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
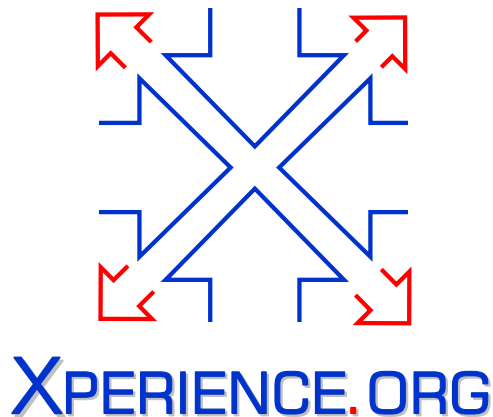
Deliverable Number:	D4.1.2
Deliverable Title:	Report or scientific publication on *Cooperative tasks* with focus on multi-agent active visual perception and scene interpretation (intention recognition and action prediction)
Type (Internal, Restricted, Public):	PU
Authors:	Tamim Asfour, Aleš Ude, Andrej Gams, Florentin Wörgötter, Mark Steedman and Rüdiger Dillmann
Contributing Partners:	KIT, JSI, UGOE, UEDIN

Contents

Chapter 1

Summary

1.1 Objective of WP4.1: Cooperative Tasks

The objective of WP4.1 as stated in the Description of Work is to study tightly coupled physical interaction (excluding explicit language) for multiple agents to accomplish a cooperative task. The goal is to study different types of cooperation, in which multiple robots are engaged in task execution to fulfill the goal. We distinguish between:

- Tightly coupled cooperative manipulation tasks, in which a direct physical interaction between multiple agents (two arms, two robots, human-robot) must take place to achieve the task goal. Examples are cooperative table/box pushing, cooperative lifting, cooperative tool use, etc.

- Loosely coupled cooperative tasks, in which multiple robots are engaged in scene interpretation and reasoning about the role of each agent involved in the execution of the task.

The main issues that will be addressed are: a) How do agents participate in a tightly coupled cooperation synchronize their actions? b) How can agents maximize the amount of mutual information for scene interpretation, c) How do agents recognize tasks that require cooperation, and d) How will the interpretation of the felt, seen and heard lead to the recognition of intention of "other" and finally to the recognition of plans.

1.2 Summary of the Results

In the first two years we developed several techniques to support both tightly and loosely coupled cooperative tasks (see deliverable **D4.1.1** and **D2.2.1**). In the following we summarize the previous contributions, which were reported in **D4.1.1** and **D2.2.1**:

- In [1], we showed how a library of movement primitives can be used for tightly-coupled interaction, where the initial motor knowledge was acquired by coaching. The key here is to enable real-time generation of movement primitives that take into account the external perturbations caused by contact with another agent.

- We demonstrated motion prediction in tightly coupled physical human-robot interaction tasks in the context of cooperative carrying of big objects. The forces resulting from the interaction between the object and the robot were used to infer the direction of human motion (leader) and mapped to commands of the robot (see deliverable **D4.1.1**).

- In [4], we demonstrated how cooperation between two independent agents (in this work each of the humanoid robot's arms is assumed to be an independent agent) can improve explorative learning of object representations.

- In [5] we addressed the problem of generating cooperative movements, 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.

D4.1.2 describes our results on improving the cooperation between a human operator and the robot. The contributions deals with both motor representations for cooperative tasks and cooperative perception to support coupled manipulation tasks. In the following we summarize the contributions:

1. The paper [WKK$^+$13] deals with autonomous learning and grounding of spatial representations. The proposed representation is grounded by exploration of the continuous state space on the sensorimotor level and by making use of common sense knowledge extracted from natural language text for discrete symbolic entities on the planning and reasoning level. The experimental evaluation on humanoid robot ARMAR-III showed that the proposed approach can reduce the need for human intervention in interactive scenarios and can thereby bootstrap human-robot interaction.

2. The next paper [WSAD13] deals with cooperative perception to support bimanual manipulation tasks. A novel saliency measure for gaze selection mechanism during bimanual manipulation tasks has been proposed to account for accuracy requirements of bimanual manipulation task in the saliency calculation. A video demonstrating the proposed approach was inclufed in deliverable **D5.3.2**.

3. In [GNIU14] we extended our previous work on synchronizing cooperative behaviors stored as DMPs [2] to show that the developed representation is suitable for the implementation of tightly coupled human-robot cooperative behaviors. This work is a substantial extension of the work which we initially reported in deliverable **D2.2.1**.

4. The paper [KBA$^+$13] investigates correlation-based learning for combining sensory information and learning with DMPs for dual-agent systems in order to solve cooperative tasks. The main reason which motivates the proposed correlation-based learning is that the agents can adapt their interaction to new situations, which might occur due to environmental changes or changes in agents behaviour.

1.3   Links to other Workpackages

WP4.1 builds on theoretical work in WP2.2 (motor representations) and contributes to the integration and demonstration work package WP5, in particular to WP5.3, as well as to WP4.2 regarding loosely coupled interaction.

Chapter 2

Description of Results

2.1 Grounding Spatial Symbols for Robot Task Planning Based on Experience

We developed methods for autonomous learning and grounding of spatial representations by exploiting two sources of experiences in order to ground spatial symbols for robot task planning [WKK$^+$13]. The proposed representation is based on the concept of Object-Action-Complexes (OACs) described in [3]. It consists of exploration of the continuous state space on the sensorimotor level and exploiting common sense knowledge extracted from natural language text for discrete symbolic entities on the planning and reasoning level. We demonstrate how spatial knowledge can be extracted from these two sources of experience: experience gathered through exploration by the robot and experience extracted from text. Robot exploration yields grounded sensory level representations that automatically encode the constraints of the robot such as the visibility of objects. Common sense spatial information is, on the other hand, extracted from large text corpora and thus provides knowledge on the symbolic level. Combining both
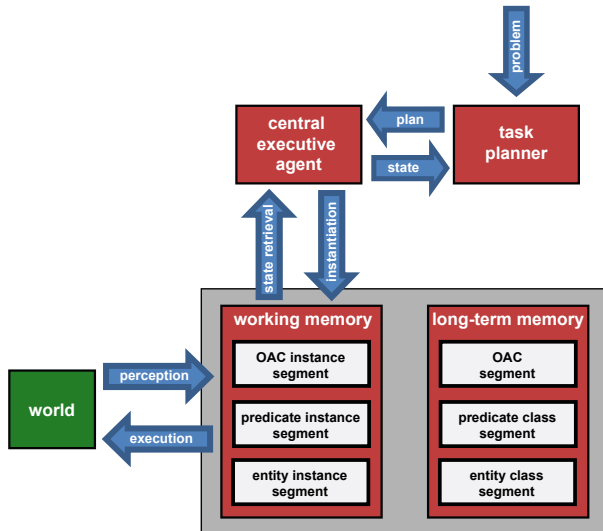

Figure 2.1: The integration between the task planning level and the sensorimotor level is established by the central executive agent (CEA). For each plan element the CEA instantiates the appropriate OAC and associated predicates and entities. Entities correspond to constants on the planning level. The goal of this work consists in learning an entity of type `location` from experience.
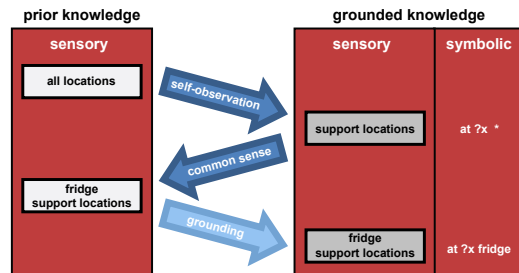
Figure 2.2: The proposed approach for acquiring grounded spatial symbols combines two sources of experience: Self-observation of the robot during the execution of kitchen tasks yields grounded representations of support locations. These support locations are associated with semantic symbols by exploiting common sense knowledge.

sources of information in a systematic way yields a solution to the grounding problem which has the potential to substantially decrease the necessary learning effort.

Although we addressed a rather specific problem of symbol grounding for robot task planning in this work, we do not consider this problem isolated but within the context of coupling high level task planning with the humanoid robot's sensorimotor knowledge. For this purpose, we developed and implemented an architecture that links these sources of information (see Figure 2.1).

The complete chain of acquiring a grounded representation of location *fridge* from experience is illustrated in Figure 2.2. The only prior knowledge at the beginning of the process consists of a common reference frame for spatial locations, which defines the space of all possible locations. By self-observation, we infer support locations in space, i. e. subspaces that frequently serve as locations for objects. Common sense reasoning then allows us to find the most likely locations for the encountered objects. This information is then used for grounding by deriving the most likely label for the regarded location.

More details can be found in the attached paper [WKK$^+$13].

2.2   Gaze Selection in Bimanual Manipulation Tasks

To support the effective collection and interpretation of visual scene information when multiple agents are performing joint actions, we developed mechanism for gaze selection during bimanual manipulation tasks [WSAD13]. Perceiving and interpreting a complete scene involves taking several snapshots from different viewing angles. To acquire such multiple views, active gaze redirections need to be performed to enable the perception and processing of relevant world entities. In this way, diverse entities in the world compete for the restricted perceptual resources (e. g. camera systems), thus necessitating an appropriate gaze selection strategy.

To this end, three different subproblems are addressed: the representation of the acquired visual input, the calculation of saliency based on this representation, and the selection of the most suitable gaze direction. As representation of the visual input, a probabilistic environmental model, which takes into account the
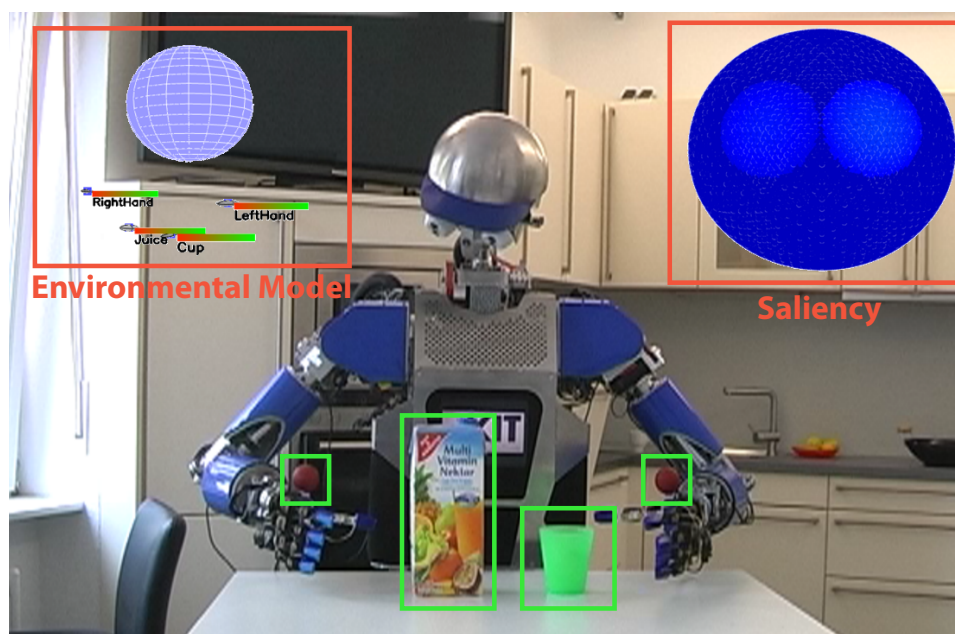

Figure 2.3: Bimanual manipulation necessitates the application of active gaze redirections in order to increase the observable area of the robot cameras. In the course of a bimanual grasping task, overall four objects need to be observed: both robot hands and both objects. In order to solve this gaze selection problem during manipulation we proposed a resource-aware mechanism, which takes into account accuracy requirements to select the next gaze direction. Based on the remaining uncertainties and the required accuracy of position estimates, the gaze selection mechanism determines gaze directions that guarantee successful task execution.

dynamic nature of manipulation tasks, has been proposed. At the core of the gaze selection mechanism, a novel saliency measure is proposed that includes accuracy requirements from the manipulation task in the saliency calculation. Finally, an iterative procedure based on spherical graphs is developed in order to decide the best gaze direction.

The developed mechanism tackles the three key problems: 1) The establishment of an environmental model covering the world's entities based on current and past sensory experience, 2) the assignment of a saliency measure that considera constraints determined by the manipulation task, and 3) the development of gaze selection mechanism that computes the optimal viewing angle. The feasibility of the approach has been experimentally evaluated in the context of bimanual manipulation tasks on humanoid robot ARMAR-III. The details are described in [WSAD13].

A video showing the complete processing chain including 1) uncertain object recognition and localization, 2) fusion in the environmental model, and 3) selection of the most suitable gaze direction based on the task acuity was attached to deliverable **D5.3.2**.

2.3   Coupling Movement Primitives: Interaction with the Environment and Bimanual Tasks

The framework of dynamic movement primitives contains many favorable properties for the execution of robotic trajectories, such as indirect dependency on time, response to perturbations, and the ability to easily modulate the given trajectories. However, the originally proposed framework is constrained to the kinematic aspect of the movement. In [GNIU14] we bridge the gap to dynamic behaviors by extending the framework with force/torque feedback. We proposed and evaluated a modulation approach, which allows interaction with the environment, where the environment can be an object, a robot, or a person. By coupling the originally independent robot trajectories, the approach also enables the execution of bimanual and tightly coupled cooperative tasks (see Fig. 2.4). We applied an iterative learning control algorithm to learn a coupling term, which is applied to the original trajectory in a feed-forward fashion and thus modifies the trajectory in accordance to the desired positions or external forces. A stability analysis was performed and results of simulated and real-world experiments using two KUKA LWR arms for bimanual tasks and interaction with the environment were provided. By expanding on the framework of dynamic movement primitives, we keep all the favorable properties of DMPs, which has been demonstrated especially in a two-agent obstacle avoidance task.

This is a substantially extended report on research first described in [2], which was attached to deliverable **D2.2.1**. In the paper attached to this deliverable we additionally describe the theoretical underpinnings of the proposed approach and its suitability for human-robot collaboration when a physical contact has been established.
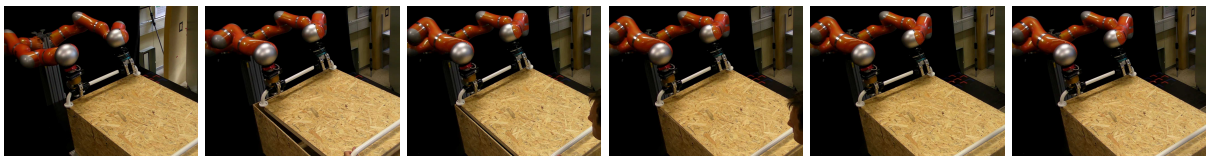

Figure 2.4: Cooperative closing of the lid. The pictures show positions of the lid after each epoch, with epoch number increasing towards the right.

2.4   Interaction Learning for Dynamic Movement Primitives Used in Cooperative Robotic Tasks

Since several years novel trajectory generation methods based on dynamic systems have been gaining importance for flexible movement control in robotics. In addition to generalization properties and robustness to perturbations, such methods also allow on-line alteration of the trajectory based on sensory feedback. For example, it is often useful to alter the trajectory of an agent as soon as an obstacle is perceived. So far dynamic systems have mainly been used for uncoupled agent systems. In [KBA+13] we

analyze tightly coupled dual agent systems where each agent has its own path plan defined by a dynamic movement primitive (DMP). In a coupled system we must ensure that both agents cooperate appropriately. This leads to situations where independent agents first have to synchronize their movements. The synchronization of movements is necessary before taking into account any sensory influences, for example to ensure obstacle avoidance, and before learning can take place. In this work we present a stability analysis of such coupled systems and evaluate learning in simulations. In the end, we also show two experiments with real robot-arms.

Results demonstrate that combining sensory information and learning with DMP can also be applied for dual-agent systems in order to solve cooperative tasks. The main reason which motivates the proposed correlation-based learning is that agents can adapt their interaction to new situations, which might occur due to environmental changes or changes in agents behaviour. Our approach creates an alternating (depending on the environment) leader/follower architecture. Thus, we do not have to explicitly define the agents' roles beforehand, but they synchronize them depending on their sensory information. Behaviour and cooperation of agents is purely based on low level sensory information without any advanced planning.

The proposed approach has many attractive properties including fast adaptation, mutual synchronization, and cooperative interaction, which can be very helpful for designing reactive, DMP-based motor control for cooperative tasks.

References

[1] D. Forte, A. Gams, J. Morimoto, and A. Ude. On-line motion synthesis and adaptation using a trajectory database. *Robotics and Autonomous Systems*, 60(10):740–757, 2012.

[2] A. Gams, B. Nemec, L. Žlajpah, A. Ijspeert, T. Asfour, and A. Ude. Modulation of motor primitives using force feedback: Interaction with the environment and bimanual tasks. In *IEEE/RSJ International Conference on Intelligent Systems and Robots*, pages 5629–5635, Tokyo, Japan, 2013.

[3] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, , T. Asfour, D. Kraft, D. Omrčen, A. Agostini, and R. Dillmann. Object-action complexes: Grounded abstractions of sensory-motor processes. *Robotics and Autonomous Systems*, 59(10):740–757, 2011.

[4] D. Schiebener, J. Morimoto, T. Asfour, and A. Ude. Integrating visual perception and manipulation for autonomous learning of object representations. *Adaptive Behavior*, 21(5):328–345, 2013.

[5] K. Yamane, M. Revfi, and T. Asfour. Planning object receiving motions of humanoid robots with human motion database. In *IEEE International Conference on Robotics and Automation*, pages 1629–1636, Karlsruhe, Germany, 2013.

Attached Papers

[GNIU14] Andrej Gams, Bojan Nemec, Auke Ijspeert, and Aleš Ude. Coupling movement primitives: Interaction with the environment and bimanual tasks. *IEEE Transactions on Robotics*, 30, 2014.

[KBA⁺13] Tomas Kulvicius, Martin Biehl, Mohamad Javad Aein, Minija Tamosiunaite, and Florentin Wörgötter. Interaction learning for dynamic movement primitives used in cooperative robotic tasks. *Robotics and Autonomous Systems*, 61(12):1450–1459, 2013.

[WKK⁺13] K. Welke, P. Kaiser, A. Kozlov, N. Adermann, T. Asfour, and M. Steedman. Grounded spatial symbols for task planning based on experience. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 484–491, Atlanta, Georgia, 2013.

[WSAD13] K. Welke, D. Schiebener, T. Asfour, and R. Dillmann. Gaze selection during manipulation tasks. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 652–659, Karlsruhe, Germany, May 2013.

Coupling Movement Primitives: Interaction with the Environment and Bimanual Tasks

Andrej Gams, Bojan Nemec, Auke J. Ijspeert and Aleš Ude

Abstract—The framework of dynamic movement primitives contains many favorable properties for the execution of robotic trajectories, such as indirect dependency on time, response to perturbations, and the ability to easily modulate the given trajectories, but the framework in its original form remains constrained to the kinematic aspect of the movement. In this paper we bridge the gap to dynamic behavior by extending the framework with force/torque feedback. We propose and evaluate a modulation approach that allows interaction with objects and the environment. Through the proposed coupling of originally independent robotic trajectories, the approach also enables the execution of bimanual and tightly coupled cooperative tasks. We apply an iterative learning control algorithm to learn a coupling term, which is applied to the original trajectory in a feed-forward fashion and thus modifies the trajectory in accordance to the desired positions or external forces. A stability analysis and results of simulated and real-world experiments using two KUKA LWR arms for bimanual tasks and interaction with the environment are presented. By expanding on the framework of dynamic movement primitives, we keep all the favorable properties, which is demonstrated with temporal modulation and in a two-agent obstacle avoidance task.

Index Terms—bimanual operation, cooperative task, interaction with environment, dynamic movement primitives.

I. INTRODUCTION

By moving beyond the structured environment of a manufacturing plant, robots are making their way into the everyday world that people inhabit – offices, hospitals, homes and other cluttered and uncontrolled environments [1], including the kitchen [2], [3]. A growing portion of robotics research already directly or indirectly deals with all aspects related to complex human environments [4]. If we envision a robotic assistant in a human environment, it will probably use its sensors and existing knowledge to generate trajectories appropriate for the given tasks, for example by generalization [5]. Despite the possibility to adapt trajectories as they are being executed [6], noise, lack of prior knowledge and errors in the perception of the environment might not make the trajectories accurate enough for the desired manipulatory actions. The generated trajectory therefore has to be adapted to the task through autonomous exploration or learning. In this paper we present a new approach to modulation and flexible learning of robot movements, which allow safe interaction with the environment and bimanual or cooperative multi-agent tasks.

Trajectory generation depends on the type of encoding approach and different encoding approaches also allow for different possibilities of modulation, interpolation, and categorization [7]. One of the approaches is the use of splines and wavelets [8], [9]. However, splines are nonautonomous representations with no attractor properties. While effective for imitation learning, they do not allow easy online modulation [10]. Rescaling the splines in space and time for generalization is possible, but it requires to explicitly recompute the spline nodes. Gaussian Mixture Regression [11] and Gaussian Mixture Models are another option. A mixture model approach was used in [12], [13] to estimate the entire attractor landscape of a movement skill from several sample trajectories. To ensure stability of the dynamical system toward an attractor point, a constraint optimization problem has to be solved in a nonconvex optimization landscape. Yet another option is the use of Hidden Markov Models [14].

We build on dynamic movement primitives (DMPs), first introduced by Ijspeert et al. [15], which model attractor behaviors of autonomous nonlinear dynamical systems with the help of statistical learning techniques. DMPs provide means to encode a trajectory as a set of differential equations that can compactly represent control policies, while their attractor landscapes can be adapted by only changing a few parameters. The latter can be exploited in several ways, for example for reinforcement learning [16]–[20], statistical generalization [5], [21], or for combining separate trajectories in a dynamic way [22], [23].

The structure of DMPs enables incorporation of sensory feedback. Modulations can affect either the transformation system or the canonical system, or both systems [10]. The use of sensory feedback was demonstrated in various applications, e. g., modulations affecting the canonical system were demonstrated on different periodic tasks [24]. Other examples include modulating the transformation system for on-line obstacle

Andrej Gams is with the Humanoid and Cognitive Robotics Lab, Dept. of Automatics, Biocybernetics and Robotics, Jožef Stean Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia and with the Biorobotics Laboratory, Institute of Bioengineering, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, email: `andrej.gams at ijs.si`

Bojan Nemec is with the Humanoid and Cognitive Robotics Lab, Dept. of Automatics, Biocybernetics and Robotics, Jožef Stean Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia, email: `bojan.nemec at ijs.si`

Auke J. Ijspeert is with the Biorobotics Laboratory, Institute of Bioengineering, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, email: `auke.ijspeert at epfl.ch`

Aleš Ude is with the Humanoid and Cognitive Robotics Lab, Dept. of Automatics, Biocybernetics and Robotics, Jožef Stean Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia, email: `ales.ude at ijs.si`

avoidance [10], [25], or for introducing an external limit [7]. An example of modulating both systems is the so-called slow-down feedback, which is used to stop the execution of the trajectory [26].

In this paper we propose a new approach to modify trajectories, i.e., the transformation system of the DMP. We propose recording the sensory feedback as the robot moves along a trajectory and then using this feedback to improve the robot's performance the next time it moves down the same trajectory. We do not modify the original trajectory, but learn a coupling term, which is fed into the original trajectory similarly to an external limit modulation [7]. The coupling term can either be the real force coupling between two manipulators/agents, or it can virtually represent an external force arising from interaction. The final shape and amplitude of the coupling term is learned in a few iterations using iterative learning control (ILC) [27]. The approach is fast and reliable for tasks which do not vary along the trials. Initial results on the approach were published in [28].

Several reasons speak for the use of iterative learning in the proposed approach. On the one hand, its appeal lies in the similarity to human learning processes, as people may practice a task many times before being able to find correct inputs to accomplish it with such a complex system as the human body [29]. In [27] the concept of ILC is well illustrated by the example of a basketball player shooting a free throw from a fixed position, who can improve the ability to score by practicing the shot repeatedly. During each shot, the basketball player observes the trajectory of the ball and consciously plans an alteration in the shooting motion for the next attempt. ILC can be applied in exactly the same manner to learning of robotic movements [30], [31].

On the other hand, ILC features several desired properties. Just as any learning system, ILC incorporates information rich error signals from previous operations for subsequent iterations. Furthermore, it only adapts the control input, and not the controller, does not require extensive training and is known to converge fast [27]. Because ILC generates its open-loop control through practice, it is also highly robust to system uncertainties [27] and can be used to achieve perfect tracking, even when the model is uncertain or unknown [32]. The novelty of the proposed approach lies in incorporating the well-defined DMP framework and the iterative learning control into a single, robust system for modification of trajectories based on force feedback, thus surpassing the kinematic domain of the DMPs.

After related work in Section II and the basic review of DMPs in Section III we present 1) force-based modulation of the DMPs at both velocity and acceleration levels; 2) coupling of DMPs for bimanual tasks (Section III); 3) learning the open coupling terms with iterative learning control (Section IV); 4) stability analysis and arguments for using both acceleration and velocity levels as compared to acceleration level only modulation (Section V). Section VI describes interaction and bimanual experiments conducted on two KUKA LWR robots in a bimanual setting, including experiments in cooperation with humans. Section VII shows that DMP properties, such as the modulation of the duration, remain intact when coupling

trajectories. Pros and cons of the approach are discussed in Section VIII and concluding remarks in Section IX.

II. RELATED WORK

Even though controlling rigid robots while in contact with the environment can be difficult, using a force feedback term to learn and improve task execution was considered in many robotic tasks, see for example [33]. The use of force feedback to change the output velocity of a manipulator was reported by Hogan [34]. On the other hand, relatively few papers discuss the use of force feedback in combination with dynamical systems or specifically DMPs.

Modifying periodic DMPs was previously demonstrated for a task of wiping a flat or curved surface [35]. Contrary to the proposed approach, complete trajectory waveforms were modified within a few periods of the task using regression methods. Formally, the approach in [35] did not rely on modulation but on learning of new trajectories, as the trajectories for the whole period of motion were constantly re-learned. The approach was expanded on by Ernesti et al. [36] to include transient motions.

Learning interaction force skills in presence of compliant external dynamics from human demonstrations using dynamical systems was shown in [37]. The authors used an interaction force encoded in terms of a parameterized time-invariant differential equation based upon the parallel force/position control law. Similarly to our proposed approach, it modulates the velocity term of its dynamical system. Applicability was shown only in virtual settings [37].

Pastor et al. [4], [38] have demonstrated an approach of modifying DMPS, which, similarly to the proposed method, relies on data from an execution of a discrete task to modify the trajectory in the next, perturbed execution. They implemented a low-level position and force control system that integrates with DMPs at the acceleration level, allowing for reactive and compliant behaviors. The key idea in their approach is that a successful demonstration provides a reference force for the following, possibly perturbed executions. A controller is used to ensure the same force profile. The approach was applied for grasping of a lamp [4] and a battery operated drill, combined with sequencing to achieve complete tasks [38].

DMPs were modulated for tightly coupled dual-agent tasks by Kulvicius et. al [39]. In their approach, the authors used virtual forces to couple DMPs at acceleration levels and applied Hebbian type learning to minimize the virtual force during the execution. Vision and touch sensors were used to determine the distance and virtual force between the two agents. An approach for bimanual operation based on dynamical systems and also applicable to DMPs was discussed by Calinon et al. [40]. The approach expresses the nonlinear force modulating the movement in the original DMP formulation as additional sets of virtual springs, adding local corrective terms that can swiftly react to perturbations during reproduction.

III. MODULATING DYNAMIC MOVEMENT PRIMITIVES

A. Dynamic Movement Primitives

DMPs have been thoroughly discussed in the literature [10], [15], [26]. Here we provide only basic information, which is

based on the formulation provided in [5], [26]. For a single degree of freedom (DOF) denoted by y, in our case one of the external task-space coordinates, a DMP is defined by the following system of nonlinear differential equations

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) + f(x), \qquad (1)$$
$$\tau \dot{y} = z. \qquad (2)$$

$f(x)$ is defined as a linear combination of nonlinear radial basis functions

$$f(x) = \frac{\sum_{i=1}^{N} w_i \Psi_i(x)}{\sum_{i=1}^{N} \Psi_i(x)} x, \qquad (3)$$
$$\Psi_i(x) = \exp\left(-h_i (x - c_i)^2\right), \qquad (4)$$

where c_i are the centers of radial basis functions distributed along the trajectory and $h_i > 0$ their widths. Provided that parameters α_z, β_z, $\tau > 0$ and $\alpha_z = 4\beta_z$, the linear part of the system $(1) - (2)$ is critically damped and has a unique attractor point at $y = g$, $z = 0$. A phase variable x is used in (1), (3) and (4). It is utilized to avoid direct dependency of f on time. Its dynamics is defined by

$$\tau \dot{x} = -\alpha_x x, \qquad (5)$$

with initial value $x(0) = 1$. α_x is a positive constant.

The weight vector $\mathbf{w}$, composed of weights w_i, defines the shape of the encoded trajectory. [15] and [5] describe the learning of the weight vector. Multiple DOFs are realized by maintaining separate sets of $(1) - (4)$, while a single canonical system given by (5) is used to synchronize them.

B. Modulation for Interaction with environment

DMPs can be modulated online to take dynamic events from the environment into account. Those online modulations are among the most important properties offered by the dynamical systems approach [10]. An example of spatial modulation is including an obstacle avoidance term in (1) [10], [25]

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) + f(x) + C_m, \qquad (6)$$

where C_m is the modulation term. In this paper we call this kind of modulation as a modulation at the *acceleration* level.

