

On-line periodic movement and force-profile learning for adaptation to new surfaces

Andrej Gams, Martin Do, Aleš Ude, Tamim Asfour and Rüdiger Dillmann

Abstract—To control the motion of a humanoid robot along a desired trajectory in contact with a rigid object, we need to take into account forces that arise from contact with the surface of the object. In this paper we propose a new method that enables the robot to adapt its motion to different surfaces. The initial trajectories are encoded by dynamic movement primitives, which can be learned from visual feedback using a two-layered imitation system. In our approach these initial trajectories are modified using regression methods. The data for learning is provided by force feedback. In this way new trajectories are learned that ensure that the robot can move along the object while maintaining contact and applying the desired force to the object. Active compliance can be used more effectively with such trajectories. We present the results for both movement imitation and force profile learning on two different surfaces. We applied the method to the ARMAR-IIIb humanoid robot, where we use the system for learning and imitating a periodic task of wiping a kitchen table.

I. INTRODUCTION

The use of a humanoid robot in a kitchen is becoming a viable scenario [1], [2]. The ability to learn new tasks in a natural way and without the need of an expert is one of the key functionalities for a truly effective and useful system. In this paper we present a two-layered system for movement imitation, which utilizes both visual and force feedback. We applied the system on a humanoid robot to achieve natural learning and effective use in a kitchen environment for the task of wiping flat or uneven surfaces.

The task of wiping a kitchen table is easy for a human, but involves several challenges for a humanoid robot. The robot has to perform periodical movement in contact with a heavy rigid object. Besides the height of the object, which can differ, depending on the height of kitchen counter or kitchen table, the location relative to the base of the robot is not fixed. Furthermore, wiping movement can be of different types. We therefore need effective methods to learn from humans and to generalize from existing knowledge.

Imitating movement with robots is a common approach [3], [4]. Different trajectory encodings have been proposed in the literature, e.g. splines [5] or dynamic movement primitives (DMP) [6]. To learn the associated parameters methods like statistical regression [7] or reinforcement learning [8] have been proposed. In this work we use dynamic movement primitives. DMPs use a set of kernel functions to

reproduce trajectories. They have favorable properties in the sense of continuous and smooth trajectories in the presence of obstacles [9] and their modulation characteristics make them especially useful for learning whole families of similar movements from a single demonstration.

For the task of learning of periodic movements we need to extract the frequency and the waveform of the demonstrated movement. A two-layered movement imitation system based on adaptive frequency oscillators in a feedback loop and a DMP formulation for waveform learning allows easy, natural, and computationally inexpensive learning and replaying of demonstrated trajectories [9]. The two-layered system of movement imitation can be used for learning of periodic trajectories in as many degrees of freedom as we can effectively measure.

When imitating human movement with a humanoid robot we can either learn movements in joint or in task space. Both have their advantages. For example when measuring the movement in joint space we have direct control over separate joints (see for example [10]). Measuring in task space, on the other hand, hands over the control of separate joints to on-line inverse kinematics algorithms. Different algorithms that try to maintain human-like postures when applying inverse kinematics algorithms exist, e.g. by minimizing the sum of exerted torques [11] or by optimizing the manipulability index [12]. As the influence of external forces on the movements is specified in task space, we specify trajectories in task space.

Contact with the environment requires tactile sensing abilities, such as for example a force-torque sensor. Controlling rigid robots while in contact with the environment is a difficult task [13], and often passive compliance elements are used to reduce the complexity. Although wiping a kitchen table, with a sponge might be successful, adding active compliance increases the versatility of feasible tasks and allows faster and more precise movement along the surface. This can be further increased by learning the force-profile of a periodic task.

The paper is organized as follows. In Section II we give a review of the two-layered movement imitation system and present force-profile learning, which allows performing periodic tasks while in contact with an object of an arbitrary shape. In Section III we present the kitchen experiment with the task of wiping an arbitrary shaped surface. We give conclusions in Section IV.

