do not address the problem of easily modifying an existing behavior to acquire a new desired outcome.

We were inspired by the efficiency of human-to-human skill transfer when developing a more effective approach to modify the existing robot behaviors. Rather than learning how to program robots, people can bring their own knowledge from interacting with each other directly into the robot domain [4]. Ideally, the human-robot interaction should focus on approaches that are intuitive for a human coach. In this paper we propose an approach for modifying existing robot behaviors based on online guidance provided by the human coach. The guidance is provided in the form of pointing gestures, i. e. the coach indicates to the robot where and how it should modify its motion. Such an interface is intuitive for humans as movement shaping through physical guidance and other means of communication is common in human motor learning [12]. It allows also non-experts to teach and alter the existing robot skills in order to obtain new desired outcomes.

A motor representation used to encode robot movements in an online coaching system must have the ability to generate smooth movements even when its parameters change online. This is important to supply an immediate feedback to the coach. Such a capability is provided by dynamic movement primitives (DMPs) [13], [14], which are defined by a set of critically damped second order linear differential equations, supplemented with a nonlinear forcing term. In this paper we focus on periodic movements [15], but the approach is fully applicable to discrete (point-to-point) movements as well. Periodic DMPs are often combined with adaptive oscillators [16]. Adaptive oscillators generate a stable limit cycle and provide the phase signal to the DMP. We assume that the initial motion pattern has been defined somehow, e. g. by kinesthetic guiding. To avoid losing postural information when using redundant robots like humanoids, the demonstrated motion pattern is encoded in the joint-space.

We developed a new DMP adaptation algorithm that can be used to modify existing motor behaviors encoded by DMPs based on human coaching gestures. The coaching gestures are specified by pointing towards the part of the movement that needs to be changed. The pointing gesture defines the direction and magnitude of change. To demonstrate the applicability of the proposed coaching approach, we implemented it both in simulation and on a real humanoid robot, where coaching gestures were obtained by Microsoft Kinect RBG-D sensor and a body tracker. The paper is organized as follows. In Section II we provide a short review of periodic DMPs. We then describe the newly developed

[1]Humanoid and Cognitive Robotics Lab & Dept. of Automatics, Biocybernetics, and Robotics, Jožef Stefan Institute (JSI), Ljubljana, Slovenia. tadej.petric@ijs.si

[2]Dept. of Brain Robot Interface (BRI), ATR Computational Neuroscience Laboratories, Kyoto, Japan

[3]Biorobotics Laboratory, Institute of Bioengineering, Ecole Polytechnique Federale de Lausanne (EPFL), Lausanne, Switzerland

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

coaching algorithm. In Section III we analyze the properties of the proposed algorithm in simulation and in Section IV we evaluate it on a real humanoid robot. Conclusions, summary and prospective future work are explained in Section V.

II. COACHING SYSTEM

The basic framework of our coaching system consists of periodic Dynamic Movement Primitives (DMPs) combined with an adaptive frequency oscillator [16], which can extract the phase and the frequency from an arbitrary periodic signal. This framework is also called a two-layered system for movement imitation [15]. In our previous work [15], [16], we proposed a learning algorithm that can be used to extract the basic frequency from the demonstrated periodic movement, learn the waveform of one period, and reconstruct the desired waveform at an arbitrary frequency. To learn a new control policy based on the human coaching gestures, this two-layered imitation system is embedded into the proposed control framework for coaching.

A. Dynamic movement primitives combined with adaptive frequency oscillators

The first layer of the imitation system is based on adaptive frequency oscillators combined with the adaptive Fourier series. The details and the properties of the learning approach are given in [16]. In summary, an adaptive frequency oscillator is defined by a set of second order differential equations

$$\dot{\phi} = \Omega - K \cdot e \cdot \sin(\phi), \tag{1}$$

$$\dot{\Omega} = -K \cdot e \cdot \sin(\phi), \tag{2}$$

where Ω is the extracted frequency, ϕ is the phase, K is the coupling constant and e is the difference between the actual and the estimated input signal. Here we denote the input signal by v and the estimated signal by $\hat{v}$. The input signal v is the signal on which the motion pattern is synchronized. Note that once the error signal e becomes zero we obtain $\dot{\Omega} = 0$ and $\dot{\phi} = \Omega$. The estimated input signal $\hat{v}$ is represented as

$$\hat{v} = \sum_{c=0}^{m} (\alpha_c \cos(c\phi) + \beta_c \sin(c\phi)). \tag{3}$$

Here m is the size of the Fourier series. Our learning algorithm simultaneously estimates the frequency Ω and the input signal $\hat{v}$, i e. the weights α_c and β_c. See [16] for details.

