SEVENTH FRAMEWORK PROGRAMME

Project Acronym:	Xperience
Project Type:	IP
Project Title:	Robots Bootstrapped through Learning from Experience
Contract Number:	270273
Starting Date:	01-01-2011
Ending Date:	31-12-2015

XPERIENCE.ORG

Deliverable Number:	D2.2.1
Deliverable Title:	Motor actions with focus on structuring of action graphs
Type (Internal, Restricted, Public):	PU
Authors:	Aleš Ude, Tamim Asfour, Minija Tamosiunaite, Florentin Wörgötter, Andrej Gams, Miha Deniša and Rüdiger Dillmann
Contributing Partners:	JSI, KIT, UGOE

Contractual Date of Delivery to the EC: 31-01-2013
Actual Date of Delivery to the EC: 31-01-2013

Contents

Chapter 1

Summary

1.1 General Objective of WP2.2: Motor Actions

From the proposal: WP2.2 is primarily concerned with how to learn and obtain complex action sequences and how to organize and structure the acquired data to 1) generate advanced motor behaviours, e.g. by blending and sequencing of behaviours in the available data sets, and 2) interact with higher-level cognitive processes that mainly use discrete representations, thus providing the bridge for planning to access representations at the sensorimotor level and vice versa.

The proposed methods like imitation learning, reinforcement learning and other exploratory approaches, which have been shown to be successful at the acquisition of motor knowledge, will be considered for implementation.

1.2 Summary of the Results

Deliverable D2.2.1 describes our results on structuring of human (or robot) trajectories in an action database. It also deals with representation, generation and adaptation of movement primitives. There are five papers and manuscripts in preparation attached to it.

- The focus of the first paper is on how to search a hierarchical database of example movements to discover new action sequences and generalize them [DU13], [2]. New action sequences are discovered based on higher-level goals specified by a user. New movements are encoded and generalized using dynamic movement primitives [5, 13].
- In the second paper we address the problem of generating cooperative movements [YRA13], which is based on simultaneously storing the trajectories of two cooperating agents in a database. At execution time the movements of a robot are generated by first recognizing the motion of a human who collaborates with a robot. The successful recognition of a human motion invokes the corresponding motion from the database, which is executed by the robot.
- While it is possible to directly use raw trajectory data stored in the database of example movements for the generation of the desired robot movements, it is often preferable to compute a more compact representation of the generated movements for on-line execution with the robot and also for learning. In the Xperience project we use dynamic systems to generate compact motor representations for learning and execution. In this deliverable we present our work on combining discrete and periodic movements in a single dynamic system [ERD+12].
- We also analyse tightly coupled dual agent systems where agents learn to cooperate and systems where an agent comes into contact with the environment. In this context we address the problem of adaptation of robot movements for robot-environment interaction and for bimanual tasks [GNv+13]. The convergence of the proposed adaptation process has been proven in the context of iterative learning control paradigm [1]. In dual agent systems each agent has its own path plan defined by a DMP. In a coupled system the agents have to equilibrate with respect to each other. We have shown analytically that in a linear-spring-coupled DMP-based system agents equilibrate into

a shared fixed point representing the two new trajectories [KBA+13]. By means of simulation we show that learning can be employed to create a system, where both agents in the end "help each other". In a real robot example we show how a robot equilibrates to cooperate with a human.

All tasks described in WP2.2 are (partially) addressed in this deliverable. The works of Section 2.1 and 2.2 deal with the structuring of motor knowledge in action graphs (**Task 2.2.2**). The works of Section 2.3, 2.4, and 2.5 address the problem of motor representations and motor learning with dynamic systems (**Task 2.2.1**). The work described in Section 2.4 uses explorative learning to generate interactive behaviors (**Task 2.2.3**). Finally, in Section 2.2 the same representation is used for the classification of human actions and for the generation of robot movements, thus addressing **Task 2.2.4**.

Chapter 2

Description of Results

2.1 Discovering New Motor Primitives in a Database of Robot Movements

We developed a new approach for discovering motor primitives in a hierarchical database of example trajectories. The example trajectories are obtained either by kinesthetic guiding as shown in Fig. 2.1 or by human demonstration. The acquired trajectories are clustered and organized in a binary tree-like structure, from which transition graphs at different levels of granularity are constructed. By searching the transition graphs and exploiting the interdependencies between the movements encoded in the graphs and the hierarchical structure of the database, parts of new movements can be discovered. From these partial paths complete new movements are constructed using optimized interpolation. The jerk of transitions is minimized in a nonlinear optimization process. This way new sets of movements can be generated, thereby reducing the amount of example trajectories that need to be demonstrated by a human teacher. By combining the results of the search and optimized interpolation with statistical generalization techniques, a complete representation of new, not directly demonstrated movement primitives can be computed.

