Human-Robot Cooperative Object Manipulation with Contact Changes

Michael Gienger[1], Dirk Ruiken[1], Tamas Bates[1,2], Mohamed Regaieg[3], Michael Meißner[1,3], Jens Kober[2], Philipp Seiwald[3], Arne-Christoph Hildebrandt[3]

***Abstract*— This paper presents a system for cooperatively manipulating large objects between a human and a robot. This physical interaction system is designed to handle, transport, or manipulate large objects of different shapes in cooperation with a human. Unique points are the bi-manual physical cooperation, the sequential characteristic of the cooperation including contact changes, and a novel architecture combining force interaction cues, interactive search-based planning, and online trajectory and motion generation. The resulting system implements a mixed initiative collaboration strategy, deferring to the human when his intentions are unclear, and driving the task once understood. This results in an easy and intuitive human-robot interaction. It is evaluated in simulations and on a bi-manual mobile robot with 32 degrees of freedom.**

I. Introduction

Handling large objects is an important task in several domains, such as production, warehouse logistics, and construction. Current systems are mainly designed to operate autonomously, and in areas that are separated from humans for safety reasons. Modern compliant robots in combination with recent regulations suggest to safely relax this strict separation, and to research and design systems that allow for physical cooperation between humans and robots [1], [2]. Such cooperative systems have the advantage of being able to exploit the cognitive abilities of humans in order to realize tasks of higher complexity.

In order to effectively support human coworkers, such systems need to be able to understand human goals and intentions, and to adapt on the fly to changes in the environment or the behavior of its coworkers. In order to work efficiently, the system must also be able to independently judge when it needs to obtain input from its coworkers and when it can act on its own. This involves some degree of shared autonomy, in which the robot occasionally defers to the humans around it, while at other times it takes its own initiative. Work on shared autonomy typically focuses on scenarios which also involve shared or traded control, as in [3], for example, to improve the safety of a human operating a semi-autonomous vehicle [4] or to improve performance in assistive and teleoperation tasks [5]. For autonomous cooperation, however, the robot does not receive explicit commands and must decide for itself when to take initiative and when to follow the human's lead.

[1]Honda Research Institute Europe, Offenbach, Germany {michael.gienger, dirk.ruiken}@honda-ri.de
[2]Technical University of Delft, Mekelweg 2, 2628CD The Netherlands {t.bates, j.kober}@tudelft.nl
[3]Technical University of Munich, Arcisstraße 21, 80333 Germany {mohamed.regaieg, philipp.seiwald}@tum.de

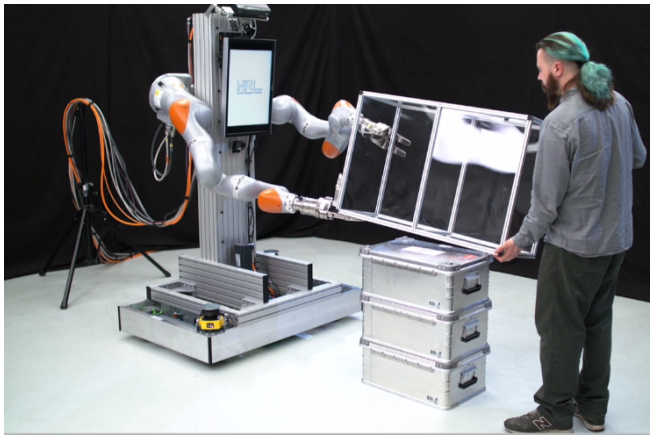

Fig. 1: Robot setup with human interaction partner

The presented system implements a kind of mixed-initiative execution strategy for sequential bi-manual tasks. In the context of load sharing, the effort-sharing strategy described in [6] analyses similar concepts. The authors explore different dynamic role-switching behavior based on sensor feedback during a joint object transportation task. Leader and follower role assignments in a table transportation task with a humanoid robot has been explored in [7]. Our system can deal with sequential manipulation tasks with contact changes (such as re-grasping). It defers to the human's leadership in the interaction when the user's intentions are unknown. As the user's goals become clear, it takes over the initiative and drives the interaction.

Joint manipulation of objects is often based on the forces and torques transmitted through the manipulated object. The robot senses the forces and torques applied to the object by the human and moves accordingly [8]–[10]. When performing human-robot cooperation with sequential manipulation actions, this path of communication might not be available anymore. There is not much literature about this type of human-robot cooperation in which a sequence of actions needs to be considered by the robot [11]. In most approaches, like [12]–[16], the robot only needs to react according to the detected situation.

In this paper, haptic feedback is used to determine which way the user wishes to rotate a jointly-held object and whether the user is ready to support and rotate it. Once the robot knows which way to perform a rotation, it plans a sequence of actions to carry it out. When the robot is confident that the user is ready, it begins executing its plan, effectively taking over leadership from the human until the

object has rotated far enough that the human will need to adjust their grasp, at which point the system defers again to the human until they are ready. Trading roles in this way feels very natural and allows the user to focus on their own part of the interaction without needing to explicitly direct the robot.