We augment this first layer by anchoring the dynamic movement primitives to the phase signal ϕ of the adaptive oscillator as in [15], [16]. This makes it possible to synchronize an arbitrary trajectory to an arbitrary periodic signal congruent with the desired behavior. The basic equations of dynamic movement primitives are summarized from [14], [13], [15]. For a single degree of freedom denoted by y, which can either be one of the internal joint-space coordinates or one of the external task-space coordinates, the following system of linear differential equations with constant coefficients, augmented by a nonlinear forceing term f, has been applied to derive DMPs

$$\dot{z} = \Omega \left(\alpha_z \left(\beta_z (g - y) - z \right) + f \right), \tag{4}$$

$$\dot{y} = \Omega z, \tag{5}$$

where α_z and β_z are the positive constants, which guarantee that the system monotonically converges to the desired trajectory, g is the center of oscillation, and f is the nonlinear part that determines the shape of the trajectory. It is given by

$$f(\phi) = \frac{\sum_{i=1}^{N} w_i \psi_i(\phi)}{\sum_{i=1}^{N} \psi_i(\phi)} r, \tag{6}$$

where r is the parameter that can be used to modulate the amplitude of the movement and ψ are the Gaussian like kernel functions given by

$$\psi_i(\phi) = \exp\left(h \left(\cos(\phi - c_i) - 1 \right) \right). \tag{7}$$

Here, h is the width and c_i is the distribution on one period. If not stated otherwise, in the following we used c_i, $i = 1, \ldots, 25$, and they were equally spread between 0 and 2π.

By applying the locally weighted regression the system can learn the shape of the trajectory on-line. The equations for incremental learning are summarized from [15], where the equations (4) and (5) were rewritten as one second order differential equation

$$f_d = \frac{\ddot{y}_d}{\Omega^2} - \alpha_z \left(\beta_z (g - y_d) - \frac{\dot{y}_d}{\Omega} \right). \tag{8}$$

Here the triplet of y_d, $\dot{y}_d$ and $\ddot{y}_d$ denotes the desired position, the velocity and the acceleration. To update the weights w_i of the kernel function ψ_i, we use the recursive least-squares method with the forgetting factor λ. In our experiments, the forgetting factor was set to $\lambda = 0.9995$. With the given target (8), the recursive algorithm updates the weights w_i using the following rule

$$P_i(t+1) = \frac{1}{\lambda} \left(P_i(t) - \frac{P_i(t)^2 r^2}{\frac{\lambda}{\psi_i(\phi(t))} + P_i(t) r^2} \right), \tag{9}$$

$$w_i(t+1) = w_i(t) + \psi_i(\phi(t)) P_i(t+1) r e_r(t), \tag{10}$$

$$e_r(t) = f_d(t) - w_i(t) r. \tag{11}$$

If not stated otherwise, we use $w_i(0) = 0$ and $P_i(0) = 1$, where $i = 1, \ldots, 25$.

In general DMPs provide a comprehensive framework for generating smooth kinematic control policies. Other important properties are: time invariance, online modulations including using a repulsive force to influence the course of the trajectory, framework for the trajectory learning, and smooth behavior in case of sudden change in the trajectory. Even though the DMP framework already posseses methods for amplitude, phase and frequency modulation, these modulations are insufficient to modify the behavior within a general coaching system. To modify the behavior online with a human in the loop, we propose an algorithm that can update the weights of the DMP based on the coaching gestures. The goal is to provide means to generate arbitrary modifications to the available movement patterns and successfully perform the desired task. The coaching system can also be used for building a library of motion patterns, which can later be used by movement generalization methods [17].

B. Coaching with Potential Fields

The primary goal of movement modeling with dynamical systems is to exploit the coupling phenomena to generate more complex behaviors [14]. We showed in the previous section how two dynamical systems can be connected together for imitation learning of periodic movements. In this section we discuss how to modify a robot trajectory online based on the input of a human coach. An ability to modulate movement trajectories online based on the human input is a very important capability for robots that interact with humans in natural environments. The proposed algorithm can modify the robot's motion online based on the human in the loop coaching gestures and is therefore an important step towards providing such a capability.

There are different ways for adding a coupling term to modify motion patterns. For example, it can be added to the transformation system or to the canonical system, or even to both [14]. For 3-D Cartesian space movements, Hoffmann et al. [18] showed that obstacle avoidance can be achieved by adding a coupling term $\boldsymbol{C}_y$ to Eq. (4)

$$\dot{\mathbf{z}} = \Omega\left(\alpha_z\left(\beta_z(\boldsymbol{g}-\mathbf{y})-\mathbf{z}\right)+\boldsymbol{C}_y+\boldsymbol{f}\right), \tag{12}$$

where $\boldsymbol{y}$, $\boldsymbol{z}$, $\boldsymbol{g}$, $\boldsymbol{C}_y$, and $\boldsymbol{f}$ are in this case three dimensional values. In [18] this equation was used to drive the robot's behavior and ensure obstacle avoidance. In our case we intend to modify the behavior permanently, therefore we use this term as input to the recursive least-squares method for updating the weights of the canonical system. Since in this case the reference trajectory is simply the output of the DMP (there is no training signal y_d, $\dot{y}_d$ and $\ddot{y}_d$), $e_r(t)$ as defined in Eq. (11) would be equal to zero if the DMP equations had not changed. However, since the differential equation (4) was changed to (12), there is an additional coupling term $\boldsymbol{C}_y$, which was not accounted for during training. Thus Eq. (11) transforms into

$$e_r(t) = C_{y,j}(t), \tag{13}$$

where $C_{y,j}(t)$ is the coupling term for the degree of freedom denoted by j.

