Demonstration based Learning and Control for Automatic Grasping

Johan Tegin and Jan Wikander Mechatronics Laboratory Machine Design KTH, Stockholm, Sweden E-mail: johant, jan@md.kth.se

Johan Tegin and Jan Wikander Staffan Ekvall and Danica Kragic

Computational Vision and Active Perception Laboratory KTH, Stockholm, Sweden E-mail: ekvall, danik@nada.kth.se Boyko Iliev Biologically Inspired Systems Lab. Applied Autonomous Sensor Systems Örebro University, Örebro, Sweden E-mail: boyko.iliev@tech.oru.se

Abstract—We present a method for automatic grasp generation based on object shape primitives in a Programming by Demonstration framework. The system first recognizes the grasp performed by a demonstrator as well as the object it is applied on and then generates a suitable grasping strategy on the robot. We start by presenting how to model and learn grasps and map them to robot hands. We continue by performing dynamic simulation of the grasp execution with a focus on grasping objects whose pose is not perfectly known.

I. INTRODUCTION

Robust grasping and manipulation of objects is one of the key research areas in the field of robotics. There has been a significant amount of work reported on how to achieve stable and manipulable grasps [1], [2], [3], [4], [5]. This paper presents a method for an initial grasp generation and control for robotic hands where human demonstration and object shape primitives are used in synergy. The methodology in this paper tackles different grasping problems but the main focus is on choosing the object approach vector, which is dependent both on the object shape and pose as well as the grasp type. Using the proposed method, the approach vector is chosen not only based on perceptional cues but also on experience that some approach vectors will provide useful tactile cues that finally result in stable grasps. Moreover, a methodology for developing and evaluating grasp schemes is presented where the focus lies on obtaining stable grasps under imperfect vision.

The presented methodology is considered in a Programming by Demonstration (PbD) framework [6], [7], where the user teaches the robot tasks by demonstrating them. The framework borrows ideas from the field of teleoperation, that provides a means of direct transmission of dexterity from the human operator. Most of the work in this field focuses however on low-level support such as haptic and graphical feedback and deals with problems such as time delays [8]. For instruction systems that involve object grasping and manipulation, both visual and haptic feedback are necessary. The robot has to be instructed what and how to manipulate. If the kinematics of robot arm/hand system is the same as for the human, a one-to-one mapping approach may be considered. This is, however, never the case. The problems arising are not only related to the mapping between different kinematic

chains for the arm/hand systems but also to the quality of the object pose estimation delivered by the vision system. Our previous results related to these problems have been presented in for example [9] and [10].

The contributions of the work presented here are: i) A suitable grasp is related to object pose and shape and not only a set of points generated along its outer contour. This means that we do not assume that the initial hand position is such that only planar grasps can be executed as proposed in [4]. In addition, grasps relying only on a set of contact points may be impossible to generate online since the available sensory feedback may not be able to estimate the exactly same points on the object's surface once the pose of the object is changed.

ii) The choice of the suitable grasp is based on the *experience*, i.e. it is learned from the human by defining the set of most likely hand preshapes with respect to the specific object. A similar idea, also using the Barrett hand [11], was investigated in [2]. Grasp preshapes are generated based on recognition of human grasps. This is of interest for humanoid robots where the current trend is to resemble human behavior as closely as possible.

iii) Finally, we evaluate the quality of different grasp types with respect to inaccuracies in pose estimation. This is an important issue that commonly occurs in robotic systems. The reasons may be that the calibration of the vision system or hand-eye system is not exact or that a detailed model of the object is not available. We evaluate how big pose estimation error different grasp types can handle.

A. Related Work

The work on automatic grasp synthesis and planning is relevant to the ideas presented here [2], [3], [4], [5]. In automatic grasp synthesis, it is commonly assumed that the position, orientation, and shape of the object is known [2]. Another assumption is that it is possible to extract the outer contour of an object and then apply a planar grasp [4]. The work on contact-level grasps synthesis concentrates mainly on finding a fixed number of contact locations without considering the hand design [1], [12].