Similarly to recent work on grasp [17], contact [18], and footstep planning for humanoid robots [19], we plan only a few steps (hand re-positions) ahead, and determine possible movements based on a set of simple physical criteria which ensure the manipulated object is always supported. This allows trajectory planning for hand movements to be very fast, as the computed trajectories are typically very short and the overall search space for valid hand positions is kept to a manageable size. This ensures that any plans can be altered on the fly without causing a noticeable delay for human coworkers, and in the event of an emergency the robot can pause and resume or re-plan its actions without any issues.

The main contribution of this paper is a novel concept for how to structure the interaction. In particular, we propose a system for human-robot cooperative object manipulation with changing contacts. First, we propose a mixed-initiative collaboration strategy by taking initiative when the system is certain about the human's intentions, and deferring to the human's leadership when their intentions are unclear. Second, three system layers and their interplay constitute to an architecture that is able to map human intentions to higher-level goals of a task, to decompose these into sequences of actions, and to propagate the actions directly to the trajectory and motion control levels. And third, the layers have been designed to be fast and responsive, so that the overall system is able to rapidly respond to a coworker's behavior. The presented concepts have been validated both in simulation and in physical robot experiments.

This paper is organized as follows: Section II gives a high-level overview of the realized system. The key elements of the underlying architecture are the interaction layer (Section III), the planning layer (Section IV), and the motion generation layer (Section V). The robot setup for our experiments is presented in Section VI. The system is evaluated is simulations and a real scenario in which a human and a robot jointly turn large boxes in Section VII.

II. System overview

The presented system is designed to support cooperation with humans through continuous intention estimation and real-time planning. Interaction, planning, and motion control are handled by three separate layers (Fig. 2). The interaction layer is the only component which has a concept of the human collaborator and handles all aspects of interacting with the human. The other layers are agnostic to the human and were made to be as fast and responsive as possible in order to account for rapid changes in the interaction.

The interaction layer uses sensor feedback and models of the interaction to estimate human intentions. These models contain the cues seen in sensor feedback when cooperatively

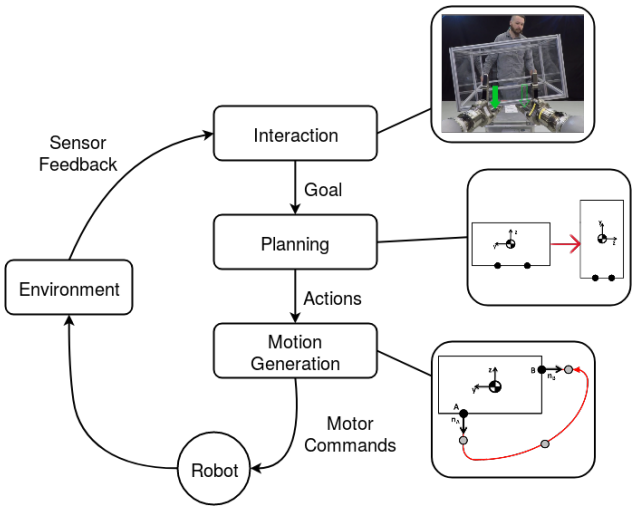

Fig. 2: System overview: The interaction layer processes sensor feedback to continuously estimate human intentions and generate matching robot goals. The planner determines the best sequence of actions to reach the estimated goal. The interaction layer synchronizes and monitors progress in the action sequence with the human while the motion generation layer turns individual actions into smooth whole body motions.

manipulating objects. Based on the estimated intentions (and the goal states fulfilling these intentions), the planning layer determines the best way to assist the human and produces a sequence of actions to reach the next estimated goal state. Finally, the motion generation layer turns the actions of the sequential plan into smooth robot motions. Throughout execution of each action, the interaction layer also ensures that action execution is synchronized with the human cooperator.

The following sections provide details on the individual layers. While the structure works for general human robot cooperation, here we focus on object manipulation tasks such as jointly turning large, bulky objects like boxes or barrels as described in Section VII.

III. Interaction Layer

The interaction layer is the core component of the presented system which coordinates all interaction with the human cooperator. It implements a mixed-initiative collaboration strategy. When the user's intentions are unknown, the system defers to the human's leadership in the interaction. As the user's goals become clear, however, the system becomes more and more autonomous, and even drives the interaction towards the recognized goals. As a result, the role of the leader and follower can switch between human and robot from action to action. To accomplish this, the interaction layer performs two roles:

1) Intention estimation: The interaction layer uses sensor feedback and models of the interaction to continuously estimate and monitor human intentions. These models can be seen as nodes of a state transition graph based on abstract object states (e.g. object rotations in 90 degree intervals)

annotated with expected sensor readings when a transition is to be taken. Based on the (current) low complexity of our models, simple filtering and thresholding of the sensor data is sufficient for reliable state and intention estimation.