A proper definition of the coupling term $\boldsymbol{C}_y$ is crucial and of course task dependent. To enable coaching by human gestures, we modified the obstacle avoidance coupling term $\boldsymbol{C}_y$ from [18] as follows

$$\boldsymbol{C}_y = \gamma\ \mathrm{s}(||\boldsymbol{o}-\boldsymbol{x}||)\ \exp(-\beta\phi)\ \boldsymbol{d}, \tag{14}$$

Here $\boldsymbol{x}$ is the Cartesian position of the end-effector, $\boldsymbol{o}$ is the center position of the perturbation potential field (defined by hand position), $\boldsymbol{d}$ is the perturbation direction (defined by the pointing gesture), γ and β are the scaling factors, ϕ is given by

$$\phi = \arccos\left(\frac{(\boldsymbol{o}-\boldsymbol{x})^T\dot{\boldsymbol{x}}}{||(\boldsymbol{o}-\boldsymbol{x})||\ ||\dot{\boldsymbol{x}}||}\right). \tag{15}$$

$\mathrm{s}(r)$ is defined as

$$\mathrm{s}(r) = \frac{1}{1+e^{\eta(r-r_m)}}, \tag{16}$$

where η is the scaling factor and r_m the distance at which the perturbation field should start affecting the robot's motion. A one degree of freedom example for the coupling term is shown in Fig. 1.

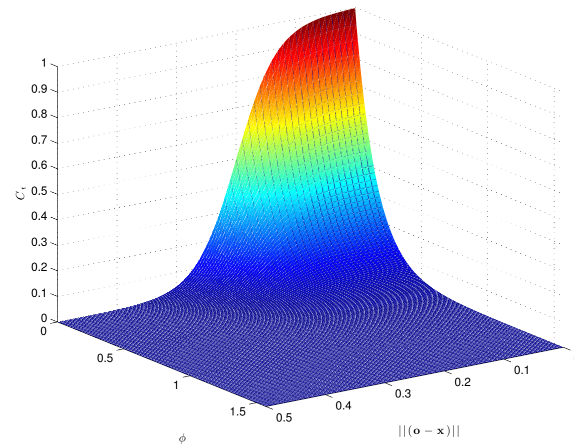

Fig. 1. One degree of freedom example coupling term C_y with parameters $\eta = 30$, $r_m = 0.2$, $\gamma = 1$, and $\beta = -20/\pi$.

To update the trajectories in joint-space while they are perturbed in task space, where the coupling term is denoted by $\boldsymbol{C}_y$, a pseudo inverse of the task Jacobian is used. This essentially maps the task space velocities into the joint space velocities with $\dot{\boldsymbol{q}} = \mathbf{J}\dot{\boldsymbol{x}}$. By applying a similar transformation to $\boldsymbol{C}_y$ we obtain

$$\boldsymbol{C}_q = \mathbf{J}^{\dagger}\boldsymbol{C}_y. \tag{17}$$

where $\boldsymbol{C}_q = [C_{q,1}\ \ C_{q,2}\ \ \ldots\ \ C_{q,k}]^T$ and k is the number of the robot's degrees of freedom. The components of (17) are now inserted into (13), which is used for updating the DMP weights w_i using (9) and (10). In this way we ensure that the joint space trajectories encoded by the DMPs are properly modified according to the coach's instructions.

Keeping the movement representation in the joint space is beneficial because our initial movement trajectories, which are encoded by DMPs, are usually acquired by kinesthetic guiding. By using joint space trajectories we avoid losing information about the selected robot configuration during human guiding on a redundant robot. Hence the DMP representation should remain in the joint space and the behavior should be modified there.

III. SIMULATION RESULTS

To show the properties of the proposed approach, we first applied it to a simulated 3 degrees of freedom planar robot. The robot was simulated using Planar Manipulator Toolbox [19]. The initial joint space trajectory was defined such that it produced a circular motion in the task space. The frequency of motion was constant and it was set to 0.5 Hz. If not stated otherwise, the coaching parameters were $\gamma = 100$, $\eta = 20$, $r_m = 0.35$, and $\beta = -8/\pi$.