Andrej Gams and Aleš Ude are with Department of Automation, Biocybernetics and Robotics, Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia andrej.gams@ijs.si

Martin Do, Tamim Asfour and Rüdiger Dillmann are with Institute for Anthropomatics, Humanoids and Intelligence Systems Laboratories, Karlsruhe Institute of Technology, Karlsruhe, Germany

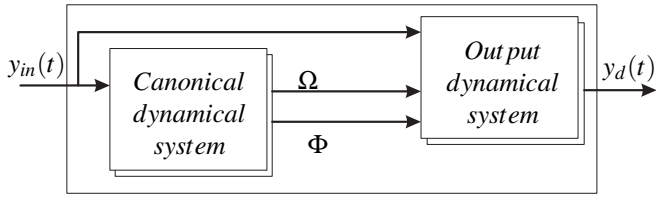


Fig. 1. Proposed two-layered structure of the control system. The input y_{in} is a measured quantity and the output is the desired trajectory y_d of the robot. The system can work in parallel for an arbitrary number of dimensions.

II. PERIODIC TRAJECTORY LEARNING

In this section we give a recap of the two-layered movement imitation system [9], which allows learning of periodic trajectories and the frequency of execution. After learning of periodic movements in task space with visual feedback we apply the system to learn the force-profile of objects by means of force feedback.

A. Learning of movements

The structure of the imitation system has two layers, as is presented in Fig. 1. The first layer – the Canonical dynamical system – extracts the fundamental frequency of the input signal. The second layer of the system is the Output dynamic system, which outputs the desired trajectory for the robot. The input signal can be any measured periodic quantity, in our case it is the visual feedback.

The Canonical dynamical system has to extract the fundamental frequency Ω and the phase signal Φ from the input signal. As the basis of the Canonical dynamical system we use a set of M adaptive frequency phase oscillators [14] in a feedback structure [15], as shown in Fig. 2.

The feedback structure of M adaptive frequency phase oscillators is governed by

$$\dot{\phi}_i = \omega_i - Ke(t) \sin(\phi_i), \quad (1)$$

$$\dot{\omega}_i = -Ke(t) \sin(\phi_i), \quad (2)$$

$$e(t) = y_{in}(t) - \hat{y}(t), \quad (3)$$

$$\hat{y}(t) = \sum_{i=1}^M \alpha_i \cos(\phi_i), \quad (4)$$

$$\dot{\alpha}_i = \eta \cos(\phi_i) e(t), \quad (5)$$

where K is the coupling strength, ϕ_i , $i = 1, \dots, M$ is the phase of a separate oscillator, $e(t)$ is the input into the oscillators, y_{in} is the input signal, M is the number of oscillators, α_i is

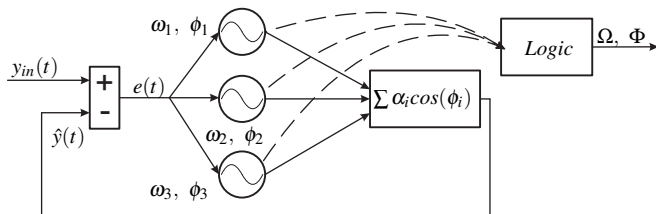


Fig. 2. Feedback structure of M nonlinear adaptive frequency oscillators.

the amplitude associated with the i -th oscillator, and η is the learning constant.

Each of the oscillators of the feedback structure receives the same input signal, which approaches zero when the weighted sum of separate frequency components \hat{y} approaches the input signal (see Fig. 2). Such a feedback structure can extract M frequency components of the input signal. Since we imitate periodic human movement and are interested only in the basic frequency, we use $M = 3$ oscillators. The feedback structure is followed by a logical algorithm, which chooses the basic or fundamental frequency Ω from the three extracted frequencies. Other approaches that do not include a logical algorithm also exist [16]. Determining the correct frequency is crucial, because it allows the learning of a single period of movement in the Output dynamical system.

The Canonical dynamical system can be used as an imitation system by itself, see [15], [16]. Adding the Output dynamical system allows exploiting the modulation and robustness properties of DMPs, which are important for movements in contact with rigid surfaces.

The Output dynamical system is a periodic DMP structure, where the waveform is anchored to the extracted phase. As illustrated in Fig. 1, the inputs are 1) the signal we are trying to learn and imitate, given by triplets of position, velocity and acceleration in discrete time steps; 2) the frequency Ω and 3) the phase Φ . The latter two are provided by the Canonical dynamical system.