For an estimated intention, a set of satisfying goal states is sent to the planning layer (Section IV) to determine how the robot can best assist the human in fulfilling their intentions and what sequence of actions would be required of the robot. While the human and the robot jointly execute this sequence, the robot continuously monitors sensor feedback to ensure that the sensor readings still agree with the estimated human intentions, and to re-plan a new sequence of actions if it becomes apparent that the human's intentions have changed.

2) Execution synchronization: When cooperating with a human, actions need to be started at the right moment for the cooperation to feel natural and be safe. If the robot waits too long before starting its next action, the human will have to wait unnecessarily. On the other hand, if an action is started before the human is ready, it could result in confusion, task failure, or even an unsafe situation. For example, when re-grasping an object, a hand can only be repositioned if it is not currently supporting the object or else the object might fall. Similarly, if the human is in the process of repositioning one of the hands, it would not be safe for the robot to reposition its hands until the human has finished.

Therefore, the interaction layer uses the sensor feedback to synchronize the execution of the planned sequence of actions with the human cooperator. Human actions are not explicitly contained or planned for in the sequence of planned actions. Instead, the actions for the robot have prerequisites (in the sensor feedback) for their execution which can be satisfied by various suitable human actions. For example, when the robot is ready to start an action, it might wait for appropriate force feedback indicating that the human is also ready to begin a joint action. Through this separation, the planner does not have to explicitly take into account human actions, but the execution is well synchronized between human and robot.

IV. Planning Layer

Once a goal is determined in the interaction layer, it is sent to the planning layer. Its role is to decompose it into a sequence of actions that drives the system to the given goal in the optimal way. To provide an example, let us consider the task of cooperatively rotating a large object as detailed in Section VII. A goal might be to rotate the object until it is upside down. The actions would then be the sequence of required changes of grasp holds and hand movements to flip the object over.

In the planning layer, a number of assumptions are made. Firstly, the state of the system is considered to be precisely measurable. We assume that the geometric shape of the object, its mass, center of mass (COM), and a feasible set of contact locations on its surface are known or can be estimated somehow. Secondly, it is assumed that the maximum applicable tangential force is limited by a given friction cone. And lastly, it is assumed that the human interaction partner is able to follow the object rotation. He is therefore not explicitly represented in the model.

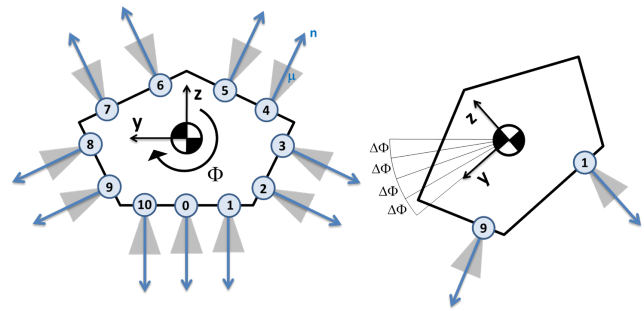

Fig. 3: Search state for an example object. The numbers in the circles enumerate the contact points. On the right, state $\boldsymbol{S} = (\Phi_{object} = -4; Contact_{Right} = 1; Contact_{Left} = 9)^T$ is shown. The object frame is located in the object's COM.

We decided on a discrete state description that contains the object rotation and the contact points of both hands. In order to keep the dimension of the search problem low, the object rotation is modeled as a planar problem with three dimensions (see Figure 3):

1) Rotation angle of the object about a fixed axis
2) Contact location for the right hand
3) Contact location for the left hand

Such search problems can be efficiently solved with methods from the class of informed search algorithms. We tried several candidates and eventually settled on the classical A* algorithm, which produced good solutions for our problem very quickly.

A key component of the selected search algorithm is the exploration strategy. For a given state, it determines the set of reachable states and assigns a cost to each transition. A simplified physics model of the object-hand interactions is the basis for this exploration. Valid successor states need to obey the following rules:

- Both hands must remain in contact with the object during rotation.
- Only one hand is allowed to change contact at a time.
- Changing the contact point is only possible if the stationary hand's friction cone is not exceeded.
- The left hand must always be to the left of the right hand.
- The distance between the hands must not go below a given threshold.
- The torque that the human must provide when the robot is changing contact must not rise above a given threshold.
- When both hands are in contact, they must support the object's COM.

A subset of these rules has been parametrized, so the search can be biased towards solutions with specific properties. One example is the torque limit that the human interaction partner feels when the robot changes its grasp (see Figure 9). We designed an admissible and monotonic heuristic function that

models the angular difference between the current and goal rotation angle. Each state change results in a transition cost corresponding to the required movement time. With these settings, the resulting A* search solution is optimal. The resulting state sequence is the basis for the motion generation layer to plan the trajectories.

