2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)
November 3 - 5, 2015, Seoul, Korea

Learning of Parametric Coupling Terms for Robot-Environment Interaction

Andrej Gams, Miha Deniša and Aleš Ude

Abstract— **In order to be effective, learning of robotic motion by demonstration should not remain limited to direct repetition of movements, but should enable modifications with respect to the state of the external environment, and generation of actions for previously unencountered situations. In this paper we propose an approach that combines these two features, and applies them in the framework of dynamic movement primitives (DMP). The proposed approach is based on the notion of motion adaptation through the use of coupling terms introduced to the DMPs at the velocity level. The coupling term is learned in a few repetitions of the motion with iterative learning control (ILC). The adaptation, which is based on force feedback, derives from either autonomous contact with the environment, or from human intervention. It can adapt to a given constraint, e.g., to a desired force of contact or to a given position. The major novelty of this paper is in extending this notion with statistical generalization between the coupling terms, allowing online adaptation of motion to a previously unexplored situation. The benefit of the approach is in reduced effort in human demonstration, because a single demonstration can be autonomously adapted to different situations with ILC, and recording the learned coupling terms builds up a database for generalization. A side-effect of learning, which takes a few iterations, is that also the coupling terms of the learning attempts can be stored in the database, allowing for different generalization queries and outcomes. In the paper we provide the details on the approach, followed by simulated and real-world evaluations.**

I. INTRODUCTION

In order to be effective and utilizable, learning by demonstration (LbD), as a popular paradigm for learning of robotic motions [1], should and has surpassed simple repetition of demonstrated motion. This was demonstrated in the literature in numerous occasions, for example, the modification of motion in order to maintain stability [2], adaptation with respect to the frame of motion [3], or modulation of duration and final goal of reaching in the framework of dynamic movement primitives (DMP) [4]. DMPs offer the basis of the approach in this paper.

The framework of DMPs, extensively utilized in LbD, offers various means of modulations, which go beyond the change of duration of motion and the final goal. For example, feedback components have been introduced for obstacle avoidance [4], cooperation [5], and also for adaptation with respect to the force of contact with the environment and thus for force-based adaptation [6], [7]. Feedback modifications have been augmented by introducing learned feed-forward components, acquired with the use of iterative learning control (ILC) [8].

*This work was supported by EU Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies under grant agreement 600578, ACAT).

All authors are with Humanoid and Cognitive Robotics Lab, Dept. for Automatics, Biocybernetics and Robotics, Jožef Stefan Institute, Ljubljana, Slovenia. name.surname@ijs.si

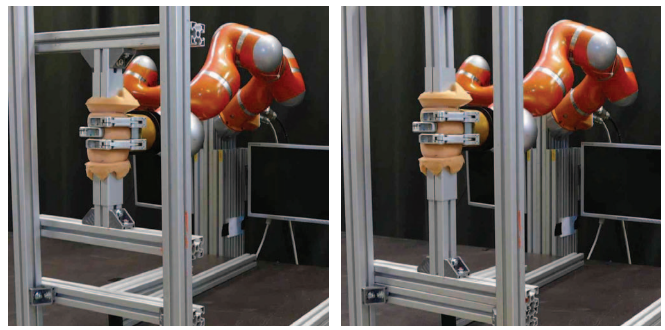

Fig. 1. Experimental setup for the evaluation of the algorithm, depicting the problem. As shown in this use-case, a single learned trajectory cannot account for all the products, as they are not all the same. The left image shows that the trajectory, which was learned for the large window-frame (on the right), is not applicable for the small one, because its bottom does not touch the stand.

Another aspect of adaptation to the external environment is with generation of new trajectories from a set of existing examples, to new, previously unexplored situations. Generation of new control policies from existing knowledge was demonstrated by Matsubara et al. [9], showing feasibility and highly extended scalability of DMPs. New DMP trajectories were generated from a library of motions using locally weighted regression (LWR) by Ude et al. [10]. As LWR is computationally demanding, Gaussian Process Regression (GPR) was used to online generalize between the weights of the DMPs by Forte et al. [11]. Generalization of DMPs was also combined with model predictive control (MPC) by Krug and Dimitrov [12]. Further, Stulp et al. have proposed learning the function approximator with one regression in the full space of phase and tasks parameters, bypassing the need for two consecutive regressions [13]. Generalization across start states was also shown by [14] and [15]. The latter, similar as [3], does not use DMPs.