The evaluation of the approach described above showed that it is possible to discover new movement primitives in a database of example movements. Unlike many other approaches in imitation learning, we assume that the training database consists of different types of movements. We combine them by exploiting the interconnections between them. The implemented search process generates new movements that are not part of the training database. Unlike previous approaches that utilized transition graphs in robotics, we do not assume a direct connection between the desired start and end point in the transition

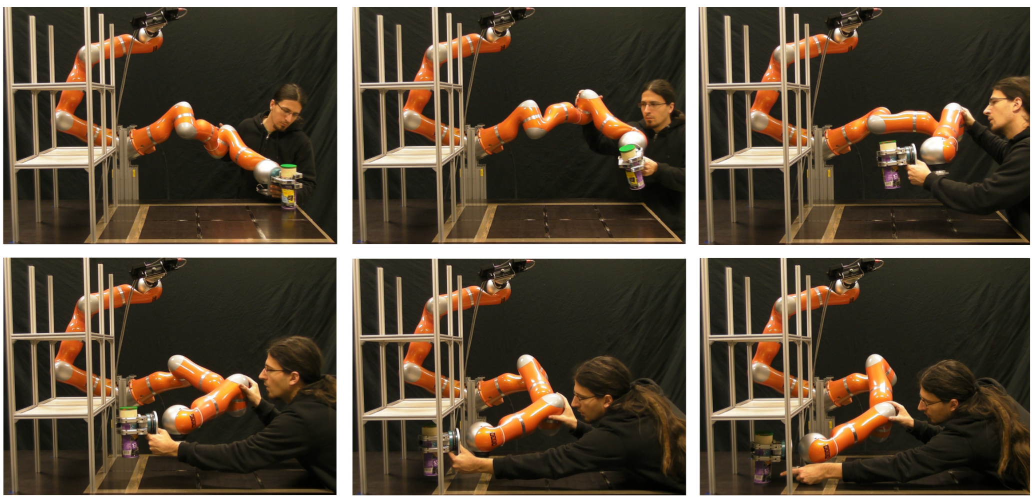

Figure 2.1: Our experimental setup consists of two Kuka LWR arms and a stereo camera. The sequence of images shows the acquisition of example movements by kinesthetic guiding, which is possible because both arms are compliant.

graph. If the connection does not exist at the specified level, we take advantage of a hierarchical database representation and look for parts of the desired path at higher levels of granularity. We combine these parts through optimized interpolation that minimizes the jerk. As a result we obtain smooth, human-like transitions in the newly generated trajectories. In our experiment, we constructed six new series of trajectories from a database consisting of six smaller series. This way we expanded 30 demonstrated reaching movements to 180 new reaching movements, each of them retaining the shape of movement and precision needed for the task. These new movements were then used as input for statistical generalization, which allowed us to synthesize movements from any starting object position. With the proposed approach we reduced the burden of demonstrating many trajectories while preserving the needed precision, shape and smoothness of movement.

More details can be found in the attached paper [DU13].

2.2 Planning Object Receiving Motions with Human Motion Database

We developed a method for planning motions of a humanoid robot that receives an object from a human, with focus on a natural object passing scenario where the human initiates the passing motion by moving an object towards the robot, which continuously adapts its motion to the observed human motion in real time. In this scenario, the robot not only has to recognize and adapt to the human action but also has to plan its motion quickly so that the human does not have to wait holding an object. We solve these issues by using a human motion database [14] obtained from two persons performing the object passing task. The rationale behind this approach is that human performance of such a simple task is repeatable, and therefore the receiver (robot) motion can be planned by looking up the passer motion in a database. We demonstrate in simulation that the robot can start extending the arm at an appropriate timing and take hand configurations suitable for the object being passed. We also perform hardware experiments of object handing from a human to a robot.