V. Motion Generation Layer

The role of the motion generation layer is to convert a planned solution into continuous task-space trajectories, and to compute the corresponding motor commands for the robot. The trajectory generation follows a receding horizon schema, and the calculations of this layer run at a high frequency (200 Hz). This allows the system to handle plan changes at any point in time and without noticeable delays.

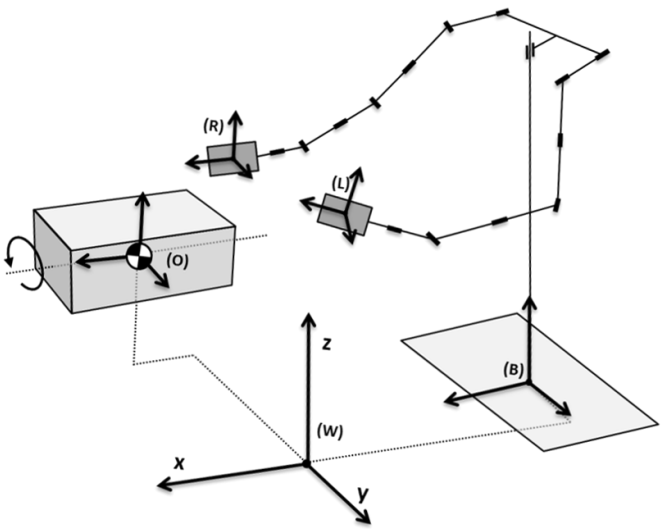

Fig. 4: Kinematic model with object-centered task description: Hand poses are represented in the frame of the object.

The kinematic model of the system is shown in Figure 4. It consists of both actuated and virtual degrees of freedom (DOF). The object (O) is modeled as a rigid body with four DOF: its position with respect to the world frame (W), and its rotation about its x-axis. The robot is modeled as a kinematic tree starting with two translational and one rotational DOF for the mobile base, and the successive joints of the robot components. The movement of the system is represented in a task space with the following parameters:

1) Rotation angle of the object (O) around its local x-axis
2) 6D pose of the left hand (L) with respect to the object
3) 6D pose of the right hand (R) with respect to the object
4) Vertical position of the object

This object-centered description, in which the end effector movement is linked to the movement of the object, follows the concepts from the dexterous manipulation domain [20].

The trajectory generation is based on fifth-order polynomials. In particular, we follow the approach taken in [21]. This elegant formulation allows for the generation of trajectories with an arbitrary number of constraints over time. Further, they are inherently smooth due to the incorporated minimum jerk model, and lead to velocity profiles similar to those of humans [22]. Constraints can be formulated on any combination of position, velocity and acceleration levels. We refer to *full constraints* as these affecting all levels, and *partial constraints* as those affecting only the position level. In order to allow for changes to the trajectories at any point in time, they are re-generated at each time step in a receding-horizon manner. Each step, the time of all constraints is shifted back by the sampling time step, and a constraint on position, velocity, and acceleration is assigned to the current time (t=0). Figure 5 shows example trajectories with different constraint types and the receding horizon. This

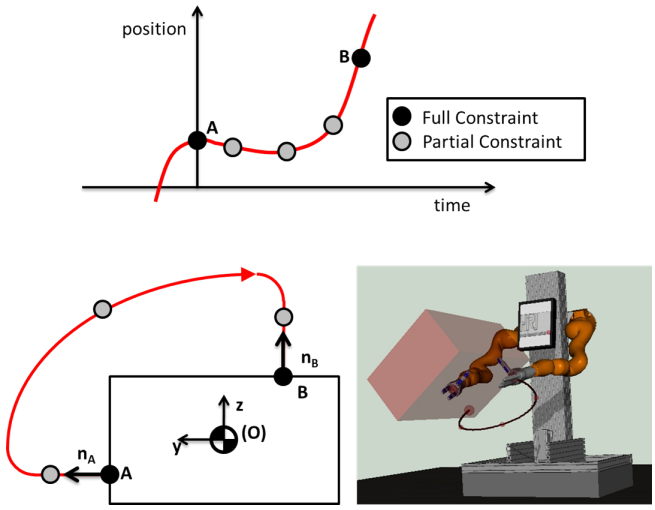

Fig. 5: Receding horizon trajectory generation. *Top:* The receding horizon model with full (affecting all properties) and partial (affecting only one) constraints. *Left:* A simple re-grasping trajectory. Full constraints in black, partial constraints in grey. *Right:* Example of a re-grasping trajectory.

representation allows for easy conversion of a search solution from the planning layer to a 3D trajectory. See Figure 5 for an example: For a contact change of the hand from A to B, a full constraint is set to the next contact wrench at B. Further, two partial constraints on the position level are placed along each contact normal. This forces the hands to get into contact without slip, and to avoid collisions with the object during the movement. Collisions are further avoided by a third partial constraint applied in the x-direction so that the hand's trajectory makes a curve away from the object during the movement. The timings of the constraints have been designed to match the robot's speed limits, and to avoid collisions for a variety of different objects. Since we did not incorporate any advanced trajectory optimization algorithm, the object's geometry is assumed to be not strongly non-convex. However, the presented concept can deal with a variety of different objects, including non-convex ones (see Section VII).