Taking into account hand kinematics and *a-priori* knowledge of the feasible grasps has been acknowledged as a more flexible and natural approach towards automatic grasp planning [13], [2]. In [13], a method for adapting a given prototype grasp of one object to another object,

such that the quality of the new grasp would be at least 75% of the quality of the original, was developed. This process however, required a parallel algorithm running on supercomputer to be computed in reasonable time. This clearly shows the need to reduce the solution space for grasping problems in order to reach solutions in an acceptable time. The method proposed in [2] presents a system for automatic grasp planning for a Barrett hand by modeling objects as sets of shape primitives; spheres, cylinders, cones, and boxes. Each primitive is associated with a set of rules to generate a set of candidate pre-grasp hand positions and configurations. Examples of robotic manipulation include [14], [15], and [16].

II. SYSTEM DESCRIPTION

In our work, robotic grasping is performed by combining a PbD framework with semi-autonomous grasping and performance evaluation. We assume that a task such as (pick up/move/put down) an object is first demonstrated to the robot. The robot recognizes which object has been moved as well as where using visual feedback. The magnetic trackers on the human hand, provide information that enables the robot to recognize the human grasp. The robot then reproduces the action [9]. The approach is evaluated using a modified and extended version of the robot grasp simulator GraspIt! [17] to allow for repetitive experiments and statistical evaluation. In the simulation experiments, we use the Barrett hand and a hybrid force/position control framework. It is shown how dynamic simulation can be used for building grasp experience, for the evaluation of grasp performance, and to establish requirements for the robot system.

The current components in our PbD system are:

- Object Recognition and Pose Estimation: Estimating the pose of an object before and after an action enables the system to identify *which* object has been moved *where*. For object recognition and pose estimation, Receptive Field Co-occurrence Histograms is used [18], [19]. It is assumed that the objects are resting on a table and can be represented by parameters (x, y and φ).
- Grasp Recognition: A data-glove with magnetic trackers provides hand postures for the grasp recognition system [10].
- 3) Grasp Mapping: An off-line learned grasp mapping procedure maps human to robot grasps as presented in Section II-A.
- 4) Grasp Planning: The robot selects a suitable grasp controller. The object will be approached from the direction that maximize the probability of reaching a successful grasp. This is presented in more detail in Section IV.
- Grasp Execution: A semi-autonomous grasp controller is used to control the hand from the planned approach position until a force closure grasp is reached, Section III.



Large Diameter Power Sphere

Fig. 1. Mapping a selection of grasps from Cutkosky's grasp hierarchy to three Barrett grasps. The robot hand configurations shown are the initial joint positions.

A. Grasp Mapping

It has been argued that grasp preshapes can be used to limit the large number of possible robot hand configurations. This is motivated by the fact that - when planning a grasp - humans unconsciously simplify the grasp choice by choosing from a limited set of prehensile postures appropriate for the object and task at hand [20]. Cutkosky [21] classified human grasps and evaluated how the task and object geometry affect the choice of grasp. The work on virtual fingers generalized the existing grasp taxonomies [22]. Based on the above work and as described in previous work [10], the current grasp recognition system can recognize ten different grasp types. The human grasps are mapped to the robot as shown in Fig. 1. The grasps refer not only to hand postures, but to grasp execution schemes that include initial position, approach vector, and hand controller.

III. GRASP CONTROL

The grasp control algorithm has to be able to cope with occlusion and limited accuracy of the vision system. The controller presented here copes with the above without exploiting the wrist or arm motion. It is assumed that there are three touch sensors mounted on the distal links of the hand and that these can detect the normal force. This type of touch sensors are available at a low cost and are easy to mount to an existing robot hand as shown in previous work [23]. Considerations on different tactile sensors are put in to perspective in [24], [25], and [26].

The hybrid force/position controller uses these tactile sensors for control of the grasp force. Position control using joint encoders maintain the desired finger configuration and hence object position. Here, an effort is made to bring the object towards the center of hand during grasping. The need of this can be exemplified by the Barrett hand for



Fig. 2. Execution of a sample task where corrective movements are used to center the object.