Another spatial modulation includes a simple repulsive force to avoid moving beyond a given position in the task space [7]. Such a repulsive force can be specified by modifying (2) into

$$\tau \dot{y} = z + h(y), \qquad (7)$$

while leaving (1) in the original form. In this paper we call this kind of modulation as modulation at the *velocity* level. A simple repulsive force to avoid hitting y_L can be defined as [7]

$$h(y) = -\frac{1}{\gamma(y_L - y)^3}, \qquad (8)$$

where y_L is the known limit. Modification of a DMP that encodes a straight trajectory from 1.3 m to 0.9 m in 5 seconds, using (7) and (8) and $\gamma = 10^5$, $y_L = 0.9$ m, results in the response as shown in red (dashed) in Fig. 1.

The external limit may be static, as shown in Fig. 1, or moving, determined for example by vision. In both cases,
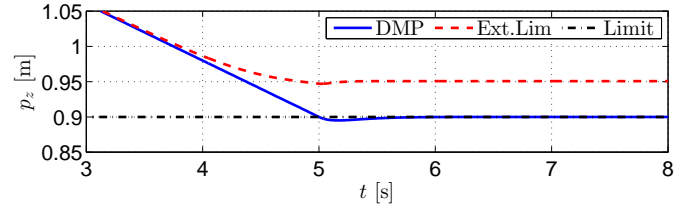

Fig. 1. Response in presence of an external limit according to (7) and (8), depicted in red (dashed), with the limit set at $y_L = 0.9$ m (black dash-dot). Original DMP trajectory in solid blue.

defining the external limit as in (7) and (8) can prevent the robot from getting into contact with objects in its environment because the repulsive force of the limit acts before actual contact takes place. We therefore propose a modification of the external limit approach, not by changing (7), but by defining a different repulsive force. Instead of using (8), we propose using the measured force F, which arises from the interaction with the environment

$$\tau \dot{y} = z + cF(t), \qquad (9)$$

where c is a scaling constant. $F(t)$ can either be the real measured force of contact or a virtual force. The virtual force can be defined as (for one DOF)

$$F(t) = kd(t), \qquad (10)$$

where k is the object (or environment) stiffness and d is the depth of penetration into the object.

A slight overshoot of forces upon environment contact appears when using the proposed velocity level modulation. To minimize this overshoot of forces (the error), we add a derivative of the measured force at the acceleration level. Similarly to PD controllers, this additional coupling introduces damping. The equation of a DMP with coupling at both the velocity and acceleration levels becomes

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) + f(x) + c_2\dot{C}, \qquad (11)$$
$$\tau \dot{y} = z + C, \qquad (12)$$
$$C = cF(t), \qquad (13)$$

with c_2 a scaling constant. Fig. 2 shows the difference of using velocity modulation, both velocity and acceleration with $c_2 = 30$, determined empirically, and only acceleration level modulation. When using velocity modulation only or acceleration modulation only, the force overshoots at time $t = 5$ s. This overshoot results in oscillations in the direction of the force. We show in Section V that adding the coupling term to both velocity and acceleration level is better than only to the acceleration level, because the latter results in significantly larger oscillations in the direction of the force. Such performance would impose restrictions on the use of the proposed iterative learning algorithm and would not allow effective learning.

Properly selected scaling factors c and c_2 ensure rapid and compliant behavior of the robot. Even so, the force F and therefore the modification of the trajectory only appears after
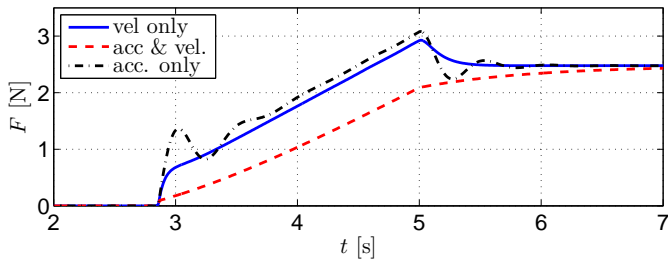
Fig. 2. Simulation results that show the difference of adding a coupling term at the velocity level (blue solid), also including the derivative of the force at the acceleration level (red dashed), or acceleration level only (black dash-dot). The red trajectory does not overshoot, while the blue and black do. Acceleration only modulation produces a more oscillatory response. As discussed in Section V, higher coupling gains are required for acceleration only modulation to achieve the same steady-state value; in this case 36 times. The trajectory was encoded to start at 1.4 m and end at 0.7 m in 5 s, while an object was encountered at 1 m.

the contact with the environment. We therefore propose to employ an ILC learning algorithm, which takes a few repetitions of the same task to learn the waveform and amplitude of what we call the *coupling term*. Using the coupling term, we can minimize the error for a desired force of contact and thus also mitigate the need for tuning the scaling factor. The learning algorithm is explained in detail in Section IV.

C. Cooperative DMPs

The force of contact with the environment can as well be the force of contact with another robot, and therefore used for bimanual or two-agent tasks. Here it is important to note that a robot with a centralized controller and accurate control for both arms (or for two agents) does not need such modifications. Well studied approaches for bimanual control exist, for example [41], [42]. However, given two independently controlled robots, possibly with conflicting trajectories, an approach for motion synchronization is needed. Examples of such are cooperation of two stand-alone robots/agents working together when carrying a large object. In our experiments we used two independently controlled robot arms for bimanual tasks.

Let us assume two separate trajectories given by two DMPs, executed by two robot arms. To keep the desired force between two agents, we introduce a coupling term. For one DOF (for clarity), this coupling term is defined as

$$F_{1,2} = F_d - (F_1 - F_2), \qquad (14)$$

where F_d is the desired coupling force and F_i is the force measured at the end-effector of the i-th agent. In simulation or if the desired distance between the two end-effectors is specified instead of the force, we introduce a virtual spring between the end-effectors of the arms that alters both trajectories. In this case the coupling term becomes

$$F_{1,2} = k(d_d - d_a), \qquad (15)$$

where d_d is the desired distance between the end-effectors and d_a is the actual difference, while k is the virtual spring constant. Measured force can be used instead of a virtual

spring. The force that acts on DMP_1 is opposite to the force acting on DMP_2

$$F_{2,1} = -F_{1,2} = -k(d_d - d_a). \qquad (16)$$

We introduce these forces, again scaled by c, into each DMP. Equations (17) – (22) define what we label *cooperative DMPs*:

$$\tau \dot{z}_1 = \alpha_z(\beta_z(g_1 - y_1) - z_1) + f_1(x) + c_2 \dot{C}_{1,2}, \quad (17)$$
$$\tau \dot{y}_1 = z_1 + C_{1,2}, \qquad (18)$$
$$C_{1,2} = c F_{1,2} \cdot l_{f1}, \qquad (19)$$
$$\tau \dot{z}_2 = \alpha_z(\beta_z(g_2 - y_2) - z_2) + f_2(x) + c_2 \dot{C}_{2,1}, \quad (20)$$
$$\tau \dot{y}_2 = z_2 + C_{2,1}, \qquad (21)$$
$$C_{2,1} = c F_{2,1} \cdot l_{f2}. \qquad (22)$$

The variable l_f defines the relation leader-follower. If $l_{f1} = l_{f2}$, then both robots will adapt their trajectories to follow average trajectories at the defined distance d_d between them (within tolerance and after learning, discussed in the next section). On the other hand, if $l_{f1} = 0$ and $l_{f2} = 1$, only DMP_2 will change the trajectory to match the trajectory of DMP_1, again at the distance d_d and again only after learning. Vice-versa applies as well. Leader-follower relation can be determined by a higher level planner, which is beyond the scope of this paper. It depends on the needs and circumstances of a specific task.

IV. LEARNING USING PREVIOUS SENSORY INFORMATION

To ensure the desired force of contact with the environment, or the desired displacement between two robots, we need to learn the terms $C_{1,2}$ and $C_{2,1}$ in such a way that $F_d = F$, where F is either the real force or defined as in (10) or (15). In the following we propose an ILC algorithm to learn $C_{1,2}$ and $C_{2,1}$. See a thorough review by Bristow et al. [27] for details on ILC. The proposed algorithm avoids the necessity to accurately model the dynamics of the robot and the environment.

In a tightly coupled bimanual task both arms are physically connected through an object and we can assume $C_{1,2} = -C_{2,1}$. Thus we need to learn only one of the two terms. In the following we denote this term by C. Upon the execution of the given task for the first time, the sensors register the resulting force. If the task was to be executed again without any difference, the sensory readings would not change, except for the noise. Therefore we propose that the second time the task is executed, the sensor measurements from the first attempt are fed into the trajectory generation in a feed-forward manner. The learning update for the coupling terms is then defined as suggested by the ILC theory [27]

$$C_i = c e_i + F_{c,i}, \qquad (23)$$
$$F_{c,i} = Q(F_{c,i-1} + L c \dot{e}_{i-1}), \qquad (24)$$
$$e_i = F_d - F_i, \qquad (25)$$

where index i denotes the i-th epoch, c is the force gain, e_i is the coupling force error calculated from the difference of the desired coupling force F_d and the measured coupling force $F_i = F_{1,i} - F_{2,i}$, $F_{c,i}$ is the learned coupling force term,

and Q and L are positive scalars. The coupling term given by (23) is known as current iteration learning control, since it incorporates instantaneous feedback in the first term and learning update in the second term. The tunable parameters are Q, L and c. In all our experiments we use $Q = 0.99$, $L = 1$ and $c = 0.5$. In the learning and subsequent execution of the learned movement we use the coupling term C_i instead of C for interaction with the environment. Similarly we use C_i instead of $cF_{1,2}$ in (19) and $-C_i$ instead of $cF_{2,1}$ in (22). Note that if the desired force $F_d = 0$, (23) takes the form $C_i = -cF + F_{c,i}$, which matches (9) in the first iteration ($i = 0$), when the learned coupling force is $F_c = 0$.

While the force depends on the execution of the trajectory and thus time, there is no need to encode the learned coupling force F_c as a vector of time-stamps and values. Just like $f(x)$, we represent F_c as a linear combination of radial basis functions along phase x.

$$F_c(x) = \frac{\sum_{j=1}^{M} a_j \Psi_j(x)}{\sum_{j=1}^{M} \Psi_j(x)} x, \qquad (26)$$

To calculate the weight parameters after the i-th epoch we use

$$\mathbf{f} = \begin{bmatrix} F_{c,i}(x_0) \\ \dots \\ F_{c,i}(x_T) \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_0 \\ \dots \\ a_M \end{bmatrix}, \qquad (27)$$

where $x_j = x(t_j)$ and t_j denotes the j-th time sample. Writing

$$\mathbf{X} = \begin{bmatrix} \frac{\Psi_1(x_0)}{\sum_{j=1}^{M} \Psi_j(x_0)} x_0 & \cdots & \frac{\Psi_M(x_1)}{\sum_{j=1}^{M} \Psi_j(x_0)} x_0 \\ \dots & \dots & \dots \\ \frac{\Psi_1(x_T)}{\sum_{j=1}^{M} \Psi_j(x_T)} x_T & \cdots & \frac{\Psi_M(x_T)}{\sum_{j=1}^{M} \Psi_j(x_T)} x_T \end{bmatrix}, \qquad (28)$$

we need to solve the following set of linear equations:

$$\mathbf{Xa} = \mathbf{f}. \qquad (29)$$

The parameters $\mathbf{a}$ are calculated in a least-squares sense. Several advantages speak in favor of encoding the coupling term in this manner. For example, the nonlinear encoding acts as a filter [7] and thus cancels out the sensor noise. The main advantage is that such coupling term depends on the same canonical system as the trajectories.

Note that separate canonical systems and therefore phases can be used for the predefined motion of the robot, given by y, and the coupling force F_c. At the end of the predefined motion, the phase x reaches practically 0 and only the linear part of the DMP remains active. In order for F_c not to go to 0 at the same time, a separate phase variable has to be kept while learning and later applying F_c. Since we are dealing with discrete, finite motions, eventually both run out.

V. STABILITY

A. Stability of coupled DMPs

Even though single DMPs are stable [10], the stability of coupled DMPs, given by (17) – (22), cannot be guaranteed without further analysis. Cooperative two robot/agent DMPs change the system from single-input-single-output (SISO) into a multiple-input-single-output (MISO) system, since their outputs are subtracted. Fig. 3 shows the resulting MISO system
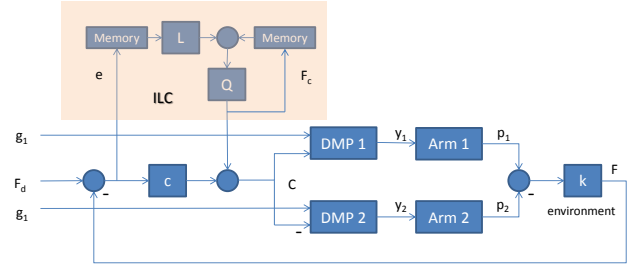

Fig. 3. MISO structure of the cooperative DMP system. Note that the coupling comes from the force F, which is (in simulation) defined as the scaled distance between the robots/agents, i.e., $F_d = kd_d$. The shaded region marks the ILC.

structure. The coupling comes from the force, which depends on the positions of the two robots as given in (15), where the actual distance is $d_a = p_1 - p_2$, with p_1 and p_2 being the positions of the two arms. In our theoretical analysis we assume that the robot tracks the desired trajectory perfectly, i.e., $p_1 = y_1$, $p_2 = y_2$, thus $d_a = y_1 - y_2$. For the given, stable DMP parameters, the gain c of the coupling term determines the behavior of the MISO system. Using the virtual spring formulation (15), we can derive the state-space system (30) – (31) from (17) – (22) with the applied feedback $C_{1,2} = -C_{2,1} = k(d_d - d_a)$, $l_{f,1} = l_{f,2} = 1$, (see Appendix)

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \qquad (30)$$
$$y(t) = \mathbf{C}\mathbf{x}(t). \qquad (31)$$

The system matrices for the controllable canonical form are given by

$$\mathbf{A} = \begin{bmatrix} -\frac{\alpha_z \tau + 2ck(c_2 + \tau)}{\tau^2} & 1 \\ -\frac{\alpha_z \beta_z \tau + 2ck}{\tau^2} & 0 \end{bmatrix}, \qquad (32)$$

$$\mathbf{B} = \begin{bmatrix} \frac{2k(c_2+\tau)}{\tau^2} & \frac{2ck(c_2+\tau)}{\tau^2} & 0 \\ \frac{2k\alpha_z}{\tau^2} & \frac{2ck\alpha_z}{\tau^2} & \frac{k}{\tau^2} \end{bmatrix}, \qquad (33)$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}. \qquad (34)$$

The input vector $\mathbf{u}$ and the scalar output y in (32) are given as $\mathbf{u} = [F_c, F_d, \alpha_z \beta_z (g_1 - g_2) + f_1(x) - f_2(x)]^T$ and $y = F$, respectively (see Fig. 3). The state vector is defined as

$$\mathbf{x} = \begin{bmatrix} F \\ \dot{F} - \frac{\alpha_z \tau + 2ck(c_2 + \tau)}{\tau^2} F \end{bmatrix}. \qquad (35)$$

Since the nonlinear parts $f_1(x)$ and $f_2(x)$ in (17) and (20) are bounded and tend to zero as the phase tends to zero, it is sufficient to prove the stability and attractor properties of the linear part of system (17) – (22). We assume environment stiffness as defined by (10).

The eigenvalues of $\mathbf{A}$ determine the stability and convergence of the linear part of differential equation system (30).
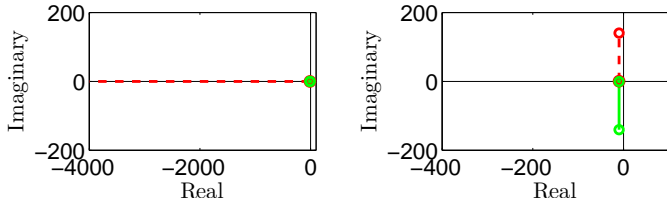
Fig. 4. Root locus plot of the coupled DMP structure with modulation at acceleration level (right) and both the velocity & acceleration levels (left). We varied the gain ck from 0 to 10000 while the gain c_2 was fixed at 1. The system has two poles denoted with red (dashed) and green. The full circles denote roots at $ck = 0$ and the empty circles at $ck = 10000$.

These eigenvalues are given as

$$\lambda_{1,2} = \frac{1}{2}\left(-\frac{\alpha_z\tau + 2ck(c_2+\tau)}{\tau^2}\pm \right. \tag{36}$$
$$\left. \sqrt{\left(\frac{\alpha_z\tau + 2ck(c_2+\tau)}{\tau^2}\right)^2 - 4\frac{\alpha_z\beta_z\tau + 2ck}{\tau^2}}\right).$$

Since all parameters $\alpha_z, \beta_z, c, k, c_2, \tau$ are positive, the eigenvalues $\lambda_{1,2}$ are negative for all cases in which both eigenvalues are real numbers. It can also happen that the eigenvalues are complex numbers, but in such cases the real part of both eigenvalues is again negative. This means that system (30) converges to a unique attractor point for all positive parameter values. We obtain complex eigenvalues only for some rather unlikely values, e. g., for very large values of τ.

We compared the performance of the proposed velocity-acceleration scheme with a scheme that uses only acceleration feedback in (11), but no feedback in (12). Root-locus in Fig. 4 shows that both schemes remain stable with increasing gain ck, but also clearly shows the main difference between them. The imaginary part of conjugate-complex eigenvalues increases only in the case when acceleration level modulation is used, whereas it remains close to zero when modulating both velocity and acceleration levels. The results clearly support the proposed velocity & acceleration level scheme, where the response is always damped, whereas the convergence is slower for modulation at acceleration level only.

B. Stability and convergence of the learning algorithm

Besides the stability and convergence to the attractor point of the coupled DMP structure, we also need to prove the stability and convergence of the proposed learning algorithm, as defined by (23) – (25). In the following we rely on the stability analysis provided by the ILC framework. A general form of ILC is defined as [27]

$$F_{c,i+1}(j) = Q(F_{c,i}(j) + Le_{i+1}(j+1)) \tag{37}$$

where $F_{c,i}(j)$ is the control input of the i-th epoch, $e_i(j) = F_d - F_i(j)$, Q is the filtering of the last control input, L is the learning parameter and j denotes the time sample. Equation (24) is exactly the same as (37) with the exception that this is a discrete time implementation, where the error derivative was replaced by the sample shift $j + 1$. The aim of the stability analysis is to find the range of parameters of the learning parameter L where the ILC remains stable. For that, state space

matrices given by (30) and (31) have to be in a discrete time form [43]. Note that from the ILC perspective, the input is the learned coupling force F_c. Therefore, a suitable discrete time representation of our system is

$$\mathbf{x}(j + 1) = \hat{\mathbf{A}}\mathbf{x}(j) + \hat{\mathbf{B}}_1 F_c(j) + \hat{\mathbf{B}}_2 F_d + \hat{\mathbf{B}}_3 \Phi(j) \tag{38}$$
$$y(j) = \hat{\mathbf{C}}\mathbf{x}(j), \tag{39}$$

where input matrices $\hat{\mathbf{B}}_1, \hat{\mathbf{B}}_2, \hat{\mathbf{B}}_3$ are formed from the first, second and the third column of the input matrix $\mathbf{B}$, respectively (see (33)). Hat symbol $(\hat{\cdot})$ denotes the discrete time counterpart of the continuous time system matrices. Input signal $\Phi(j)$ is defined as $\alpha_z\beta_z(g_1 - g_2) + f_1(x(j)) - f_2(x(j))$. A commonly accepted framework to examine the stability of a discrete time plant controlled with ILC in time-domain is a lifted or supervector representation of the system dynamics [27], [44]. The supervector representation of the discrete time system results in $T + 1$ dimensional input and output vectors and $(T + 1) \times (T + 1)$ system matrix

$$\begin{bmatrix} F_i(0) \\ F_i(1) \\ F_i(2) \\ \vdots \\ F_i(T) \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{B}}_1 & 0 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{A}}\hat{\mathbf{B}}_1 & \hat{\mathbf{C}}\hat{\mathbf{B}}_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^{T-1}\hat{\mathbf{B}}_1 & \hat{\mathbf{C}}\hat{\mathbf{A}}^{T-2}\hat{\mathbf{B}}_1 & \cdots & 0 \end{bmatrix}$$
$$\begin{bmatrix} F_{c,i}(0) \\ F_{c,i}(1) \\ F_{c,i}(2) \\ \vdots \\ F_{c,i}(T) \end{bmatrix} + \begin{bmatrix} r(0) \\ r(1) \\ r(2) \\ \vdots \\ r(T) \end{bmatrix}, \tag{40}$$

$$\begin{bmatrix} r(0) \\ r(1) \\ r(2) \\ \vdots \\ r(T) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{C}} \\ \hat{\mathbf{C}}\hat{\mathbf{A}} \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^2 \\ \vdots \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^T \end{bmatrix} \begin{bmatrix} F(0) \\ 0 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{B}}_2 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{A}}\hat{\mathbf{B}}_2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^{T-1}\hat{\mathbf{B}}_2 & \cdots & 0 \end{bmatrix} \begin{bmatrix} F_d \\ F_d \\ F_d \\ \vdots \\ F_d \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{B}}_3 & \cdots & 0 \\ \hat{\mathbf{C}}\hat{\mathbf{A}}\hat{\mathbf{B}}_3 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^{T-1}\hat{\mathbf{B}}_3 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \Phi(x(0)) \\ \Phi(x(1)) \\ \Phi(x(2)) \\ \vdots \\ \Phi(x(T)) \end{bmatrix},$$

where the contribution of initial condition $F(0)$ and inputs $F_d, \Phi(x(t))$ is treated as an exogenous signal $\mathbf{r}$ and T is the number of time samples. Note that inputs $\Phi(x(t))$ are bounded because the nonlinear part of DMP ($f_1(x)$ and $f_2(x)$) is given as linear combination of radial basis functions, which are bounded. The system matrix in (40), which we denote by $\mathbf{P}$, is a lower triangular Toeplitz matrix, where the coefficients are Markov parameters [27]. With the supervector notation of

the ILC matrices $\mathbf{Q}$ and $\mathbf{L}$ and by inserting (40) into (37), we obtain

$$\mathbf{F}_{c,i+1}(j) = \mathbf{Q}(\mathbf{I} - \mathbf{LP})\mathbf{F}_{c,i}(j) + \mathbf{QL}(\mathbf{F}_d - \mathbf{r}), \quad (41)$$

where i denotes the iteration index of the learning controller, $\mathbf{I}$ is a diagonal matrix of dimension $(T+1) \times (T+1)$ and $\mathbf{F}_d = [F_d, \ldots, F_d]^{\mathrm{T}}$. The ILC system is asymptotically stable if and only if [27]

$$\rho(\mathbf{Q}(\mathbf{I} - \mathbf{LP})) < 1, \quad (42)$$

where ρ denotes the maximum absolute value of the matrix eigenvalue. If the ILC system is asymptotically stable, the asymptotic error when T tends to infinity is

$$\mathbf{e}_\infty = [\mathbf{I} - \mathbf{P}[\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{LP})]^{-1}\mathbf{QL}](\mathbf{F}_d - \mathbf{r}). \quad (43)$$

For $\mathbf{Q} = \mathbf{I}$, the error $e_i(j)$ will converge to 0. Applying ILC controller with $\mathbf{Q} = 0.99\mathbf{I}$ and $\mathbf{L} = L\mathbf{I}$ to the coupled DMP system shows that the stability can be guaranteed for the learning controller gains within the range $L = [0, 2.09]$, where $T \leq 500$ and the gain ck was set to 100. Given these settings and $L = 1$, $g_1 - g_2 = 5$, $f_1(x) - f_2(x) = 0$, $F_d = 0$, the force F will converge to the desired force F_d with maximal error norm $0.062813N$.

In the ILC stability analysis we assumed that given equal control signals, the plant always returns the same outputs, which can not be always guaranteed in the case of changing environmental dynamics. A key question is therefore whether or not the proposed scheme remains asymptotically stable to plant perturbations. In [27] and [44] it was shown that the ILC is inherently stable to the plant dynamics variation. If we want to further increase the robustness to plant perturbations, the most direct way is to decrease the $\mathbf{Q}$ filter gain. On the other hand, decreasing $\mathbf{Q}$ increases steady state learning error. As a consequence, the selection of $\mathbf{Q}$ is a tradeoff between performance and robustness. Robustness was experimentally evaluated and the results are presented in the next Section.

VI. EVALUATION

A. Experimental Setup

We performed several simulated and real world experiments. The real world experiments were performed on two KUKA LWR arms with 7 DOFs each. Both arms are shown in Fig. 16. Trajectory calculation was performed on a client computer using Matlab/Simulink. The desired task space coordinates were sent to a server computer at 200 Hz via UDP. The server computer, running an xPC Target application at 2 kHz sent these commands to the KUKA controller, utilizing KUKA Fast Research Interface (FRI). It also sent the measured actual robot positions and forces back to the client PC.

B. Contact with the Environment

Contact with the environment is crucial for many robotic tasks. It needs to be safe for both the robot and the environment, which consequently means that the forces should be kept low. We applied the proposed algorithm to produce a desired force of contact $F = 15$ N upon impact with a table

(see Fig. 6). The movement was repeated 10 times. Fig. 5 shows the results of the real-world experiment. The top plot shows the trajectories of all epochs. The original trajectory was defined to start at 1.0 m and end at 0.6 m in 5 s. The table was at just under 0.67 m. The trajectory in the 1-st epoch is in green and the trajectory after learning in red. While the position trajectories practically overlap, the initial and final force trajectories are considerably different. Note that here it is crucial that the DMP was modulated with the measured force already in the first epoch, otherwise the resulting forces would be far greater and could damage the robot. The bottom plot shows the forces, with the force of the first epoch in green and the final force after 10 epochs in red. The reference force is set to appear after the impact is detected. Note that this is later anticipated by the ILC and the trajectory is altered before the actual contact.

C. Bimanual Tasks

We applied the proposed approach to couple two trajectories in simulation. For the left robot, the original trajectory was equal to $p_{x,L} = 0.75$ m, $p_{y,L} = 0.4 + 0.2\sin(t\pi/2)$ m, and $p_{z,L} = (0.7 + 0.12t)$ m. For the right robot, the trajectory was defined as $p_{x,R} = 0.75$ m, $p_{y,R} = 0.4$ m, and $p_{z,R} = (0.7 + 0.14t)$ m. The desired distance between the robots was set to $d_d = 0.8$ m and the virtual coupling spring constant to 20 N/m. The top plot in Fig. 7 shows the distance between the robots along the trajectories for each of the 10 learning epochs. It is evident that the error is considerably reduced after very few epochs; epochs 1 and 2 are marked. The final distance along the trajectory between the robots, marked with red, has a maximal error of less than 0.003 m, appearing at the very start of the motion.

The bottom plot shows the Root-Mean-Square (RMS) error of the distance for the cases of ideal trajectories (no noise) and for the case when noise of distance estimation was added to show the robustness. The noise was added on the position
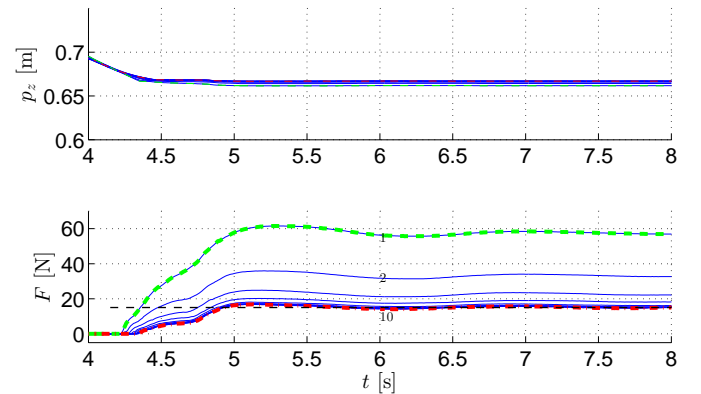

Fig. 5. Real world results of adaptation to environment. Top plot: trajectories of motion, with the trajectory of the 1-st epoch in green (dashed) and the final, 10-th epoch in red (dash-dot). Bottom plot: measured forces with the force of the 1-st epoch in green (dashed) and the last, 10-th in red (dash-dot). The black dashed line shows the desired force, set to appear after the impact is detected,
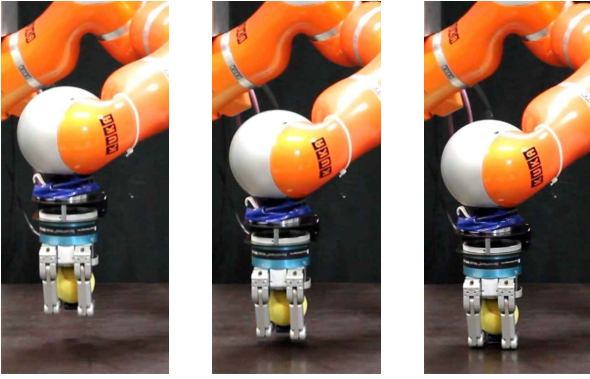
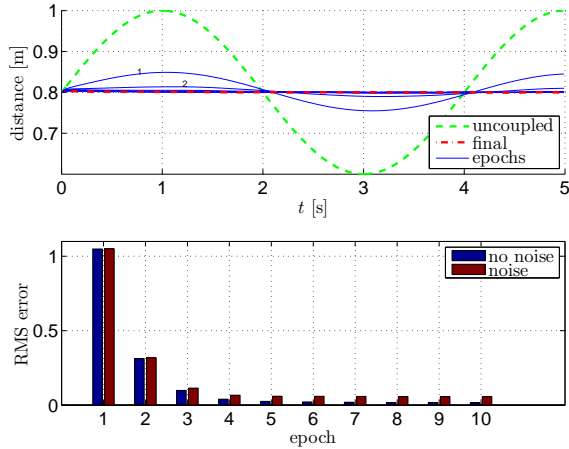Fig. 6. Image sequence showing the collision of the robot with the table.