In this paper we combine the adaptation of dynamic movement primitives through coupling terms [8] and online statistical generalization of DMP parameters using Gaussian process regression into an approach to enable modifications with respect to the state of the external environment, and generation of actions to previously un-encountered situations. In our approach we propose generalizing between the weights of the coupling terms and not the DMP trajectories. Several reasons motivate and speak in favor of such an approach. First, in order to generalize from existing knowledge, one needs to acquire the library of motions, which can be tedious and demanding [10], or may even require the acquisition of parameters, which are not attainable from a purely human demonstration, such as the force profiles [7]. With the proposed approach, a single demonstration is needed. This demonstration is then autonomously adapted through learned feed-forward components of the coupling terms, to comply with the given constraint – a force of contact, an end or intermediate position, etc. Thus, the effort of the human is considerably reduced and the database may be expanded completely autonomously.

Second, the adaptation of trajectories, through ILC, can be influenced by the human, who can push or pull on the robot and thus modify the trajectory in a manner of a coach, not unlike the approach in [16]. This provides effective means for knowledge transfer and human-robot interaction for either domestic or industrial environments. Given that the approach combines a learned feed-forward and a feedback component, which is used for instantaneous reaction, it allows increased means of safe human-robot cooperation.

Finally, the learning, which takes a few iterations, also produces coupling terms, which can be stored in the database, allowing for different generalization queries and outcomes. It should also be noted, that the generalization may provide an insufficient coupling term. Such a coupling term may still serve as the initial approximation for further refinement of the trajectory with ILC to the given constraint.

Other methods of learning the trajectories could be utilized. Reinforcement learning methods have proven well suited for the task of trajectory learning, and have provided numerous solutions, including in combination with generalization [17], [18]. The advantage of using ILC is that it takes only a fraction of repetitions needed for explorative methods, which often require plenty.

While far from the only means of adaptation and generalization of DMPs, our approach provides an effective way of autonomous and/or human-guided knowledge acquisition and adaptation of trajectories to the current state of the environment. The approach can be utilized to teach future robotic household assistants, or in industry, where it can be used to cope with the ever increasing demands of small-batch and mass-customized production, as it greatly reduces the needed time and cost for reprogramming the robot to a new product – two of the great obstacles hindering robotic production in small and medium enterprises. An illustration of the problem and the experimental setup for the evaluation of the approach are depicted in Fig. 1.

In the following we first provide an overview of the approach in Section II, followed by the details on the building blocks – coupled DMPs and ILC in Section III; and statistical generalization using GPR in Section IV. Experimental evaluation results are provided in Section V, and a discussion concludes the paper in Section VI.

II. ADAPTATION OF MOTION

In this Section we provide a short recap of the whole approach, given in Fig. 2 as a pseudo-algorithm. Individual blocks are explained in the following Sections.

The major novelty of the approach is in generalizing between the coupling terms of the DMPs. Given a database of coupling terms, the approach can provide an estimate of the motion on-line. If the database is not present, or relatively small, the learning time is reduced by taking the generalization outcome as the initial approximation for the ILC.

The coupling terms can be learned autonomously, where the robot executes the motion and modifies it based on the feedback information with ILC. The approach also allows for human intervention, where a person can push or pull on the robot in the manner of a tutor, coaching the robot through several iterations towards the desired target. While this might not provide the exact same reference in every iteration, it has been shown in [8] that human intervention and ILC can be used for modification of trajectories.

procedure GeneralizeCouplingTerm
    acquire one demonstration trajectory
        by kinesthetic guiding or imitation;
    set a target, i. e., a query point for generalization
    if no coupling term in database
        while error above predefined threshold
            run the trajectory with ILC on the coupling term
    else provide a generalized coupling term
    if error above threshold
        run the trajectory with ILC on the
        coupling term until error below threshold
end

Fig. 2. A pseudo-algorithm for the proposed approach. A target of generalization (a query is), for example, the window-frame size.

III. COUPLED DYNAMIC MOTION PRIMITIVES

In this Section we provide a recap on coupled dynamic movement primitives as they were introduced in [8]. The difference to standard DMPs is in a coupling term at the velocity level.