Inverse kinematics is used to project the task space trajectories into the robot's joint space. The task descriptors in

the relative reference frames closely follow the formulations in [23]. Joint limit and singularity avoidance criteria are projected into the null space. Non-articulated DOFs do not contribute. The resulting joint space movement is sent to all actuators of the system.

VI. Robot Setup

The experimental evaluation has been conducted with the two-arm mobile humanoid robot shown in Figure 1 and 4. It is built on an omni-directional platform that can instantaneously translate and rotate on the horizontal plane. The platform has a load capacity of one ton. It carries a vertical linear actuator with a max. speed of 0.5 m/sec, a max. load of 100 kg and a travel of approx. 1 m. Two Kuka LBR iiwa 820 with a payload of 14 kg are attached to the slide. They are spatially inclined in order to reduce the risk of kinematic singularities, and to maximize the overlap between the workspace of both arms. A six-axis force torque sensor and a Schunk dexterous 3-finger hand with seven DOF and six tactile pads is mounted to the flange of each of the robot arms. The complete system has 32 articulated DOF.

The trajectory and inverse kinematics calculations run at 200 Hz on a standard PC. The resulting joint-level motor commands are distributed to the different robot components and controlled locally. In addition, a Cartesian end effector compliance controller is running on the 7-DOF robot arms, which makes the system behave comfortably in the interaction and absorbs small disturbances.

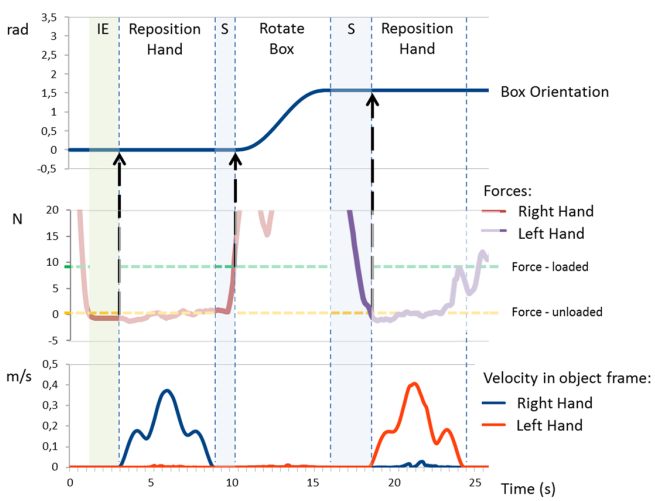

Fig. 7: Example action sequence for a clockwise box rotation showing the box orientation (*top*), sensed forces at the hands (*middle*), and hand velocities in the object frame (*bottom*).

VII. Evaluation

The robot setup described in the previous section was used to evaluate the presented concepts in a human-robot interaction task of jointly rotating a large box. The box has a size of $1 \times 0.64 \times 0.36$ m, and weighs approx. 9.4 kg.

Figure 6 shows the sequence of actions to carry out one 90-degree rotation of the box. The robot used only force feedback to interpret inputs from the user and estimate their intentions. Forces at the hands were estimated from joint torque readings in the arms. These were used to determine if the human is ready and in which direction they wished to rotate the box. The corresponding thresholds have been carefully tuned by hand. Almost no false positives have been found to compromise the reliability of the estimation. The corresponding data is plotted in Figure 7. In the first shaded segment (IE), the system defers the initiative to the human and waits until the measured forces indicate that the human wants to rotate the box (Fig. 6-2). It then repositions its hand to prepare for the rotation (Fig. 6-3). The robot then waits until the human applies a load in the desired rotation direction (Fig. 7, S). Once this is measured, the system takes initiative and rotates the box to the goal (Fig. 6-3). After the rotation is done, the robot defers leadership back to the human, and the interaction continues. Users reported that the interaction was intuitive and comfortable overall, but noted that the actions required of them to provide the initial conditions for starting a rotation were not immediately obvious without instruction.

In addition, simulations with a variety of different object shapes (box, cylinder, L-shape) were performed to validate the planning and motion generation layers. A set of 16 contact points is approximately uniformly distributed around the circumference of the objects. The discretization of the object rotation angle is 30 degrees. The goal is to turn the objects about 180 degrees. The results (see Figures 8 and 9) show that the system is able to find collision-free solutions to efficiently rotate all test objects while ensuring they remain safely supported during the manipulation.

A second set of simulations has been conducted to show the flexibility obtained by the parametrization of the planning layer. The planner can for instance be configured to limit the maximum torque the human will be exposed to while the robot is regrasping. In the upper row of Figure 9, the search did not consider any torque limit. In the lower row, a limit of 1.5 Nm has been applied. Figure 9 shows the difference in behavior when manipulating two otherwise identical cylinders. The sequence shows nicely that the support hand locations are closer under the object's center of mass, as one would expect. If the object's weight is known, or estimated via the robot's force sensing abilities, the robot can automatically adjust its behavior to provide better support for heavier objects. This allows the incorporation of additional criteria, e.g. ergonomic indicators, directly into the planning layer.