A periodic DMP structure is given as a second order system of differential equations

$$\dot{s} = \Omega \left(\alpha_s (\beta_s (g - y) - s) + \frac{\sum_{i=1}^N \psi_i w_i r}{\sum_{i=1}^N \psi_i} \right), \quad (6)$$

$$\dot{y} = \Omega s. \quad (7)$$

The system above includes a nonlinear term specified by basis functions ψ_i . The nonlinear term provides the desired waveform of the output by multiplying the Gaussian-like kernel functions ψ_i and the weights w_i . In (6) Ω is the frequency given by the Canonical dynamical system, and $\alpha_s = 12$ and $\beta_s = 3$ are positive constants which ensure that the system monotonically varies to the trajectory that is oscillating around an anchor point g . N is the number of kernel functions and r is the amplitude scaling factor. The kernel functions are given by

$$\psi_i = \exp(h(\cos(\phi - c_i) - 1)), \quad (8)$$

The parameters of the kernel functions are h , which determines their width, and c_i , which distribute them over the course of one period. c_i are equally spaced between 0 and 2π in N steps.

The waveform of the learned movement is updated incrementally in every cycle as is explained for the force profile learning in the next section. We use visual feedback to learn

the initial periodic movement, which we later modify by force profile adaptation.

B. Force profile adaptation

Learning of a movement that brings the robot into contact with the environment must be based on force control, otherwise there can be damage to the robot or the object to which the robot applies its force. In the task of wiping a table or any other object of arbitrary shape constant contact with the object is required. To teach the robot the necessary movement, we decoupled the learning of the movement from the learning of the shape of the object. We first apply the described two-layered movement imitation system to learn the desired trajectories by means of visual feedback. We then use force-feedback to adapt the motion to the shape of the object that the robot acts upon.

Periodic movements can be of any shape, yet wiping can be effective with simple one dimensional left-right movement, or circular movement. Once we are satisfied with the learned movement, we can reduce the frequency of the movement by modifying the Ω value. The low frequency of movement and consequentially low movement speed reduce the possibility of any damage to the robot. When performing playback we modify the learned movement with an active compliance algorithm. The algorithm is based on the velocity-resolved approach [17]. The end-effector velocity is calculated by

$$\mathbf{v}_r = \mathbf{S}_v \mathbf{v}_v + \mathbf{K}_F \mathbf{S}_F (\mathbf{F}_m - \mathbf{F}_0). \quad (9)$$

Here \mathbf{v}_r stands for the resolved velocities vector, \mathbf{S}_v for the velocity selection matrix, \mathbf{v}_v for the desired velocities vector, \mathbf{K}_F for the force gain matrix, \mathbf{S}_F for the force selection matrix, and \mathbf{F}_m for the measured force. \mathbf{F}_0 denotes the force offset which determines the behavior of the robot when not in contact with the environment. To get the desired positions we use

$$\mathbf{Y} = \mathbf{Y}_r + \mathbf{S}_F \int \mathbf{v}_r dt. \quad (10)$$

Here \mathbf{Y}_r is the desired initial position and $\mathbf{Y} = (y_j)$, $j = 1, \dots, 6$ is the actual position/orientation. Using this approach we can modify the trajectory of the learned periodic movement as described below.

Equations (9 – 10) become simpler for the specific case of wiping a flat surface. By using a null matrix for \mathbf{S}_v , $\mathbf{K}_F = \text{diag}(0, 0, k_F, 0, 0, 0)$, $\mathbf{S}_F = \text{diag}(0, 0, 1, 0, 0, 0)$, the desired end-effector height z in each discrete time step Δt becomes

$$\dot{z}(t) = k_F (F_z(t) - F_0), \quad (11)$$

$$z(t) = z_0 + \dot{z}(t) \Delta t. \quad (12)$$

Here z_0 is the starting height, k_F is the force gain (of units kg/s), F_z is the measured force in the z direction and F_0 is the force with which we want the robot to press on the object. Such formulation of the movement ensures constant movement in the $-z$ direction, or constant contact when an object is encountered. Another simplification is to use

the length of the force vector $F = \sqrt{F_x^2 + F_y^2 + F_z^2}$ for the feedback instead of F_z in (11). This way the robot can move upwards every time it hits something, for example the side of a sink. No contact should be made from above, as this will make the robot press up harder and harder.