Fig. 3. Grasp controllers is performed in x-space: total grasp force, stability, centering, and spread. The distal links have thin tactile sensors mounted to them (red).

which the grasp is typically of higher quality when all fingers have approximately the same closing angle, rather than when the object is far from the palm center. This behavior can be seen in the example task shown in Fig. 2. Here, the Barrett hand is modeled such that the two joint angles of each finger have a fixed relation. All control is performed using Matlab. Alternative low-level controllers have been investigated in e.g. [27].

A. Controller Design

To enable a more intuitive formulation of the controller – as opposed to decentralized control of reference trajectories and/or torques – a linear transform T is used to relate the original joint angles q to new more intuitive control variables x; x = Tq, see Fig. 3 and [28] for more detail.

Controlling spread (x_1) separately and the force, centering and stability according to Fig. 3, the transform becomes:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1/2 & 1/2 & 1\\ 0 & 1/2 & 1/2 & -1\\ 0 & 1 & -1 & 0 \end{bmatrix}.$$
 (1)

The control forces f are computed using a P-controller f = De where D contains controller gains and e is an error vector with force and position errors. The joint torques used to control the hand in simulation become $F = T^T f = T^T De$. The error e is computed using the desired [des] and actual [act] variable values as

$$e = \begin{bmatrix} e_1 & e_2 & e_3 & e_4 \end{bmatrix}^T$$

$$e_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} e_x$$

$$e_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} e_f$$

$$e_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} e_x$$

$$e_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} e_x$$

$$e_f = f_{des} - f_{act} = f_{des} - T^{-T} F_{act}$$

$$e_x = x_{des} - x_{act} = x_{des} - Tq_{act}.$$
(2)

IV. GRASP PLANNING

Each object is represented by its appearance (textural properties) for recognition and its *shape primitive* for grasp



Fig. 4. Up: The real objects. Mid: The modeled objects. Down: The object primitives used for training.

planning, Fig. 4. Recent progress presented in [29] shows a promising method for retrieving shape primitives using vision, although the method is restricted to objects with uniform color. The planning is performed using a simple search technique where many different approach vectors are tested on the object. The training can be performed on either the primitive object model or the full object model, and in the experiments we have evaluated both methods.

Two approaches were used for planning. For *Power Grasps* the hand was moved along an approach vector until or just before contact, and then the controller is engaged. For *Precision Grasps* the approach is the same, but with the addition that the hand is retracted a certain distance a number of times. After each retraction, the controller is engaged with the goal of reaching a fingertip (precision) grasp.

For power grasps, the three parameters (θ, ϕ, ψ) describing the approach direction and hand rotation are varied. For precision grasps, a fourth parameter *d* describes the retract increment. The number of evaluated values for the variables are $\theta=9$, $\phi=17$, $\psi=9$, d=6. For the precision grasps the search space was hence 8262 grasps which required about an hour of training using kinematic simulation. For the power grasp simulations, 1377 approach vectors were evaluated. The 5 s long grasping sequence is dynamically simulated in 120 s (Intel P4, 2.5 GHz, Linux). The quality measures for each grasp is stored in a *grasp experience* database.

A. Grasp Quality Measures

To evaluate grasps, the 6-D convex hull spanned by the forces and torques resistible by the grasp is analyzed using GraspIt!. The ϵ -L1 quality measure is the smallest maximum wrench the grasp can resist and is used for power grasps. For precision grasps, a grasp quality measure based on the volume of the convex hull was used, volume-L1. These grasp quality measures require full knowledge of the world, and can thus only be used in simulation.



Fig. 5. Left: The human moves the rice box. The system recognizes what object has been moved and which grasp is used. Right: The robot grasps the same object using the mapped version of the recognized grasp.