2.3 New Formulation for Encoding of Periodic Movements and their Transients

Present formulations of periodic dynamic movement primitives (DMPs) do not encode the transient behavior required to start the rhythmic motion, although these transient movements are an important part of the rhythmic movements (i.e. when walking, there is always a first step that is very different from the subsequent ones). An ad-hoc procedure is then necessary to get the robot into the periodic motion. In this contribution we present a novel representation for rhythmic Dynamic Movement Primitives (DMPs) that encodes both the rhythmic motion and its transient behaviors. As with previously proposed DMPs, we use a dynamical system approach where an asymptotically stable limit cycle represents the periodic pattern. Transients are then represented as trajectories converging towards the limit cycle, different trajectories representing varying transients from different initial conditions. Our approach thus constitutes a generalization of previously proposed rhythmic DMPs. Experiments conducted on the humanoid robot ARMAR-III demonstrate the applicability of the approach for movement generation.

2.4 Modulation of Motor Primitives for Bimanual Tasks and Interaction with the Environment

Movements that take place in contact with the environment or with another agent cannot be learned in advance from human demonstration or kinesthetic guidance. Instead of exploring a database of available movements, the previously learned movements need to be adapted on-line to account for contacts affecting the motion as they occur during the task execution. The framework of dynamic movement primitives [5, 4] allows the generation of discrete and periodic trajectories, which can be modulated in various aspects. In the attached paper [GNv+13] we propose and evaluate a modulation approach which extends the framework to allow interaction with objects and the environment. It is a variant of iterative learning

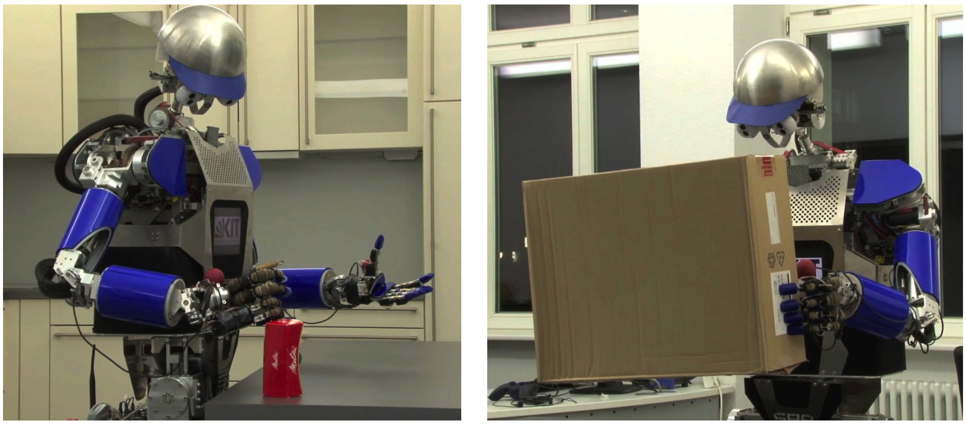

Figure 2.2: Adaptation of colliding movement (left) and synchronization of the left arm movement with the right arm movement (right).

control method, which is based on the notion that a performance of a system that executes the same task multiple times can be improved by learning from previous executions [1]. The developed algorithm enables the coupling of independently executed robotic trajectories and thus simplifies the performance of bimanual and cooperative tasks. In a few iterations the algorithm learns the necessary coupling term to modify the trajectory in accordance to the desired position or external force. In the attached paper we show that both the coupling and the learning algorithms are numerically stable.

One of the advantages of the proposed algorithm is that it builds an internal environment model. Once the appropriate behavior has been learned, it can act also without sensors, e. g. without a force-torque sensor. On the other hand, sensory feedback assures that it will gradually adapt to new situations as they occur and improve the performance of the task after a few task executions. The strengths of the algorithm, which fits in the scope of the iterative learning control theory, are shown in bimanual and two-agent obstacle avoidance tasks, where no higher-level cognitive reasoning or planning is required. Two KUKA LWR arms and the ARMAR-III humanoid robot (see Fig. 2.2) were used to verify the effectiveness of the approach.