The learning of the force profile is done by modifying the weights w_i for the selected degree of freedom y_j in every time-step by incremental locally weighted regression [7]. The input in this case is $Y(t)$ from (10). The target for learning is determined by $f_{targ} = \frac{1}{\Omega^2} \ddot{y}_j - \alpha_s (\beta_s (g - y_j) - \frac{1}{\Omega} \dot{y}_j)$, which is obtained by matching y from (6 – 7) to y_j , s to $\frac{\dot{y}_j}{\Omega}$, and \dot{s} to $\frac{\ddot{y}_j}{\Omega}$. Given the target data $f_{targ}(t)$ and $r(t)$, with the frequency Ω and the phase Φ the values set for the playback, w_i , $i = 1, \dots, N$ are updated in recursion by

$$w_i(t+1) = w_i(t) + \Psi_i P_i(t+1) r(t) e_r(t), \quad (13)$$

$$P_i(t+1) = \frac{1}{\lambda} \left(P_i(t) - \frac{P_i(t)^2 r(t)^2}{\frac{\lambda}{\Psi_i} + P_i(t) r(t)^2} \right), \quad (14)$$

$$e_r(t) = f_{targ}(t) - w_i(t) r(t). \quad (15)$$

The recursion is started with $w_i = 0$ and $P_i = 1$, where $i = 1, \dots, N$. P is in general the inverse covariance matrix and $\lambda = 0.99$ is the forgetting factor.

The \mathbf{K}_F matrix controls the behaviour of the movement. The correcting movement has to be fast enough to move away from the object if the robot hand encounters sufficient force, and at the same time not too fast so that it does not produce instabilities due to the discrete-time sampling when in contact with an object. A dead-zone of response has to be included, for example $|F| < 1$ N, to take into account the noise. We empirically set $k_F = 20$, and limited the force feedback to allow maximum linear velocity of 120 mm/s.

Feedback from a force-torque sensor is often noisy due to the sensor itself and mainly due to vibrations of the robot. A noisy signal is not the best solution for the learning algorithm because we also need time-discrete first and second derivatives. The described active compliance algorithm uses the position of the end-effector as input, which is the integrated desired velocity and therefore has no difficulties with the noisy measured signal.

Having adapted the trajectory to the new surface enables very fast movement with a constant force profile at the contact of the robot/sponge and the object, without any time-sampling and instability problems that may arise when using compliance control only. Furthermore, we can still use the compliant control once we have learned the shape of the object. Active compliance, combined with a passive compliance of a sponge, and the modulation and perturbation properties of DMPs, such as slow-down feedback [9], allow fast and safe execution of periodic movement while maintaining a sliding contact with the environment.

III. EXPERIMENTAL EVALUATION

In this section we describe the experimental setup and present the results on the ARMAR-IIIb humanoid robot.

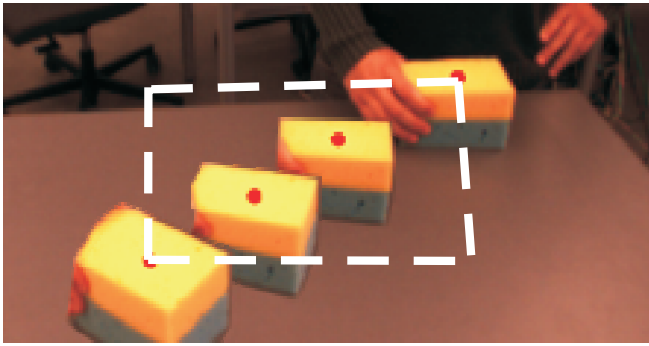


Fig. 3. Area for movement demonstration is determined by measuring the bottom-left most and the top-right most positions within a given time frame. These coordinates make a rectangular area (marked with dashed lines) where the robot tracks the demonstrated movements.

A. Experimental setup

1) *ARMAR-IIIb humanoid robot*: The humanoid robot ARMAR-IIIb, which serves as the experimental platform in this work, is a copy of the humanoid robot ARMAR-IIIa [1]. From the kinematics point of view, the robot consists of seven subsystems: head, left arm, right arm, left hand, right hand, torso, and a mobile platform. The head has seven DOF and is equipped with two eyes, which have a common tilt and can pan independently. Each eye is equipped with two digital color cameras, one with a wide-angle lens for peripheral vision and one with a narrow-angle lens for foveal vision. The upper body of the robot provides 33 DOF: 2·7 DOF for the arms and three DOF for the torso. The arms are designed in an anthropomorphic way: three DOF for each shoulder, two DOF in each elbow and two DOF in each wrist. Each arm is equipped with a five-fingered hand with eight DOF. The locomotion of the robot is realized using a wheel-based holonomic platform.