B. Grasp Retrieval

At run-time, the robot retrieves the approach vector for the highest quality grasp from the grasp experience database. As the highest quality grasp is not necessarily the most robust with respect to position and model errors, the grasp should be chosen taking also those parameters into account, see Sec. V-B. In a PbD scenario, the mapping from human to robot grasp is many-to-one. But if the robot acts autonomously, i.e. *explores* the environment and performs grasp on unknown objects, the grasp type is not defined and the best grasp can be chosen from among all possible grasps.

V. EXPERIMENTAL EVALUATION

This section provides experiments that demonstrate i) robot grasping given the current state of the environment and the *grasp experience* database, and ii) how errors in pose estimation affect the grasp success. The objects were placed on a table, Fig. 5 (left). Fig. 5 (right) shows the results of object recognition and pose estimation process. The human teacher, wearing a data-glove with magnetic trackers, moves an object. The move is recognized by the vision system and so is the grasp the teacher used. This information is used to generate a suitable robot grasp (grasp mapping) that controls the movement of the robot hand in the simulator.

A. Dynamic Simulation

Grasping the rice box with a fingertip grasp was dynamically simulated using the controller from Sections III and III-A. Of the 1377 approach vectors, 1035 were automatically discarded because the hand interfered with the table upon which the box is placed while approaching the object, or that the object was obviously out of reach. The remaining 342 initial robot hand positions were evaluated and resulted in 171 force closure grasps, 170 failed grasp attempts, and one simulation error. The top three hand initial positions and the resulting grasps are shown in Fig. 6. Some sample data from the third best simulation, Fig. 6 c) and f), is shown in Fig. 7. The desired grasping force is set to 5 N. A low-pass filter is used for the tactile sensor signal.



Fig. 6. The top three approach positions and the final grasps for fingertip grasping of the rice box. These results show that it is important to consider the dynamics when designing grasp execution schemes and for analyzing the grasp formation process. In several simulations the fingers stop after contacting the box (as they should), but when the grasping force is increased, the box slides on the low friction proximal links and also on the resting surface until it comes in contact with the high friction tactile sensors.



Fig. 7. Data logged from the grasp simulation in Fig. 6 c) and f). The first 1.3 seconds the fingers close under force control. The force at that time is used as the start value for the force controller that ramps the grasp force to 5 N. The joint angle values show that the joint angles are getting closer to equal with time.

B. Introducing Errors in the Pose Estimation

To evaluate the performance under imperfect pose estimation, we have simulated errors in pose estimation by providing an object pose with an offset. In the first simulation experiment, grasping the rice box with a fingertip grasp, the object was translated a certain distance in a *random* direction. As a result, the robot interpreted the situation as if the object was in another position than that for which the grasp was planned. This was repeated 50 times for five different vector lengths: 0, 1, 2, 3, and 4 cm. In total, 250 grasps from 201 positions. Figs. 9 and 10 show the grasp success rates for various grasps and objects, under increasing error in position estimation. A grasp is considered successful if it results in force-closure.

For the second experiment, the scenario was grasping a



Fig. 8. Grasp success as a function of initial bottle position. Grasp success is here defined as reaching a force closure grasp. The (x,y)=(0,0) position is right in front of the palm with a fraction of a millimeter of space between the palm and bottle. The inset shows the final grasp from (x,y)=(40,40).

bottle using a wrap grasp. In the initial position, the bottle was centered with respect to the palm and a fraction of a millimeter away. It was then re-positioned at positions in a grid with 20 mm spacing and another grasp was performed. For this scenario, the position of the bottle does not to be highly accurate, see Fig. 8.

Using kinematic simulation, we have evaluated how an error in rotation estimate affects the grasp formation. As expected, for symmetric objects like the orange and the bottle this type of error has no effect. Table I shows the rotation tolerance for different objects and grasp types. For grasping the mug, the rotation estimation is absolutely crucial. Thus, this type of grasp should be avoided for this object.

Object	Grasp Type	Rot. Err. Tolerance [degrees]
Zip Disc Box	Wrap	3
Rice Box	Wrap	17
Mug	Wrap	12
Mug	Precision Disc	0
Mug	Two Finger Thumb	6

TABLE I ROTATION ERROR TOLERANCES.