A. Coupled DMPs

Standard dynamic movement primitives (DMPs) are based on a damped spring-mass model [4]. To couple them to an external signal, we add the coupling term u at the velocity level, i. e., in (2), and a scaled derivative ($c_a\dot{u}$, c_a being a constant) at the acceleration level for less oscillations. For a single degree of freedom (DOF) this results in

$$\tau\dot{z} = \alpha_z(\beta_z(g-y)-z)+f(x)+c_a\dot{u}, \quad (1)$$

$$\tau\dot{y} = z+u. \quad (2)$$

Other than the coupling term, the rest of the DMP is the same as a classical DMP. $f(x)$ is defined as a linear combination of nonlinear radial basis functions (Ψ_i)

$$f(x) = \frac{\sum_{i=1}^{N} w_i\Psi_i(x)}{\sum_{i=1}^{N}\Psi_i(x)}x, \quad (3)$$

$$\Psi_i(x) = \exp\left(-h_i\,(x-c_i)^2\right), \quad (4)$$

where c_i are the centers of radial basis functions. $h_i > 0$ are their widths. Provided that parameters $\alpha_z,\ \beta_z,\ \tau > 0$ and $\alpha_z = 4\beta_z$, the system (1) – (2) is critically damped and has a unique attractor point at $y = g,\ z = 0$. A phase variable x is utilized to avoid direct dependency of f on time. Its dynamics is defined by

$$\tau\dot{x} = -\alpha_x x, \tag{5}$$

with initial value $x(0) = 1$. α_x is a positive constant. The details on the stability of coupled DMPs are provided in [8].

The coupling term u is composed of a feedback component, and a feed-forward component, learned with the use of ILC. Note that the feed-forward component of the coupling term is also encoded in a set or radial basis functions, and that it is subject to the same phase x as the original trajectory.

B. Iterative Learning Control

Iterative learning control (ILC) is a method that provides a way to minimize feedback error by adding a feed-forward term. The feedback error signal is used to improve the performance of motion with respect to the given reference, i. e., a given force amplitude or a given position, in the next repetition (iteration) of the same behavior. It updates the coupling term u after every attempt at a task. We propose to use the current-iteration ILC, which is defined by the formula [19]

$$u_{j+1}(k) = \underbrace{Q(s)(u_j(k) + Le_j(k+1))}_{\text{feedforward term}} + \underbrace{Ce_{j+1}(k)}_{\text{feedback term}}, \tag{6}$$

where u denotes the coupling term, k-th time sample is denoted k, and the learning iteration by j. $C > 0$ is the feedback gain. Q and L filters are defined differently in this particular case, with $L = 1$ a constant, determined empirically. $Q(s)$, on the other hand, is a low-pass analog filter, with a cut-off frequency of 2π rad/s, determined empirically. As higher frequency components of the error signal are not reduced with ILC, which might lead to oscillations, this low-pass filter is used to cancel them out [20]. ILC uses the prediction of the error $e(k+1, j)$ in the $j+1$ iteration. This serves to anticipate the error caused by the action taken at the k-th time step of the current iteration. ILC modifies the coupling term in the next iteration based on the coupling term and feedback error in the previous iteration.

Since the coupling term is composed of both feedback and feed-forward components, it also allows instantaneous reaction, which can be used for safer human-robot interaction, as the robots immediately gives way upon contact. In essence, the coupling with the DMP implements an admittance control scheme, where the velocity of the robot depends on the measured error, which is given by the difference of the desired and the actual force $e = F_d - F_a$. Note that positions can also be used instead of forces, and virtual forces are then used, modeled by a linear spring $F_v = k_v p$, with k_v the virtual spring gain and p the difference in positions.

The feed-forward component of the coupling term, updated with ILC, is after every iteration encoded in a set of radial basis functions (RBF) Ψ_j

$$u(x) = \frac{\sum_{j=1}^{M} a_j \Psi_j(x)}{\sum_{j=1}^{M} \Psi_j(x)} x, \tag{7}$$

with a_j the weights of the RBF. Having the coupling term encoded as such allows us to use Gaussian process regression for statistical generalization between acquired coupling terms.