Fig. 2 shows simulation results where the coaching point was defined with the parameters $\boldsymbol{o} = [1.8, -0.6]$ $\boldsymbol{d} = [0, 1]^T$. The coaching command was inserted at the selected position after 5 seconds. On the right plot we can see the evolution of the task space trajectory. Here the red line shows the initial task space motion and the green line the task space motion after coaching. The grey lines show the evolution of DMP modification in time. It can be seen that the coaching command was only acting at the desired location and therefore it did not affect the rest of the initial trajectory. This can also be seen in the bottom plot left where the scale of the coupling term used in recursive least squares is shown. Here we can see that the coupling term only acts when the end-effector is close to the perturbation point. In addition, we can see that the coupling term is iteratively converging towards zero. The first three plots on the left show the joint trajectories in time. We can see that they are modified only when the end-effector is near to the perturbation point and that they remain smooth.

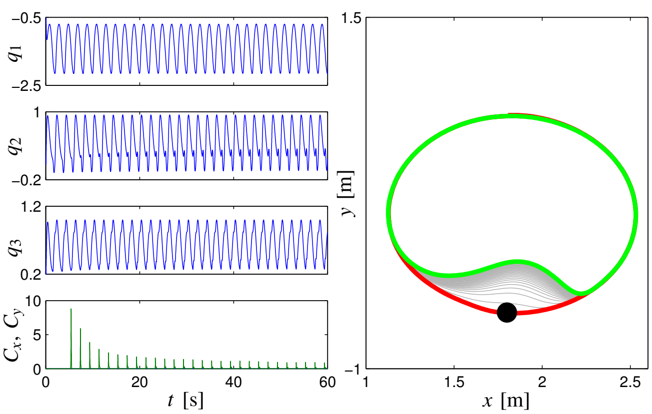

Fig. 2. Simulation results where the circular motion in task space was pushed in. Coaching command parameters were $\boldsymbol{o} = [1.8, -0.6]$ and $\boldsymbol{d} = [0, 1]^T$. The coaching point was activated after 5 seconds.

Similar results can also be observed in Fig. 3, where the initial task space trajectory was pushed out. In this case the coaching point parameters were $\boldsymbol{o} = [1.8, 0.8]$ $\boldsymbol{d} = [0, 1]^T$. Again the coaching point was inserted after 5 seconds. We can see the same basic performance as in the case of Fig. 2. This two study cases clearly show that we can easily modify the task space behavior at the desired point to achieve the desired course of movement, even though the trajectories are encoded in the joint space. These two case studies show that we can smoothly and iteratively modify the task space behavior in an arbitrary direction.

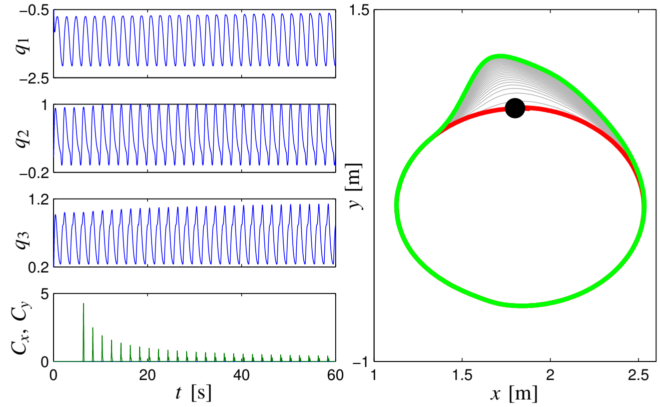

Fig. 3. Simulation results where the circular motion in task space was pushed out. Coaching command parameters were $\boldsymbol{o} = [1.8, 0.8]$ and $\boldsymbol{d} = [0, 1]^T$. The coaching command was activated after 5 seconds.

To further support the last statement we show in Fig. 4 an experiment where the coaching command is kept at the constant distance to the end-effector. In other word, the perturbation point moves along the trajectory and the perturbation direction $\boldsymbol{d}$ is in this case focused towards the centre of the circle. The coaching begins after 5 seconds. Here we can see in the right plot that the motion is constantly modified and directed towards the center of the circle. The final task space trajectory is indicated with the green line and the initial trajectory with the red line. In the first three plots left we can also see that as expected, the amplitude of motion is decreasing for all three joints.

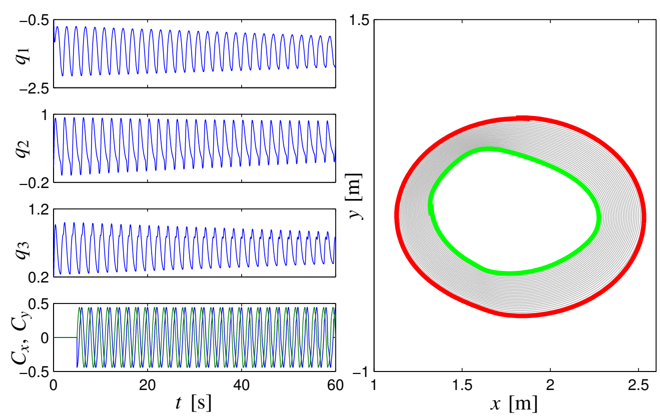

Fig. 4. Simulation results where the circular motion in task space was pushed in all the time, i. e. the direction of coaching command was towards the center of the circle all the time. The coaching began after 5 seconds.