2) *Vision and force feedback*: In order to obtain reliable motion data of a human wiping demonstration through observation by the robot, we exploited the color features of the sponge to track its motion. Using the stereo camera setup of the robot, the implemented blob tracking algorithm based on color segmentation and a particle filter framework provides a robust location estimation of the sponge in 3D. The resulting trajectories were captured with a frame rate of 30 Hz.

For learning of movements we first define the area of demonstration by measuring the lower-left and the upper-right position within a given time-frame, as is presented in Fig. 3. All tracked sponge-movement is then normalized and given as offset to the central position of this area.

For measuring the contact forces between the object in the hand and the surface of the plane a 6D-force/torque sensor is used, which is mounted at the wrist of the robot.

3) *The learning scenario*: Our kitchen scenario includes the ARMAR-IIIb humanoid robot wiping a kitchen table. First the robot attempts to learn wiping movement from human demonstration. During the demonstration of the desired wiping movement the robot tracks the movement of

the sponge in the demonstrator's hand with his eyes. The robot only reads the coordinates of the movement in a horizontal plane, and learns the frequency and waveform of the movement. The waveform can be arbitrary, but for wiping it can be simple circular or one-dimensional left-right movement. The learned movement is encoded in the task space of the robot, and an inverse kinematics algorithm controls the movement of separate joints of the 7-DOF arm. The robot starts mimicking the movement already during the demonstration, so the demonstrator can stop learning once he/she is satisfied with the learned movement. Once the learning of periodic movement is stopped, the term $F - F_0$ in (11) provides velocity in the direction of $-z$ axis, and the hand holding the sponge moves towards the kitchen table or any other surface under the arm. As the hand makes contact with the surface of an object, the vertical velocity adapts. The force profile is learned in a few periods of the movement. The operator can afterwards stop force profile learning and execute the adjusted trajectory at an arbitrary frequency.

B. Evaluation

1) *Learning of periodic movement*: The two-layered movement imitation system extracts the frequency and learns one period of the waveform. As we can see from the results in Fig. 4, the system extracts three frequency components, of which one is the offset (0 Hz). As we expect relatively simple waveform for wiping movements we only use $M = 3$ oscillators and always choose the middle of the three frequencies for the fundamental frequency. This way the chosen frequency is never associated with the offset or the higher frequency components present. We simply reset the oscillators if more than one frequency component converges to the 0 Hz frequency (see also [9]). The learning of the waveform happens simultaneously with the learning of the frequency. Once we are satisfied with the performed movement, we stop the learning. After learning we can modulate the frequency value Ω .

Fig. 5 shows the adaptation of the waveform during learning for both dimensions in the xy plane. The input signal is transformed from task-space coordinates to the normalized deviation from the center of the visual learning frame (see Fig. 3). As we can see the waveform adapts quickly. Some delay can be observed, which can be attributed to the slow frame-rate of the visual feedback at 30 Hz, and the learning algorithm itself. Once we are satisfied with the waveform we

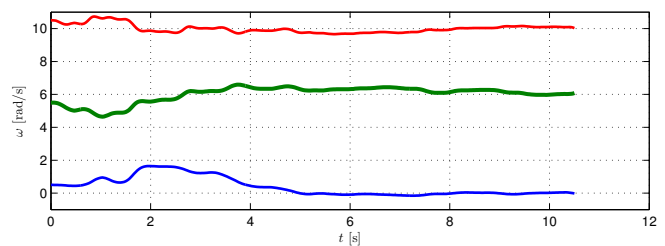


Fig. 4. Result of frequency adaptation. The middle ω was chosen as the fundamental frequency Ω .