Fig. 7. Simulated results of cooperative DMPs. The top plot shows the distance between the robots, where the green (dashed) line marks the distance during the trajectory of uncoupled robots, the red (dash-dot) line the distance after 10 epochs and the blue lines the distance in each epoch; epochs 1 and 2 are marked with numbers. The bottom plot shows the RMS error after each epoch for trajectories with and without noise.
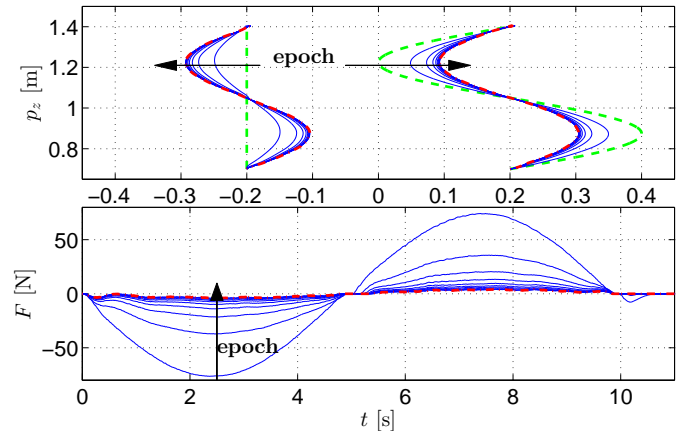

Fig. 8. Real world experimental results for cooperative DMPs. Note that the $p_{y,L}$ and $p_{y,R}$ trajectories are for presentation purposes depicted at ± 0.2 m, although the trajectories were performed at ± 0.4 m. The top $p_y - p_z$ plot shows the trajectories, with the green dashed lines showing the original trajectories and the red dash-dot lines showing the final trajectories. The bottom plot shows the resulting force, the final in red dash-dot.

of the arms, with the maximal noise amplitude at 0.01 m. We can see that the approach is hardly affected by the noise.

We performed a similar experiment on the real robots, which we tightly coupled by both of them rigidly holding a stick. The motions were the same as for the virtual experiment with the difference in the p_z direction, defined the same for both robots at $p_{z,L} = p_{z,R} = (0.7 + 0.07t)$ m. The duration of motion was set to 10 s. A full sinusoidal wave was performed by the left robot with $p_{y,L} = 0.4 + 0.2\sin(t\pi/5)$ m. The task was to modify the trajectories so that the force along the stick on the robots will be minimal. 7 learning epochs were conducted. Fig. 8 shows the results. The top plot shows the trajectories in a $p_y - p_z$ plot. The green dashed lines show the original uncoupled trajectories and the red dash-dot lines the final trajectories. The bottom plot shows the resulting measured force. Both robots adapted, i.e., $l_{f1} = l_{f2} = 1$. The force scaling factor was set empirically.

The two image sequences shown in Fig. 9 compare the execution of independent and cooperative DMPs, where cooperative DMPs were learned in 7 epochs. The top row shows the execution of the original, independent trajectories, where

we can see that the distance between the robot end effectors is changing, which can be observed from the length of the stick on the right side of the right robot. The bottom row shows the execution of cooperative trajectories, where the distance between the robots is kept constant. Fig. 8 shows the resulting forces along the stick.

In another experiment, we coupled two independently controlled KUKA LWR robots, and combined the task with adaptation to external, human interference, which is not fully repeatable due to the human in the loop. The task demanded that the robots – together with the human – place a lid on a wooden box, the robots holding one side, the human another side. The fit of the lid was very tight. The initial trajectories were learned by demonstration. Then, the box was moved 12 cm in $-p_x$ (backward) and 7 cm in p_y (robot's right) direction from the demonstrated position and our proposed approach was used to correct for the misplacement of the box in 8 epochs. p_x and p_y directions were corrected, while p_z (up-down) was not. The task was to minimize the force the robots exert to each-other and towards the human, i.e., $F_{d,x,y} = 0$ for both robots. Fig. 10 shows the resulting forces and positions of both robots.

We can see from the force plots in Fig. 10, that the person had to push on the lid in the first three epochs (see F_y plots). We can also see in the left F_z plot, that the lid did not fit in the box in the first 4 epochs. Once it did, the forces were reduced to desired values at 0N. Negative F_z is the force the human is exerting on the lid, while positive values indicate simply the weight of the lid. Fig. 11 shows the position of the lid after the demonstration and after the first 5 epochs. The experiment is also depicted in the accompanying video.

D. Effect of changes in epochs

In a real-world scenario with two independent agents performing a cooperative task, a systematic error of the system could influence the repeatability of the execution during
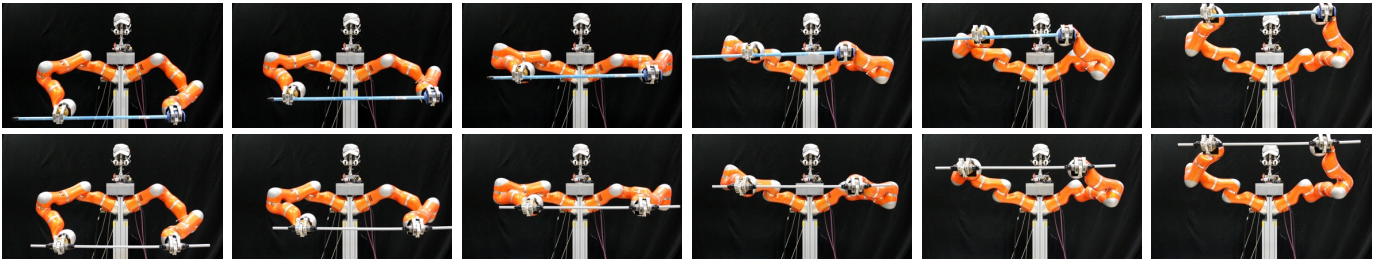
Fig. 9.   The execution of a task as performed by two independent DMPs (top row) and the execution with cooperative DMPS after learning (bottom row)
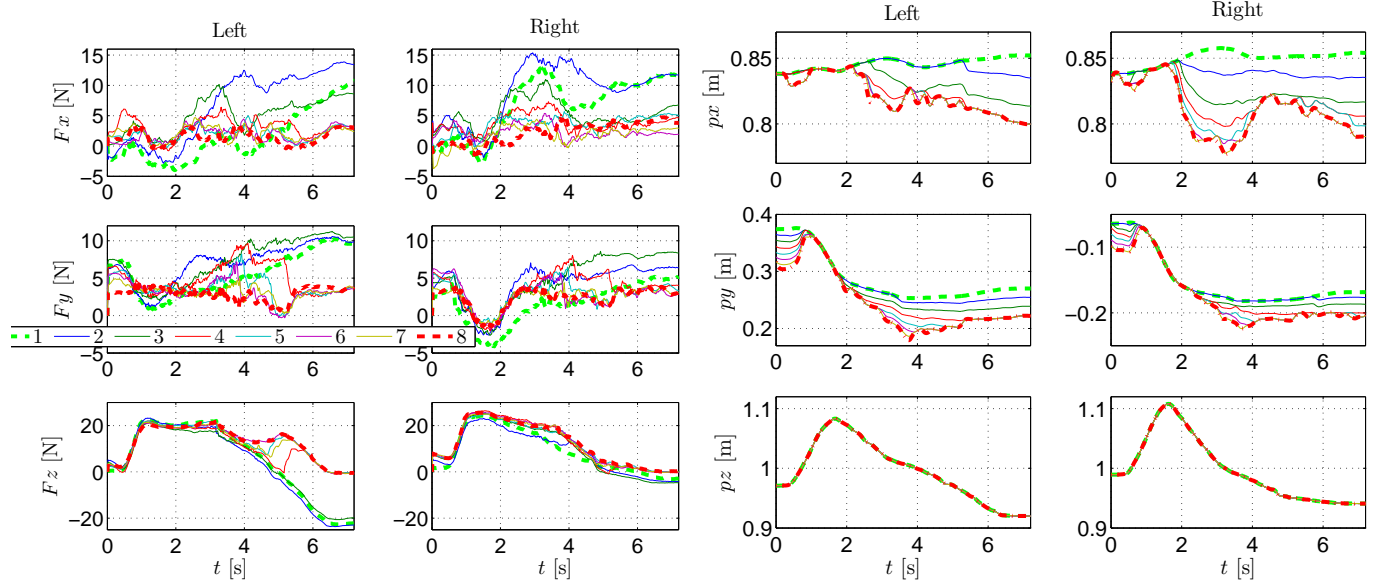

Fig. 10.   Results of adaptation of force in both robots in the left figure and of position in the right figure. The legend denotes the colors of separate epochs for all 12 plots. The 1-st is marked with green dashed and the last with red dash-dot line.
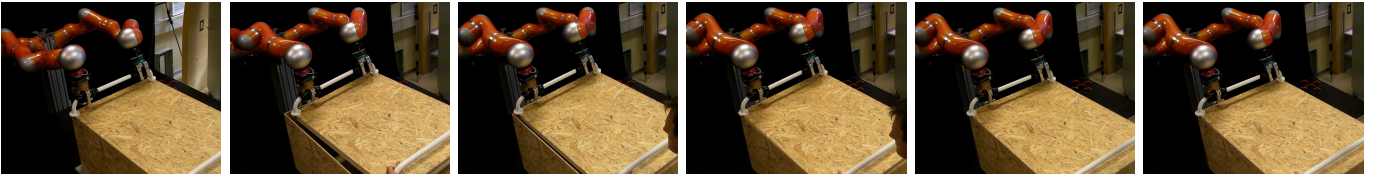

Fig. 11.   The position of the box, lid, and the robots in the demonstration in the leftmost picture. Positions after each epoch, with epoch number increasing towards the right. Only 5 epochs are shown because the pictures of the final position after epoch 5 are practically identical.

epochs. Examples of such could be the slipping of the wheels of wheeled robots cooperatively moving an object, faulty sensors, miscalibration ... In the experiment two robots are coupled to maintain a common distance of 0.4 m. The original trajectory of the right robot is a straight vertical line, and of the left robot a sinusoidal line. The left robot is the leader and does not adapt trajectories. It also drifts in the first five epochs ($i = 1, ..., 5$), which we simulate with $y_L(t) = 0.4 + \sin(t\pi/2)0.2 + (i-1)0.005t + (i-1)\sin(t\pi/2)0.002$. After the fifth epoch, the drifting stops. We also simulate noise.

Figure 12 shows the trajectories of both robots, with original trajectories in dashed green and final trajectories in dashed red. Adaptation through all epochs can be clearly seen. The final trajectories practically maintain the desired distance at all times as can be seen in the RMS error results in Fig. 13. Figure 13 also depicts RMS errors for using only a feedback controller for coupling while the leader robot drifts, i.e., in the first five epochs. The results clearly show the advantage

of using ILC even if the trajectories are smoothly changing from trial to trial.

E. Obstacle Avoidance in Bimanual Tasks

When one of the arms in a bimanual task encounters an obstacle, both arms have to adapt. Similar experiments but with acceleration level coupling and hebbian-type learning of a filter gain were discussed in [39]. If the arms are coupled using our proposed approach, the feedback term will move the arm that does not encounter an obstacle, but considerable force will appear between the tightly coupled arms. If the obstacle is repeatable over epochs, the proposed ILC approach can be used to minimize the force. In our experiments we applied the same online obstacle avoidance algorithm as presented in [10], [25]. We used it to control the left robot arm.

The cooperative DMPs were set in a leader-follower relation, the left robot being the leader ($l_{f,L} = 0$). From the start both robots had identical 20 s vertical trajectories
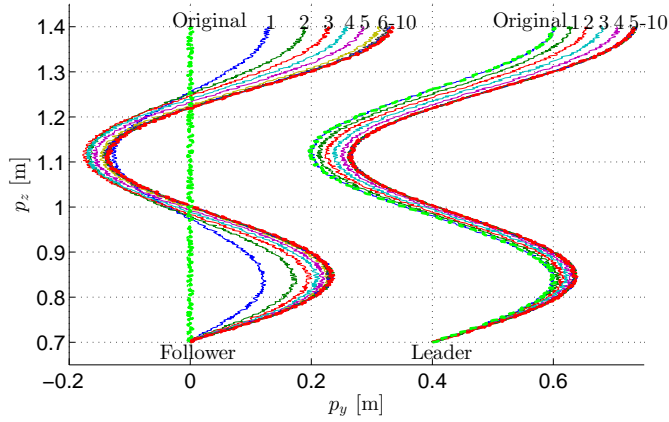
Fig. 12. Simulated results for cooperative DMPs. The leader robot does not adapt its trajectories. It also drifts in the first 5 epochs. The follower robot, which starts from a straight trajectory, adapts in every epoch and converges to the final trajectory after the drift of the leader robot stops. RMS error for the follower robot trajectories is shown in Fig. 13.
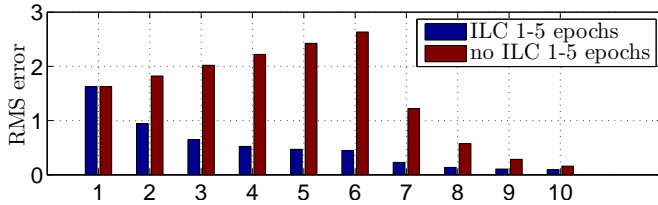

Fig. 13. RMS error for simulated trajectories from Fig 12. Despite the drift, the RMS error is being clearly reduced. RMS error is far bigger if only a feedback controller (no ILC) is used.

$(y_{L,R} = \pm 0.4$ m), but the left robot encountered an obstacle at $p_x = 0.7$ m, $p_y = 0.45$ m, $p_z = 1$ m and therefore had to apply obstacle avoidance. Our proposed algorithm was utilized to minimize the forces between the robots, rigidly connected with a stick. Fig. 14 shows the results of learning to minimize the forces between the robots in 7 epochs. The top plot shows the $p_y - p_z$ trajectory plot. The trajectories are for presentation again depicted at $y_{L,R} \pm 0.2$ m, but they were executed at $y_{L,R} = \pm 0.4$ m. The dashed green lines show the original trajectories. The black lines connecting the robots show the connecting stick every 5 s. The bottom plot shows the resulting forces between the robots, in p_y direction (blue), and the resulting torques around the global z (vertical) axis.

Fig. 15 shows the results of a similar real world experiment, where additionally the right robot encounters an obstacle at $p_x = 0.75$ m, $p_y = -0.4$ m, $p_z = 0.9$ m. The obstacle is set so that the robot must avoid it in the $-x$ direction. The resulting movement leads to a rotation of the stick between the robots, namely around the world z axis. The rotation of the object was a direct result of cooperation and no higher level planners were applied. The results indicate the ability of the algorithm to provide trajectories that can guide wide objects through narrow passages, e. g., a long board through a door, without any higher-level planning. Note that the coupling between the robots was in all task space degrees of freedom. In the top 3-D plot we can also notice the initial oscillations. These are the

result of both obstacle avoidance and cooperative terms acting on the trajectory of the right robot. The oscillations disappear by the final, 7-th epoch, marked with red. The bottom plot shows the resulting forces in the p_y direction and the resulting torque around the world z axis. Fig. 16 shows the two robots avoiding 2 obstacles. Note that in a simulated scenario, the robots could get stuck in a local minimum, where both the obstacle avoidance and the coupling terms would provide the same, excluding modulation values. In a real world scenario that is unlikely, even more so due to the fact that the coupling acts on both the velocity and acceleration.

VII. MODULATING THE DURATION OF COUPLED TRAJECTORIES

The introduction of the coupling term does not affect other DMP modulation properties, as was already demonstrated with obstacle avoidance. In the following we show how we can modulate the duration of coupled DMPs.

The property of not being directly dependent on time but on the phase x of the movement, allows the modulation of a DMP trajectory duration by changing a single parameter, i. e., parameter τ. Coupled DMPs preserve this property if a simple scaling factor is added to (11) or (17) and (20), respectively.

For example, if $\tau_{\text{new}} = 2\tau$ and the rest of the DMP parameters remain unchanged, the new DMP trajectory will take twice as long to execute. Other than duration, given correct initial conditions (position and velocity), the trajectory will remain unchanged. The coupling term F_c, which couples the trajectory to the environment or another robot, also depends on the phase x, see (26). Let us assume that the same τ governs the duration of both the trajectory and the coupling term F_c. By changing the duration of the coupled trajectory with $\tau_{\text{new}} = 2\tau$, the behavior of the robot will remain the same even though F_c was learned for τ, with the only change in (11), where the term $c_2\dot{C}$ must be changed to $\frac{\tau_{\text{new}}}{\tau}c_2\dot{C}$.

Fig. 17 shows the results of modulating τ for the case of interaction with the environment. A DMP trajectory was
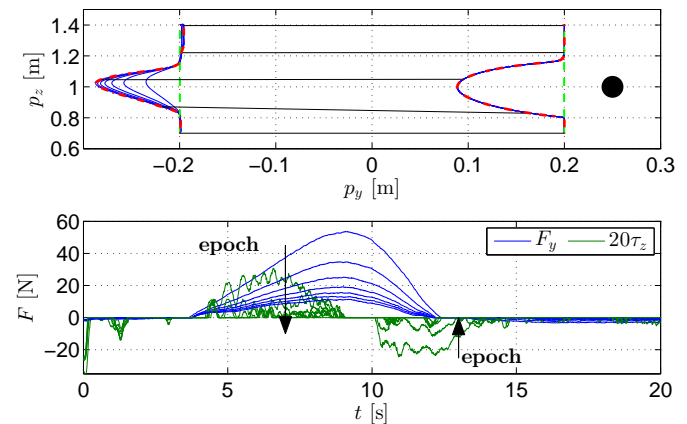

Fig. 14. Real world results of obstacle avoidance with the right robot following the left one. The top plot shows the $p_y - p_z$ trajectories, the original trajectories depicted with green dashed lines. The bottom plot shows the resulting forces and the resulting torques (scaled 20 times for presentation purposes).
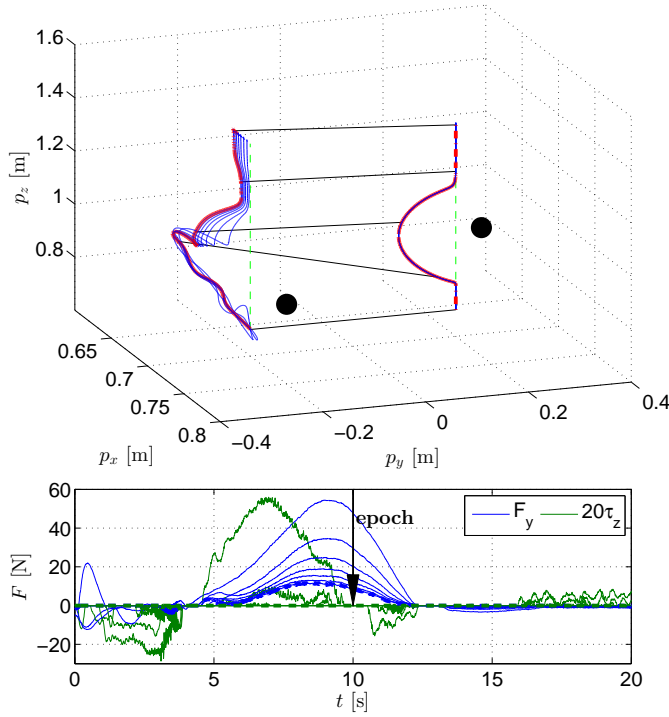
Fig. 15. Real world results of double obstacle avoidance of cooperative DMP trajectories. The left robot is the leader, which encounters an obstacle, but the follower also encounters an obstacle. The top plot shows the trajectories, the final, 7-th plot marked in red dash-dot line. Dashed green lines show the original trajectories. The bottom plot shows the resulting force in p_y direction and the resulting torque (scaled 20 times for presentation purposes) around the world z-axis. Dashed lines show the forces and torques in the final epoch.

Fig. 17. Simulation results obtained by modulating the duration of an environment-coupled DMP. The original trajectory and the trajectory with the changed duration are in the top plot. Both trajectories, where the modulated trajectory time scale is changed with the same ratio as the duration, are presented in the bottom plot.
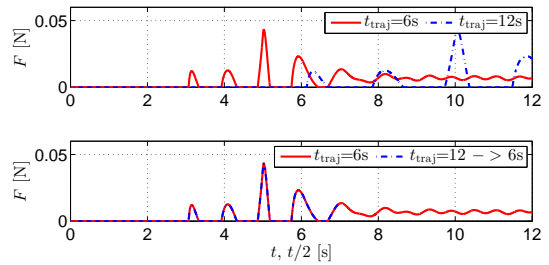
Fig. 18. Simulation force results obtained by modulating the duration of an environment-coupled DMP. The measured force during the execution of the original trajectory and the force measured along the trajectory with the changed duration are in the top plot. Both forces, where the modulated trajectory time scale is changed with the same ratio as the duration, are presented in the bottom plot.

encoded to reach from $p_z = 1.4$ m to $p_z = 0.7$ m, and the coupling term $F_c(x)$ learned to stop at the obstacle at $p_z = 1$ m, but not press on it ($F_d = 0$). The top plot shows the trajectory for $\tau = 6$ in red, and the temporally modulated trajectory for $\tau_{new} = 2\tau = 12$, in dashed blue. The bottom plot shows the same trajectories, but the modulated (dashed blue) trajectory is plotted against $t/2$, i.e., the time axis is squeezed. The trajectories match perfectly. The same initial position and velocity conditions were applied for both trajectories. Fig. 18 shows the force results for the same scenario, also showing a perfect matching.
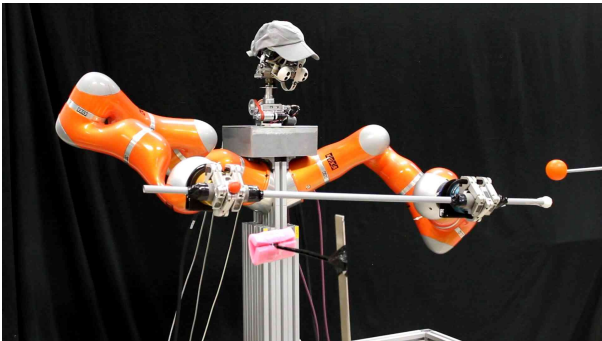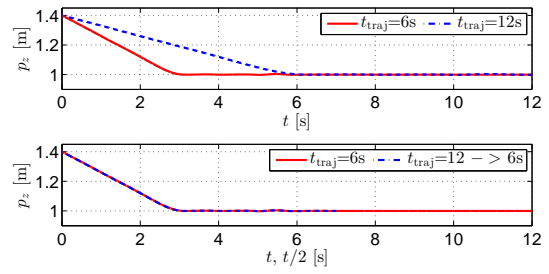

Fig. 16. Simultaneous avoidance of two obstacles. The leader robot arm (left) encounters an obstacle to its left side (orange ball). Before that, the follower robot encounters an obstacle in front of itself (pink foam).

VIII. DISCUSSION

In this paper we have shown that the proposed approach can effectively be utilized to achieve the desired force contact behavior for both interaction with the environment and cooperative/bimanual tasks. Because it generates an internal environment model, i. e., learns the predictive coupling term to achieve the desired behavior, it can prevent hard contacts with the environment, which can arise during pure feedback control. The robot learns to anticipate when a contact will occur, and prepares appropriately. The sensory feedback is always present in the system and assures that the robot gradually adapts to a different configuration, should the need arise. In the following we discuss some issues of the proposed method and compare it to similar approaches in the literature. We also briefly discuss non-stationary conditions.

The current-iteration iterative learning control algorithm requires two important tunable parameters, namely Q and L. The force gain parameter c determines the responsiveness of the system in the first iteration when the coupling term F_c equals 0. Parameters were tuned using *heuristic* approach [45], where the choice of Q is a tradeoff between the stability region and steady-state error. L is calculated according to (42). As stated in the literature [27], a combination of ILC and feedback controller can also be applied to reject noise and nonrepeating disturbances. We showed in Figs. 7, 12 and 13, that the approach is beneficial in the presence of both noise and systematic errors.

Considering that our approach can modify any trajectory, to achieve the desired behavior, the approach is in its essence general. While it is true that it requires a few repetitions of the task to learn the behavior, the same can be claimed for any learning scenario. Since many objects in a human home, such as furniture, are stationary, the learning process needs to take place only once for a specific task. The feedback controller and continuous learning over the epochs can account for changes in the environment. When considering the algorithm for non-stationary conditions, for example when operating in contact with a human or when dealing with moving objects, the proposed algorithm is also applicable as we have shown in our experiments.

Contrary to our approach, which changes the reference trajectory to achieve the desired interaction dynamics, Cheah & Wang [46] showed how to apply ILC to learn the target impedance model. To improve stability of interaction, Yang et al. [47] moved beyond standard ILC and proposed a learning controller for interaction tasks by adapting feedforward force and impedance. The advantage of changing reference trajectories – like in our work – is that we can anticipate contacts before they arise. We have shown in our experiments that the proposed approach can cope with gradual changes in the environment.

With respect to similar algorithms in the literature, the algorithm in [38] uses a similar setting. A trajectory is executed once and the measured signals are used as referential signals for the controller. This is also the fundamental difference to our approach, as the first execution gives the reference, while in our case the reference can be anything, applied to any trajectory. While the papers [38] and [4] show impressive disturbance rejection results, true generalization remains subject to the first, successful and referential execution.

In the approach by Pastor et al. [38], the output of the controller is fed into the acceleration level of the DMP to generate an improved movement. Notably, the measured force is equivalent to acceleration and therefore it makes sense to couple the DMP at the acceleration level. However, we have shown that using only the acceleration level of the DMP for coupling results in greater oscillations in the direction of the coupling force. As we can see from the root-locus plot of the coupled system (see Fig. 4), coupling at the velocity level results in better damping of the system. In view of the proposed ILC algorithm, this has an effect on the stability of learning. If only a feedback controller at the velocity level is used, a certain error is expected to appear in case of changed conditions every time, even if the error is repeatable in consecutive motions. This is demonstrated in the obstacle avoidance task, as shown in Fig. 14. Since the original trajectories are perfectly parallel, the first epoch, while $F_c = 0$, can be considered the same as the approach by Pastor et. al [4], but with coupling at both the velocity and acceleration levels. If no learning were present, the measured force between the robots would remain the same throughout the epochs. Reducing the force in consecutive executions is the real advantage of the proposed algorithm.

Kulvicius et al. [39] proposed to couple DMPs at the acceleration level. In this paper we demonstrated that it is beneficial to couple DMPs at velocity and acceleration level. Their approach uses a modified DMP representation, which is explicitly dependent on time. In our work we keep DMPs phase dependent, which allowed us to implement velocity scaling (Section VII). Most importantly, instead of learning a predictive term (26), [39] applies Hebbian type learning to determine a filter gain, which uses coupling force error as the input. Such a formulation cannot learn to anticipate coupling forces across learning epochs. Using the well-defined ILC framework from control theory, we were also able to prove the convergence and stability of the proposed scheme without any linearization assumptions.

IX. Conclusion

In this paper we presented a new approach for learning coupling terms for interactive and cooperative DMPs. Introducing force feedback into the well defined framework of the DMPs is one of the key advantages of the proposed approach. It enables learning of coupling terms that establish desired contact forces with the environment and the adaptation of trajectories for cooperative task execution, essentially bridging the gap from the purely kinematic domain of the DMPs to dynamic behavior.

We have shown that both the coupling and the learning algorithms are stable; that it is important that the coupling terms are added at the appropriate level, i.e., velocity and acceleration; that it is robust to noise and systematic errors; and that it can be applied to use real force feedback. The latter was demonstrated in a number of simulation and real-world experiments, where the approach was applied to actual interaction and bimanual cooperation tasks, including cooperation with a human. The low number of learning epochs also makes on-line learning of the coupled/interactive trajectories a viable possibility.

Appendix

In the following we will derive state space representation for the coupled DMP system. By writing (17) and (18) and (20) and (21) as second order equations we get

$$\ddot{y}_1 + \frac{\alpha_z}{\tau}\dot{y}_1 + \frac{\alpha_z\beta_z}{\tau^2}y_1 = \frac{c_2+\tau}{\tau^2}\dot{C}_{1,2} + \frac{\alpha_z}{\tau^2}C_{1,2} + \frac{\alpha_z\beta_z}{\tau^2}g_1$$
$$+ \frac{f_1(x)}{\tau^2}, \qquad (44)$$
$$\ddot{y}_2 + \frac{\alpha_z}{\tau}\dot{y}_2 + \frac{\alpha_z\beta_z}{\tau^2}y_2 = \frac{c_2+\tau}{\tau^2}\dot{C}_{2,1} + \frac{\alpha_z}{\tau^2}C_{2,1} + \frac{\alpha_z\beta_z}{\tau^2}g_2$$
$$+ \frac{f_2(x)}{\tau^2}. \qquad (45)$$

Applying Laplace transform [43] to both differential equations yields

$$(s^2 + a_1 s + a_2)\mathcal{Y}_1 = (b_1 s + b_2)\mathcal{C}_{1,2} + b_3\mathcal{X}_1, \quad (46)$$
$$(s^2 + a_1 s + a_2)\mathcal{Y}_2 = (b_1 s + b_2)\mathcal{C}_{2,1} + b_3\mathcal{X}_2, \quad (47)$$

where signals are now Laplace transform of the continuous time signals assuming zero initial conditions[1] ($\mathcal{Y} = \mathcal{L}(y), \mathcal{C} =$

---

[1]We achieve this by subtracting the initial values from the corresponding signals.