IV. ROBOT EXPERIMENTS

To show the applicability of the proposed approach in real world, we implemented it on the JST-ICORP/SARCOS humanoid robot CBi [20]. We used the Microsoft Kinect sensor and the associated body tracker to capture human coaching gestures [21]. Fig. 5 shows the experimental setup, where the body tracking results can be seen on the display in the background.

To acquire the human coaching gestures in the coordinate system of the robot, we calibrated the Microsoft Kinect sensor to the robot base coordinate system. To obtain the appropriate transformation matrix, we recorded at least four pairs of points in both coordinate systems. For this purpose the human coach placed his hand at the same location as the robot's end-effector and the position of the human hand and the robot's end-effector were measured in the Kinect's and robot base coordinate system, respectively. The transformation matrix was calculated using least-squares fitting of two points set as described in [22].

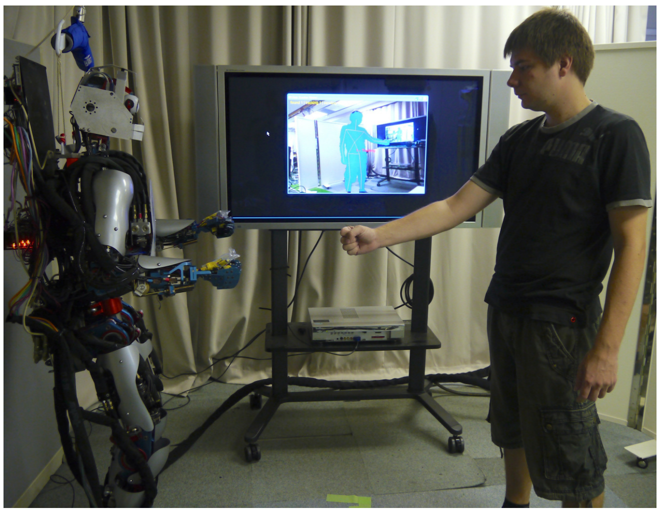

Fig. 5. Experimental setup, where a human coach is modifying the robot's motion. The human coaching gesture is captured using Microsoft Kinect sensor.

To make coaching as intuitive as possible, we developed an interface where the human coach can modify the trajectory by either pushing it away from him using his right hand or attracting it towards him with his left hand. The coaching direction was calculated using the wrist and the elbow location. For the right hand, which pushes the trajectory away form the coach, the direction is given by

$$\boldsymbol{d}_R = \frac{\boldsymbol{x}_{w,R} - \boldsymbol{x}_{e,R}}{||\boldsymbol{x}_{w,R} - \boldsymbol{x}_{e,R}||}, \tag{18}$$

where the $\boldsymbol{x}_{w,R}$ and the $\boldsymbol{x}_{e,R}$ are the Cartesian positions of the right hand wrist and the right hand elbow in the robot's base coordinate system. For attracting the trajectory towards the coach, the direction is given by

$$\boldsymbol{d}_L = \frac{\boldsymbol{x}_{e,L} - \boldsymbol{x}_{w,L}}{||\boldsymbol{x}_{e,L} - \boldsymbol{x}_{w,L}||}. \tag{19}$$

Here $\boldsymbol{x}_{w,L}$ and the $\boldsymbol{x}_{e,L}$ are respectively the Cartesian positions of the left hand wrist and the left hand elbow in the robot's base coordinate system.

Since Microsoft Kinect sensor relies on depth information and our humanoid robot has similar body proportions as a human, the body tracker sometimes becomes confused if the human approaches the robot very closely. For this reason the human coach did not approach the robot too closely in our experiments. Instead, the center of the potential field generated by each hand was moved slightly away from the respective hand. For the right hand, the origin of the potential field defined by the coaching gesture was moved in the direction of the coaching gesture

$$\boldsymbol{o}_R = \boldsymbol{x}_R + \xi_R \boldsymbol{d}_R, \tag{20}$$

where ξ_R is the scalar that defines the distance between the hand and the center of the coaching point in the direction of $\boldsymbol{d}_R$. Similar equation is used also for the left hand which attracts the trajectory towards the hand.

$$\boldsymbol{o}_L = \boldsymbol{x}_L - \xi_L \boldsymbol{d}_L. \tag{21}$$

Here, the effective coaching point is moved in the opposite direction of perturbation $\boldsymbol{d}_L$. With such modifications the effective origins of potential fields are always in front of the human hands in the direction of pointing at the distance defined by ξ_R and ξ_L.

To determine which hand is active, we use the distance between both wrist positions $\boldsymbol{x}_{w,L}$, $\boldsymbol{x}_{w,R}$ and the robot's end-effector position $\boldsymbol{x}$. The active hand is the one which is closer to the robot's hand position.