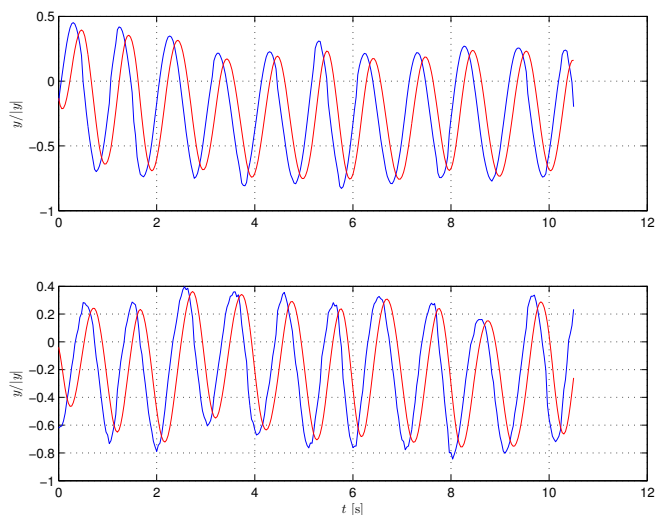


Fig. 5. Visual feedback results (blue) and the learned waveform (red). The learned waveform has a delay of a few time-samples, which at 30 Hz leads to approx. 0.2 s. The waveform is relatively non-complex and learned very fast.

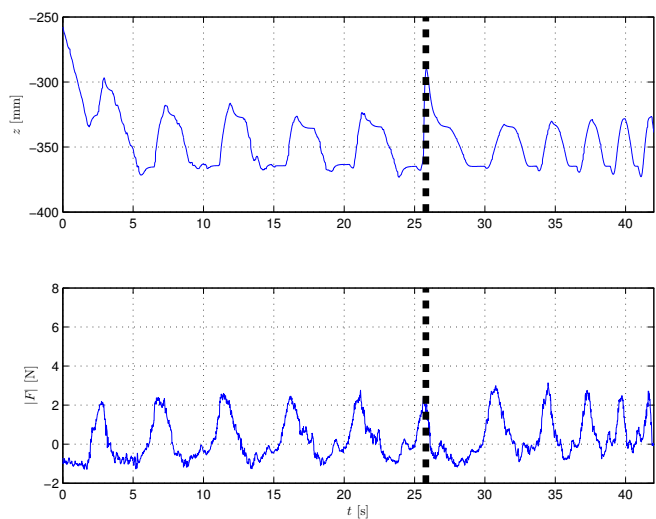


Fig. 6. Results of learning the force profile on a flat surface. Because of an unknown TCP after grasping a sponge, and changes in the orientation due to joint limits, the height of the movement changes for approx. 5 cm during one period to maintain contact with the surface. The values were attained trough robot kinematics. A dashed vertical line marks the end of learning of the force profile. Increase in frequency can be observed in the end of the plot. The increase was added manually. An overshoot at the time of the end of learning is due to an implementation error of the switch of the phase.

stop the learning, and can then arbitrarily scale and offset the learned movement.

2) *Learning of force profile*: Once we stop the movement learning we start learning the force-profile, with which we learn the trajectory required to maintain constant contact between the sponge and the robot. A video showing the experiment can be found at <http://www.ijs.si/~aude/ForceLearning.mov>.

Fig. 6 shows the results of learning the force-profile for a flat surface. As the robot grasps the sponge, its orientation

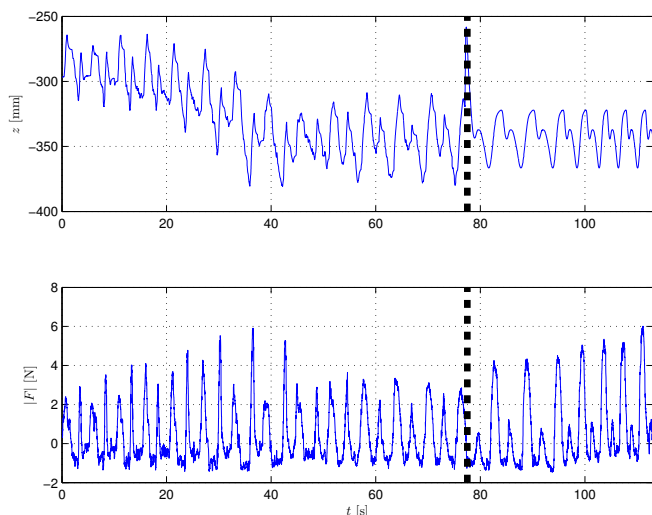


Fig. 7. Results of learning the force profile on a bowl-shaped surface. From the start the height decreases and then assumes a bowl-shape with an additional change of direction, which is the result of the compliance of the wiping sponge and the zero-velocity dead-zone. A dashed vertical line marks the end of learning of the force profile. An overshoot at the time of the end of learning is due to an implementation error of the switch of the phase.