$\mathcal{L}(C), \mathcal{F} = \mathcal{L}(F), \mathcal{X}_i = \mathcal{L}(\alpha_z\beta_z g_i + f_i(x)))$ and parameters a_1, a_2, b_1, b_2, b_3 are

$$a_1 = \frac{\alpha_z}{\tau}, \ a_2 = \frac{\alpha_z\beta_z}{\tau^2},$$

$$b_1 = \frac{c_2 + \tau}{\tau^2}, \ b_2 = \frac{\alpha_z}{\tau^2}, \ b_3 = \frac{1}{\tau^2}.$$

Lets rewrite both outputs in transfer function notation

$$\mathcal{Y}_1 = H_1\mathcal{C}_{1,2} + H_2\mathcal{X}_1, \tag{48}$$

$$\mathcal{Y}_2 = H_1\mathcal{C}_{2,1} + H_2\mathcal{X}_3, \tag{49}$$

where $H_1 = (b_1 s + b_2)/(s^2 + a_1 s + a_2)$ and $H_2 = b_3/(s^2 + a_1 s + a_2)$. Subtracting (49) from (48) and multiplying the result by k we obtain

$$\mathcal{F} = k(\mathcal{Y}_1 - \mathcal{Y}_2) = k(H_1(\mathcal{C}_{1,2} - \mathcal{C}_{2,1}) + H_2(\mathcal{X}_1 - \mathcal{X}_2)). \tag{50}$$

Now we can apply feedback. First we subtract (19) and (22), assume $l_{f1} = l_{f2} = 1$, and use (23) and (25) to get the relation

$$\mathcal{C}_{1,2} - \mathcal{C}_{2,1} = 2(\mathcal{F}_c + c\mathcal{F}_{1,2}) = 2\mathcal{F}_c + 2c(\mathcal{F}_d - \mathcal{F}), \tag{51}$$

and insert result into (50)

$$\mathcal{F} = 2kH_1\mathcal{F}_c + 2ckH_1(\mathcal{F}_d - \mathcal{F}) + kH_2(\mathcal{X}_1 - \mathcal{X}_2) \tag{52}$$

Solving for $\mathcal{F}$ yields

$$\mathcal{F} = \frac{2kH_1}{1 + 2ckH_1}\mathcal{F}_c + \frac{2ckH_1}{1 + 2ckH_1}\mathcal{F}_d \tag{53}$$

$$+ \frac{kH_2}{1 + 2ckH_1}(\mathcal{X}_1 - \mathcal{X}_2).$$

The output force is therefore the sum of three transfer functions multiplied with inputs $\mathcal{F}_c, \mathcal{F}_d$ and $\mathcal{X}_1 - \mathcal{X}_2$. By inserting the definitions of H_1 and H_2 back into (54) we obtain

$$\mathcal{F} = \frac{2k(b_1 s + b_2)}{s^2 + (a_1 + 2ckb_1)s + (a_2 + 2ckb_2)}\mathcal{F}_c$$

$$+ \frac{2ck(b_1 s + b_2)}{s^2 + (a_1 + 2ckb_1)s + (a_2 + 2ckb_2)}\mathcal{F}_d \tag{54}$$

$$+ \frac{k}{s^2 + (a_1 + 2ckb_1)s + (a_2 + 2ckb_2)}(\mathcal{X}_1 - \mathcal{X}_2).$$

Finally, we substitute a_1, a_2, b_1, b_2, b_3 and rewrite the transfer function (54) in a controllable canonical state space form and introduce initial conditions

$$\mathbf{A} = \begin{bmatrix} -\frac{\alpha_z\tau + 2ck(c_2+\tau)}{\tau^2} & 1 \\ -\frac{\alpha_z\beta_z\tau + 2ck}{\tau^2} & 0 \end{bmatrix},$$
$$\tag{55}$$

$$\mathbf{B} = \begin{bmatrix} \frac{2k(c_2+\tau)}{\tau^2} & \frac{2ck(c_2+\tau)}{\tau^2} & 0 \\ \frac{2k\alpha_z}{\tau^2} & \frac{2ck\alpha_z}{\tau^2} & \frac{k}{\tau^2} \end{bmatrix}, \tag{56}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \tag{57}$$

$$\mathbf{x}(0) = \begin{bmatrix} F(0) \\ \dot{F}(0) - \frac{\alpha_z\tau + 2ck(c_2+\tau)}{\tau^2}F(0) \end{bmatrix}, \tag{58}$$

$$F(0) = k(y_1(0) - y_2(0)). \tag{59}$$

REFERENCES

[1] O. Khatib, K. Yokoi, O. Brock, K. Chang, and A. Casal, "Robots in human environments," in *Proceedings of the First Workshop on Robot Motion and Control (RoMoCo)* , 1999, pp. 213–221.

[2] 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 International Conference on Humanoid Robots (Humanoids)*, 2006.

[3] R. B. Rusu, B. Gerkey, and M. Beetz, "Robots in the kitchen: Exploiting ubiquitous sensing and actuation," *Robotics and Autonomous Systems*, vol. 56, no. 10, pp. 844–856, 2008.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[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] Y. Wada and M. Kawato, "A via-point time optimization algorithm for complex sequential trajectory formation," *Neural Networks*, vol. 17, no. 3, pp. 353–364, 2004.

[10] 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.

[11] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation: An approach based on hidden Markov model and Gaussian mixture regression," *IEEE Robotics and Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010.

[12] S. Khansari-Zadeh and A. Billard, "Imitation learning of globally stable non-linear point-to-point robot motions using nonlinear programming," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, 2010, pp. 2676–2683.

[13] E. Gribovskaya, S. Khansari-Zadeh, and A. Billard, "Learning non-linear multivariate dynamics of motion in robotic manipulators," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 80–117, 2011.

[14] T. Inamura, I. Toshima, H. Tanie, and Y. Nakamura, "Embodied symbol emergence based on mimesis theory," *International Journal of Robotics Research*, vol. 23, no. 4-5, pp. 363–377, 2004.

[15] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, Washington, DC, 2002, pp. 1398–1403.

[16] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Networks*, vol. 21, pp. 682–697, 2008.

[17] J. Kober and J. Peters, "Learning motor primitives for robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 2009, pp. 2112–2118.

[18] F. Stulp, E. A. Theodorou, and S. Schaal, "Reinforcement learning with sequences of motion primitives for robust manipulation," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1360–1370, 2012.

[19] M. Tamosiunaite, B. Nemec, A. Ude, and F. Wörgötter, "Learning to pour with a robot arm combining goal and shape learning for dynamic movement primitives," *Robotics and Autonomous Systems*, vol. 59, no. 11, pp. 910–922, 2011.

[20] J. Kober, A. Wilhelm, E. Oztop, and J. Peters, "Reinforcement learning to adjust parametrized motor primitives to new situations," *Autonomous Robots*, vol. 33, pp. 361–379, 2012.

[21] 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.

[22] B. Nemec and A. Ude, "Action sequencing using dynamic movement primitives," *Robotica*, vol. 30, no. 05, pp. 837–846, 2012.

[23] T. Kulvicius, K. Ning, M. Tamosiunaite, and F. Wörgötter, "Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 145–157, 2012.

[24] 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.

[25] 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.

[26] 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.

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

[28] A. Gams, B. Nemec, L. Žlajpah, M. Waechter, A. Ijspeert, T. Asfour, and A. Ude, "Modulation of motor primitives using force feedback: Interaction with the environment and bimanual tasks," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 5629–5635.

[29] G. Heinzinger, D. Fenwick, B. Paden, and F. Miyazaki, "Stability of learning control with disturbances and uncertain initial conditions," *IEEE Transactions on Automatic Control*, vol. 37, no. 1, pp. 110–114, 1992.

[30] J. Wallén, M. Norrlöf, and S. Gunnarsson, "Arm-side evaluation of ILC applied to a six-degrees-of-freedom industrial robot," in *Proceedings of the 17th World Congress The International Federation of Automatic Control*, 2008, pp. 13 450–13 455.

[31] A. Tayebi and S. Islam, "Adaptive iterative learning control for robot manipulators: Experimental results," *Control Engineering Practice*, vol. 14, no. 7, pp. 843–851, 2006.

[32] H.-S. Ahn, Y.-Q. Chen, and K. Moore, "Iterative learning control: Brief survey and categorization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1099–1121, 2007.

[33] L. Villani and J. De Schutter, "Force control," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds.   Springer, 2008.

[34] N. Hogan, "Impedance control: An approach to manipulation I II III," *Journal of Dynamic Systems, Measurement and Control*, no. 109, pp. 1 – 24, 1985.

[35] A. Gams, M. Do, A. Ude, T. Asfour, and R. Dillmann, "On-line periodic movement and force-profile learning for adaptation to new surfaces," in *2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Nashville, TN, 2010, pp. 560–565.

[36] J. Ernesti, L. Righetti, M. Do, T. Asfour, and S. Schaal, "Encoding of periodic and their transient motions by a single dynamic movement primitive," in *2012 IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012, pp. 57–64.

[37] V. Koropouli, D. Lee, and S. Hirche, "Learning interaction control policies by demonstration," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 344–349.

[38] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards associative skill memories," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012, pp. 309–315.

[39] 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*, 2013.

[40] S. Calinon, Z. Li, T. Alizadeh, N. G. Tsagarakis, and D. G. Caldwell, "Statistical dynamical systems for skills acquisition in humanoids," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012, pp. 323–329.

[41] P. Chiacchio, S. Chiaverini, and B. Siciliano, "Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system," *J. Dyn. Sys., Meas., Control*, vol. 118, no. 4, 1996.

[42] N. Likar, B. Nemec, and L. Žlajpah, "Virtual mechanism approach for dual-arm manipulation," in *Robotica*, 2013.

[43] M. Gopal, *Modern Control System Theory*.   Wiley, 1993.

[44] K. Moore, Y. Chen, and H.-S. Ahn, "Iterative learning control: A tutorial and big picture view," in *2006 45th IEEE Conference on Decision and Control*, San Diego, CA, 2006, pp. 2352–2357.

[45] M. Norrlöf and S. Gunnarsson, "Experimental comparison of some classical iterative learning control algorithms," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 4, pp. 636–641, 2002.

[46] C.-C. Cheah and D. Wang, "Learning impedance control for robotic manipulators," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 3, pp. 452–465, 1998.

[47] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schäffer, and E. Burdet, "Human-like adaptation of force and impedance in stable and unstable interactions," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 918–930, 2011.

Andrej Gams received his diploma degree in electrical engineering in 2004 and his Ph.D. degree in robotics from the University of Ljubljana, Slovenia, in 2009.

He is currently a Research Fellow at the Department of Automatics, Biocybernetics, and Robotics at Jožef Stefan Institute in Ljubljana, Slovenia. He was a postdoctoral researcher at Biorobotics Laboratory, EPFL, Lausanne, Switzerland in 2012-13. His research interests include imitation learning, control of periodic tasks and humanoid cognition.

Dr. Gams received the Best paper research Award at the 19th International Workshop in Alpe-Adria-Danube region in 2010, and the Jožef Stefan Golden Emblem award for his Ph.D. thesis in 2012. He received the SCIEX NMS-CH fellowship for postdoctoral studies at EPFL, Switzerland in 2012. He has been a program committee member of several conferences, including Humanoids 2011.

Bojan Nemec received his diploma degree in electrical engineering in 1979, M.Sc degree in 1982 and his Ph.D. degree in robotics from the University of Ljubljana, Slovenia, in 1988.

He is currently a senior research associate at the Dept. of Automatics, Biocybernetics and Robotics, Jožef Stefan Institute. He spent his sabbatical leave at the Institute for Real-Time Computer Systems and Robotics, University of Karlsruhe in 1993. His research interests include robot control, robot learning, service robotics and sports biomechanics.

Dr. Nemec received NTF award for best paper in category of Entertainment Robots and Systems at IROS 2009. He has been a program committee member of several conferences and general chair of the Workshop in Alpe-Adria-Danube region in 2013.

Auke Jan Ijspeert (M'00) received the B.Sc./M.Sc. degree in physics from the Ecole Polytechnique Fdrale de Lausanne (EPFL), Lausanne, Switzerland, and the Ph.D. degree in artificial intelligence from the University of Edinburgh, Edinburgh, U.K. He is currently a Professor with the EPFL, where he is also the Head of the Biorobotics Laboratory.

His research interests are at the intersection between robotics, computational neuroscience, nonlinear dynamical systems, and applied machine learning. He is interested in using numerical simulations and robots to get a better understanding of sensorimotor coordination in animals and in using inspiration from biology to design novel types of robots and adaptive controllers.

Dr. Ijspeert received the Best Paper Award at the IEEE International Conference on Robotics and Automation in 2002, the Industrial Robot Highly Commended Award at the Eighth International Conference on Climbing and Walking Robots in 2005, and the Best Paper Award at the IEEE-RAS International Conference on Humanoid Robots in 2007. He has been a program committee member of more than 40 conferences.

Aleš Ude received a diploma degree in applied mathematics from the University of Ljubljana, Slovenia, in 1990 and a Dr. Eng. sciences degree from the Faculty of Informatics, University of Karlsruhe, Germany, in 1995.

He is currently the head of Humanoid and Cognitive Robotics Lab, Department of Automatics, Biocybernetics, and Robotics, Jožef Stefan Institute, Ljubljana, Slovenia. He is also associated with the ATR Computational Neuroscience Laboratories, Kyoto, Japan. His research interests include autonomous robot learning, imitation learning, humanoid robot vision, perception of human activity, humanoid cognition, and humanoid robotics in general.

Dr. Ude is a recipient of the Science and Technology Agency fellowship for postdoctoral studies in ERATO Kawato Dynamic Brain Project, Kyoto, Japan. He has also received several fellowships from the Japan Trust International Research Cooperation Programme. He was a general chair of Humanoids 2011 conference. In recent years he has been a member of program committees of all major robotics conferences including ICRA, IROS, and RSS.

Interaction learning for dynamic movement primitives used in cooperative robotic tasks

Tomas Kulvicius [a,*], Martin Biehl [c], Mohamad Javad Aein [a], Minija Tamosiunaite [a,b], Florentin Wörgötter [a]

[a] Georg-August-Universität Göttingen, Bernstein Center for Computational Neuroscience, Department for Computational Neuroscience, III Physikalisches Institut - Biophysik, Friedrich-Hund Platz 1, D-37077 Göttingen, Germany
[b] Department of Informatics, Vytautas Magnus University, Vileikos g. 8, Kaunas, Lithuania
[c] Adaptive Systems Research Group, The University of Hertfordshire, School of Computer Science College Lane, Hatfield, Hertfordshire AL10 9AB, United Kingdom

HIGHLIGHTS

- We present a tightly-coupled robotics systems based on Dynamic Movement Primitives.
- We provide an analytical stability analysis for an equilibration of coupled system.
- We introduce sensory feedback with a predictive learning for the agent interaction.
- We show that such a mechanism allows us to learn an adaptive, sensor-driven interaction.
- We demonstrate that agents learn to cooperate when adding adaptive sensor control.

ARTICLE INFO

ABSTRACT

Since several years dynamic movement primitives (DMPs) are more and more getting into the center of interest for flexible movement control in robotics. In this study we introduce sensory feedback together with a predictive learning mechanism which allows tightly coupled dual-agent systems to learn an adaptive, sensor-driven interaction based on DMPs. The coupled conventional (no-sensors, no learning) DMP-system automatically equilibrates and can still be solved analytically allowing us to derive conditions for stability. When adding adaptive sensor control we can show that both agents learn to cooperate. Simulations as well as real-robot experiments are shown. Interestingly, all these mechanisms are entirely based on low level interactions without any planning or cognitive component.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Novel trajectory generation methods such as Dynamic Movement Primitives (DMPs [1,2]) or Gaussian Mixture Models (GMMs [3–5]) can generalize over different start and end points of the movement trajectory and they can efficiently emulate different trajectory shapes also allowing us to combine them in a dynamic way [6,7]. 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, like obstacle avoidance [8–15], grasping and object manipulation [16,17], locomotion and crawling [18,19], drumming [20], Ball-in-a-Cup game [21].

So far DMPs and GMMs have mainly been used for uncoupled agent systems. In this study, we analyze tightly coupled dual agent systems where each agent has its own path plan defined by a DMP. Note that in simulations we couple agents by a stiff virtual spring whereas in real robot scenario agents are coupled by a rigid object, i.e., a trace. 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 will 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.

In the following we will describe our framework for interactive DMPs also introducing the learning method. Mathematical derivations are given and a detailed analysis of signals and learning statistics is performed using simulations. In the end we then show experiments with a real robot-arm. Finally, in the discussion section, we will relate and compare our method to other existing approaches.

2. Methods

2.1. Dynamic Movement Primitives (DMPs)

To describe the movement trajectory of an agent we use the method for generating movement sequences proposed in [7] which is a modification of the original dynamic movement primitives (DMPs, [1,22,23,2]). 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 [7]. Similar to the original approach, modified DMPs are based on differential equations and consist of two dynamic systems: the transformation system and the canonical system. The transformation system is described as follows:

$$\dot{z} = \alpha(\beta(r - y) - z) + f, \tag{1}$$

$$\dot{y} = z, \tag{2}$$

$$\dot{r} = \begin{cases} (g - s)/T, & \text{if } t \leq T \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where α and β are time constants (in this study we used $\alpha = 0.75$, $\beta = \alpha/4$), $\dot{z}$, $\dot{y}$ and y correspond to acceleration, velocity and position, respectively. Here r defines a piecewise-linear goal function where s and g are the known start and goal states (start/end-point) and T is the duration of the movement. Initially we set $y_0 = r_0 = s$, $\dot{y}_0 = 0$ and $\dot{z}_0 = 0$.

The canonical system is described by a sigmoidal decay function:

$$\dot{\xi} = -\frac{\alpha_\xi \exp(\alpha_\xi(T - t))}{(1 + \exp(\alpha_\xi(T - t)))^2}, \tag{4}$$

where α_ξ is a time constant and defines the steepness of the sigmoidal function (in this study we used $\alpha_\xi = 1.0$) centered at time moment T. Initially we set $\xi_0 = 1$. The nonlinear function f is given by

$$f = \alpha_w \frac{\sum_{i=1}^{n} \psi_i \omega_i \xi}{\sum_i \psi_i}, \tag{5}$$

with

$$\psi_i = \exp\left(\frac{-(\frac{t}{T} - c_i)^2}{2\sigma_i^2}\right), \tag{6}$$

where ψ_i denote Gaussian kernels, c_i and σ_i is the center and width of the ith kernel, respectively. Kernels are placed evenly along the trajectory in time and spaced between 0 and 1, where 0 denotes the beginning of the movement trajectory and 1 the end. The shape of the movement trajectory is defined by weights ω_i and in our study they were generated manually but in general case they can be obtained by imitation learning [1,2]. Here we use α_w as a general scaling factor for all learned weights and in this study we set it to 1. Note that here DMPs are time dependent. This can be changed to phase-based DMPs without problems as shown in one of our older studies [7]. However, this is not relevant for the current investigations. In simulations and a real robot experiment we used a sampling rate of 200 Hz.

2.2. Interactive DMPs

We model the two agent system as two point particles coupled by 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 ith coordinate ($i = 1, 2, 3$ correspond to X, Y and Z-coordinates, respectively) of the jth particle ($j = 1, 2$ correspond to agent P and Q, respectively) as $y_{i,j}$ and the corresponding velocities as $z_{i,j}$. Assuming that the particles have mass m Newton's equation of motion is

$$m\dot{z}_{i,j} = F_{i,j}^S + F_{i,j}^D, \tag{7}$$

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

$$F_{i,1}^S = k\, direction\, (d - offset) = -F_{i,2}^S, \tag{8}$$

where $direction = (y_{i,1} - y_{i,2})/offset$ and $offset = \sqrt{\sum_l (y_{l,1} - y_{l,2})^2}$. Here d denotes the spring length when relaxed and k is the spring constant. In this study we used $d = 50$ cm and $k = 0.95$ N/cm. Note that in our simulations we omitted masses by setting them to $m = 1$ for both robots. In general, robots would adapt to any masses due to learning. Also, in simulations we chose k in order to obtain a relatively hard stiffness for coupling, whereas in the real robot scenario (as shown later) there is no need to tune k.

As explained above, the position of the agent is defined by a DMP and we denote its force $F_{i,j}^D$ by

$$F_{i,j}^D = m(\alpha\, (\beta\, (r_{i,j} - y_{i,j}) - z_{i,j}) + f_{i,j} + u^A + u^I). \tag{9}$$

Here u^A is a reactive term which is used for obstacle avoidance and in this study is fixed, whereas u^I is an interactive term and is learnt. Definitions of these terms are given below.

Also we have for the accelerations and velocities:

$$\dot{z}_{i,j} = \frac{1}{m}(F_{i,j}^S + F_{i,j}^D), \tag{10}$$

$$\dot{y}_{i,j} = z_{i,j}. \tag{11}$$

3. Implementation

3.1. Task definition

In the interaction learning scenario, as explained above, we have two identical agents (P and Q) 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 as shown in Fig. 2B.

3.2. Definition of the sensor inputs

We consider two types of sensor inputs. (1) Two avoidance sensors (touch and vision, with fixed characteristics used for obstacle avoidance) and (2) two interaction-relevant sensors (displacement and force used-for interaction learning).

Obstacle avoidance is implemented in the conventional way (potential field approach [8,9,12]) and is used to create realistic situations for interaction learning. We use two types of sensors which generate a compound avoidance signal: touch and vision. Normally the agent should be able to use the gradually rising vision

signal alone to avoid the obstacle. The binary touch signal is used as an emergence fall-back. For example, if the robot–robot interaction massively pushes one robot into an obstacle, then this situation can be recovered by the reaction to touch. Specifically we define:

(1a) The touch sensor obtains value 1 if the agents hits the obstacle and 0 if there is no collision:

$$A_T(t) = \begin{cases} 1, & \text{if } \delta(t) < \Theta_T \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

where δ is the minimal distance from the agent's center point to the obstacle, and Θ_T is the threshold for collision detection.

(1b) The visual sensor is triggered whenever the obstacle appears within the vision field (defined by a threshold) of the agent and is described by

$$A_V(t) = \begin{cases} \delta(t)/\Theta_V, & \text{if } \delta(t) < \Theta_V \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

where Θ_V defines the radius of the visual field. In this study we used $\Theta_T = 5$ cm and $\Theta_V = 20$ cm.

In general, we use a filter to smooth sensor signals and obtain as final input:

$$u^A(t) = a u^A(t - \Delta t) + (1 - a)(\Gamma_T A_T(t) + \Gamma_V A_V(t)), \tag{14}$$

with $a = 0.98$ the filter parameter. Here $\Gamma_{T,V}$ are weights which define the strength and direction of the obstacle avoidance reaction where positive weights were used to generate leftward/upward movements and negative weights for rightward/downward movements. Values for $\Gamma_{T,V}$ can change for different experiment and are, thus, given below. In this study we used $\Delta t = 0.005$ s.

(2a) The displacement sensor D is defined by

$$D(t) = \begin{cases} \eta(t)(y_a(t) - y_p(t)), & \text{if } |y_a(t) - y_p(t)| < \Theta_D \\ 0, & \text{otherwise,} \end{cases} \tag{15}$$

where $\eta(t) = 1$, if $u^A(t) < \epsilon$ and $\eta(t) = 0$, otherwise. In this study we used $\epsilon = 10^{-4}$. Thus, η acts as an inhibition term where an agent, that encounters an obstacle will not react to any push or pull produced by the other agent, as the need to avoid the obstacle is fundamental. This way the agents obtain their roles (leader and follower) naturally depending on the situation and there is no need to define them in advance. Here y_a is the actual trajectory of the agent defined by Eq. (11) and y_p is the planned trajectory obtained without spring force (i.e., in Eq. (10) we set $F^S = 0$). Note that here we use the threshold Θ_D for the displacement sensor in order to compensate for tracking errors, since in the real robot applications the actual and the planned paths will never match exactly. In this study we used $\Theta_D = 1$ cm. Also, the displacement signal does not influence the trajectories of the robots. It is only used for learning.

(2b) The force sensor signal F is defined by the force F^S of the spring model (see Eq. (8)):

$$F(t) = \eta(t)F^S(t). \tag{16}$$

This signal is filtered with

$$u^I(t) = a u^I(t - \Delta t) + (1 - a)\rho(t)F(t), \tag{17}$$

with $a = 0.98$ and ρ a weight, which will be changed by the learning rule described next. Note that in real robot experiment (as shown later) F^S is obtained from a force sensor of the robot. Also, please keep in mind that the agents are independent and have their respective sensors.

3.3. Learning rule

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) to grow weight ρ [24].

For this, we use a correlation based learning rule (Hebbian type [25]).

$$\dot{\rho} = \mu D F, \tag{18}$$

where μ is the learning rate. Learning stops as soon as $D = 0$, i.e., as soon as the displacement sensor is not triggered anymore and the predictive response has fully taken over.

4. Results

This section will start with an analytical stability analysis of the coupled agent system. This will be followed by simulation results, first for 1D cases and then for a 3D case. Finally we will show a real robot experiment.

4.1. Stability analysis

Such coupled DMP-based systems are quite interesting as coupling leads to mutual influence of one DMP onto the other. The question arises, thus, under which conditions these systems are stable and how they converge along the trajectory. In Appendix A we provide analytical details addressing this question. They are based on the, quite conventional, approximation of each agent as a point-source (end point of the robot's end-effector) and a mutual compliance that is modeled by a spring. Robot experiments shown here confirm that these approximations are justified for systems that have limited intrinsic dynamics (kinematically stiff conventional robots). As solutions are complex, here we will just summarize the result. We discuss the only case that $m_1 = m_2 = m$. Generalization to different masses are straightforward as only their relation m_1/m_2 is relevant.

Conventionally (uncoupled) DMPs are used with parameters α and $\beta = \alpha/4$ as this leads to critical damping along the DMP-trajectory and, thus, to optimally fast convergence [1]. Fixed point analysis (see Appendix A) of the coupled system reveals that for these parameters the system will oscillate along the direction of the spring. One has to choose $\alpha > \sqrt{8k}/\sqrt{m}$ and $\beta_c = (m\alpha^2 - 8k)/(4m\alpha)$ to achieve critical damping along the spring, but will receive now a slightly suboptimal (over-damped) behavior along the DMP trajectories. From analysis we also found that depending on the parameters there exist either one stable fixed point or two fixed points, one stable and the other unstable. The stable one represents the desired solution (agents approach goal-points). This analysis is valid for all start and goal positions and without any constraints on the system.

4.2. Interaction learning without obstacle avoidance

First of all we will present interaction learning between two agents without obstacle avoidance when the agents initially have different paths (paths were generated by setting DMP weights manually). In this case we are going to learn the interaction only for the Y-coordinate (1D case). Simulation results of such an experiment are shown in Fig. 1. Signal development during the learning process is given in panels A1–A3 (we show signals only for the agent Q, because signals of agent P are identical with an inverted sign). The first few seconds show the control case (before learning, 0–5 s). The first five learning trials follow (5–30 s). Note that here one trial lasts 5 s, which given a sampling rate of 200 Hz corresponds to 10^3 time steps). One can see that already during the first learning trial (5–10 s) forces are significantly reduced compared
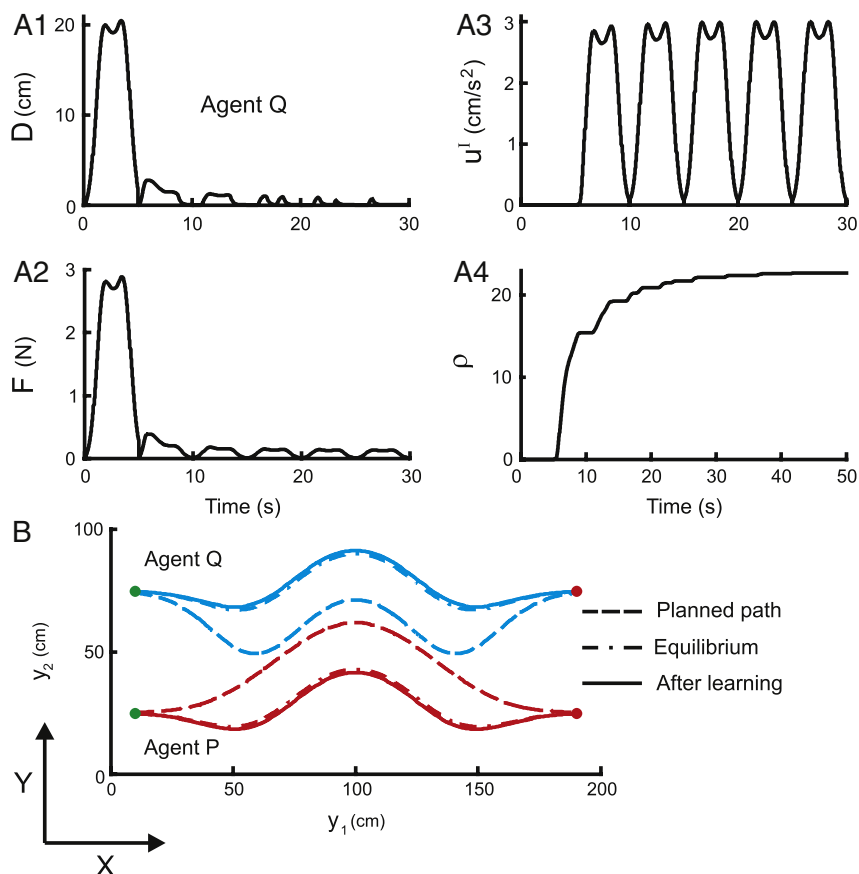
Fig. 1. Simulation results from interaction learning without obstacles. In panels **A1–A4** signals only for the agent Q are shown. **(A1)** Displacement signal D; **(A2)** predictive force signal F; **(A3)** output signal u^I; **(A4)** weight ρ; **(B)** trajectories. Green and red dots correspond to the start and end points, respectively. The learning rate was $\mu = 0.04$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

to the control trial (0–5 s). The displacement signal (panel A1) was fully avoided and the weights (panel A4) stabilized after eight learning trials. Resulting trajectories are presented in panel B. We observe that the agents converge to the "natural" equilibrium, which is the one that would also be obtained by purely passive equilibration, i.e., terms u^A and u^I are equal to zero all the time. The main gain from employing learning, however, is that this way the forces between both agents will be minimized, which is not the case for passive equilibration (in real robot scenarios, if the forces are high then the robots might lose or damage the object by which they are coupled).