To show the applicability of the interface for online modification of the initial rhythmic movement using human in the loop coaching gestures, we first provide an example of pulling-in the task space trajectory. The parameters were set to $\gamma = 10$, $\eta = 10$, $r_m = 0.15$ and $\beta = -10/\pi$. Fig. 6 shows the task space motion of the robot's end-effector in the $x-y$ plane. We can see a successful modification of the motion based on the human coaching gestures. In Fig. 7 we show the corresponding joint space trajectories as a function of time. The teaching of the new motion pattern begins after 5 seconds, which is indicated with the first vertical line. We can see that the joint space trajectory was modified successfully to achieve the desired task space motion. In Fig. 7 we can see that at approximately 50 seconds the human coach stopped

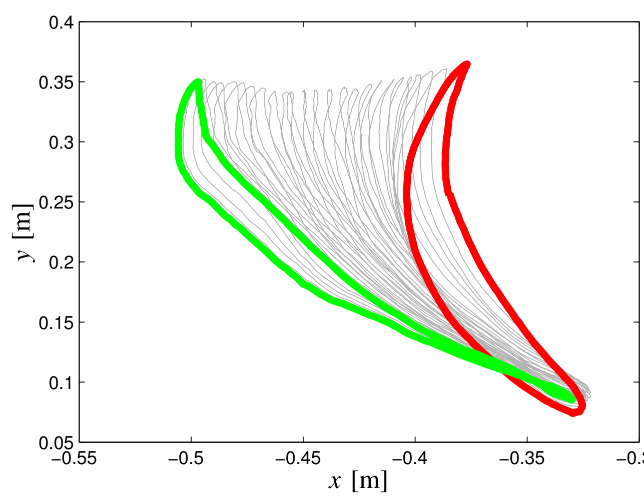

Fig. 6. Task space motion of the robot's end-effector, where human coach was modifying the motion pattern. The initial trajectory is in red and the final trajectory is in green. The time evolution of the trajectory modification is indicated with grey line.

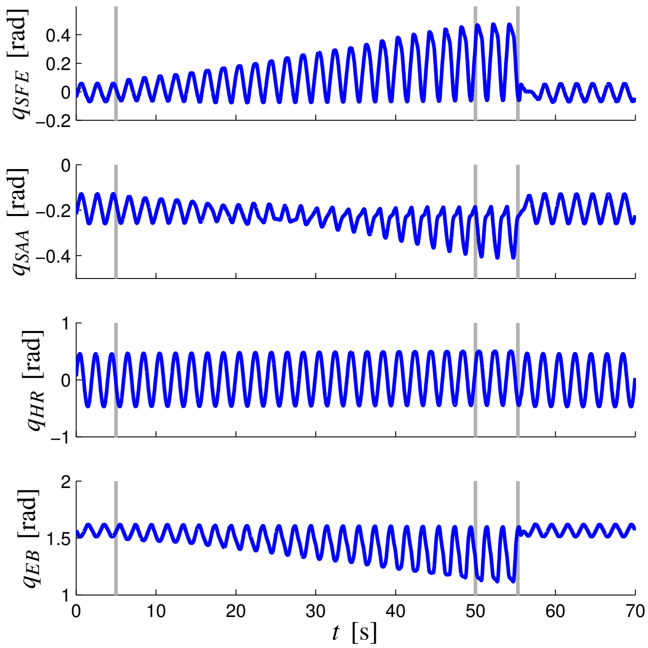

Fig. 7. Joint space motion in time of the robot's right hand, while coaching. Vertical lines indicate the important events described in text.

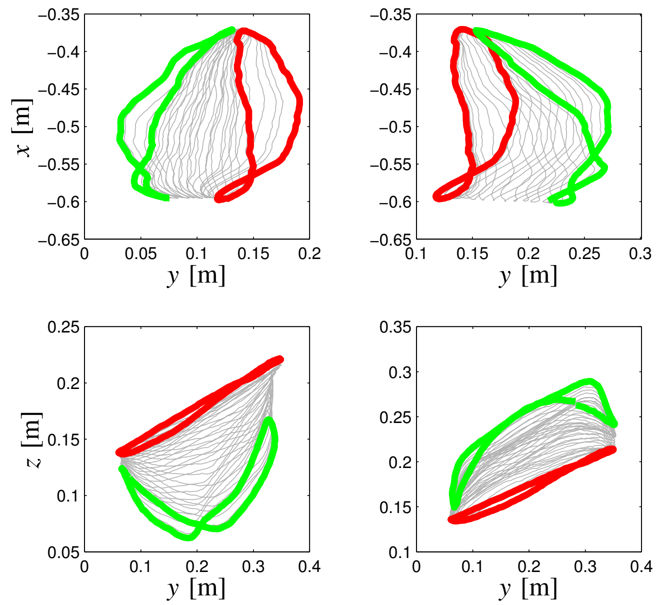

Fig. 9. Four different modifications of the original motion, which are also shown in the supplemental video. The top left graph corresponds to example 1, the top right graph to example 2, the bottom left graph to example 4 and the bottom right graph to example 5 in the supplemental video.

modifying the behavior and at approximately 55 seconds the new motion pattern was switched back to the original motion pattern. At this point the difference between original motion trajectory and the modified motion trajectory is even more evident. The snapshots showing the original and the modified trajectory of the humanoid arm movement are shown in Fig. 8. This experiment is also shown in the supplementary video.