IV. STATISTICAL GENERALIZATION OF COUPLING TERMS

While ILC can be used to adapt a DMP to variations in the environment and/or the task, some learning iterations are still needed. These re-executions can be avoided by using statistical generalization. First, let's assume a set of learned coupling terms $\mathbf{F}_u$ (given by a set of $\mathbf{a}_N$, N being the number of the coupling terms) that transition relatively smoothly between each other as a function of query points c. These queries describe certain variations in the task, e. g., final position of the movement, speed of execution, obstacle position, desired contact force, etc. The example set can be used with Gaussian Process Regression (GPR) [21] to calculate an appropriate coupling term for any query point within the training space.

In formal terms, our goal is to learn a function

$$\mathbf{F}_u : c \longmapsto [\mathbf{a}], \tag{8}$$

which uses the database of N learned coupling terms $\mathbf{u}$ to define a new coupling term, defined as the weights $\mathbf{a}$ of RBF terms (7), adapted to the new query point c.

Let r be one of the components of the output vector $\mathbf{a}$. For this one component, training data can be written as $\{r_k, c_k\}_{k=1}^{N}$, where r_k are now scalar valued outputs. Let $\boldsymbol{r} = [r_1, \ldots, r_N]^T$ be the vector combining all the training outputs. Let's also assume that we obtain a new set of inputs $c^* = [c_1^*, \ldots, c_K^*]^T$ for which the corresponding outputs $\boldsymbol{r}^* = [r_1^*, \ldots, r_K^*]^T$ should be computed. Assuming that the mean of training outputs $\{r_k\}$ is zero[1], Gaussian Process Regression can be applied to compute $\boldsymbol{r}^*$ as follows

$$\boldsymbol{r}^* \quad = \quad \boldsymbol{\Sigma}(\boldsymbol{C}^*, \boldsymbol{C}) \cdot \left[\boldsymbol{\Sigma}(\boldsymbol{C}, \boldsymbol{C}) + \sigma_n^2 \mathbf{I}\right]^{-1} \boldsymbol{r}. \tag{9}$$

Here $\boldsymbol{C} = \{c_1, \ldots, c_N\}$, $\boldsymbol{C}^* = \{c_1^*, \ldots, c_K^*\}$, σ_n is the noise variance of the output data and

$$\begin{aligned}
&\boldsymbol{\Sigma}(\{c_1, \ldots, c_K\}, \{c_1', \ldots, c_{K'}'\}) = \\
&\qquad = \begin{bmatrix} \mathrm{cov}(c_1, c_1') & \ldots & \mathrm{cov}(c_1, c_{K'}') \\ \vdots & \ldots & \vdots \\ \mathrm{cov}(c_K, c_1') & \ldots & \mathrm{cov}(c_K, c_{K'}') \end{bmatrix},
\end{aligned} \tag{10}$$

$$\mathrm{cov}(c_i, c_j') = \sigma_f^2 \exp\left(-\frac{\|c_i - c_j'\|^2}{2l^2}\right), \tag{11}$$

[1] In general the mean $\bar{r} = \frac{1}{N}\sum_{k=1}^{N} r_k$ is not equal to zero and should be subtracted from the training data. $\bar{r}$ should later be added to the estimated output values (9).

where σ_f is the signal variance and l the characteristic length-scale, i. e., roughly the distance that one has to move in the input space before the value of the output signal changes significantly. σ_n, σ_f, and l are called hyperparameters and should be estimated from the training data. This can be accomplished by maximizing the following log marginal likelihood

$$\log\left(p(\boldsymbol{r}|\boldsymbol{C},\sigma_l,\sigma_f,l)\right) = -\frac{1}{2}\boldsymbol{r}^T[\boldsymbol{\Sigma}(\boldsymbol{C},\boldsymbol{C})+\sigma_n^2\mathbf{I}]^{-1}\boldsymbol{r} - \frac{1}{2}\log\left(\det\left[\boldsymbol{\Sigma}(\boldsymbol{C},\boldsymbol{C})+\sigma_n^2\mathbf{I}\right]\right) - \frac{N}{2}\log 2\pi. \tag{12}$$

A variety of standard nonlinear optimization routines can be used to perform the needed optimization.