4.3. Interaction learning with obstacle avoidance

In the next experiment we have a scenario where the agents initially have the same planned trajectories but they will encounter obstacles along their paths. So in this case agents have to learn to help each other to avoid obstacles by moving to the same direction. As in the previous experiment we are going to learn interaction only for the Y-coordinate.

Results for interaction learning with obstacle avoidance are presented in Fig. 2. As in the previous case signals for the control case (before learning, 0–5 s) and the first three learning trials (5–20 s) are shown in panels A1–A3. Here we can see that before learning agents will be pushed into the obstacles (see touch signals A_T, inset of panel A1) whereas – as learning proceeds – agents are learning to help each other and the obstacles are not touched anymore. Weight development is shown in panel A4. In this case it took one trial longer for agent Q than for agent P (17 and 16 trials, respectively) until weights finally stabilized since agent Q

was pushed stronger away by agent P due to its obstacle avoidance reaction. Resulting trajectories are shown in panel B where we can see that the roles of agents interchange depending on their sensor inputs, i.e., the agent which is going to avoid the obstacle becomes the leader and the other agent is the follower (which learns to follow the leader). For comparison we also show the case when the roles of the agents were predefined at the beginning (P = master and Q = slave). Resulting trajectories are presented in panel C where we can see that in this case only agent Q was learning to help the other agent. As a consequence agent Q was never able to avoid the obstacle (see also touch signal in the inset). These results clearly demonstrate that presented non-rigid Leader/Follower mechanism is advantageous compared to a fixed Master/Slave architecture.

4.4. Statistical evaluation

We evaluated our model statistically in a 3D scenario with obstacles (see Fig. 3C1) where the agents initially had different paths (most general case). In this case we were learning weights for the interaction for all three dimensions (X, Y and Z-coordinates). We analyzed the robustness of the learning and the influence of the learning rate μ by the number of learning trials (experiences) needed to learn interacting. Signals and resulting trajectories (see the supplementary video, Appendix B) from a single simulated experiment are shown in Fig. 3A–C, where in panels A1–A3 we show signals only for agent P (signals for agent Q look similar). Again, we show the control case (before learning, 5–10 s) and the first four learning trials. As in the previous examples, we see that the amplitude of the displacement signals for all three dimensions is
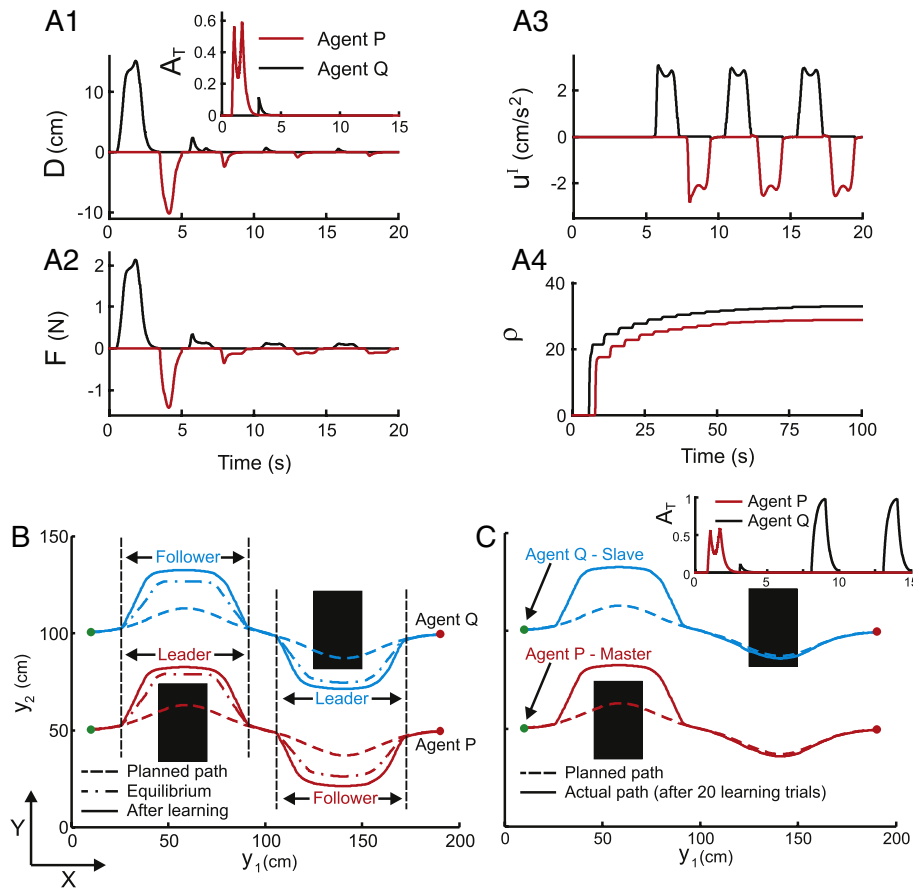
Fig. 2. (**A, B**) Simulation results from interaction learning with obstacles without predefined roles of the agents and **C** with initially predefined roles ($P = $ master and $Q = $ slave). Note that in simulations we let the robots go through obstacles, whereas in the real robot experiment this would not be the case and the behavior would much depend on the obstacle avoidance module, e.g., if the reflex is strong enough, then the robot would avoid an obstacle without toggling it. (**A1**) Displacement signals D; (**A2**) predictive force signals F; (**A3**) output signals u^I; (**A4**) weights ρ; (**B, C**) trajectories. The learning rate was $\mu = 0.4$. Weights for obstacle avoidance were tuned experimentally and were as follows: $\Gamma^P_{T,y_2} = 2$, $\Gamma^P_{V,y_2} = 5$, $\Gamma^Q_{T,y_2} = -2$, $\Gamma^Q_{V,y_2} = -5$ and $\Gamma^{P/Q}_{T/V,y_1} = 0$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

decreasing as learning proceeds. In this case, for agent P it took longer compared to agent Q (17 and 9 trials, respectively) until weights finally stabilized due to differences in trajectories and configuration of the obstacles.

To gather statistics we changed start and end positions of the coupled agents in every learning trial. We considered learning as finished and successful if (1) the displacement signal was not triggered and (2) the weights were not changing anymore within five consecutive trials. We found that weights converged and learning was successful in all 100 experiments. The influence of the learning rate is shown in Fig. 3D where we can observe, as expected, that higher learning rates lead to fewer learning experiences needed to learn interacting. Results also demonstrate that depending on the differences in trajectories and configuration of the obstacles the number of learning experiences between agents can vary. Note that in general too high learning rate will potentially lead to the one-shot learning where the learnt reaction might be not optimal for that particular case, i.e., overlearning.

4.5. Robot experiments

Finally, we performed two robot experiments (similar to the simulations) with KUKA light-weight arms [26]. In the first experiment we demonstrate the human–robot interaction, whereas in the second experiment we let two robots interact with each other. The results of these experiments are presented below.

4.5.1. Human–robot interaction

Here we let a human and a robot interact carrying a tray with bottles. Hence, here we do not consider two equilibrating DMPs (two robots), but only one. The goal is to avoid the red bar (left) not hitting it with the tray when moving along a curved trajectory (see Fig. 4B, T0). As in the simulations the robot has to learn to move in the same direction by reacting to the force sensor thereby helping the human avoiding the obstacle. Signals and resulting trajectories are shown in Fig. 4 and are similar to the experiments presented above. Note that here learning was applied only for the Y-coordinate. In this case learning stopped and weights stabilized after three learning trials (T1–T3, see the supplementary video, Appendix B).

4.5.2. Robot–robot interaction

In the second experiment, we let two KUKA robot-arms interact with each other where both robots were learning in this case. The experimental setup is shown in Fig. 5A. Robots were coupled by a wooden (painted in red) bar. The goal, as in the previous experiments, was to avoid obstacles (a box and a bottle) without hitting them when moving along a planned path. The positions of the obstacles were fixed and the paths for obstacle avoidance was predefined (as the planned trajectories) such that the robots were able to avoid obstacles without touching them when being decoupled. As paths are initially independent of each other, this, however, leads to the situation that prior to learning obstacles will be
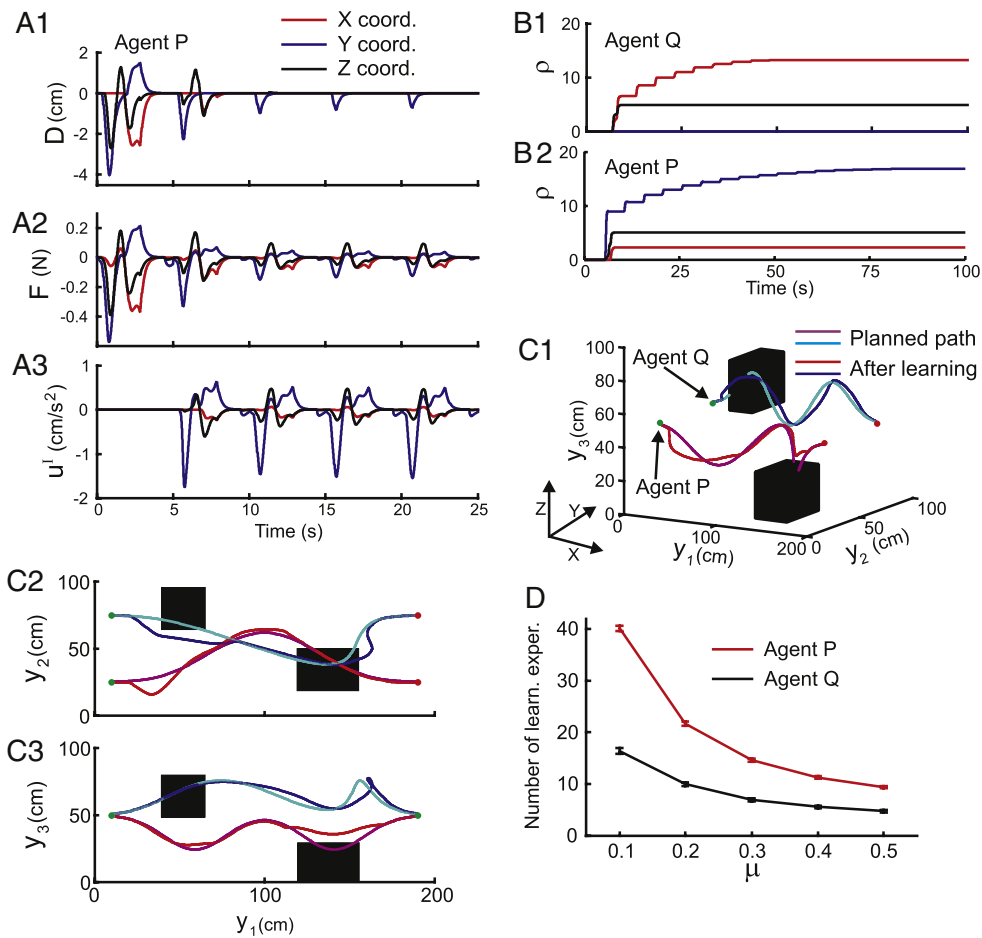
Fig. 3. **(A–C)** Simulation results from interaction learning with obstacles in a 3D scenario. In panels **A1–A3** signals only for agent P are shown. **(A1)** Displacement signals D; **(A2)**; predictive force signals F; **(A3)** output signals u^I; **(B1, B2)** weights ρ; **(C1–C3)** trajectories. The learning rate was $\mu = 0.2$. Weights for obstacle avoidance were tuned experimentally and were as follows: $\Gamma^P_{T,y_3} = 2$, $\Gamma^P_{V,y_3} = 2.5$, $\Gamma^Q_{T,y_2} = -2$, $\Gamma^Q_{V,y_2} = -2.5$, $\Gamma^P_{T/V,y_{1/2}} = 0$ and $\Gamma^Q_{T/V,y_{1/3}} = 0$. **(D)** Statistics for the 3D simulation as shown in panels A–C obtained from 100 experiments. In this case we changed start and end positions of the coupled agents in every trial (from a uniform distribution with interval $[-5; 5]$ cm for all, X, Y and Z positions), whereas the position of the obstacles was always the same. The average number of learning experiences needed to learn to interact is plotted vs. the learning rate μ. Error bars show confidence intervals of the mean (95%).

hit, when robots are coupled. Resulting signals are shown in panels C and D where we show one control case (equilibration, 0–20 s) followed by four learning trials and one post-learning trial. Note that here we have a full 3D case. In this case for the agent Q, learning stopped and weights stabilized after three trials and for agent P after four trials. Trajectories for planned, equilibrated and post-learning paths are shown in panels B1 and B2 (see the supplementary video, Appendix B).

5. Discussion

In this study we presented a combination of sensory driven interaction learning with dynamic movement primitives in a dual tightly-coupled agent system. Of importance for motivating the here-used learning mechanism is the following. Learning does not require manual tuning of parameters for sensors in order to produce appropriate behavior, but let the system find the right parameters by itself. Another advantage of using learning is that learning makes the system easily transferable to different agents/robots with different sensor–motor embodiments. The third reason which motivates the here-used learning is that the agent can adapt their interaction to new situations, which might occur due to environmental changes or changes in agents behavior. In the following we will compare our method to other existing approaches.

Previously, many different learning techniques, such as local/global regression techniques [1,2,27–29]) or reinforcement learning methods [30–34], were successfully applied to the DMP framework in order to learn movement trajectory and/or goal. Different from these approaches, we do not learn DMP weights ω, which encode the shape of the trajectory, but instead weight ρ of the interactive term u^I by which the trajectory is locally influenced. It is, of course, possible to combine ω- and ρ-learning. In addition to this, our approach can also be applied to other trajectory-shaping methods (like GMMs [3–5]). In this case the reactive term will have to be added to velocity, instead of acceleration as done here.

As already mentioned above, so far sensor-driven DMPs and GMMs have mainly been used in uncoupled systems. Different from this in the current study we were concerned with constructing sensor-driven *interactive* dynamic movement primitives in order to use them for cooperative tasks. We were interested in understanding the passive (equilibration) DMP-properties of coupled agents as well as their potential for conjoint adaptation. Conventionally, there are two architectures existing for introducing coupling: (1) master/slave and (2) non-master/slave [35]. In the first case one needs in advance to explicitly define master and slave, which is often a drawback of this architecture. The position of the slave (force-controlled) is defined by the position of the master (position controlled). The second architecture is a centralized approach where some reference frame is used to control both agents
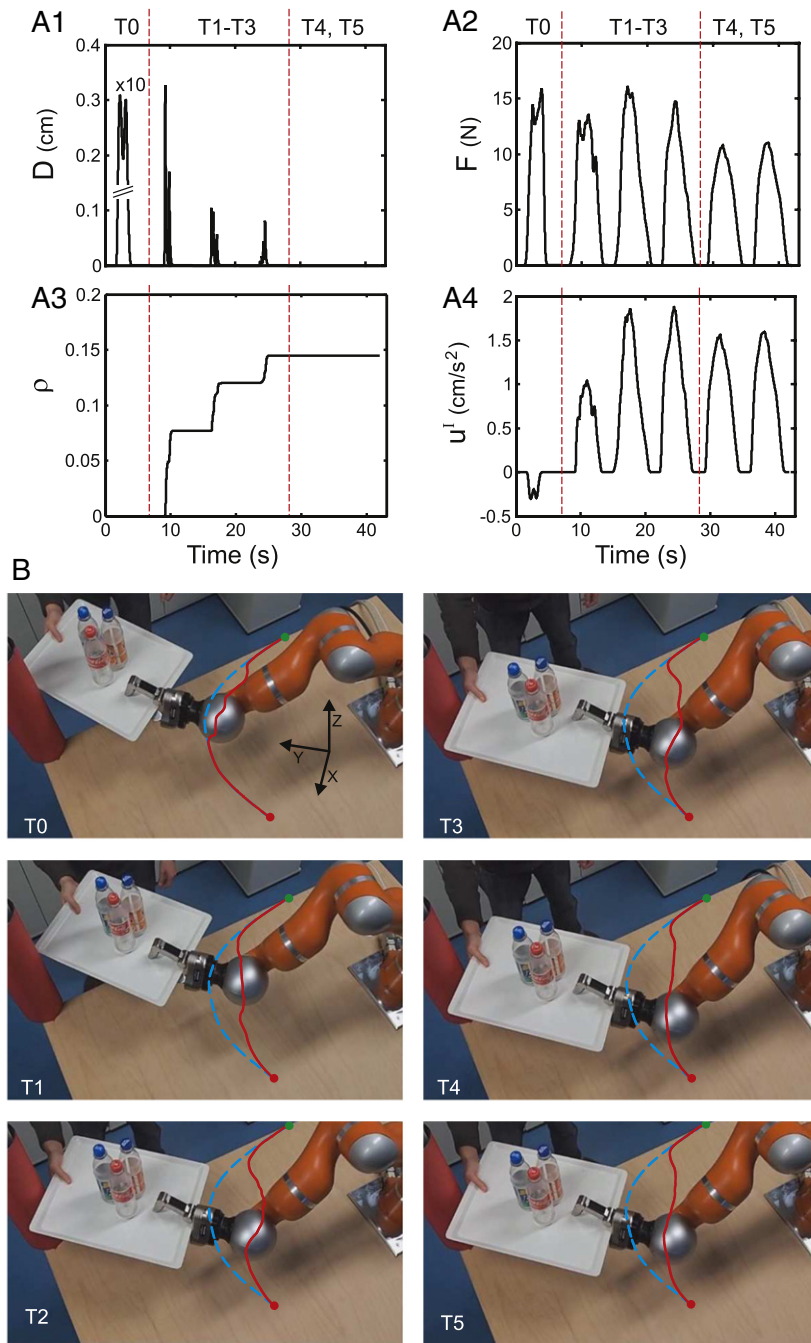
Fig. 4. Results from human–robot interaction learning. **(A)** Signal development from six trials (separated by dashed lines). Trial T0 is a control case-path-persistence behavior (no learning). **(A1)** Displacement signal D; **(A2)** predictive force signal F; **(A3)** weight ρ; **(A4)** output signal u^I. The 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. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

at the same time. Such architectures usually are realized by using position/force [35–38], impedance control or variable impedance control with virtual stiffness [39,40]. Our approach is similar to the impedance control, however, instead of setting/adapting stiffness parameters, we learn (in a model-free way) the appropriate reaction in order to minimize external forces between agents and help to cooperate with each other.

Our approach creates an alternating (depending on the environment) leader/follower architecture. Thus, we do not have to explicitly define the agents' roles beforehand, but they obtain them depending on their sensory information. This way the one agent that first encounters the obstacle becomes the leader, whereas the

other agent becomes the follower and learns to help the leader. Only in the real robot scenario presented above, we do indeed have a predefined master/slave architecture. However, here we wanted to show the example of the interaction learning where the robot learns to interact and help human. So here, only one (the robot) of the two agents was learning.

Another aspect of traditional master/slave architectures is that there is an important constraint which must be satisfied: the distance between the master and slave should be equal to the length of the object. In our approach we do not need to define this constraint. Agents act autonomously but this can also lead to situations where the object will be lost or deformed at the start of
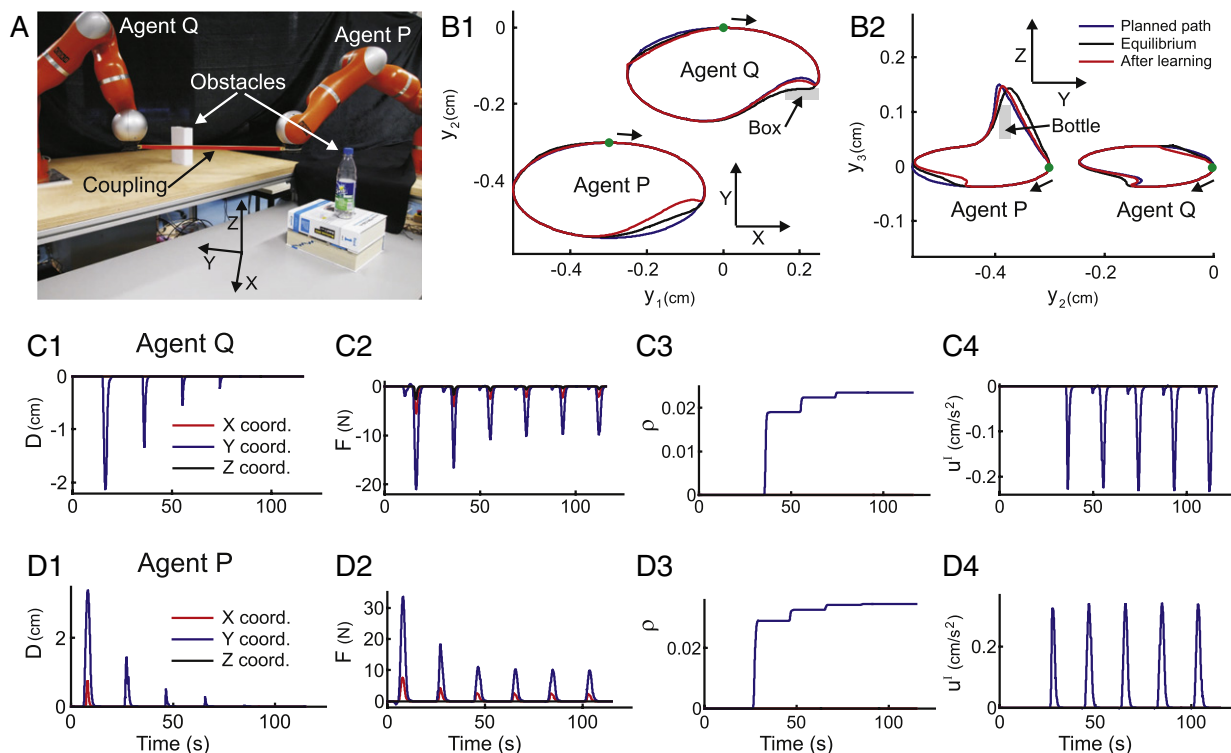
Fig. 5. Results from robot–robot interaction learning. **(A)** Experimental setup. **(B)** Trajectories for planned, equilibrated (before learning) and post-learning paths. Note that here robots started and ended at the same point marked by a green dot, whereas arrows show the direction of the movement. **(C1, D1)** Displacement signal D; **(C2, D2)** predictive force signal F; **(C3, D3)** weight ρ; **(C4, D4)** output signal u^l. The learning rate was $\mu = 0.001$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

learning. The trade-off between autonomy and potential damage cannot be resolved up front for all situations. In general, however, a "floating" master/slave architecture, such as the one present here, appears advantageous if the roles and/or the environmental constraints are not known in advance, i.e., for truly autonomous robots in a dynamic environment.

Also, different from the above mentioned tightly-coupled agent interaction approaches we use learning in order to acquire cooperate behavior instead of pre-programming it. There is a recent study by Gribovskaya et al. [41], which more closely relates to our method, where adaptive impedance control is employed for the agent interaction. Here the impedance parameters are adapted by using an iterative algorithm based on an error function. However, in this case robot-leader and robot-follower are again predefined in advance (master/slave architecture).

The way we couple DMPs with sensory inputs in order to produce an on-line reaction is similar to the approaches presented in [8,21,9,16,29,12,17]. Here, we show that such coupling can also be applied for dual-agent systems in order to solve cooperative tasks. Different from our approach, in [8,9,16,12] there is no learning applied and the reaction is generated by manually defined potential fields with fixed parameters which makes them incapable to adapt their behavior to new situations. [21,29,17] as in our case use learning, however, different from our approach, learning acts on DMP weights [21,29] and not on sensory terms as in our case. Also, learning in those approaches is performed in several phases, for example, first of all DMP weights are learnt to obtain basic behavior and only afterward they are modified by sensory feedback [21].

As explained above, the trajectory planning and on-line modification in our case was done in a task-space taking into account collision avoidance only for an end-effector of the manipulator. However, due to the interaction trajectories of the end-effectors might be altered in such a way that it will lead to link-collisions. To prevent such situations one can augment the system by adding

potential fields on the manipulator links [42] or use an inverse-kinematics model in which the null-space is constrained to avoid link-collisions [43,8].

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.

Acknowledgments

Appendix A. Stability analysis

To investigate how the coupled agents converge toward the goal we look at the fixed points of the dynamical system in Eqs. (10) and (11). We set $f_{i,j}[t] = 0$ and $r_{i,j}[t] = g_{i,j}$ because we are interested in the behavior as $t \to \infty$. Then without loss of generality we choose

coordinates such that $g_{1,1} = g_{1,2} = g_x$, $g_{3,1} = g_{3,2} = g_z$ and $g_{2,1} = a/2 = -g_{2,2}$, for some constant $a \geq 0$. In other words, we align the y-axis with the spring in the goal position. Note that all parameters (i.e., α, β, m, k, d, a) are assumed positive.

For convenience we use center of mass coordinates in this section. Let $\vec{y}_j = (y_{1,j}, y_{2,j}, y_{3,j})^T$, $\vec{z}_j = (z_{1,j}, z_{2,j}, z_{3,j})^T$. Then the transformation of the new coordinates is given by

$$
\begin{aligned}
\vec{y}_1 &= \vec{R} + \vec{r}, \\
\vec{y}_2 &= \vec{R} - \vec{r}, \\
\vec{z}_1 &= \vec{V} + \vec{v}, \\
\vec{z}_2 &= \vec{V} - \vec{v}.
\end{aligned}
\tag{A.1}
$$

Here $\vec{R} = (R_x, R_y, R_z)^T$ and $\vec{V} = (V_x, V_y, V_z)^T$ are the center of mass positions and their velocities respectively, $\vec{r} = (r_x, r_y, r_z)^T$ the relative coordinates, $\vec{v} = (v_x, v_y, v_z)^T$ their respective velocities. Standard linear stability analysis [44] of the transformed dynamical system reveals that depending on parameters, there is either only one stable fixed point or additionally an unstable one. In any case they share all coordinates besides r_y, these are as expected:

$$
\begin{aligned}
\vec{R}^* &= (g_x, 0, g_z)^T, \quad \vec{V}^* = (0, 0, 0)^T \\
\vec{v}^* &= (0, 0, 0)^T, \quad \vec{r}^* = (0, r_y^*, 0)^T.
\end{aligned}
\tag{A.2}
$$

One fixed point (FP1) always exists and is stable throughout the parameter range.

$$
FP1 : r_y^* = \frac{am\alpha\beta + 2dk}{2m\alpha\beta + 4k}.
\tag{A.3}
$$

This corresponds to both agents being as close to the goal position as the spring allows, e.g., let $d = a$ then $r_y^* = a/2 = g_{2,1}$ such that each agent's position coincides with its goal position. If $am\alpha\beta < 2dk$ there is another, unstable, fixed point (FP2) at

$$
FP2 : r_y^* = \frac{am\alpha\beta - 2dk}{2m\alpha\beta + 4k}.
\tag{A.4}
$$

This corresponds to the situation where the agents switch around ($r_y^* < 0$) and block each other from reaching the goal by compressing the spring in between them. The fixed point FP2 disappears when the pull toward the goal position is stronger than the spring can compensate ($am\alpha\beta > 2dk$).

In order to analyze the asymptotic dynamics toward FP1 we look at the linearization of the dynamical system at FP1. The linearized twelve dimensional first order system can be rewritten as six second order equations of $R_x, R_y, R_z, r_x, r_y, r_z$. Each equation is that of a free damped harmonic oscillator. The equations are

$$
\begin{aligned}
\ddot{\vec{R}} &= -\alpha\beta\vec{R} - \alpha\dot{\vec{R}}, \\
\ddot{r}_x &= -\frac{a\alpha\beta(m\alpha\beta + 2k)}{am\alpha\beta + 2dk}r_x - \alpha\dot{r}_x, \\
\ddot{r}_y &= -\frac{m\alpha\beta + 2k}{m}r_y - \alpha\dot{r}_y, \\
\ddot{r}_z &= -\frac{a\alpha\beta(m\alpha\beta + 2k)}{am\alpha\beta + 2dk}r_z - \alpha\dot{r}_z.
\end{aligned}
\tag{A.5}
$$

From here one can follow any text book on mechanics (e.g. [45]) to derive the DMP parameter that leads to critical damping β_c for each coordinate. For R_x, R_y and R_z we find $\beta_c^R = \alpha/4$ which corresponds to the standard value of β for uncoupled DMPs. For r_y we find $\beta_c^y = (m\alpha^2 - 8k)/(4m\alpha)$ which is smaller than β_c^R if $k > 0$. For r_x and r_z the expression is quite complicated

$$
\beta_c^{x,z} = \beta_c^y + \frac{1}{8}\sqrt{\frac{am^2\alpha^4 - 16akm\alpha^2 + 32dkm\alpha^2 + 64ak^2}{am^2\alpha^2}}
$$

and always larger than β_c^y. Note that for $a = d$, we obtain $\beta_c^{x,z} = \beta_c^R$. The goal in choosing β is that no oscillations occur, which means that if we cannot damp all coordinates critically, we prefer to overdamp them. In all cases above overdamping is achieved if $\beta < \beta_c$;

therefore, the smallest β_c, i.e., β_c^y should be chosen for the coupled system. Because we need $\beta > 0$ for a converging DMP, we must require $\alpha > \sqrt{8k/m}$ in this case.