Fig. 9 shows four different modifications of the original motion. In the top row we can see the horizontal pushing and pulling of the motion in the x-y plane and in the bottom row the vertical pushing and pulling in y-z plane. As we can see, the human coach was successful at modifying the movement of the robot in the desired direction using either the pushing or pulling technique, i. e. using either the right or the left hand to define the coaching gesture.

V. CONCLUSION AND FUTURE WORK

In this paper we developed a new coaching methodology that makes use of coaching gestures to modify an existing movement encoded by a periodic DMP. The DMP modification method is based on a recursive least-squares technique for updating the weights of periodic DMPs. With the developed system a human teacher can iteratively modify the previously acquired trajectories. It operates online and can therefore provide an immediate feedback to the coach. We presented simulation case studies where we successfully modified the joint space trajectories to obtain the new desired task space motions. The same method was also applied to the JST-ICORP/SARCOS humanoid robot CBi, where the human coach modified the humanoid robot's behavior to obtain the desired outcome. The proposed approach can easily be extended to discrete DMPs.

The main limitation of the coaching interface was the inability of the body tracker to distinguish between the robot and the human arm when a human teacher was close to the robot. Although the use of Microsoft Kinect sensor is beneficial because it can be used without much preparations, i. e. no markers or other special equipment is necessary, we believe that marker-based systems with more accurate tracking would provide a better and more accurate interface to modify the humanoid robot's movements. With a more reliable tracking of human coaching gestures, we could achieve similar results on the real robot as showed in Fig. 4, which is based on simulated data. The implementation and evaluation of the proposed algorithm with a more accurate body tracking system is an important goal of our future work. On the other hand, it is important that the human interface stays as intuitive as possible.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies) under grant agreements no. 270273, Xperience and no. 600716, CoDyCo. It was also supported by MEXT KAKENHI Grant Number 23120004; by JSPS and SRA: Japan-Slovenia research Cooperative Program; by MIC-SCOPE; by JST-SICP; by SRPBS, MEXT; by contract with the Ministry of Internal Affairs and Communications entitled 'Novel and innovative R&D making use of brain structures'.

REFERENCES

[1] M. Nakatani, K. Suzuki, and S. Hashimoto, "Subjective-evaluation oriented teaching scheme for a biped humanoid robot," in *IEEE-*