While calculating the needed hyperparameters $\{\sigma_n,\sigma_f,l\}$, done through minimizing (12), is one of the most computationally expensive part, calculation of the inverse matrix $[\boldsymbol{\Sigma}(\mathbf{C},\mathbf{C})+\sigma_n^2\mathbf{I}]^{-1}$ also demands a lot of computation time. However, both of these two calculations only depend on the training data and can be done offline. Once these calculations are done and the Gaussian Process has been *trained*, new coupling terms for arbitrary queries $\boldsymbol{c}^*$ can be calculated in real time by a few simple matrix multiplications.

V. EXPERIMENTAL EVALUATION

We tested the approach in a real-world scenario, applicable to customized assembly. In customized or small batch assembly, every product might be different, for example, special window-frame sizes might be required. In such a case, robotic assembly would require reprogramming for every instance. In our approach, we use compliant behavior of the robot to enable coaching through physical intervention, and in a few iterations achieve the desired behavior.

Autonomous adaptation is also possible through force feedback. We demonstrate that in simulation (for robot safety concerns), where we show the adaptation to the object-placing height. We also show how the learning epochs can be used for generalization.

A. Experimental Setup and the Task

Our experimental setup consists of a Kuka LWR-4 robot with a Bartett-hand 262 gripper, controlled from Matlab through Kuka's Fast Research Interface (FRI). The experimental setup is depicted in Fig. 1, which also clearly demonstrates the problem at hand.

The task of the robot was to pick up a window-frame and place it at the mounting position. One demonstration trajectory was acquired with kinesthetic guiding. For a new size of the window-frame, held by the robot, a new trajectory is required, otherwise the robot will not place it to the correct position. The right image of Fig. 1 shows how it should be placed.

B. Real-world Results

The original demonstration trajectory, appropriate for the window-frame of the height $c_1 = 75$ cm, was adapted through human intervention in three repetitions (epochs) for the size of $c_2 = 70$ cm, and in four epochs for the sizes

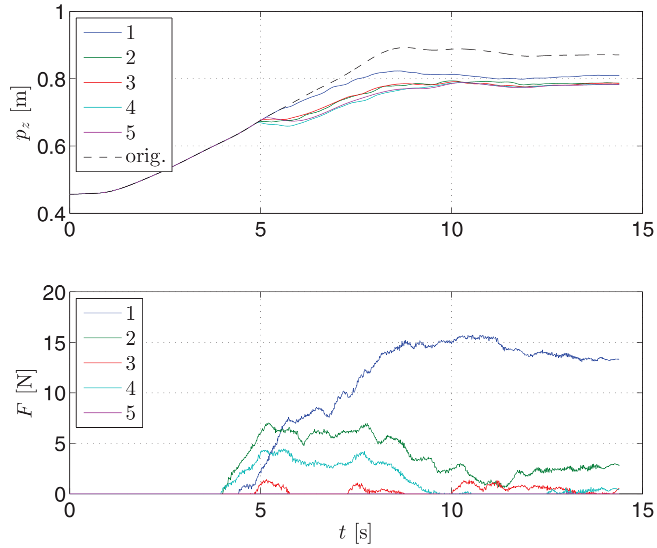

Fig. 3. Adaptation of the trajectory through physical human-robot interaction for $c = 55$ cm in the top plot. Note that the results depict the end-effector position. After the last epoch the robot settles so that the edge of the window-frame is pressed against the stand. All 5 epochs are shown, the fifth without any intervention. The bottom plot shows the measured forces for the same epochs. T

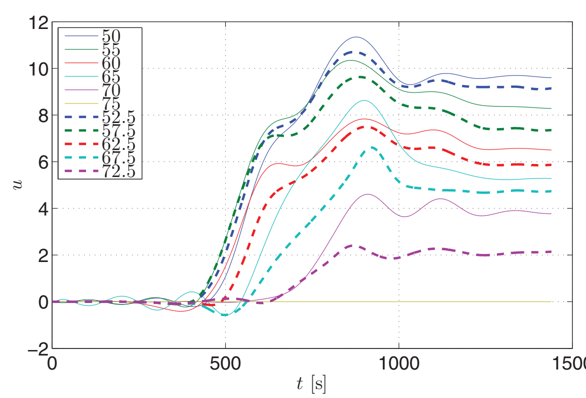

Fig. 4. Coupling terms of the database (solid thin lines) and the generalized coupling terms in dashed bold lines. The legend depicts the query point for separate lines.