2.5 Tightly-Coupled Agent Interaction Learning

Novel trajectory generation methods such as Dynamic Movement Primitives (DMPs, [5]) or Gaussian Mixture Models (GMMs, [6, 7]) can generalize over different start and end points of the movement trajectory and they can efficiently emulate different trajectory shapes also allowing to combine them in a dynamic way [9, 8]. Such methods also allow an on-line alteration of the trajectory, if need be. For example, it is clearly useful to alter the trajectory of an agent as soon as an obstacle (a path disturbance) is sensed. Such problems have been addressed by using sensory feedback and applied in a variety of different applications. So far DMPs and GMMs have mainly been used for uncoupled agent systems. In this study, we analyse tightly coupled dual agent systems where each agent has its own path plan defined by a DMP. In a coupled system the problem exists that both agents might not cooperate. This leads to the situations that agents will first have to equilibrate with respect to each other. Only on top of this any sensor influence – for example for obstacle avoidance – and/or learning can take place. As shown here analytically both agents will indeed equilibrate into a shared fixed point representing the two new trajectories. This leads to the situation that sensor reactions and learning can operate in a stable way also in the dual agent system. Specifically, we show that learning can be employed to create a system, where both agents in the end "help each other". Probably one interesting aspect of this approach is that, due to the intrinsic attractor properties of DMPs, these systems do not need any conventional active control-components (impedance control, servoing, etc.), while still performing remarkably well.

To describe the movement trajectory of an agent we use the method for generating movement sequences proposed in [8] (for more details see [KBA+13]) which is a modification of the original dynamic movement primitives (DMPs, [5]). Here we use modified DMPs since they have faster convergence at the end-point compared to the original DMP formulation and allow smooth joining of movement sequences with non-zero velocities at the joining point [8]. We model the two agent system as two point particles coupled by