and location are unknown to the robot, and the tool center point (TCP) changes. Should the robot simply perform a planar trajectory it would not ensure constant contact with the table. As we can see from the results, the hand initially moves down until it makes contact with the surface. The force profile later changes the desired height by approx. 5 cm within one period. After the learning (stopped manually, marked with a vertical dashed line) the robot maintains such a profile. A manual increase in frequency was introduced to demonstrate the ability to perform the task at an arbitrary frequency. The bottom plot shows the measured length of the force vector $|F|$. As we can see the force vector keeps the same force profile, even though the frequency is increased. No increase in the force profile proves that the robot has learned the required trajectory.

Fig. 7 shows the results for a bowl-shaped object. As we can see from the results the height of the movement changes for more than 6 cm within a period. The learned shape (after the vertical dashed lined) maintains the shape of a bowl, but has an added local minimum. This is the result of the dead-zone within the active compliance, which comes into effect when going up one side, and down the other side of the bowl. No significant change in the force profile can be observed in the bottom plot after a manual increase in frequency. Some drift, as the consequence of an error of the sensor and of wrist control on the robot, can be observed.

IV. CONCLUSION

We have shown the use of a two-layered imitation system with decoupled periodic movement learning and force profile learning. The main contribution of the paper is a new algorithm for force profile learning. We have demonstrated the usefulness of our approach for the implementation of a