VIII. Conclusion

We have presented an interactive human-robot collaboration system for joint manipulation of large objects. This unique system features several novelties. First, it implements a mixed-initiative collaboration strategy by taking initiative when it can be certain about the human's intentions, and deferring to the human's leadership when their intentions are unclear. This shifting of roles produced a very comfortable and natural-feeling interaction for simple joint object

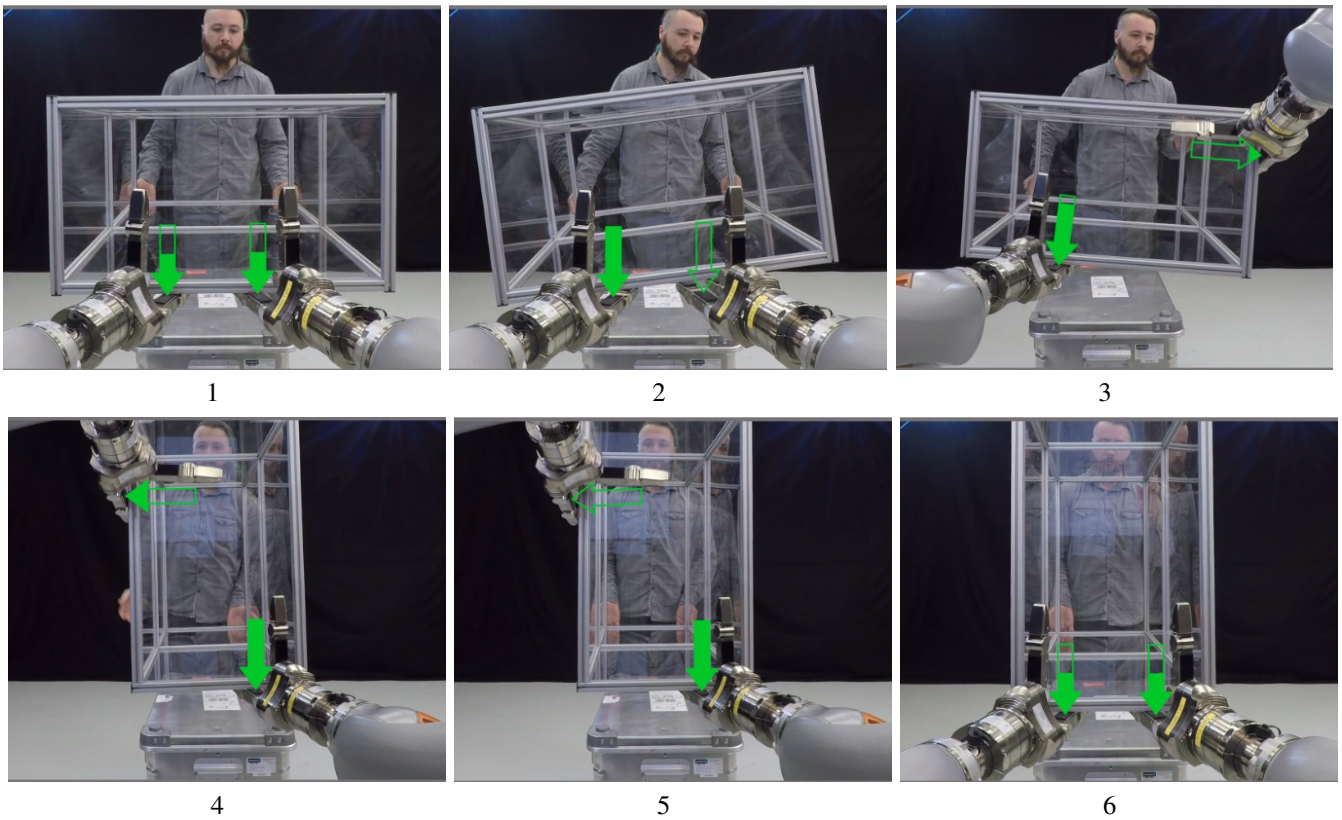

Fig. 6: Cooperative rotation of a large box. Sensed normal forces on the robot's grippers are illustrated with green arrows. The robot is supporting the box (1). It senses that the right hand is unloaded (2) and repositions it to support a predicted following rotation (3). Once the torque applied by the human to the box is sensed, the robot assists in the rotation (3 to 4). As soon as the left hand is unloaded (5), the hand can safely be repositioned to support the bottom of the box (6).