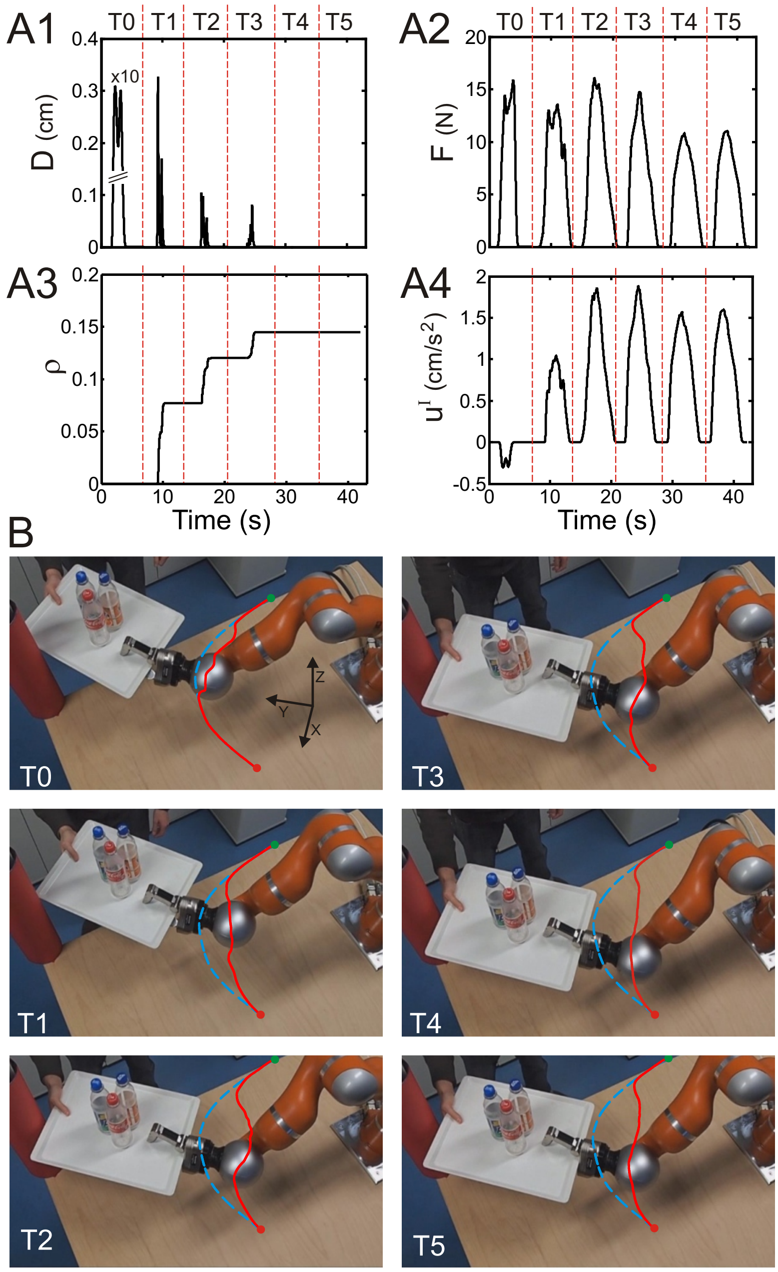

Figure 2.3: Results from interaction learning obtained with a real robot. **A)** Signal development from six trials (separated by dashed lines). Trial T0 is a control case - path-persistence behavior (no learning). **A1)** Displacement signal D; **A2)** predictive force signal F; **A3)** weight ρ; **A4)** output signal u^I. Learning rate was $\mu = 0.04$. **B**) Trajectories for control case (T0), learning process (T1-T3) and post-learning (T4, T5). Dashed and solid lines represent planned and actual paths, respectively.

a spring. Here we treat agents with equal mass. Each agent is subject to a primary force generated by a dynamic movement primitive, which can be viewed as the control signal. We denote the i-th coordinate (i=1,2,3 correspond to X, Y and Z-coordinates, respectively) of the j-th particle (j=1,2 correspond to agent P and Q, respectively) as $y_{i,j}$ and the corresponding velocities as $z_{i,j}$. Assuming the particles have mass m Newton's equation of motion is

$$m\dot{z}_{i,j} = F^S_{i,j} + F^D_{i,j}, \tag{2.1}$$

where $F^S_{i,j}$ are the forces acting due to the spring coupling and $F^D_{i,j}$ the forces from the DMP. The spring forces can be written as

$$F_{i,1}^{S}=-k\left(y_{i,1}-\left(y_{i,2}+d\frac{y_{i,1}-y_{i,2}}{\sqrt{\sum_{l}(y_{l,1}-y_{l,2})^{2}}}\right)\right)=-F_{i,2}^{S}. \tag{2.2}$$

Here d denotes the spring length when relaxed and k is the spring constant. As explained above, the position of the agent is defined by a DMP and we denote its force $F_{i,j}^{D}$ by

$$F_{i,j}^{D}=m\left(\alpha\left(\beta\left(r_{i,j}-y_{i,j}\right)-z_{i,j}\right)+f_{i,j}+u^{A}+u^{I}\right), \tag{2.3}$$

where u^A and u^I additional terms for obstacle avoidance and interaction learning, respectively. For more details please see [KBA$^+$13]. Also we have for the accelerations and velocities:

$$\dot{z}_{i,j}=\frac{1}{m}(F_{i,j}^{S}+F_{i,j}^{D}), \tag{2.4}$$

$$\dot{y}_{i,j}=z_{i,j}. \tag{2.5}$$

Obstacle avoidance is implemented in the conventional way (potential field approach, [10, 3, 11]) and is used to create realistic situations for interaction learning. In the interaction learning scenario, as explained above, we have two agents which are physically coupled via the linear spring. Initially the agents are going to follow their planned path, so in case the agents have different paths or the path gets changed due to obstacle avoidance forces between agents will increase due to the coupling. The goal is to learn to interact in a way that the forces between agents are minimized. For example, if agent P is going to avoid the obstacle, then agent Q has to learn interacting and helping agent P by moving to the same direction. For learning we make use of the physical fact that the position signal follows the acceleration-dependent signal. Hence force (acceleration-dependent) is predictive for a displacement (position-dependent) that will arise later. We can use the displacement signal to learn a predictive reaction in response to the (earlier occurring) force signal. Thus, for learning the sensor signal D (displacement) is paired with the sensor signal F (force) [12]. Learning stops as soon as $D = 0$, i.e., as soon as the displacement sensor is not triggered any more and the predictive response has fully taken over (see [KBA$^+$13] for more details).

We present results from interaction learning with obstacle avoidance obtained on a KUKA light-weight robot arm, where we let a human and a robot interact carrying a tray with bottles. The goal is to avoid the red bar (left) not hitting it with the tray when moving along a curved trajectory (see Fig. 2.3 B, T0). The robot has to learn to move in the same direction by reacting to the force sensor and help the human avoiding the obstacle. Signals and resulting trajectories are shown in Fig. 2.3. Note that here the learning was applied only for Y-coordinate. In this case learning stopped and weights stabilized after three learning trials (T1-T3, see also supplementary video). Similar experiments can be performed with the KUKA arm also in more dimensions but the behavior does not changes in any significant way.