Appendix B. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.robot.2013.07.009.

References

[1] J.A. Ijspeert, J. Nakanishi, S. Schaal, Movement imitation with nonlinear dynamical systems in humanoid robots, in: Proc. 2002 IEEE Int. Conf. Robotics and Automation, 2002, pp. 1398–1403.

[2] S. Schaal, P. Mohajerian, J.A. Ijspeert, Dynamics systems vs. optimal control—a unifying view, Progress in Brain Research 165 (2007) 425–445.

[3] S.M. Khansari-Zadeh, A. Billard, BM: an iterative method to learn stable non-linear dynamical systems with gaussian mixture models, in: Proc. 2010 IEEE Int. Conf. Robotics and Automation, 2010, pp. 2381–2388.

[4] S.M. Khansari-Zadeh, A. Billard, Imitation learning of globally stable non-linear point-to-point robot motions using nonlinear programming, in: Proc. 2010 IEEE Int. Conf. Intelligent Robots and Systems, 2010, pp. 2676–2683.

[5] S.M. Khansari-Zadeh, A. Billard, Learning stable non-linear dynamical systems with gaussian mixture models, IEEE Transactions on Robotics 27 (2011) 943–957.

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

[7] T. Kulvicius, K.J. Ning, M. Tamosiunaite, F. Wörgötter, Joining movement sequences: modified dynamic movement primitives for robotics applications exemplified on handwriting, IEEE Transactions on Robotics 28 (1) (2012) 145–157. http://dx.doi.org/10.1109/TRO.2011.2163863.

[8] D.-H. Park, H. Hoffmann, P. Pastor, S. Schaal, Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields, in: Proc. 8th IEEE-RAS Int. Conf. Humanoid Robots, 2008, pp. 91–98.

[9] H. Hoffmann, P. Pastor, D.-H. Park, S. Schaal, Biologically-inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance, in: Proc. 2009 IEEE Int. Conf. Robotics and Automation, 2009, pp. 1534–1539.

[10] N. Ratliff, M. Zucker, J.A. Bagnell, S. Srinivasa, Chomp: gradient optimization techniques for efficient motion planning, in: Proc. 2009 IEEE Int. Conf. Robotics and Automation, 2009, pp. 489–494.

[11] S. Calinon, I. Sardellitti, D.G. Caldwell, Learning-based control strategy for safe human–robot interaction exploiting task and robot redundancies, in: Proc. 2010 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2010, pp. 249–254.

[12] A.S. Phung, J. Malzahn, F. Hoffmann, T. Bertram, Get out of the way — obstacle avoidance and learning by demonstration for manipulation, in: Proc. 11th IFAC World Congress, 2011, pp. 11514–11519.

[13] M. Mühlig, M. Gienger, J.J. Steil, Interactive imitation learning of object movement skills, Autonomous Robots 32 (2) (2012) 97–114.

[14] T. Matsubara, S.-H. Hyon, J. Morimoto, Learning parametric dynamic movement primitives from multiple demonstrations, Neural Networks 24 (5) (2011) 493–500.

[15] S.-M. Khansari-Zadeh, A. Billard, A dynamical system approach to realtime obstacle avoidance, Autonomous Robots 32 (2012) 433–454.

[16] O. Krömer, R. Detry, J. Piater, J. Peters, Adapting preshaped grasping movements using vision descriptors, in: Proc. 11th Int. Conf. Simulation of Adaptive Behavior: from Animals to Animats, 2010, pp. 156–166.

[17] P. Pastor, L. Righetti, M. Kalakrishnan, S. Schaal, Online movement adaptation based on previous sensor experiences, in: Proc. 2011 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2011, pp. 365–371.

[18] L. Righetti, J.A. Ijspeert, Programmable central pattern generators: an application to biped locomotion control, in: Proc. 2006 IEEE Int. Conf. Robotics and Automation, 2006, pp. 1585–1590.

[19] S. Degallier, L. Righetti, L. Natale, F. Nori, G. Metta, J.A. Ijspeert, A modular bio-inspired architecture for movement generation for the infant-like robot iCub, in: Proc. 2nd IEEE RAS/EMBS Int. Conf. Biomedical Robotics and Biomechatronics, 2008, pp. 795–800.

[20] S. Degallier, C.P. Santos, L. Righetti, J.A. Ijspeert, Movement generation using dynamical systems: a humanoid robot performing a drumming task, in: Proc. 6th IEEE-RAS Int. Conf. Humanoid Robots, 2006, pp. 512–517.

[21] J. Kober, B. Mohler, J. Peters, Learning perceptual coupling for motor primitives, in: Proc. 2008 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2008, pp. 834–839.

[22] J.A. Ijspeert, J. Nakanishi, S. Schaal, Learning attractor landscapes for learning motor primitives, in: Proc. 15th Advances in Neural Information Processing Systems, 2003, pp. 1547–1554.

[23] S. Schaal, J. Peters, J. Nakanishi, J.A. Ijspeert, Learning movement primitives, in: Proc. 2003 Int. Symp. Robotics Research, 2004, pp. 561–572.

[24] B. Porr, F. Wörgötter, Isotropic sequence order learning in a closed loop behavioural system, Philosophical Transactions of the Royal Society A: Mathematical Physical and Engineering Sciences 361 (2003) 2225–2244.

[25] D.O. Hebb, The Organization of Behavior, Wiley & Sons, New York, 1949.

[26] Kuka Robot Systems. http://www.kuka-robotics.com.

[27] B. Nemec, M. Tamosiunaite, F. Woergoetter, A. Ude, Task adaptation through exploration and action sequencing, in: Proc. 9th IEEE-RAS Int. Conf. Humanoid Robots, 2009, pp. 610–616.

[28] A. Ude, A. Gams, T. Asfour, J. Morimoto, Task-specific generalization of discrete and periodic dynamic movement primitives, IEEE Transactions on Robotics 26 (2010) 800–815.
[29] A. Gams, M. Do, A. Ude, T. Asfour, R. Dillmann, On-line periodic movement and force-profile learning for adaptation to new surfaces, in: Proc. 10th IEEE-RAS Int. Conf. Humanoid Robots, 2010, pp. 560–565.
[30] J. Peters, S. Schaal, Reinforcement learning of motor skills with policy gradients, Neural Networks 21 (4) (2008) 682–697.
[31] E. Theodorou, J. Buchli, S. Schaal, Reinforcement learning of motor skills in high dimensions: a path integral approach, in: Proc. 2010 IEEE Int. Conf. Robotics and Automation, 2010, pp. 2397–2403.
[32] P. Kormushev, S. Calinon, D.G. Caldwell, Robot motor skill coordination with EM-based reinforcement learning, in: Proc. 2010 IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2010, pp. 3232–3237.
[33] M. Tamosiunaite, B. Nemec, A. Ude, F. Wörgötter, Learning to pour combining goal and shape learning for dynamic movement primitives, Robotics and Autonomous Systems 59 (11) (2011) 910–922.
[34] J. Kober, J. Peters, Policy search for motor primitives in robotics, Machine Learning 84 (1–2) (2011) 171–203.
[35] M. Uchiyama, P. Dauchez, Symmetric kinematic formulation and non-master/slave coordinated control of two-arm robots, Advanced Robotics 7 (4) (1993) 361–383.
[36] E. Nakano, S. Ozaki, T. Ishida, I. Kato, Cooperational control of the anthropomorphous manipulator 'MELARM', in: Proc. 4th Int. Symp. Industrial Robots, 1974, pp. 251–260.
[37] S. Fujii, S. Kurono, Coordinated computer control of a pair of manipulators, in: Proc. IFTOMM World Congr., 1975, pp. 411–417.
[38] M. Uchiyama, N. Iwasawa, K. Hakomori, Hybrid position/force control for coordination of a two-arm robot, in: Proc. IEEE Int. Conf. Robotics and Automation, vol. 4, 1987, pp. 1242–1247.
[39] F. Caccavale, P. Chiacchio, A. Marino, L. Villani, Six-DOF impedance control of dual-arm cooperative manipulators, IEEE/ASME Transactions on Mechatronics 13 (5) (2008) 576–586.
[40] T. Tsumugiwa, R. Yokogawa, K. Hara, Variable impedance control with virtual stiffness for human–robot cooperative task (human–robot cooperative peg-in-hole task), in: Proc. 41st SICE Ann. Conf., vol. 4, 2002, pp. 2329–2334.
[41] E. Gribovskaya, A. Kheddar, A. Billard, Motion learning and adaptive impedance for robot control during physical interaction with humans, in: Proc. IEEE Int. Conf. Robots and Automation, 2011, pp. 4326–4332.
[42] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, International Journal of Robotics Research 5 (1) (1986) 90–98.
[43] A.A. Maciejewski, C.A. Klein, Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments, International Journal of Robotics Research 4 (3) (1985) 109–117.
[44] S.H. Strogatz, Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering, first ed., Westview Press, 2001.
[45] P.A. Tipler, G. Mosca, Physics for Scientists and Engineers: Standard Version, fifth ed., W.H. Freeman, 2003.

Martin Biehl received his M.S. degree in Physics from Vienna University of Technology, Austria, in 2009. Currently he pursues a Ph.D. degree at the Adaptive Systems Research Group, The University of Hertfordshire. His research focuses on autonomy, interaction, and information theory of sensor actor systems.

Mohamad Javad Aein received Bachelor of Science degree in Electrical Engineering from Amirkabir University of Technology (AUT), Tehran, Iran, in 2006. He received his Master of Science degree in Control Systems in 2009 at the same university for his work on soft tissue modeling and needle insertion in Robotic surgery. From 2009 to 2011, he worked as research assistant in Advanced Intelligent Systems Research Center (AISRC) in Tehran, Iran, being engaged in numerous modeling and control-related projects. Currently he is doing Ph.D. at the Department of Computational Neuroscience, University of Göttingen, Germany. His research interests include control of robotic manipulators, trajectory planning and behavior-based robotic manipulations.

Minija Tamosiunaite has received a Ph.D. in Informatics in Vytautas Magnus University, Lithuania, in 1997. Currently she works as a senior researcher at the Bernstein Center for Computational Neuroscience, Inst. Physics 3, University of Göttingen. Her research interests include machine learning, biological signal analysis, and application of learning methods in robotics.

Tomas Kulvicius received a Ph.D. degree in Computer Science (2010) from the University of Göttingen, Germany. In his Ph.D. thesis he investigated development of receptive fields in closed loop learning systems. Currently he is a Postdoctoral Researcher at the Department for Computational Neuroscience in the University of Göttingen. His research interests include closed loop behavioral systems, learning algorithms, receptive fields, robotics, dynamic movement primitives, biosignal analysis, biological system modeling.

Florentin Wörgötter studied biology and mathematics at the University of Düsseldorf, Germany. He received his Ph.D. for work on the visual cortex from the University of Essen, Germany, in 1988. From 1988 to 1990, he was engaged in computational studies with the California Institute of Technology, Pasadena. Between 1990 and 2000, he was a Researcher at the University of Bochum, Germany, where he was investigating the experimental and computational neuroscience of the visual system. From 2000 to 2005, he was a Professor of computational neuroscience with the Psychology Department, University of Stirling, U.K., where his interests strongly turned toward "Learning in Neurons". Since July 2005, he has been the Head of the Computational Neuroscience Department at the Bernstein Center for Computational Neuroscience, Inst. Physics 3, University of Göttingen, Germany. His current research interests include information processing in closed-loop perception–action systems, sensory processing, motor control, and learning/plasticity, which are tested in different robotic implementations. His group has also developed the RunBot, which is a fast and adaptive biped-walking robot.

Grounded Spatial Symbols for Task Planning Based on Experience

Kai Welke[1], Peter Kaiser[1], Alexey Kozlov[1], Tamim Asfour[1],
Mike Lewis[2], Mark Steedman[2]
[1]Institute for Anthropomatics, Karlsruhe Institute of Technology
Adenauerring 2, 76131 Karlsruhe, Germany
[2]School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh, EH8 9AB, Scotland, United Kingdom
{welke,asfour}@kit.edu, {steedman}@inf.ed.ac.uk

Abstract— **Providing autonomous humanoid robots with the abilities to react in an adaptive and intelligent manner involves low level control and sensing as well as high level reasoning. However, the integration of both levels still remains challenging due to the representational gap between the continuous state space on the sensorimotor level and the discrete symbolic entities used in high level reasoning. In this work, we approach the problem of learning a representation of the space which is applicable on both levels. This representation is grounded on the sensorimotor level by means of exploration and on the language level by making use of common sense knowledge. We demonstrate how spatial knowledge can be extracted from these two sources of experience. Combining the resulting knowledge in a systematic way yields a solution to the grounding problem which has the potential to substantially decrease the learning effort.**

I. INTRODUCTION AND RELATED WORK

Establishing robotic systems that offer a level of autonomy suitable for real world applications requires bringing together expertise and approaches from a variety of different research fields. One of the most challenging problems therewith consists in integrating high level artificial intelligence (AI) with low level robot control.

The main challenge arises form the representational discontinuity between the continuous state spaces of robot control and the discrete symbolic representation used in most AI approaches. In order to fill-in this representational gap, the concept of object-action complexes (OACs) has been proposed as representation for all levels of the processing hierarchy ([1]). The OAC follows the affordance concept and tightly couples perception and action within a single representation. Applications of the OAC concept on several levels of the hierarchy including high level planning have been demonstrated ([2], [3]).

Bridging the gap between low level and high level processing requires defining a path from the continuous world to the symbolic representation and vice versa. While the OAC formalization takes into account the major processes within a cognitive architecture such as learning, predicting, and execution, the underlying structure in terms of appropriate state spaces needs to be defined in a problem specific way.

In this work, we focus on deriving representations of the spatial domain enabling the connection of high level planning with the sensori-motor level in humanoid robots. The need of such a representation can be visualized by considering the following action defined within a PDDL [4] domain specification:

```
(:action putdown
 :parameters ( ?x ?y ?z )
 :precondition (and (inHand ?x ?z)
                    (hand ?z)
                    (location ?y)
                    (graspable ?x))
 :effect (and (handEmpty ?z )
              (at ?x ?y)
              (not(inHand ?x ?z))))
```

The action `putdown` describes the process of putting an object `?x` held in the hand `?z` to a location `?y`. Two properties of this action render it a good example for the proposed work: First, the putdown action is required in several assistance tasks such as setting the table or stowing away. Second, the action has a direct reference to the spatial domain by means of the location `?y`. The spatial parameter `?y` appears in the binary predicate `at ?x ?y` which is necessary to describe the effect on the world state.

The goal now consists in establishing a representation for the parameter `?y` which is valid on the semantic as well as on the sensori-motor level. On the bottom-up path, this representations needs to support the observation of the world change triggered by the action in terms of the predicate `at ?x ?y`. On the top-down path, the execution of the action needs to be parameterized with the appropriate spatial location from the continuous domain in order to achieve this world change.

The simultaneous task and motion planning (STAMP) field tackles the problem of combining task planning and metric level. The goal consists in combining task and collision-free motion planning in a consistent fashion ([5],[6],[7],[8]). In contrast to STAMP, where full knowledge of the metrics and geometry as well as full knowledge of the task planning domain is assumed, our research focus lies on exploiting experience to improve the learning process of such representations.

Semantic information in spatial representations has been exploited in semantic maps of the environment in the navigation and mapping field [9]. Such semantic maps usually describe topological relations between semantic places in

the environment. Either these places are directly perceivable using appropriate models or detectors (e.g. [10]) or inferred using a known model of the binding from semantics to detectable places ([11], [12], [13], [14]).

In contrast to these approaches, the idea of exploiting experience in order to learn these semantic bindings of places stands at the core of our approach. Without the use of detectors for places, we make use of available common sense knowledge in order to establish the binding between semantics and the explored metrical representation. The extracted common sense knowledge is transferable from one environment to another and thus provides a consistent binding to the symbolic world. This transferability is essential for establishing task planning across large domains.

II. THE SYSTEM CONCEPT

A. Conceptual assumptions

The focus of this work lies in the acquisition of grounded spatial representations from experience. Obviously this spatial knowledge is only a small fraction of the overall knowledge required to operate the system in an autonomous way. In order to clearly outline our approach, we assume prior knowledge to be present in several forms on the system. The prior knowledge assumed in this work includes

- Object knowledge: We assume extensive prior knowledge on objects in the world. This knowledge includes object models for recognition and localization as well as the associated class labels. More precisely, we know the models and class labels for common manipulable kitchen objects such as cups, plates, milk, or juice available from the KIT object model database [15].
- Action knowledge: We assume that the robotic platform is able to perform basic actions. We explicitly make use of the grasp and putdown actions during the exploration phase. Further, locomotion abilities are necessary in order to allow learning of larger scale spatial domains.

In order to apply the gathered knowledge in task planning, it is necessary to have full knowledge of the planning domain. In order to execute the plan, a sensori-motor representation of all involved actions and all predicates needs to be available. All non-locational constants, such as class labels of objects, need to be grounded on the sensori-motor level. The rules in terms of pre- and postconditions of actions have to be known, but can also be learned by exploration [16]. The missing piece, the combined sensori-motor representation of locations and the associated symbolic constant is learned by our approach.

B. System architecture

While addressing a quite specific problem of symbol grounding for task planning in this work, we do not consider this problem isolated but within the context of a systematic way of coupling high level task planning with the sensori-motor level on humanoid robots. For this purpose, we developed and implemented an architecture that couples these levels. Relevant components within this architecture are illustrated in Fig. 1, where the focus lies on how sensori-motor
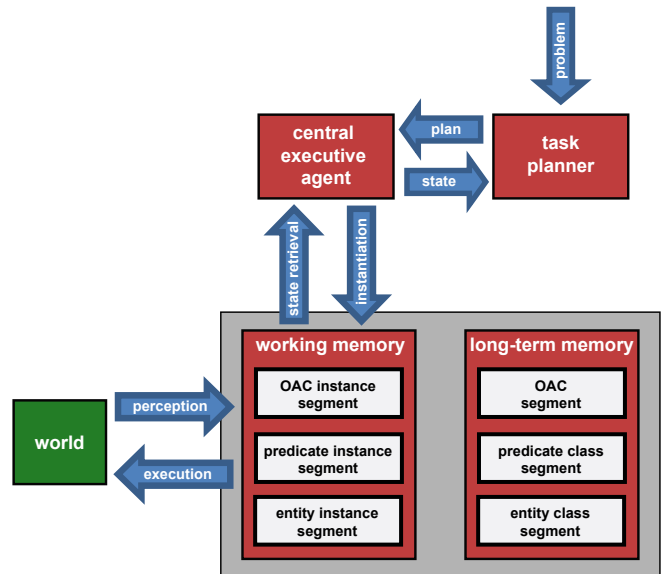

Fig. 1: The integration between the task planning level and the sensori-motor level is established by the central executive agent (CEA). For each plan element the CEA instantiates the appropriate OAC and associated predicates and entities. Entities correspond to constants on the planning level. The goal of this work consists in learning an entity of type `location` from experience.

representations are made available for planned task execution and plan monitoring. The major involved components are the task planner, the central executive agent (CEA) and the memory system. In our current implementation we make use of the PKS planner for STRIPS like task planning with the support of plan monitoring ([17], [18]). The CEA is the mediator between planning level and memory system. Its purpose is bidirectional: Bottom-up it translates the content of the working memory (WM) to a world state in PDDL format in order to enable planning and plan monitoring. Top-down the CEA translates plans from the PDDL domain to OAC representations. In a sequential manner it instantiates OACs from the long-term memory (LTM) in the WM according to the active plan element. Further, in order to allow state monitoring, it creates instances of required WM elements associated with the OAC. Such elements include unary, binary, or n-ary predicates as required for the PDDL world state description and entities which correspond to constants on the planning level such as objects and locations. The underlying execution and perception mechanisms take care of keeping these constants and predicates consistent with the real world.

Taking into account this architecture the goal now consists in learning the representation of an entity class for locations which can be used as constant of type `location ?y` and can be used with the predicate `at ?x ?y`. While the class stored in LTM should be valid for different tasks and objects, the instantiation of this entity in the WM is specific for the current task and object.

III. ACQUIRING SPATIAL KNOWLEDGE FROM EXPERIENCE

The acquisition of spatial knowledge from experience has several advantages over resorting to manually generated spatial representations. The most important benefit lies in the improvement of the system's autonomy by establishing the required processes for acquiring such representations on the robot.

In this work, we exploit two different sources of experience: experience gathered through exploration on the robot system and experience available in common sense knowledge. The exploration on the robot yields embodied sensory-level representations that already encode the constraints of the platform such as the visibility of objects. Spatial information from common sense knowledge on the other hand is extracted from large text corpora and thus provides knowledge on the symbolic level. In the following, we introduce approaches that make available both sources of knowledge for the acquisition of spatial knowledge. We will show how to combine the gathered knowledge in order to acquire grounded spatial symbols in Section IV.

A. Spatial knowledge from exploration

The goal of the exploration consists in incrementally learning a spatial model of the environment with respect to the set of known objects. More precisely, the developed approach allows inferring common object locations based on object detection and localization results from multiple episodes. In order to keep the exploration effort low, we employ self-observation of the robot. While the robot is controlled through human interaction in our kitchen scenario, it records all encountered and manipulated objects with location, label, and current task.

1) Metric spatial representation: In order to represent the encountered objects, we employ probabilistic and continuous space representations, which are similar to those proposed by Stulp et el. in their concept of ARPlace [19]. More precisely, the object positions are described by a probability density function (PDF) in 3D space. This approach allows avoiding a prior space discretization, while simultaneously providing a natural way to incorporate object localization uncertainty. To represent the learned object position PDF, we use the Gaussian Mixture Model (GMM):

$$f(\vec{x}) = \sum_{i=1}^{N} w_i \mathcal{N}(\vec{x}; \vec{\mu_i}, \vec{\Sigma_i}), \qquad \forall w_i > 0 \qquad (1)$$

The GMM has an important property of being a universal PDF approximator [20], which means that it can approximate any given distribution with an arbitrary precision. From a practical point-of-view, GMM is of particular interest because of its ability to cope with multi-modality and moderate storage requirements.

2) Learning common locations: Each time an object is encountered in the world, the spatial representation is updated. Initially, we start with an empty GMM η. The observed object position is modeled as a Gaussian $\mathcal{N}(\vec{x}; \vec{\mu_o}, \Sigma_o)$
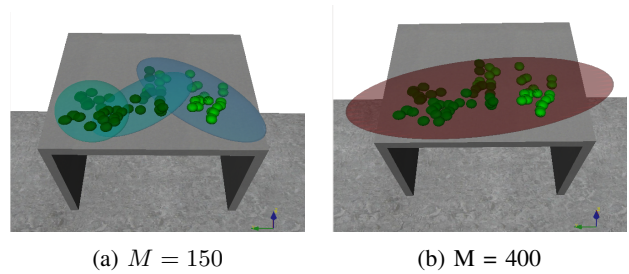

(a) $M = 150$ (b) M = 400

Fig. 2: Different clusterings of the same position distribution achieved by setting *(a)* low and *(b)* high value of the deviation threshold M.

encoding the localization uncertainty in 3D Cartesian space [21]. This Gaussian is added as a new component with a constant weight (e.g., 1) to the GMM: $\eta \leftarrow \eta \cup (1, \vec{\mu_o}, \Sigma_o)$. At the same time, the following three operations are applied to the existing components:

- **Aging** Since older observations are assumed to be less relevant than the recent ones, the weights of corresponding GMM components are reduced by multiplying with the discount coefficient $\gamma \in [0, 1]$:

$$\forall i \ w_i \leftarrow \gamma \cdot w_i \qquad (2)$$

- **Pruning** Components with weights below the threshold W_{prune} are removed from the mixture:

$$\forall i: \ w_i < W_{prune} \quad \eta \leftarrow \eta \ \setminus \ (w_i, \vec{\mu_i}, \Sigma_i) \quad (3)$$

- **Merging** two components which are considered "similar" in terms of distance d are replaced with their *moment-preserving merge*:

$$d(i,j) < D_{min}: \ \eta \leftarrow \eta \ \setminus \ (w_i, \vec{\mu_i}, \Sigma_i), (w_j, \vec{\mu_j}, \Sigma_j)$$
$$\eta \leftarrow \eta \ \cup \ (w_m, \vec{\mu_m}, \Sigma_m) \qquad (4)$$

The calculation of $(w_m, \vec{\mu_m}, \Sigma_m)$ is performed according to [22].

The resulting representation encodes the spatial distribution of common object locations. The aging factor accounts for changes in the scene, while pruning and merging keep the representation compact. An example for common locations on the table is illustrated in Fig. 2.

3) Querying: To make spatial knowledge accessible, it should be provided at a suitable abstraction level for the task at hand. For this purpose, we implemented a query interface which allows for two types of generalization:

- **Spatial generalization** Spatial generalization allows combining several neighbored observation to a single cluster. For this purpose, three established GMM reduction algorithms were implemented (West [23], Runnalls [24], Williams [22]). The level of generalization can be adjusted by specifying the stop condition of the GMM reduction. The stop condition is either defined by a target number of clusters or by the maximum
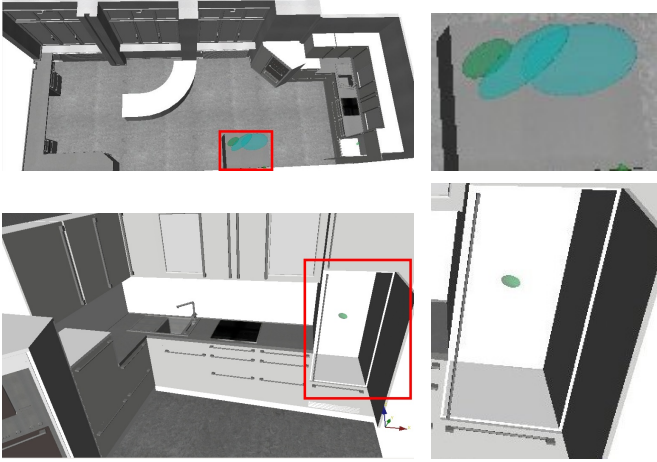
Fig. 3: Common places learned from four ARMAR-III kitchen demonstrations after spatial generalization.

deviation M within a cluster. An example of the spatial generalization is illustrated in Fig. 2.

- **Ontological generalization** In addition to positions of objects (e. g. Cup), places for abstract classes (e.g. Food) can be queried as well. This is achieved by using a simple class ontology with parent-child relations.

4) Common places in the kitchen domain: The learning algorithm and clustering approach described above were applied on object locations collected during the demonstrations of the humanoid robot ARMAR-III ([25], [26]) in the kitchen. In the scenario, the robot localized and manipulated objects in the fridge and on the table. Figure 3 illustrates a spatial generalization query on the representation resulting from four ARMAR-III demonstrations.

B. Spatial relations from human knowledge

Besides learning from the robot's own experience, we would like to gain information on spatial relations from human knowledge. Human knowledge could tell the robot that *milk* is usually kept in refrigerators. Hence, there is a certain probability that a spatial cluster containing positions of *milk* is a refrigerator.

In this section we propose a method to infer a set of likely locations for a given object. In order to reduce the amount of prior knowledge in our system, we do not use common sense databases like *Open Mind Common Sense* [27] that were explicitly created with artificial intelligence in mind, but prefer to extract the relevant relations from text sources written by humans for humans [?].

1) Extracting spatial relations from text: Spatial relations are linguistically expressed using spatial prepositions:

- *The milk is **in** the refrigerator*
- *Take a knife **from** the drawer*.

In this work, we propose to determine the conditional probability of a location given an object based on the number of matching prepositional relations in a text corpus. We are aiming for prepositions like *in* and *on*, but do not predefine a set of valid prepositions.
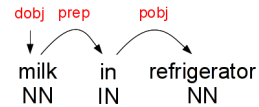

Fig. 4: A syntactic ngram containing two content-words and a preposition. The words are equipped with a part-of-speech tag and a dependency label[1].

Let N_{obj} be the frequency of occurrence of obj in prepositional contexts and let $N_{obj,loc}$ be the number of those prepositional contexts where obj and loc occurr together. The conditional probability $P(loc|obj)$ can then be approximated as follows:

$$P(loc|obj) = \frac{P(obj, loc)}{P(obj)} \approx \frac{N_{obj,loc}}{N_{obj}} \tag{5}$$

Working on the whole vocabulary of the corpus makes the values of the conditional probability difficult to compare. As we know the set L of possible locations in the kitchen from the planning domain specification, we can formulate the restricted conditional probability:

$$P_L(loc|obj) = \frac{P_L(obj, loc)}{P_L(obj)} \approx \frac{N_{obj,loc}}{\sum_{l \in L} N_{obj,l}} \tag{6}$$

These formulas imply the assumption that a text corpus is a suitable foundation for estimating $P(loc|obj)$. See section III-B.4 for a discussion.

2) The Text Corpus: In this paper we propose to extract spatial relations from the *Google Books Ngrams Corpus* [28], in the following referred to as the *Google Corpus*. This corpus contains a representation of 3.5 million English books with a total size of about 345 billion words. It does not contain the raw text. Several preprocessing steps have been applied to the sentences:

1) Parsing into dependency trees
2) Extracting *syntactic ngrams*, i.e. n content-words long subpaths of the dependency trees (see Fig. 4)
3) Counting the frequency of occurrence of each syntactic ngram

We are using the corpus in its *arcs*-variant, which only includes syntactic ngrams with two content-words plus possible non-content-words like prepositions or conjunctions. Overall, the preprocessing makes the Google Corpus convenient for conducting analysis on the frequency of grammatical structures.