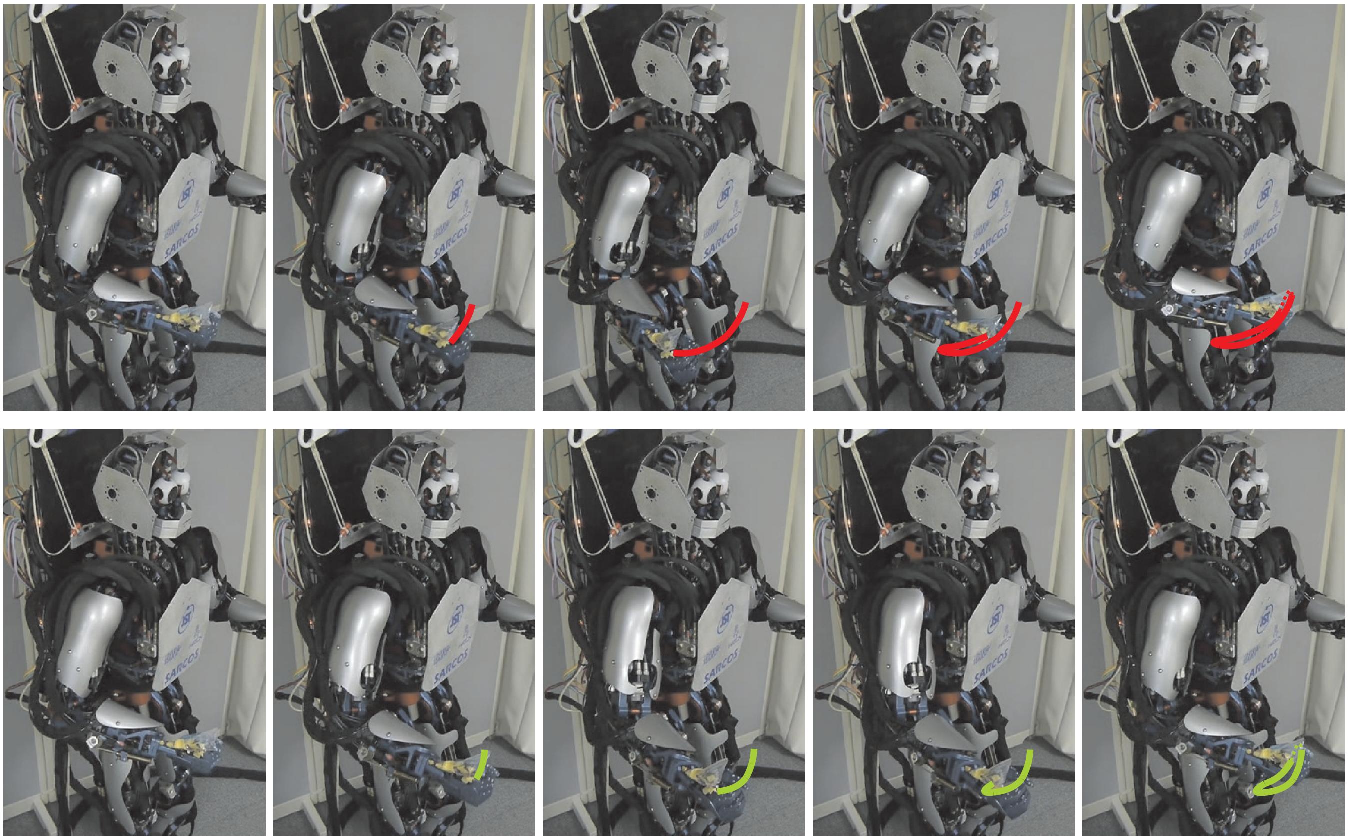

Fig. 8. A sequence of still photos showing the original motion in the top row and the final modified motion in the bottom row. The photos frame rate is 0.4 per second.

RAS International Conference on Humanoid Robots (Humanoids), Karlsuhe, Germany, 2003.

[2] A. Gruebler, V. Berenz, and K. Suzuki, "Coaching robot behavior using continuous physiological affective feedback," in *2011 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Bled, Slovenia, 2011, pp. 466–471.

[3] M. N. Nicolescu and M. J. Mataric, "Natural methods for robot task learning: Instructive demonstrations, generalization and practice," in *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 2003, pp. 241–248.

[4] M. Riley, A. Ude, C. Atkeson, and G. Cheng, "Coaching: An approach to efficiently and intuitively create humanoid robot behaviors," in *2006 6th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Genoa, Italy, 2006, pp. 567–574.

[5] D. Lee and C. Ott, "Incremental kinesthetic teaching of motion primitives using the motion refinement tube," *Autonomous Robots*, vol. 31, no. 2-3, pp. 115–131, 2011.

[6] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in Cognitive Sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[7] A. Billard and K. Dautenhahn, "Experiments in learning by imitation – grounding and use of communication in robotic agents," *Adaptive Behavior*, vol. 7, no. 3-4, pp. 415–438, 1999.

[8] A. Ude, C. G. Atkeson, and M. Riley, "Programming full-body movements for humanoid robots by observation," *Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 93–108, 2004.

[9] T. Asfour, P. Azad, F. Gyarfas, and R. Dillmann, "Imitation learning of dual-arm manipulation tasks in humanoid robots," *International Journal of Humanoid Robotics*, vol. 5, no. 02, pp. 183–202, 2008.

[10] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Berlin, Heidelberg: Springer Verlag, 2008.

[11] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010.

[12] R. Schmidt and T. Lee, *Motor Control and Learning: A Behavioral Emphasis*. Champaign, IL: Human Kinetics Publishers Ltd., 2011.

[13] S. Schaal, P. Mohajerian, and A. Ijspeert, "Dynamics systems vs. optimal control – a unifying view," *Progress in Brain Research*, vol. 165, pp. 425–445, 2007.

[14] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural Computation*, vol. 25, no. 2, pp. 328–373, 2013.

[15] A. Gams, A. J. Ijspeert, S. Schaal, and J. Lenarčič, "On-line learning and modulation of periodic movements with nonlinear dynamical systems," *Autonomous robots*, vol. 27, no. 1, pp. 3–23, 2009.

[16] T. Petrič, A. Gams, A. J. Ijspeert, and L. Žlajpah, "On-line frequency adaptation and movement imitation for rhythmic robotic tasks," *The International Journal of Robotics Research*, vol. 30, no. 14, pp. 1775–1788, 2011.

[17] A. Ude, A. Gams, T. Asfour, and J. Morimoto, "Task-specific generalization of discrete and periodic dynamic movement primitives," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 800–815, 2010.

[18] H. Hoffmann, P. Pastor, D.-H. Park, and S. Schaal, "Biologically-inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance," in *IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 2009, pp. 2587–2592.

[19] L. Žlajpah, "Simulation in robotics," *Mathematics and Computers in Simulation*, vol. 79, no. 4, pp. 879–897, 2008.

[20] G. Cheng, S.-H. Hyon, J. Morimoto, A. Ude, J. G. Hale, G. Colvin, W. Scroggin, and S. C. Jacobsen, "CB: A humanoid research platform for exploring neuroscience," *Advanced Robotics*, vol. 21, no. 10, pp. 1097–1114, 2007.

[21] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE MultiMedia*, vol. 19, no. 2, pp. 4–10, 2012.

[22] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, 1987.