In summary, in this study we stressed the importance of combining sensory information with dynamic movement primitives and learning in a dual tightly-coupled agent system where the behavior and cooperation of agents is purely based on low level sensory information without any advanced planning. We believe that the here arising attractive properties, like fast adaptation, mutual equilibration, and cooperative interaction, can be very helpful for designing reactive, DMP-based motor control for cooperative tasks. It should also be easier to introduce planning as well as other (higher) cognitive traits into such systems as their sensory-reactions and low-level learning makes them already “well-behaved” from the beginning.

References

[1] D. A. Bristow, M. Tharayil, and A. G. Alleyne. A survey of iterative learning control: A learning-based method for high-performance tracking control. *IEEE Control Systems Magazine*, 26(3):96–114, 2006.

[2] M. Deniša and A. Ude. New motor primitives through graph search, interpolation and generalization. In S. Lee, K.-J. Yoo, and J. Lee, editors, *Frontiers of Intelligent Autonomous Systems*, pages 137–148. Springer, 2013.

[3] 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*, pages 1534–1539, Kobe, Japan, 2009.

[4] A. J. Ijspeert, J. Nakanishi, and S. Schaal. Learning rhythmic movements by demonstration using nonlinear oscillators. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 958–963, Lausanne, Switzerland, 2002.

[5] A. J. Ijspeert, J. Nakanishi, and S. Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *IEEE International Conference on Robotics and Automation*, pages 1398–1403, Washington, DC, 2002.

[6] S. M. Khansari-Zadeh and A. Billard. BM: An iterative method to learn stable non-linear dynamical systems with gaussian mixture models. In *IEEE International Conference on Robotics and Automation*, pages 2381–2388, Anchorage, Alaska, 2010.

[7] S. M. Khansari-Zadeh and A. Billard. Learning Stable Non-Linear Dynamical Systems with Gaussian Mixture Models. *IEEE Transactions on Robotics*, 27(5):943–957, 2011.

[8] T. Kulvicius, K. J. Ning, M. Tamosiunaite, and F. Wörgötter. Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting. *IEEE Transactions on Robotics*, 28(1):145–157, 2012.

[9] B. Nemec and A. Ude. Action sequencing using dynamic movement primitives. *Robotica*, 30(5):837–846, 2012.

[10] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal. Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields. In *8th IEEE-RAS International Conference on Humanoid Robots*, pages 91–98, Daejeon, Korea, 2008.

[11] A. S. Phung, J. Malzahn, F. Hoffmann, and T. Bertram. Get out of the way - obstacle avoidance and learning by demonstration for manipulation. In *11th IFAC World Congress*, pages 11514–11519, 2011.

[12] B. Porr and F. Wörgötter. Isotropic sequence order learning in a closed loop behavioural system. *Roy. Soc. Phil. Trans. Mathematical, Physical & Engineering Sciences*, 361:2225–2244, 2003.

[13] A. Ude, A. Gams, T. Asfour, and J. Morimoto. Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, 26(5):800–815, 2010.

[14] K. Yamane, Y. Yamaguchi, and Y. Nakamura. Human motion database with a binary tree and node transition graphs. *Autonomous Robots*, 30:87–98, 2011.

Attached Papers

[DU13] Miha Deniša and Aleš Ude. Synthesis of new dynamic movement primitives through search in a hierarchical database of example movements. Submitted to *Journal of Intelligent and Robotic Systems*, 2013.

[ERD+12] Johannes Ernesti, Ludovic Righetti, Martin Do, Tamim Asfour, and Stefan Schaal. Encoding of periodic and their transient motions by a single dynamic movement primitive. In *IEEE-RAS International Conference on Humanoid Robots*, pages 57–64, Osaka, Japan, 2012.

[GNv+13] Andrej Gams, Bojan Nemec, Leon Žlajpah, Auke Ijspeert, Tamim Asfour, and Aleš Ude. Modulation of motor primitives for bimanual tasks and interaction with the environment. In *IEEE/RSJ International Conference on Intelligent Systems and Robots (submitted)*, Tokyo, Japan, 2013.

[KBA+13] 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*, under revision, 2013.

[YRA13] Katsu Yamane, Marcel Revfi, and Tamim Asfour. Planning object receiving motions of humanoid robots with human motion database. In *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013.