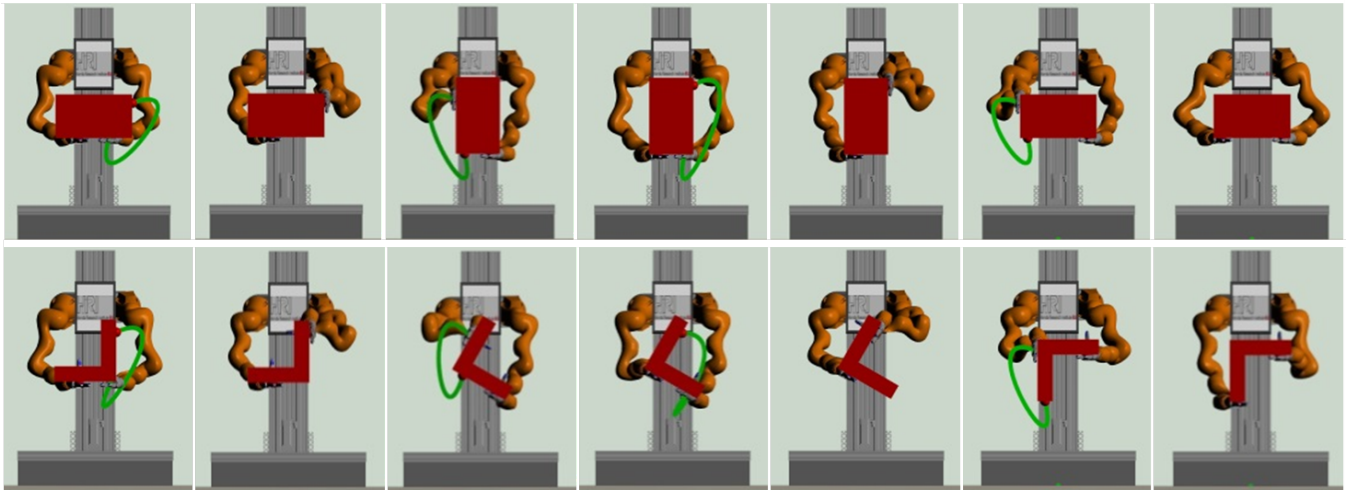

Fig. 8: Simulated rotation of a rectangular box (*top*) and an L-shaped object (*bottom*). The green line shows the position component of the end effector trajectory.

manipulation tasks. Second, the underlying architecture is able to decompose higher-level goals within the task into sequences of actions, and to propagate them directly to the trajectory and motion control levels. Third, the system has been designed for fast and responsive for interactive scenarios. It operates autonomously, adapting its current goals and plan based on an estimate of the human's intentions obtained from sensor feedback. This opens the door to tackle sequential collaborative tasks with contact changes, as shown in the evaluation section. Future work will focus on more complex tasks, a more comprehensive model of the human's intentions, and an evaluation of our concepts in user studies.

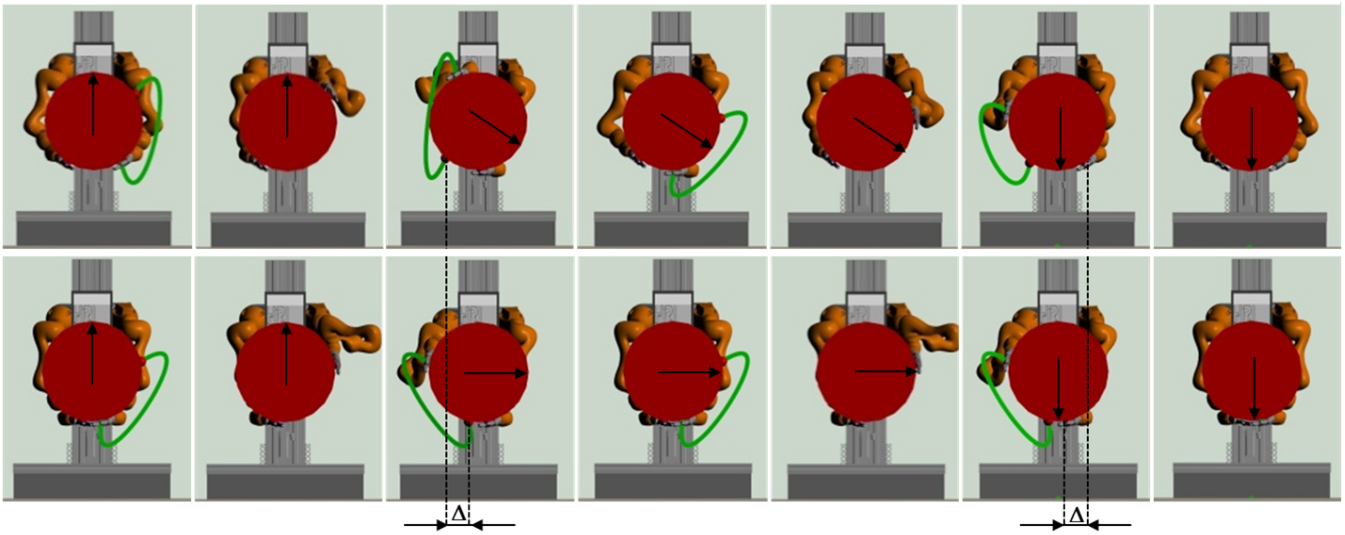

Fig. 9: Rotation of an object with high (*top*) and low (*bottom*) torque limits. It can be seen that a wider grasp can be chosen for the lower torque limit. The difference in hand positions marked by Δ.