$c_i = [65,60,55,50]$ cm, $i = 3,4,5,6$. The adaptation of the trajectory through physical human-robot interaction for the query point of 55 cm is presented in the top plot of Fig. 3, while the bottom plot depicts the measured forces of human-robot interaction in the p_z direction of the robot's coordinate system. These forces at the end effector were estimated from the robot's joint torque-sensors. The complete database consisted of 6 coupling terms (one being simply zero), each encoded in RBFs. The database of real-world coupling terms, acquired through physical human interaction is depicted in thin solid lines in Fig. 4. Using this database we calculated the hyperparameters for the GPR. We then used GPR for the window-frame sizes in-between the training data, i. e., for $\mathbf{c} = [52.5,57.5,62.5,67.5,72.5]$ cm. The generated, generalized coupling terms are depicted in bold dashed lines in Fig. 4.

As we can see from the results, the database did transition

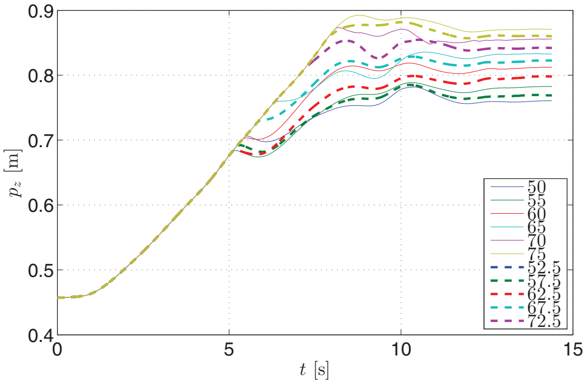

Fig. 5. Robot trajectories of the database depicted in thin solid lines. Generalized trajectories of motion in bold dashed lines. The legend depicts the query point for separate lines.

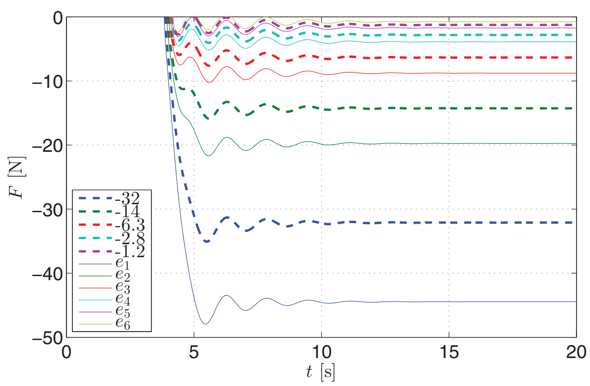

Fig. 6. Simulated force trajectories during autonomous adaptation, where the robot used the force of contact to the environment to adapt its trajectory. The adaptation epochs in thin solid lines, marked with $e_1, \ldots, e_6$ represent the learning database. The bold dashed lines show the forces that appear when trajectories were adapted with generalized coupling terms for the query points listed in the legend (in N).

smoothly, but the effect of human coaching, is clearly seen is some oscillations. These are a direct result of the non-constant human coaching. Never the less, the results clearly show that the trajectory was adapted to the final configuration, with the window frame touching the stand (see also the attached video). Furthermore, the experiment was conceived to show that a smaller window-frame can remain closer to the final placing position, so that the complete trajectory (after the point of interaction) is adapted, and not only the very end. This becomes specifically important, if – due to the nature of the construction process – the area around the work-piece is very limited. The trajectories of motion after human coaching and the generalized trajectories are depicted in Fig. 5.

Fig. 7 shows three series of still images. The first line shows one epoch of trajectory coaching through physical human-robot interaction. The second line shows the final positions of the database. The third line shows the final generalized positions. The acquisition of the base, the database and the generalized trajectories are all depicted in the accompanying video.

C. Generalization of Learning Examples

During the adaptation of motion we collect several trajectories and coupling terms, which can be associated to new query points. These coupling terms can be used for generalization to new desired forces. In the following we present a simulated use-case of adaptation with respect to the force of contact with an object.

The original trajectory of motion, going straight downwards, was adapted to reach a minimal force of contact with an object in its path, at an unknown height. An example of a real-world scenario would be placing an object on the table. Fig. 6 shows the measured forces during the adaptation in thin solid lines. Bold dashed lines show the coupling terms generalized to the forces in between the forces achieved during the training. As we can see from the results, the coupling term can be generalized to new positions despite the unequal spreading of the trajectories in this database.