In the Google Corpus, each syntactic ngram is stored in a distinct line. The information that is relevant in this paper is the ngram itself and its frequency of occurrence. The entry for the exemplary ngram in Fig. 4 looks as follows:

milk/NN/dobj/0 in/IN/prep/1 refrigerator/NN/pobj/2 160

The syntactic ngram's path consists of three nodes, each containing the following relevant fields:

---

[1] *NN* - noun, *IN* - preposition
dobj - direct object, *prep* - preposition, *pobj* - prepositional object

TABLE I: Restricted conditional probability $P_L(loc|obj)$. Darker colors indicate higher probabilities, omitted probabilities are zero.

	cellar	counter	cupboard	dishwasher	drawer	freezer	microwave	oven	refrigerator/fridge	shelf	table
beer	0.0763	0.0518	0.0095						0.6045		0.2579
bread	0.0033	0.0235	0.0515		0.0017	0.0065		0.2566	0.0046	0.0181	0.6343
cereal			0.4045							0.1685	0.4270
coffee	0.0076	0.1458	0.0108				0.0120	0.0089		0.0203	0.7945
cup		0.0728	0.0337	0.0029	0.0066	0.0025	0.0172	0.0278	0.0059	0.0405	0.7901
dough		0.1293				0.0376		0.2674	0.3834		0.1823
juice	0.0190	0.0306				0.0146	0.0146	0.0889	0.6064		0.2259
knife		0.0723			0.2752			0.0089	0.0036	0.0197	0.6203
meat	0.0194	0.0180	0.0143			0.0680	0.0132	0.1180	0.1309	0.0028	0.6154
milk	0.0275	0.0299	0.0141			0.0255	0.0319	0.1054	0.3832	0.0238	0.3586
pot	0.0103	0.1154	0.0291					0.1195	0.0139	0.0734	0.6385
wine	0.4128	0.0061	0.0144					0.0112	0.0387	0.0050	0.5119

- The word that the node represents in the original sentence (e.g. *milk*).
- The Penn-Treebank part-of-speech tag [29] for this word (e.g. *NN*).
- The basic Stanford-dependencies label [30] for the node's grammatical function (e.g. *dobj*).

The final number is the frequency of occurrence of the syntactic ngram. This exemplary ngram occurred 160 times in the Google Corpus.

3) Extracting Relations from the Corpus: We are interested in extracting prepositional relations between objects and locations. Referring to the above exemplary line from the Google Corpus, we are looking for lines that match the following pattern:[2]

$$[object]/\text{NN}/\bullet/\bullet \quad \bullet/\bullet/\text{prep}/\bullet \quad [location]/\text{NN}/\bullet/\bullet. \qquad (7)$$

After searching and accumulating prepositional contexts with regard to pattern (7), the probability of a location given an object can be approximated using (6).

4) Evaluation: Table I shows the restricted conditional probabilities $P_L(loc|obj)$ as defined in (6) of a set of objects (y-axis) given a predefined set of possible locations (x-axis). The table shows that the proposed method of extracting prepositional contexts from a text corpus is able to infer reasonable values for $P_L(loc|obj)$. Exemplary conclusions that can be drawn from the results include:

- Refrigerators are a likely location for beer, juice and milk.
- Cups and coffee may be found on tables.
- Apart from the oven, bread could be on a table or in a cupboard.

IV. GROUNDED SPATIAL SYMBOLS

In this section we will outline how symbols for locations can be obtained which are grounded in language as well as in the continuous domain. For this purpose, the two sources of experience introduced in the previous section, exploration and common sense knowledge, are combined. In the following we will show how the predicate `at ?x fridge` can be inferred from experience. Since we assumed to have a representation of all involved objects on the sensory
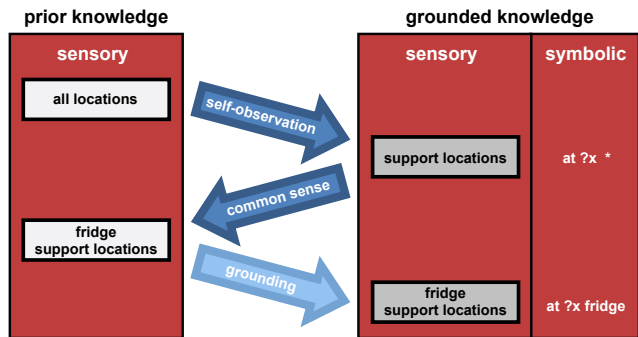

Fig. 5: The proposed approach for acquiring grounded spatial symbols combines two sources of experience: Self-observation of the robot during the execution of kitchen tasks yields grounded representations of support locations. These support locations are associated with semantic symbols by exploiting common sense knowledge.

as well as on the symbolic level, we know all enumerations of the parameters `?x` and the associated object models. The problem of evaluating the above predicate then boils down to inferring a grounded representation of the location constant `fridge`. Note that the constant `fridge` does not refer to an object but to the support locations offered by the fridge in terms of the `at` predicate as defined in our PDDL definition of the putdown action in section I.

The greedy acquisition of a representation of the fridge locational constant grounded in the spatial domain would require providing a large set of spatial locations corresponding to the fridge together with the symbolic tag. Collecting this data on the robot either requires the involvement of a teacher, or the evaluation of the symbolic binding via higher level inference (e.g. by exploiting knowledge such as temperature in the fridge) in order to assert the validity of the symbolic tag. Such grounding processes are slow and costly in terms of resources. By exploiting experience, a good prior for such constants can be achieved, substantially decreasing the grounding effort.

The complete chain of acquiring a grounded representation of `fridge` from experience is illustrated in Fig. 5. The only prior knowledge at the start of the process consists in a
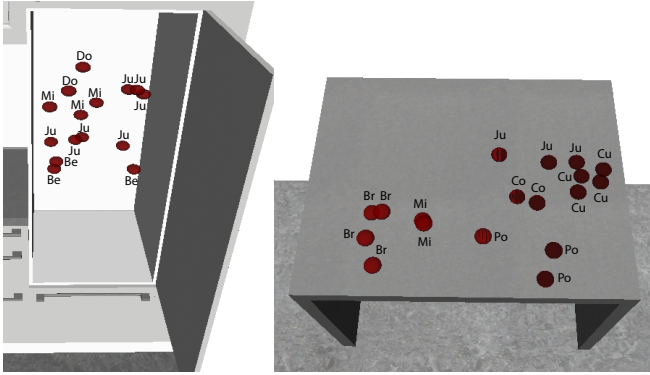
---

[2]"$\bullet$" denotes a wildcard

Fig. 6: Simulation of the exploration phase with the following objects: **be**er, **br**ead, **co**ffee, **cu**p, **do**ugh, **ju**ice, **mi**lk, **po**t. The pickup and putdown poses were chosen at random in the fridge and on the table.

common reference frame for spatial locations that defines the space of all possible locations. Through self-observation and by applying common sense knowledge, we calculate a prior for fridge support locations. The final grounding step again involves means of ascertaining the gained representation similar to the greedy approach but with a prior that eases the grounding process. In the following, the single steps towards this prior are discussed in detail.

A. Exploration of grounded support locations

The first step in our approach consists in determining grounded support locations in the environment by exploration. This exploration is realized by means of self-observation as introduced in Section III-A. The resulting locations collected during the execution of tasks on the robot are stored using the proposed metric spatial representation. Thereby, each experienced location is accompanied with the label of the occupying object and the action that has been executed on the object. In order to extract support locations from this data, only these locations are considered which correspond to the actions pickup or putdown. Since the support locations are collected during self-observation, the resulting representation naturally includes reachability constraints of the experimental platform. With this representation, we acquired a grounded concept of support locations for the covered objects ?x denoted with at ?x * in Fig. 5. An example of such a representation based on simulated data is illustrated in Fig. 6.

B. Common sense knowledge for symbol binding

In the second step of our approach, we employ common sense knowledge in order to establish a symbolic binding for support locations within the representation explored in the previous step. In our example, the goal consists in inferring a good prior for at ?x fridge based on the explored at ?x *. The major challenge here consists in establishing a representation of the location constant fridge that is grounded in the sensory as well as in the language domain. Here we exploit that spatial relations between objects and

locations are part of human common sense knowledge and are accessible on the linguistic level in terms of prepositional structures as detailed in Section III-B. For each location stored in the representation from step one we can conclude the probability of belonging to a fridge location by accessing the associated object class label and querying the common sense knowledge encoded in Table I. By assuming that locational symbols stay constant in the local neighborhood (e.g. neighbored locations of a fridge location are also likely to be fridge locations), evidence for the symbol can be propagated from location to location. This can be efficiently implemented by making use of the spatial generalization query introduced in Section III-A.3. For each resulting soft cluster, the associated likelihood of belonging to the location fridge is collected over all object locations belonging to the cluster by means of a linear opinion pool. Figure 7 illustrates the result of this process for the simulated data from Fig. 6. Using an appropriate deviation threshold for spatial generalization yields two clusters: one in the fridge and one on the table. The probability of the fridge cluster belonging to the constant fridge is calculated with the above procedure and amounts to $P(Fridge|O1) = 0.53$. The same approach for the table cluster yields $P(Fridge|O2) = 0.15$. The application of common sense knowledge in this way allows exploiting negative examples. For instance, clusters that contain a high frequency of cups are not likely to be located in the fridge according to Table I.

Associating the explored objects with location symbols by exploiting common sense knowledge yields the representation we were seeking: a representation of the locational constant fridge which is grounded within the metric spatial representation. This representation can be used to establish the world state in terms of the predicate at ?x fridge. For objects ?x that lie in the fridge cluster this predicate is valid. Further, the explored locations associated with the cluster can be used to parameterize the action putdown ?x fridge ?z. This process of course involves thresholding the probability values in Fig. 7 which might not be trivial in all cases. Nevertheless, we achieved a good prior for the symbolic binding which allows quite efficient disambiguation in further grounding processes.

V. DISCUSSION

A. Contributions

In this work we presented an approach for learning a representation of space applicable on the task planning as well as on the sensori-motor level. In order to establish a symbolic link to the continuous world, we exploited two sources of experience: experience from exploration and experience from common sense knowledge.

While this work developed the approach in a quite exemplary manner, based on the fridge example, the general concept is applicable to most typical places in human made environments. The common sense knowledge provides transferable concepts for places. Considering the table location in Table I implies that the table is a quite versatile support surface. Based on this knowledge we would assign the
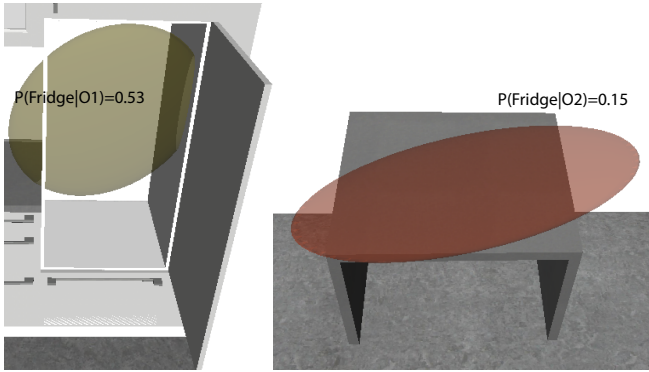
Fig. 7: Result of spatial generalization with deviation threshold $M = 400$. The cluster in the fridge $O1$ has a much higher probability of being the fridge as the cluster $O2$ on the table. This result is achieved only based on the observation of objects and the extraction of prepositional contexts.

TABLE II: Most frequent prepositions for locations

	first	second
cellar	in - 69%	from - 27%
counter	on - 81%	at - 9%
cupboard	in - 71%	from - 29%
dishwasher	in - 100%	
drawer	in - 86%	from - 14%
freezer	in - 96%	from - 4%
microwave	in - 100%	
oven	in - 81%	from - 9%
refrigerator/fridge	in - 80%	from - 20%
shelf	on - 95%	from - 5%
table	on - 73%	at - 9%

symbol `table` to places which are used to support numerous different objects, independent of the current domain.

The application of common sense knowledge was demonstrated on simulated data in order to show the feasibility of the approach. The same could be done on the real data collected through exploration. However, the real kitchen data only covered the object juice in the fridge, which would result in a quite simple query to the common sense knowledge (e.g. $P(Fridge|O1) = 0.6$). Further, not all objects on the table had a significant amount of occurrences in the corpus. This stems from the fact, that often classes are used in language instead of single instances of the object (e.g. cereal vs. vitalis cereal). This problem could be addressed by means of the ontological generalization as explained in Section III-A.

B. Outlook

So far, we used the predicate `at` in order to express that an object is at a specific location. However, this preposition is not quite common and would probably not be applied in order to describe a location in the fridge. Rather, we would use prepositions which also encode the function of the location such as `in`, `on`, or `from`. In the proposed approach we used all prepositions to query for locations in order to get a significant amount of occurrences of a location independent of its function. In task planning however, there can be a huge

difference between putting something on a place or in a place (e.g. open the door before putting in). This problem can be addressed by making use of the corpus again. As can be seen in Table II the query yields the correct prepositions for fridge and table and thus also allow to infer this functional aspect from common sense knowledge.

REFERENCES

[1] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrčen, A. Agostini, and R. Dillmann, "Object-action complexes: Grounded abstractions of sensorimotor processes," *Robotics and Autonomous Systems*, vol. 59, pp. 740–757, 2011.

[2] C. Geib, K. Mourao, R. Petrick, N. Pugeault, M. Steedman, N. Krueger, and F. Wörrgötter, "Object action complexes as an interface for planning and robot control," in *IEEE International Conference on Humanoid Robots (Humanoids)*, 2006.

[3] R. Petrick, D. Kraft, K. Mourão, N. Pugeault, N. Krüger, and M. Steedman, "Representation and Integration: Combining Robot Control, High-Level Planning, and Action Learning," in *International Cognitive Robotics Workshop (CogRob 2008) at ECAI 2008*, 2008, pp. 32–41.

[4] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "Pddl—the planning domain definition language," New Haven, CT: Yale Center for Computational Vision and Control, Tech. Rep., 1998.

[5] S. Cambon, R. Alami, and F. Gravot, "A hybrid approach to intricate motion, manipulation and task planning," *The International Journal of Robotics Research*, vol. 28, no. 1, pp. 104–126, 2009.

[6] J. Wolfe, B. Marthi, and S. Russell, "Combined task and motion planning for mobile manipulation," in *International Conference on Automated Planning and Scheduling*, Toronto, Canada, 05/2010 2010.

[7] L. Kaelbling and T. Lozano-Perez, "Hierarchical task and motion planning in the now," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1470–1477.

[8] E. Plaku, "Planning in discrete and continuous spaces: From ltl tasks to robot motions," in *Advances in Autonomous Robotics*, ser. Lecture Notes in Computer Science, G. Herrmann, M. Studley, M. Pearson, A. Conn, C. Melhuish, M. Witkowski, J.-H. Kim, and P. Vadakkepat, Eds. Springer Berlin Heidelberg, 2012, vol. 7429, pp. 331–342.

[9] B. Kuipers, "The spatial semantic hierarchy," *Artificial Intelligence*, vol. 119, pp. 191–233, 2000.

[10] A. Pronobis, O. M. Mozos, B. Caputo, and P. Jensfelt, "Multi-modal semantic place classification," *The International Journal of Robotics Research, Special Issue on Robotic Vision*, vol. 29, no. 2-3, pp. 298–320, Feb. 2010.

[11] A. Nüchter, H. Surmann, K. Lingemann, and J. Hertzberg, "Semantic scene analysis of scanned 3d indoor environments," in *Eighth International Fall Workshop on Vision, Modeling, and Visualization*, 2003.

[12] O. M. Mozos, P. Jensfelt, H. Zender, G.-J. M. Kruijff, and W. Burgard, "From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots," in *IEEE International Conference on Robotics and Automation (ICRA) Workshop: Semantic information in robotics*, 2007, pp. 33–40.

[13] C. Galindo, J.-A. Fernández-Madrigal, J. González, and A. Saffiotti, "Robot task planning using semantic maps," *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 955–966, Nov. 2008.

[14] P. Viswanathan, D. Meger, T. Southey, J. Little, and A. Mackworth, "Automated spatial-semantic modeling with applications to place labeling and informed search," in *Canadian Conference on Computer and Robot Vision*, 2009, pp. 284–291.

[15] A. Kasper, Z. Xue, and R. Dillmann, "The kit object models database: An object model database for object recognition, localization and manipulation in service robotics," *The International Journal of Robotics Research*, vol. 31, no. 8, pp. 927–934, 2012.

[16] K. Mourão, L. S. Zettlemoyer, R. P. A. Petrick, and M. Steedman, "Learning strips operators from noisy and incomplete observations," in *Uncertainty in Artificial Intelligence (UAI)*, 2012.

[17] R. P. A. Petrick and F. Bacchus, "A knowledge-based approach to planning with incomplete information and sensing," in *International Conference on Artificial Intelligence Planning and Scheduling (AIPS-2002)*, M. Ghallab, J. Hertzberg, and P. Traverso, Eds. Menlo Park, CA: AAAI Press, Apr. 2002, pp. 212–221.

[18] ——, "Extending the knowledge-based approach to planning with incomplete information and sensing," in *International Conference on Principles of Knowledge Representation and Reasoning (KR-2004)*, D. Dubois, C. Welty, and M.-A. Williams, Eds. Menlo Park, CA: AAAI Press, June 2004, pp. 613–622.

[19] F. Stulp, A. Fedrizzi, L. Msenlechner, and M. Beetz, "Learning and Reasoning with Action-Related Places for Robust Mobile Manipulation," *Journal of Artificial Intelligence Research (JAIR)*, vol. 43, pp. 1–42, 2012.

[20] V. Maz'ya and G. Schmidt, "On approximate approximations using gaussian kernels," *IMA Journal of Numerical Analysis*, vol. 16, pp. 13–29, 1996.

[21] K. Welke, "Memory-based active visual search for humanoid robots," Ph.D. dissertation, Karlsruhe Institute of Technology (KIT), Computer Science Faculty, Institute for Anthropomatics (IFA), 2011.

[22] J. Williams and P. Maybeck, "Cost-function-based gaussian mixture reduction for target tracking," in *Sixth International Conference of Information Fusion*, vol. 2, 2003, pp. 1047–1054.

[23] M. West, "Approximating posterior distributions by mixtures," *Journal of the Royal Statistical Society (Ser. B)*, 1993.

[24] A. Runnalls, "Kullback-leibler approach to gaussian mixture reduction," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 43, no. 3, pp. 989–999, 2007.

[25] T. Asfour, K. Regenstein, P. Azad, J. Schröder, N. Vahrenkamp, and R. Dillmann, "ARMAR-III: An integrated humanoid platform for sensory-motor control," in *IEEE International Conference on Humanoid Robots (Humanoids)*, 2006, pp. 169–175.

[26] T. Asfour, P. Azad, N. Vahrenkamp, K. Regenstein, A. Bierbaum, K. Welke, J. Schröder, and R. Dillmann, "Toward humanoid manipulation in human-centred environments," *Robotics and Autonomous Systems*, vol. 56, no. 1, pp. 54–65, 2008.

[27] P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu, "Open mind common sense: Knowledge acquisition from the general public," in *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE.* Springer, 2002, pp. 1223–1237.

[28] Y. Goldberg and J. Orwant, "A dataset of syntactic-ngrams over time from a very large corpus of english books," in *Second Joint Conference on Lexical and Computational Semantics*, 2013, pp. 241–247.

[29] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a Large Annotated Corpus of English: The Penn Treebank," *Computational linguistics*, vol. 19, pp. 313–330, 1993.

[30] M.-C. de Marneffe and C. D. Manning, "Stanford typed dependencies manual," 2008.

Gaze Selection during Manipulation Tasks

Kai Welke, David Schiebener, Tamim Asfour and Rüdiger Dillmann

Karlsruhe Institute of Technology (KIT)

Institute for Anthropomatics (IFA)

{welke,schiebener,asfour,dillmann}@kit.edu

Abstract— A major strength of humanoid robotics platforms consists in their potential to perform a wide range of manipulation tasks in human-centered environments thanks to their anthropomorphic design. Further, they offer active head-eye systems which allow to extend the observable workspace by employing active gaze control. In this work, we address the question where to look during manipulation tasks while exploiting these two key capabilities of humanoid robots.

We present a solution to the gaze selection problem, which takes into account constraints derived from manipulation tasks. Thereby, three different subproblems are addressed: the representation of the acquired visual input, the calculation of saliency based on this representation, and the selection of the most suitable gaze direction. As representation of the visual input, a probabilistic environmental model is discussed, which allows to take into account the dynamic nature of manipulation tasks. At the core of the gaze selection mechanism, a novel saliency measure is proposed that includes accuracy requirements from the manipulation task in the saliency calculation. Finally, an iterative procedure based on spherical graphs is developed in order to decide for the best gaze direction. The feasibility of the approach is experimentally evaluated in the context of bimanual manipulation tasks on the humanoid robot ARMAR-III.

I. INTRODUCTION

The anthropomorphic design of humanoid robots makes these platforms most suitable for manipulation tasks in human-centered environments and facilitates human-robot interaction. The integration of active head-eye systems in such platforms is a direct consequence of the anthropomorphic design. While the application of head and eye movements to fixate, to saccade, or to perform smooth pursuit plays an important role in interaction, there is also a technical benefit in using active systems. In contrast to passive camera systems where an increase in the field-of-view leads to a loss in the resolution of details, the application of active systems allows to increase the observable area while keeping the details. During manipulation this behavior is desirable, especially if the task involves multiple objects which are spatially distributed. Fig. 1 illustrates such a setup, where the goal of a bimanual manipulation task is pouring of juice into a glass. By fixating the objects sequentially using active gaze control, almost the full camera resolution is available to perform reliable object recognition and pose estimation.

The classical approach to solving complex manipulation tasks involves the sense-plan-act scheme. Thereby, entities of the world are visually captured in an internal representation which is then used as basis for action sequencing and motion planning followed by the execution of the resulting trajectories. On humanoid robots such an approach bears several

Fig. 1. ARMAR-IIIa [1] performing bimanual visual servoing in order to pour juice into a cup. During this task, both target objects as well as the hands of the robot need to be observed by the visual system. Due to the large workspace, the selection of appropriate gaze directions is required in order to cover all task relevant objects and thus allow successful execution of the task.

problems that render its applicability in real world scenarios difficult. First, the complexity of humanoid platforms leads to divergence between the kinematic and dynamic model and the real execution. Thus, the planned trajectories are usually not executed with the required accuracy. Second, the internal representation of the scene is also affected by inaccuracies. These stem on the one hand from noisy measurements of the perception and on the other hand from unpredictable behavior in unconstrained environments. Consequently, in order to achieve robust execution of manipulation tasks, a continuous adaptation of the internal representations is favorable over a sense-plan-act approach.

In order to take into account the inaccuracies of perception and execution, a common approach consists in formulating the processes involved in the overall task in a probabilistic fashion [2]. The derivation of an internal representation then becomes a probabilistic inference problem where appropriate models for uncertainty in the perception and execution processes need to be provided. In our approach, the environment is represented using a spatial environmental model of all entities involved in the manipulation task. Each entity corresponds to an object in the real world, i.e. cups, juice boxes, or hands of the robot, and is accompanied with uncertainties

about its current state. Appropriate observation models for the visual perception are introduced and applied in a data fusion scheme thus reducing the amount of uncertainty over time. Further, the motion of all objects is predicted in order to account for the dynamic nature of manipulation tasks. This is essential since e.g. the arms of the robot move during most manipulation tasks.

During a manipulation task, the different world entities compete for being fixated by the active camera system. Each fixation of an entity allows a more accurate representation of its state within the environmental model by fusing the new observation with the past sensor data. Depending on the manipulation task the requirements on the accuracy of the environmental model might differ significantly. While e.g. transportation tasks usually do not require a precise estimation of the object's position, other tasks such as grasping will fail if the estimated poses of the object and the robot hand are not accurate enough. Consequently, we propose to include these accuracy requirements in the gaze selection mechanism by fixating objects accordingly. In order to include this task specific guidance in the gaze selection strategy, we introduce the *task acuity* in the calculation of saliencies. By implementing a gaze selection mechanism on top of the task acuity, the active perceptual process can be configured in order to guarantee a specific accuracy for each element in the environmental model as suitable for the manipulation task.

This paper is organized as follows: In the next section, related work is discussed and the novel aspects of the proposed work in the paper are highlighted. Subsequently, in Section III, the proposed gaze selection mechanism is introduced including the environmental model, the saliency measure based on the task acuity and the decision for the most feasible gaze direction. The proposed mechanism is then put into the context of a bimanual manipulation task, and appropriate motion models are defined in Section IV. The achieved results are discussed in Section V, before the contribution of the proposed work is summarized in Section VI.

II. RELATED WORK

In the context of human visual processing, the problem of gaze selection is often referred to as overt visual attention. The most prominent computational model for visual attention has been proposed by Itti et al. ([3]) followed by several extensions (e.g. [4]) and implementations on robotic platforms (e.g. [5]). An extensive review of such approaches can be found in [6]. In contrast to this line of research, where the goal consists in mimicking the human visual attention processes, our work focuses on establishing a technically motivated approach which allows to support manipulation in a real world environment while making use of active camera systems.

Another line of research deals with active visual search, where the goal consists in detecting and recognizing objects in the extended observable area. Thereby, models for search targets are usually made available as cue for the search task

([7], [8]). Recently, active visual search has been extended by means of integrating a spatial memory that allows to fuse visual information over several gaze directions ([9], [10]). Further, in [11], the active visual search task is extended to a treasure hunting task involving not only gaze selection but also locomotion of the robot in order to detect the object. It has been shown that such systems already can be applied in manipulation tasks ([12]). Nevertheless, constraints arising from manipulation tasks are not taken into account. This applies to the requirement of a dynamic environmental model as well as to the inclusion of accuracy requirements in the saliency measure as proposed in our approach.

The competition for the limited resources of the visual perception system of humanoid robots stands at the core of several gaze selection approaches in the context of humanoid locomotion. During locomotion, usually at least two different perceptual tasks compete: the self-localization of the robot and the obstacle avoidance. In [13], a gaze selection mechanism is proposed which minimizes the self-localization uncertainty as well as the obstacle avoidance uncertainty. In [14], the authors approach the gaze selection problem in a RoboCup scenario, where self-localization, obstacle avoidance, and ball detection compete for the limited resources. Thereby, the environmental model allows for dynamic entities using occupancy grid mapping techniques. The gaze selection mechanism aims to reduce the uncertainty within the grid based representation. For this purpose, a saliency measure is proposed based on the Shannon entropy. Being tailored for locomotion, these approaches do not propose any mechanism to include constraints from manipulation tasks in the gaze selection.

Another possibility of calculating gaze sequences for a manipulation task consists in using the knowledge from a motion planning step. In [15], the knowledge from planning is used to determine the position of objects in the scene and thus adapt the gaze accordingly. In [16], the gaze direction is planned together with the robot motion under consideration of visibility constraints.

In contrast to these approaches and as motivated in the introduction, we seek to establish a gaze selection mechanism which can handle inaccurate or incomplete world knowledge. Thus, we propose an online approach to gaze selection in contrast to the offline calculation of gaze sequences based on the a-priori model during motion planning. Therefore, we establish an environmental model which is updated online and can handle dynamic world entities. This update is formulated as probabilistic inference process. Similar to [13] and [14], the goal of redirecting the gaze is then the reduction of uncertainty in this model. The main contribution of this work consists in the generation of a task specific gaze sequence by introducing the task acuity as saliency measure. The task acuity allows to configure the required accuracy for entities in the environmental model. Based on this saliency measure, an approach for selecting the optimal gaze direction during manipulation is introduced.
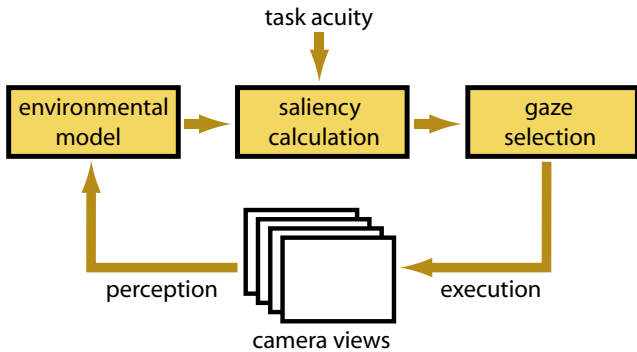
Fig. 2. The proposed approach generates gaze sequences in a perception-action loop. The processing chain includes fusion in the environmental model, calculation of the saliency under consideration of task constraints, and selection of the best gaze direction.
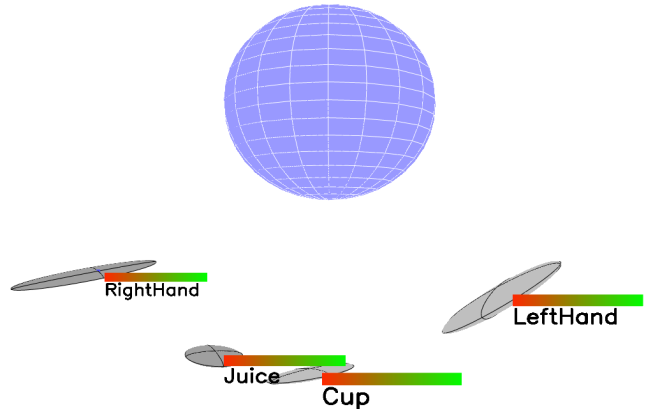

Fig. 3. The environmental model represents the current state estimation of the pose of objects relevant to the manipulation task in a fixed ego-centric reference frame. Each object is associated with a label, a pose estimate, and a recognition certainty. The figure illustrates the environmental model during a bimanual manipulation task involving both hands and two objects. The estimated position uncertainties are indicated by ellipsoids corresponding to the covariance matrix. For each object the recognition certainty is visualized with a bar, where green denotes certainty close to one.