Acknowledgment

The authors thank A. Bendig and D. Luttropp for their hardware and IT support.

References

[1] S. N. Yu, S. Y. Lee, C. S. Han, K. Y. Lee, and S. H. Lee, "Development of the curtain wall installation robot: Performance and efficiency tests at a construction site," *Autonomous Robots*, vol. 22, no. 3, pp. 281–291, 2007.

[2] T. Wojtara, M. Uchihara, H. Murayama, S. Shimoda, S. Sakai, H. Fujimoto, and H. Kimura, "Human–robot collaboration in precise positioning of a three-dimensional object," *Automatica*, vol. 45, no. 2, pp. 333–342, 2009.

[3] D. A. Abbink, M. Mulder, and E. R. Boer, "Haptic shared control: smoothly shifting control authority?," *Cognition, Technology & Work*, vol. 14, no. 1, pp. 19–28, 2012.

[4] W. Schwarting, J. Alonso-Mora, L. Pauli, S. Karaman, and D. Rus, "Parallel autonomy in automated vehicles: Safe motion generation with minimal intervention," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1928–1935, IEEE, 2017.

[5] D. Gopinath, S. Jain, and B. D. Argall, "Human-in-the-loop optimization of shared autonomy in assistive robotics," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 247–254, 2017.

[6] A. Mörtl, M. Lawitzky, A. Kucukyilmaz, M. Sezgin, C. Basdogan, and S. Hirche, "The role of roles: Physical cooperation between humans and robots," *Int J Rob Res*, vol. 31, pp. 1656–1674, nov 2012.

[7] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *International Symposium on Robot and Human Interactive Communication RO-MAN*, pp. 962–967, IEEE, 2012.

[8] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 3, pp. 2985–2991, IEEE, 2003.

[9] V. Duchaine and C. M. Gosselin, "General model of human-robot cooperation using a novel velocity based variable impedance control," in *Second Joint EuroHaptics Conference, 2007 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2007.*, pp. 446–451, IEEE, 2007.

[10] J. Dumora, F. Geffard, C. Bidard, T. Brouillet, and P. Fraisse, "Experimental study on haptic communication of a human in a shared human-robot collaborative task," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5137–5144, IEEE, 2012.

[11] V. V. Unhelkar, X. J. Yang, and J. A. Shah, "Challenges for communication decision-making in sequential human-robot collaborative tasks," in *Workshop on Mathematical Models, Algorithms, and Human-Robot Interaction at R:SS*, 2017.

[12] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4326–4332, IEEE, 2011.

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

[14] L. Peternel, T. Petrič, E. Oztop, and J. Babič, "Teaching robots to cooperate with humans in dynamic manipulation tasks based on multi-modal human-in-the-loop approach," *Autonomous Robots*, vol. 36, no. 1-2, pp. 123–136, 2014.

[15] L. Rozo, S. Calinon, D. G. Caldwell, P. Jimenez, and C. Torras, "Learning physical collaborative robot behaviors from human demonstrations," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 513–527, 2016.

[16] D. J. Agravante, A. Cherubini, A. Bussy, and A. Kheddar, "Human-humanoid joint haptic table carrying task with height stabilization using vision," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 4609–4614, IEEE, 2013.

[17] N. Vahrenkamp, D. Berenson, T. Asfour, J. Kuffner, and R. Dillmann, "Humanoid motion planning for dual-arm manipulation and regrasping tasks," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 2464–2470, IEEE, 2009.

[18] S. Tonneau, A. Del Prete, J. Pettré, C. Park, D. Manocha, and N. Mansard, "An efficient acyclic contact planner for multiped robots," *IEEE Transactions on Robotics*, 2018.

[19] A.-C. Hildebrandt, M. Klischat, D. Wahrmann, R. Wittmann, F. Sygulla, P. Seiwald, D. Rixen, and T. Buschmann, "Real-time path planning in unknown environments for bipedal robots," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 1856–1863, 2017.

[20] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, pp. 255–262, IEEE, 2000.

[21] C. Sung, T. Kagawa, and Y. Uno, "Whole-body motion planning for humanoid robots by specifying via-points," *International Journal of Advanced Robotic Systems*, vol. 10, no. 7, p. 300, 2013.

[22] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *Journal of neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[23] M. Gienger, M. Toussaint, and C. Goerick, "Whole-body motion planning–building blocks for intelligent systems," in *Motion Planning for Humanoid Robots*, pp. 67–98, Springer, 2010.