The results depict oscillations, which are in this case the result of the modeling of the force with a simple linear spring, without any damping. The same control algorithm as for the real-world scenario was used. It should also be noted that when human intervention changes the trajectories, adaptations will be far less uniform and the generalization from the learning database might not produce accurate trajectories. Even so, these can be used as the initial approximation for further adaptation.

VI. DISCUSSION AND CONCLUSIONS

Acquiring a database for generalization is tedious and can be demanding. If the robot can do it by itself, a person only needs to teach it the basic trajectory. Even only adapting trajectories through physical interaction seems favorable to complete trajectories. It should be noted, though, that a too strong a push or a push in a wrong direction will change the trajectories for the worse. Wasting feasible trajectories seems a giant loss. That is why in our approach we show how learning trajectories can still be used. Even if the generalization result is not exactly was was opted for, it can still serve as the initial approximation in learning.

Building huge databases for every possible scenario, even if it is autonomous, might not strike one as the best possible approach. While data storage has been made extremely available with contemporary hardware, calculation times might eventually surpass the usefulness. In GPR, where the hyper-parameters are calculated off-line, this might not represent a major obstacle, as the calculations might be completed in the robot's downtime.

When comparing the proposed approach to simple changing of the goal of the DMP, it should be noted that in our case – specifically when modifying trajectories through physical contact, the user has control over the complete trajectory and can adapt any part of it, not just the end point. This seamless modification of trajectories is also what can be beneficially put to use in industrial environments, for example for customized industrial assembly. The cost of special personal to reprogram the robot is thus almost completely excluded.

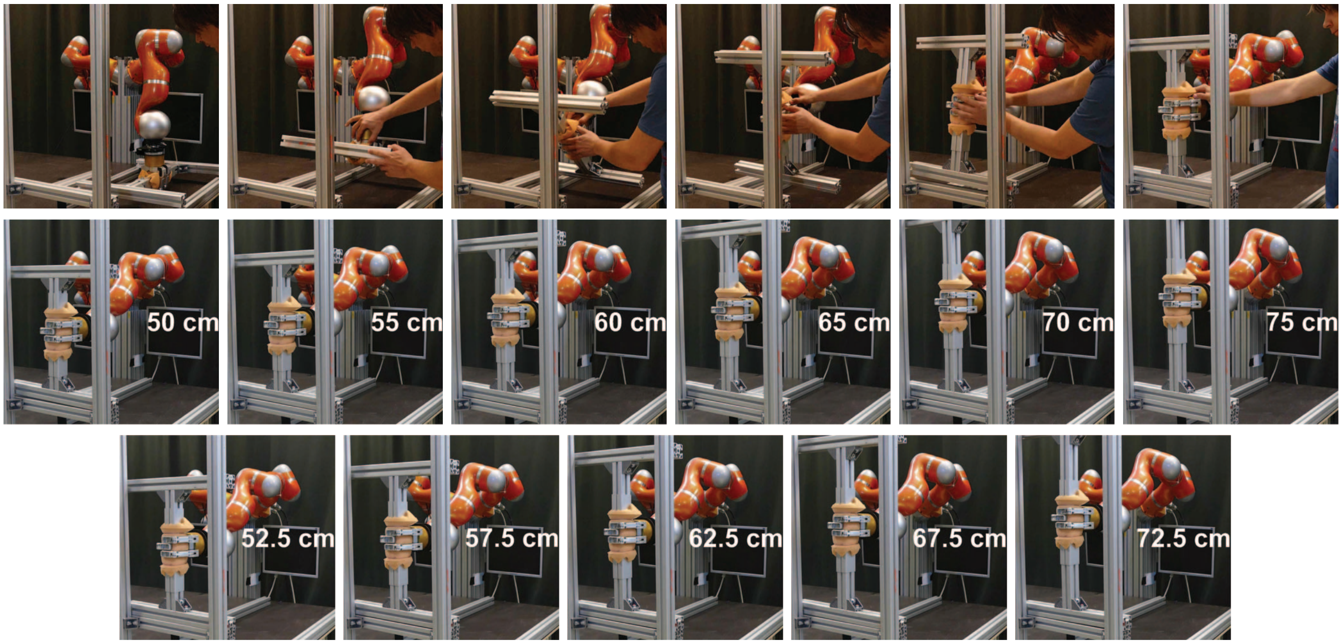

Fig. 7. Top row: image sequence of coaching a trajectory through physical human-robot interaction for $c = 55$ cm. Middle row: The final positions of the trained trajectories, i. e., the database. Bottom row: The final positions of the generalized trajectories, there the query was set between the database queries.