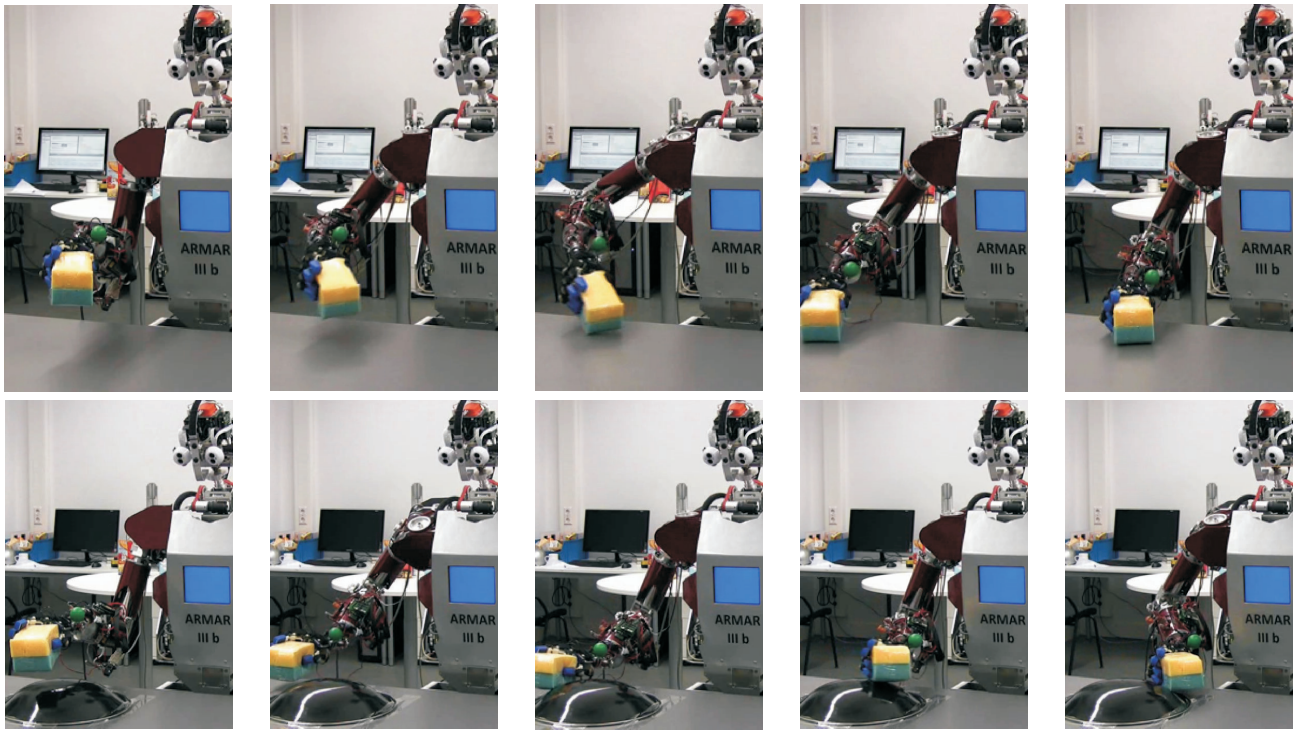


Fig. 8. A sequence of still photos showing the adaptation of wiping movement via force-feedback to a flat surface, as of a kitchen table, in the top row, and adaptation to a bowl-shaped surface in the bottom row. A video showing the experiment can be found at <http://www.ijs.si/~aude/ForceLearning.mov>.

wiping behaviour. Learning of the force-profile (or of the shape of the object), in combination with both active and passive compliance, allows faster and more robust execution of tasks. Our approach thus implements an important behaviour for the future.

ACKNOWLEDGMENT

The work described in this paper was partially conducted within the EU Cognitive Systems projects PACO-PLUS (FP6-027657) and GRASP (FP7- 215821) funded by the European Commission. It was partially funded by the Slovenian Research Agency grant Z2-3672.

REFERENCES

- [1] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann. ARMAR-III: An Integrated Humanoid Platform for Sensory-Motor Control. In *IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids 2006)*, Genoa, Italy, 2006.
- [2] R. B. Rusu, B. Gerkey, and M. Beetz. Robots in the kitchen: Exploiting ubiquitous sensing and actuation. *Robot. Auton. Syst.*, 56(10):844–856, 2008.
- [3] S. Schaal. Is imitation learning the route to humanoid robots? *Trends in cognitive sciences*, 6:233–242, 1999.
- [4] A. Gams, L. Žlajpah, and J. Lenarčič. Imitating human acceleration of a gyroscopic device. *Robotica*, 25(4):501–509, 2007.
- [5] Hiroyuki Miyamoto, Stefan Schaal, Francesca Gandolfo, Hiroaki Gomi, Yasuharu Koike, Reiko Osu, Eri Nakano, Yasuhiro Wada, and Mitsuo Kawato. A kendama learning robot based on bi-directional theory. *Neural Networks*, 9(8):1281–1302, 1996.
- [6] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 1398–1403, Washington, DC, 2002.
- [7] C.G. Atkeson, A.W. Moore, and S. Schaal. Locally weighted learning. *AI Review*, 11:11–73, 1997.
- [8] J. Kober and J. Peters. Learning Motor Primitives for Robotics. In *Proceedings of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2112 – 2118, 2009.
- [9] A. Gams, A. Ijspeert, S. Schaal, and J. Lenarčič. On-line learning and modulation of periodic movements with nonlinear dynamical systems. *Autonomous robots*, 27(1):3–23, 2009.
- [10] A. Ude, C.G. Atkeson, and M. Riley. Programming full-body movements for humanoid robots by observation. *Robotics and autonomous systems*, 47:93–108, 2004.
- [11] O. Khatib, J. Warren, V. De Sapio, and L. Sentis. Human-like motion from physiologically-based potential energies. In J. Lenarčič and C. Galletti, editors, *On Advances in Robot Kinematics*, pages 149–163. Kluwer Academic Publishers, 2004.
- [12] A. Gams and J. Lenarčič. Humanoid arm kinematic modeling and trajectory generation. In *The First IEEE/RAS-EMBS Conference on Biomedical Robotics and Biomechanics (BioRob)*, Pisa, Italy, page 5. [Piscataway]: IEEE, 2006.
- [13] M. Vukobratovic, D. Surdilovic, Y. Ekalo, and D. Katic. *Dynamics and Robust Control of Robot-Environment Interaction*. World Scientific Publishing, 2009.
- [14] L. Righetti, J. Buchli, and A.J. Ijspeert. Dynamic hebbian learning in adaptive frequency oscillators. *Physica D*, 216(2):269–281, 2006.
- [15] L. Righetti and A. J. Ijspeert. Programmable Central Pattern Generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006.
- [16] T. Petrič, A. Gams, and L. Žlajpah. Frequency extraction based on adaptive fourier series - application to robotic yoyo. In *Proceedings of Icinco, to appear*, 2010.
- [17] L. Villani and De Schutter J. *Handbook of Robotics*, chapter Force Control. Springer, 2008.