III. GAZE SELECTION DURING MANIPULATION TASKS

The proposed gaze selection mechanism adapts the gaze of the active camera system online in a perception-action loop. The processing steps are illustrated in Fig. 2. First, the processed camera views are fused in the environmental model. Then, the saliency of all entities in the environmental model is calculated. Thereby, the task acuity allows to configure the accuracy of the active perceptual process. Finally, a selection mechanism steers the gaze redirection according to the saliency measure. All three steps of the processing chain are discussed in detail in the following sections.

A. Environmental Model Representation

In order to support manipulation, the environmental model needs to cover task relevant objects such as the hands of the robot or manipulation targets as well as their relevant properties. While the selection of these objects and properties is task specific and thus varies, all manipulation tasks share the common goal of physically interacting with the environment. Consequently, all entities stored within the environmental model need to provide at least means to direct interaction toward them.

To support the physical interaction, the environmental model is organized as a spatial memory covering 6D pose information for each entity. For most manipulation tasks, the number of objects that need to be considered and represented in the model is limited. Consequently, we choose a sparse landmark-based approach to represent the environmental model. For each landmark, its 3D position x accompanied with the location uncertainty Σ_x and its orientation in quaternion representation q is stored. The resulting environmental memory is a collection of the N task relevant entities:

$$M = (m_1, \cdots, m_N),$$

where each entity m_i is represented as:

$$m_i = (x_i, \Sigma_{x_i}, q_i).$$

In order to update the content of the environmental model, stereo-based object localization is performed in the current view of the cameras. For this purpose, we make use of the approach proposed in [17] for textured objects and in [18] for uniformly colored objects. The localization process provides the position z and the orientation q_z for each object. The uncertainty in the localization process is modeled as additive Gaussian noise in the position domain with the covariance matrix Σ_z. In order to approximate the 3D localization uncertainty, we assume 2D additive Gaussian noise of localization in each stereo image which is passed through the epipolar geometry using the unscented transform [19]. Further, we calculate a scalar value $\epsilon \in [0, 1]$ that quantifies the confidence of the object recognition and localization process.

The update of the environmental model based on the object localization result is implemented as probabilistic inference process. The correspondence between localized object and memory entity is solved on the spatial domain using the maximum a-posteriori estimate. Since only normally distributed random variables are involved, the update process is realized using Kalman filtering. The prediction step of the Kalman filter incorporates the motion and motion uncertainty of the memory entity, while the update step fuses the predicted estimate and the current observation. For the prediction step, we provide a motion model for each entity in the task. These motion models are task specific and will be further defined for the application in bimanual visual servoing in Section IV.

In the update step we incorporate the confidence of the current object localization ϵ as proposed in [20]. Instead of using just the Kalman gain matrix K, the position and uncertainty estimation is updated using $\epsilon \cdot K$. The resulting position estimates are illustrated in Fig. 3. The orientation of

each entity is updated by applying spherical linear interpolation to the stored and observed quaternions. The interpolation parameter κ is derived according to the predicted variance of the stored entity position ($\overline{\Sigma}_x$) and observed position variance (Σ_z). Thereby, we use the radius of a sphere with the same volume as the uncertainty ellipsoid as quantification for the amount of uncertainty:

$$\kappa = \frac{|\overline{\Sigma}_x|^{\frac{1}{6}}}{|\overline{\Sigma}_x|^{\frac{1}{6}} + |\Sigma_z|^{\frac{1}{6}}}.$$

As for the position, we incorporate the confidence of correct recognition and thus interpolate the orientation with the factor $\epsilon \cdot \kappa$.

B. Saliency Calculation

The saliency measure in our work encodes the necessity to fixate a location in the observable area. It forms the basis for deciding the optimal gaze in the gaze selection step. Thus, the definition of the saliency measure is the most crucial element in implementing the gaze selection strategy.

As already discussed in the introduction, each manipulation task has specific requirements on the perceptual process. In order to allow the inclusion of constraints from manipulation tasks, we propose a saliency measure which can be configured for specific tasks. For this purpose, we introduce the task acuity a_i which allows to specify the required accuracy of an entity m_i within the environmental model. More precisely, the task acuity is interpreted as desired upper bound for the uncertainty in the localization of a memory entity. As such, the accuracy of the localization estimate resulting from the active perceptual process becomes the main driving force for gaze selection. In the following, we will derive a consistent way to embed the task acuity in a saliency measure for gaze selection.

The gaze selection strategy aims at reducing the overall localization uncertainty within the environmental model. For this purpose, the saliency is calculated for each memory entity m_i stored in the environmental model. A memory entity with high localization uncertainty Σ_{x_i} should thereby be assigned with a high saliency value in order to express the necessity for revalidation. In order to quantify the amount of uncertainty, the covariance matrix Σ_{x_i} needs to be mapped to a scalar value. A natural quantification of the amount of uncertainty is provided by the differential entropy, which is a generalization of the Shannon entropy to continuous probability distributions. Given that the localization uncertainty in our work is normally distributed, its differential entropy can be calculated in closed form using

$$u_i(t) = \frac{1}{2} \log \left[(2\pi e)^3 |\Sigma_{x_i}(t)| \right], \tag{1}$$

where $\Sigma_{x_i}(t)$ is the location uncertainty corresponding to the memory entity m_i.

Using $u_i(t)$ as saliency measure would result in a gaze selection strategy which aims at reducing the uncertainty of all memory entities irrespective of the manipulation task's requirements. In order to include these requirements, we
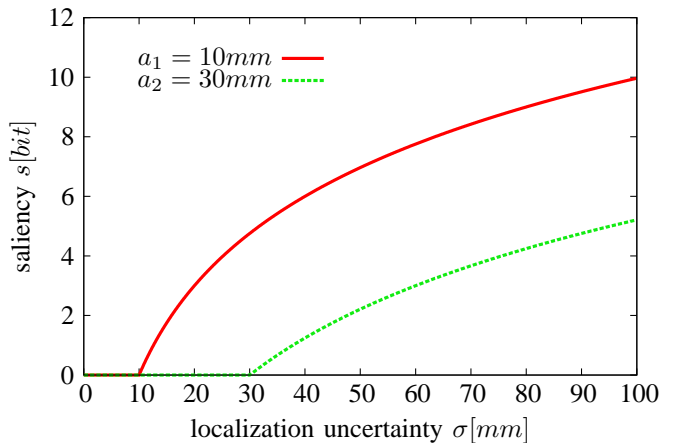

Fig. 4. Saliency in relation to localization uncertainty σ and task acuity a. The saliency measure drops to zero once the uncertainty reaches the requested task acuity.

express the task acuity $a_i(t)$ in the same domain as $u_i(t)$ using again the differential entropy

$$b_i(t) = \frac{1}{2} \log \left[(2\pi e a_i(t)^2)^3 \right]. \tag{2}$$

The resulting measure $b_i(t)$ encodes the desired upper bound for the entropy of the position estimate.

Putting equations (1) and (2) together yields the final saliency measure $s_i(t)$, where saliency is defined as difference between entropy resulting from the localization uncertainty and minimal entropy desired by the manipulation task

$$s_i(t) = u_i(t) - b_i(t). \tag{3}$$

The saliency measure $s_i(t)$ in relation to the localization uncertainty and the effect of a fixed task acuity a are illustrated in Fig. 4. The plot was generated using a univariate localization uncertainty with standard deviation σ for two different task acuities $a_1 = 10mm$ and $a_2 = 30mm$. The logarithmic shape of the entropy measure is desirable, since it results in a smaller validation effect for entities with higher localization uncertainty. The task acuity acts as shift and cutoff for the saliency and assures that once the localization uncertainty reaches the requested task acuity the saliency drops to zero. For values smaller zero we set $s_i(t) = 0$ in order to avoid negative saliency.

In summary, the combination of differential entropy and task acuity yields a consistent integration of accuracy requirements in the saliency calculation. Moreover, the introduction of the task acuity renders the differential entropy usable at all for a saliency measure. A major drawback of the differential entropy, its definition on the interval $(-\infty, \infty)$, is compensated with the inclusion of the task acuity in equation (3). The task acuity limits the differential entropy to $[0, \infty)$ making it suitable for saliency calculation.

C. Gaze Selection

On a humanoid robot, a gaze can generally be realized by specifying the 6D pose of the camera system plus the version

and vergence parameters of the active cameras. Selecting gazes in this space would require to initiate full-body motions of the robot in order to achieve the optimal gaze. During a manipulation task, it is obviously not suitable to realize a gaze direction in this way, since its execution would interfere with the execution of the task. Rather, in order to not affect the manipulation task, we only consider the active head-eye system of the robot for realizing the optimal gaze.

For the selection of gaze directions, we further simplify the head-eye system in order to achieve a representation of gaze directions which allows computationally feasible online performance. For this reason, the gaze of the system is represented on a unit sphere with an origin at the center between both active cameras. The sphere representation allows to encode a gaze direction with the zenith θ and azimuth ϕ of the corresponding spherical polar coordinates. We omit the rotation around the tangential plane to the sphere, since this degree of freedom is hard to realize when using only the head-eye system. The unit sphere is represented using a spherical graph as illustrated in Fig. 5, where each of the equidistantly distributed nodes corresponds to one viewing direction of the active head-eye system.

In order to determine the optimal gaze, each node of the graph is assigned a rating based on the saliency measure introduced in the last section. The rating for a node with coordinates (θ, ϕ) is calculated as a weighted sum of saliencies:

$$r_{(\theta,\phi)} = \sum_{i \in 1 \dots N} v_i(\theta, \phi) \cdot s_i. \qquad (4)$$

The weight $v_i(\theta, \phi)$ encodes the visibility in the cameras of each environmental memory entity with the current gaze direction. In order to determine this visibility, a simplified camera model is used which approximates the view frustum of the cameras by a single cone. For each memory entity, such a cone is intersected with the sphere of gaze directions. Entities which are situated close to the limits of the cone are likely to be partially occluded and subject to lens distortion effects. Consequently, the visibility is attenuated with increasing distance to the cone center since localization performance decreases towards the limits. An example of resulting spherical graphs is illustrated in Fig. 5. For both cases, four memory entities were involved in the task. In the first case, two entities are situated at the same position in the center, resulting in an increased rating of the region. The second case illustrates how overlapping visibility regions can generate maxima on the sphere by considering the sum of saliencies. A gaze direction towards such a maximum allows to fixate multiple objects at the same time.

The representation of possible viewing directions as nodes in a spherical graph allows straightforward and efficient determination of optimal gazes by detecting the maximum on the spherical graph. However, as pointed out earlier, it is also a simplification of the active head-eye system since it assumes a fixed reference frame for the camera system during calculation of the rating. In reality however, the cameras move resulting in inaccurate approximations of the
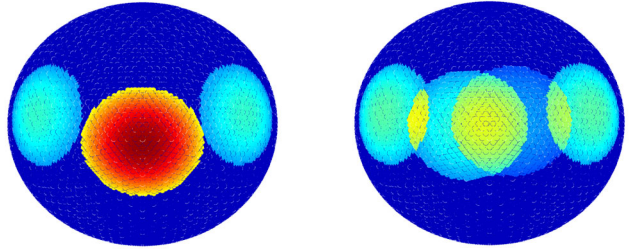

Fig. 5. Gaze directions are represented as a spherical graph with equidistantly distributed nodes. The nodes are rated according to the saliency of visible memory entities when taking the corresponding gaze direction. Thereby, the visibility of the objects in the cameras is approximated by viewing cones with decreasing localization reliability towards the limits. The node with the maximum rating is chosen as optimal gaze direction. Both graphs illustrate an example task involving four objects, where in the left graph two objects completely overlap in the center.

real saliency distribution. In order to compensate this effect, we use an iterative procedure for calculating the corrected optimal gaze direction: First, the rating in equation (4) is performed using the current posture of the head-eye system. Then, a candidate gaze direction is determined by searching the node with maximum rating on the graph. Using this gaze direction as input, the new posture of the head-eye system is calculated using inverse kinematics. With this new posture, the rating procedure is repeated with a sphere centered at the new reference frame. The iterative procedure stops when the posture of the head-eye system does not change anymore. The optimal gaze direction is then approximated by the node corresponding to the maximum peak on the spherical graph. In practice, it never occurred that more than one additional iteration was required, as the posture of the head-eye system does not change significantly during the iterations.

IV. Application in bimanual Manipulation

In this section, the proposed method for gaze selection is applied in a bimanual manipulation task on the humanoid robot ARMAR-III. In previous work, we demonstrated the execution of bimanual tasks using visual servoing techniques [21]. Thereby, the wide operational space necessitates head-eye movements in order to observe all objects involved in the task. In the previous work, the gaze selection was accomplished in a manner specific for the task. Based on such an application we will demonstrate, how the proposed gaze selection mechanism allows to produce gaze sequences for a given task in a more consistent and general way.

In the following, a brief introduction to the implementation of the bimanual manipulation task is given. Subsequently, the motion models required to complete the definition of the gaze selection mechanism for this task are introduced.

A. Bimanual Visual Servoing

In our previous work we solved bimanual manipulation tasks such as pouring or carrying big objects using position-based visual servoing. The benefits of applying visual ser-

voing techniques lie in their robustness towards inaccuracies in the kinematic model of the system. Thus, task execution based on visual servoing shares the goal with our approach in being applicable in the presence of inaccuracies, making it well suited for complex integrated platforms such as humanoid robots. In contrast to planned motions, trajectories resulting from visual servoing are not guaranteed to be collision-free, thus limiting its applicability to tasks which do not include possible collision with obstacles. Nevertheless, the application of visual servoing provides a feasible test-bed for the proposed gaze selection strategy for two reasons: First, it allows the execution of manipulation tasks including multiple objects. Second, the feedback from the perceptual processes can easily be integrated in the execution by directly using the content of the environmental model as input. For the integration with motion planning, a suitable plan monitoring and re-planning step would need to be implemented which goes beyond the scope of this paper.

For bimanual manipulation we observe the two robot hands and two target objects with the proposed gaze selection mechanism. Their position and orientation from the environmental model are then used in the position-based visual servoing approach. The trajectory is generated by successively reducing the distance of robot hands and target objects using differential inverse kinematics as discussed in [21].

To realize the desired gaze directions, we use the active head of ARMAR-III offering 3 DoF in the neck, a common tilt and a separate pan for both cameras. The joint angles for these 6 DoF are calculated by solving the inverse kinematics problem using optimization. We use an objective function that assures the correct gaze direction and generates natural looking postures. The inverse kinematics solution is calculated using gradient-free local optimization. The kinematic model for the head-eye system is calibrated offline using the approach proposed in [22]. The same model is used to retain stereo perception while the extrinsic camera parameters change.

B. Motion models

The prediction of motion within the environmental model is necessary since not all objects are visible to the cameras all of the time during the manipulation task. Thereby, two kinds of motion need to be considered: The motion of the head-eye system and the motion of entities physically controlled by the robot such as its hands. Both motions can be approximated by reading the joint encoders of the robot. Due to remaining inaccuracies in the positioning and in the kinematic model of the system, these measurements are not entirely correct, thus necessitating the inclusion of motion uncertainty in the motion model.

In order to calculate uncertainties implied by the head motion, we use the frame of the left camera as reference. This reference frame changes during head-eye movements. In order to cope with the inaccuracies in the kinematic model and in the positioning, we assume additive Gaussian noise in the joint angles of the head-eye system. Using the unscented

transform, this noise is passed through the system in order to retrieve an estimate of the uncertainty implied by the head motion to the position of an entity. The resulting covariance matrix is used in the Kalman filter prediction step for the entities.

In order to define motion models for objects in the scene, we differentiate between the objects which are controlled by the robot, as e.g. its hands, and objects which are target of the manipulation. In the current setup, the objects the robot wants to manipulate are assumed to be static, i.e. they do not move on their own. Consequently, the pose and the associated uncertainty do not change over time and no additional uncertainty needs to be considered in the motion model. In contrast, for the hands of the robot again the inaccuracies in the kinematic model need to be considered with respect to the joint encoder readings of the arm. As for the head-eye system, we make use of the unscented transform in order to calculate the uncertainty of motion implied by the model inaccuracies. The pose of the objects is then predicted using the resulting covariance matrix within the Kalman filter prediction step.

V. Experimental Evaluation

A. Experimental Setup

In the following, the proposed gaze selection mechanism is evaluated in a typical kitchen environment task on the humanoid robot ARMAR-IIIa. A complex task in this environment which involves multiple objects and requires two arms is the pouring scenario, where the robot pours juice from a container in one hand into a cup held in the other hand. In the context of gaze selection, the first phase of this task, the approach and grasping of both objects with the five-fingered hand is the most demanding part, since four distinct objects need to be observed: the cup, the juice, and both hands. Consequently, we restrict the experiments to this first phase.

For all experiments, we used the green cup and the vitamin juice placed on a table in front of ARMAR-IIIa as shown in Fig. 1. The positions of cup and juice were varied within the workspace of the robot. The desired grasps for both objects and thus the target poses for visual servoing were predefined relative to the objects' local coordinate frames.

In order to initiate the task execution, an estimate of the position of both hands and both objects needs to be provided as prior in the environmental model. For the hands, we use the pose from the kinematic model as initial estimate. For both objects, a position on the table in front of the robot is provided as initial estimate. We choose a conservative initial localization uncertainty with a standard deviation of $500\,mm$ in all directions for the hands whereas the objects are assigned with a higher uncertainty corresponding to a standard deviation of $1000\,mm$. The execution itself is started once the localization uncertainty of all objects drops below a standard deviation of $d_{max} = 50\,mm$.
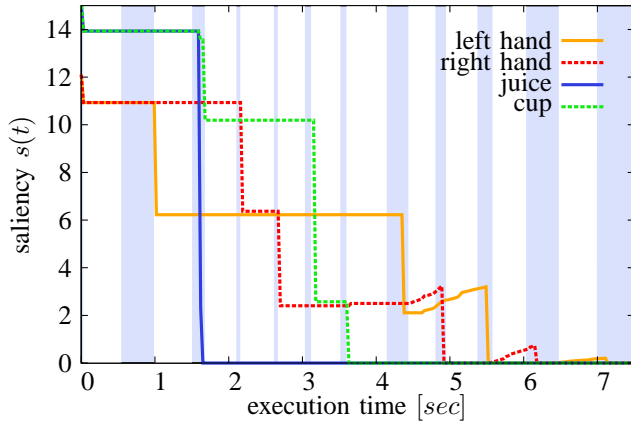
Fig. 6. The bimanual visual servoing task requires pose estimates of both hands and both involved objects with the necessary accuracy. The saliency measure $s(t)$ encodes the necessity to perform a localization of an element. For a task acuity of $a = 5\,mm$, the plot shows the development of the saliency over a complete approach and grasp phase. Blue regions indicate phases where localizations are performed while white regions denote head movements to selected gaze directions.

B. Saliency during Manipulation

Fig. 6 illustrates the course of the saliency measure $s_i(t)$ for all objects involved in the task over one complete approach and grasp phase. The task duration from the initial localization of all objects until successful grasp execution is about seven seconds, where the first four seconds are required in order to successively reduce the uncertainty of all objects under the limit d_{max}. Once the uncertainty drops below this limit, the visual servoing procedure is started until the target is reached and the grasp is executed.

During the execution of the manipulation task, successive fixations of the involved objects are performed according to the gaze selection mechanism. After each redirection of the gaze, the object localization modules are triggered in order to determine the pose of all visible objects and to update the environmental model. The localization is stopped once a new gaze direction is requested by the gaze selection mechanism. The time intervals, when localization is performed are marked with blue background in Fig. 6. The update of the environmental model is performed delayed, once the object localization processes finish the computation of the pose.

The plot clearly illustrates how the proposed approach allows to reduce the localization uncertainty by actively redirecting the gaze appropriately. Each localization results in a reduction of the uncertainty of the observed entity. Once the desired accuracy, defined by the task acuity a is reached, the saliency $s(t)$ drops to zero. For the cup and the juice box, the saliency drops to zero after a few localizations and remains there. For the two robot hands, the uncertainty in the pose estimate increases due to the movement of the robot arms, accompanied by an increase of the associated saliency. As expected, the gaze selection mechanism compensates this increase by initiating additional localizations of the robot hands.
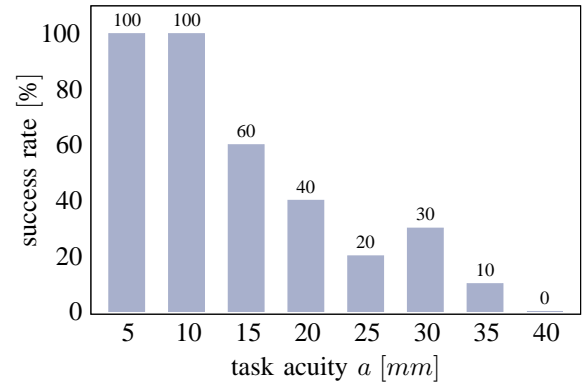

Fig. 7. Success rate of the bimanual visual servoing and grasping task in relation to the selected task acuity. For each setting of the task acuity 10 trials were executed. For a task acuity of $5\,mm$ and $10\,mm$ all trials could be carried out successfully. With increasing task acuity, the success rate drops.
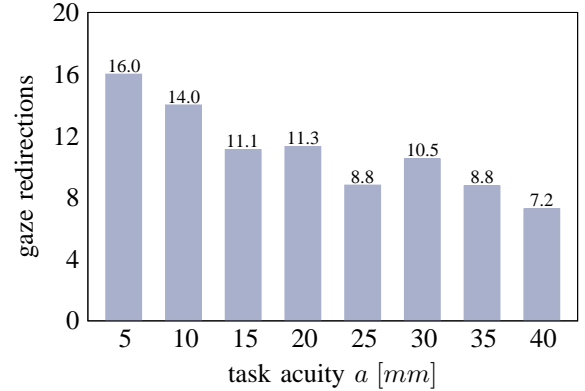

Fig. 8. Number of head movements in relation to the selected task acuity. For lower task acuity, more head movements need to be executed in order to achieve the required acuity of the environmental model.

C. Influence of the Task Acuity

In order to evaluate the feasibility of the task acuity as means of integrating manipulation task constraints with the gaze selection approach, the bimanual task was performed several times with varying task acuity settings. Thereby, a task acuity from the range $a_{min} = 5\,mm$ to $a_{max} = 40\,mm$ was used with an increment of $5\,mm$. For each setting from this range, the manipulation task was executed ten times. After each execution, the success was assessed by lifting both objects. The resulting success rate in relation to the tested task acuity is illustrated in Fig. 7. For the task acuity settings of $5\,mm$ and $10\,mm$ the execution succeeded in all ten trials. The success rate drops with increasing task acuity until no successful execution is possible with the maximum tested task acuity of $40\,mm$. These results are into accordance with our expectations, since a minimal required localization accuracy of $10\,mm$ for both – robot hand and object – seems to be feasible in order to produce a stable grasp of the object.

In addition to the success rate we investigated the number of gaze redirections required to perform the manipulation task in relation to the task acuity. While higher task acuity

obviously lead to better pose estimates, it also necessitate the execution of more gaze shifts. An appropriate selection of the task acuity should minimize the number of gaze redirections required but still retain the ability to successfully accomplish the task. As illustrated in Fig. 8 the number of required gaze redirections drops with increasing task acuity. Considering the number of redirections, the optimal choice for the task acuity in the bimanual visual servoing and grasping task amounts to $a = 10\,mm$.

VI. Conclusion

In this work, we introduce a gaze selection approach tailored for manipulation tasks on humanoid robots. The applicability in a manipulation task influences the proposed approach in several ways: First, the proposed environmental model allows for dynamic entities by the inclusion of motion models. Second, the saliency calculation includes accuracy constraints from the manipulation task by means of the task acuity. Finally, the gaze selection and redirection is implemented using only the DoF of the head-eye system in order to not interfere with the manipulation task.

The gaze selection approach was evaluated in a bimanual visual servoing task involving four objects that would fail without the application of active gaze control. Using the proposed gaze selection approach, the task could be accomplished with a success rate of 100%. Further, we could demonstrate that the inclusion of the task acuity allows to intuitively configure the perceptual processes. The optimal trade-off between accuracy and number of required gaze redirections was achieved for a task acuity of $a = 10\,mm$. Being able to perform the task with this accuracy is feasible as well as intuitive.

While the evaluation in the context of a bimanual visual servoing task is suitable to demonstrate the feasibility of the approach, it only covers a fraction of possible applications for the proposed gaze selection mechanism. For complex manipulations involving obstacles and dexterous abilities, motion planning is required in order to achieve an executable and collision-free trajectory. Having performed motion planning, the motion models as well as the required task acuity could be directly derived from the resulting trajectory and its relation to the world model.

In summary, the described gaze selection approach enables the robot to exploit two of its key capabilities, manipulation and active gaze control, in an integrated fashion. The inclusion of constraints based on the task acuity allows the adaptation of the generated gaze sequence in order to support successful task execution. Thus, the proposed approach substantially contributes in increasing the autonomy of humanoid platforms.

ACKNOWLEDGMENT

References

[1] T. Asfour, P. Azad, N. Vahrenkamp, K. Regenstein, A. Bierbaum, K. Welke, J. Schröder, and R. Dillmann, "Toward humanoid manipulation in human-centred environments," *Robotics and Autonomous Systems*, vol. 56, no. 1, pp. 54–65, 2008.

[2] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, ser. Intelligent Robotics and Autonomous Agents. MIT Press, 2005.

[3] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *pami*, vol. 20, no. 11, pp. 1254–1259, 1998.

[4] L. Itti, N. Dhavale, and F. Pighin, "Realistic avatar eye and head animation using a neurobiological model of visual attention," in *Proc. SPIE 48th Annual International Symposium on Optical Science and Technology*. SPIE Press, 2003, pp. 64–78.

[5] A. Ude, V. Wyart, L.-H. Lin, and G. Cheng, "Distributed visual attention on a humanoid robot," in *humanoids*, 2005, pp. 381–386.

[6] S. Frintrop, E. Rome, and H. Christensen, "Computational visual attention systems and their cognitive foundations: A survey," *ACM Transactions on Applied Perception*, vol. 7, no. 1, pp. 1–39, 2010.

[7] C. Breazeal and B. Scassellati, "A context-dependent attention system for a social robot," in *International Joint Conference on Artificial Intelligence*, 1999, pp. 1146–1153.

[8] M. Bjorkman and D. Kragic, "Combination of foveal and peripheral vision for object recognition and pose estimation," in *humanoids*, 2004, pp. 5135–5140.

[9] D. Figueira, M. Lopes, R. Ventura, and J. Ruesch, "Towards a spatial model for humanoid social robots," in *Progress in Artificial Intelligence*. Springer Berlin / Heidelberg, 2009, pp. 287–298.

[10] K. Welke, "Memory-based active visual search for humanoid robots," Ph.D. dissertation, Karlsruhe Institute of Technology (KIT), Computer Science Faculty, Institute for Anthropomatics (IFA), 2011.

[11] O. Stasse, T. Foissotte, D. Larlus, A. Kheddar, and K. Yokoi, "Treasure hunting for humanoid robots," in *Workshop on Cognitive Humanoids Vision*, 2009.

[12] B. Rasolzadeh, M. Björkman, K. Huebner, and D. Kragic, "An active vision system for detecting, fixating and manipulating objects in real world," *The International Journal of Robotics Research*, vol. 29, pp. 133–154, 2009.

[13] F. Seara, J. Strobl, K. H. Martin, and E. Schmidt, "Task-oriented and situation-dependent gaze control for vision guided autonomous walking," in *humanoids*, 2003.

[14] S. Kohlbrecher, A. Stumpf, and O. von Stryk, "Grid-based occupancy mapping and automatic gaze control for soccer playing humanoid robots," in *Proc. 6th Workshop on Humanoid Soccer Robots*, 2011.

[15] K. Okada, M. Kojima, S. Tokutsu, Y. Mori, T. Maki, and M. Inaba, "Task guided attention control and visual verification in tea serving by the daily assistive humanoid HRP2JSK," in *iros*, sept. 2008, pp. 1551 –1557.

[16] P. Michel, C. Scheurer, J. Kuffner, N. Vahrenkamp, and R. Dillmann, "Planning for robust execution of humanoid motions using future perceptive capability," in *Proceedings of the IEEE/RSJ IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, October 2007, pp. 3223–3228.

[17] P. Azad, T. Asfour, and R. Dillmann, "Stereo-based 6D Object Localization for Grasping with Humanoid Robot Systems," in *iros*, San Diego, USA, October 2007, pp. 919–924.

[18] ——, "Accurate Shape-based 6-DoF Pose Estimation of Single-colored Objects," in *iros*, St. Louis, USA, October 2009, pp. 2690–2695.

[19] S. J. Julier and J. K. Uhlmann, "A new extension of the kalman filter to nonlinear systems," 1997, pp. 182–193.

[20] T. Kirubarajan and B. Y. Shalom, "Probabilistic data association techniques for target tracking in clutter," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 536–557, 2004.

[21] N. Vahrenkamp, C. Böge, K. Welke, T. Asfour, J. Walter, and R. Dillmann, "Visual Servoing for Dual Arm Motions on a Humanoid Robot," in *humanoids*, Paris, France, Dec. 2009, pp. 208–214.

[22] K. Welke, M. Przybylski, T. Asfour, and R. Dillmann, "Kinematic calibration for saccadic eye movements," Institute for Anthropomatics, Universität Karlsruhe, Tech. Rep., 2008.