While we have shown real-world experiments with the robot, further research questions remain open. For one, the approach needs statistical evaluation and formalized approaches to parameter setting, despite the relatively low and manageable number. Detailed comparisons with other approaches are also on the list of future work. Despite this, the major novelty – generalization of coupling terms – has been shown effective and it opens the possibilities of autonomous database expansion through learning. While the approach was demonstrated for a singled DOF, extending it to multiple DOF is straightforward, since we can decouple the measured forces in separate directions.

REFERENCES

[1] R. Dillmann, "Teaching and learning of robot tasks via observation of human performance," *Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 109–116, 2004.

[2] A. Gams, J. van den Kieboom, F. Dzeladini, A. Ude, and A. J. Ijspeert, "Real-time full body motion imitation on the coman humanoid robot," *Robotica*, vol. 33, pp. 1049–1061, 6 2015.

[3] S. Calinon, D. Bruno, and D. G. Caldwell, "A task-parameterized probabilistic model with minimal intervention control," in *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)*, Hong Kong, China, May-June 2014, pp. 3339–3344.

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

[5] T. Kulvicius, M. Biehl, M. J. Aein, M. Tamosiunaite, and F. Wörgötter, "Interaction learning for dynamic movement primitives used in cooperative robotic tasks," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1450–1459, 2013.

[6] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, "Online movement adaptation based on previous sensor experiences," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, 2011, pp. 365–371.

[7] F. Abu-Dakka, B. Nemec, J. Jrgensen, T. Savarimuthu, N. Krger, and A. Ude, "Adaptation of manipulation skills in physical contact with the environment to reference force profiles," *Autonomous Robots*, pp. 1–19, 2015.

[8] A. Gams, B. Nemec, A. Ijspeert, and A. Ude, "Coupling movement primitives: Interaction with the environment and bimanual tasks," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 816–830, 2014.

[9] T. Matsubara, S.-H. Hyon, and J. Morimoto, "Learning parametric dynamic movement primitives from multiple demonstrations," *Neural Networks*, vol. 24, no. 5, pp. 493–500, 2011.

[10] 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.

[11] D. Forte, A. Gams, J. Morimoto, and A. Ude, "On-line motion synthesis and adaptation using a trajectory database," *Robotics and Autonomous Systems*, vol. 60, no. 10, pp. 1327 – 1339, 2012.

[12] R. Krug and D. Dimitrov, "Model predictive motion control based on generalized dynamical movement primitives," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 1, pp. 17–35, 2015.

[13] F. Stulp, G. Raiola, A. Hoarau, S. Ivaldi, and O. Sigaud, "Learning compact parameterized skills with a single regression," in *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Oct 2013, pp. 417–422.

[14] M. Parlaktuna, D. Tunaoglu, E. Sahin, and E. Ugur, "Closed-loop primitives: A method to generate and recognize reaching actions from demonstration," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, May 2012, pp. 2015–2020.

[15] S. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, Oct 2011.

[16] T. Petrič, A. Gams, L. Žlajpah, A. Ude, and J. Morimoto, "Online approach for altering robot behaviors based on human in the loop coaching gestures," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[17] J. Kober, E. Oztop, and J. Peters, "Reinforcement learning to adjust robot movements to new situations," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

[18] B. da Silva, G. Konidaris, and A. Barto, "Learning parameterized skills," in *International Conference on Machine Learning (ICML)*, 2012, pp. 1679 – 1686.

[19] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[20] M. Norrlöf and S. Gunnarsson, "Time and frequency domain convergence properties in iterative learning control," *International Journal of Control*, vol. 75, no. 14, pp. 1114–1126, 2002.

[21] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*. Cambridge, MA, USA: MIT Press, 2006.