As expected, power grasps are more robust to position errors than precision grasps. The precision grasps target details of an object, e.g., the bottle cap or the ear of the mug. Thus, the grasps are much more sensitive to position inaccuracies. It is interesting to see that the dynamic simulation and the controller previously outlined yields significantly better results than that from purely kinematic simulation. This is a motivation for continuing the investigations on the dynamics of the grasp formation process.

The bottle and the mug have been trained both using a primitive model and using the real model. Training on the primitive model does not decrease the grasp success rate much, especially not for the bottle. However, the primitive



Fig. 9. The effect of position errors on the grasp success rate. For these results, the training and evaluation was performed using kinematic simulation only.



Fig. 10. The need for using dynamic simulation in grasp formation analysis is obvious. The grasp is the same as seen in Fig. 6 c) and f). (Due to some problems with the simulator, a limited number of samples were used in the evaluation of dynamic grasping. For the 0, 1, 2, 3, and 4 cm random displacement, the number of trials were 50, 14, 18, 18, and 12 respectively (instead of 50). Still, these samples were truly random and we believe that the number of trials is high enough to draw conclusions.)

model of the mug is, unlike the real mug, not hollow, which causes problems for some of the precision grasps trained on the primitive.

C. Discussion

The success rate of the presented system depends on the performance of four subparts: i) object recognition, ii) grasp recognition, iii) pose estimation of the grasped object, and iv) grasp execution. As demonstrated in previous papers, [18], [10], the object recognition rate for only five objects is around 100 %, and the grasp recognition ratio is about 96 % for ten grasp types. Therefore, the performance in a static environment may be considered close to perfect with respect to the first steps. As the object pose and possibly the object model is not perfectly known, some errors were introduced that indicate the needed precision in the pose estimation under certain conditions. Initial results suggest that for certain tasks, grasping is possible even when the object's position is not perfectly known.

If a high quality dynamic physical modeling is essential, for example when grasping compliant objects or for advanced contact models, other simulation tools than GraspIt! may be more suitable, see e.g. [30].

VI. CONCLUSIONS

A framework for generating robot grasps based on object models, shape primitives and/or human demonstration have been presented and evaluated. The focus lies on the choice of approach vector which depends on the object's pose and grasp type. The approach vector is based on perceptional cues and on experience that some approach vectors will provide better tactile cues that result in stable grasps. Another issue considered is obtaining stable grasps under imperfect vision, something that has not been thoroughly investigated in the literature.

Simulating results were necessary for generating insight into the problem and for performing the statistical evaluation for the grasp experience, since i) the world must be reset after each grasp attempt, and ii) computing the grasp quality measure requires perfect world knowledge. The proposed strategies have been demonstrated in simulation using tactile feedback and hybrid force/position control of a robot hand. The functionality of the proposed framework for grasp scheme design has been shown by successfully reaching force closure grasps using a Barrett hand and dynamic simulation.

Future work include further grasp execution scheme development and implementation. Furthermore, to ensure truly secure grasping outside the simulator, the grasping scheme must also comprise a grasp quality evaluation method that does not use information available in simulation only. Preferably such a measure would also depend upon the task at hand.

The grasp experience database contains not only a record of success rates for different grasp controllers but also the object-hand relations during an experiment. In this way, we can specify under what conditions the learned grasp strategy can be reproduced in new trials. The results of the experimental evaluation suggest that the outlined approach and tools can be of great use in robotic grasping, from learning by demonstration to robust object manipulation.

ACKNOWLEDGMENTS

This work was supported by EU through the project PACO-PLUS, FP6-2004-IST-4-27657, by the Knowledge Foundation through AASS at Örebro University, and by the Swedish Foundation for Strategic Research through the Centre for Autonomous Systems at KTH. Thanks to Johan Sommerfeld for helpful software.

REFERENCES

- A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'00, pp. 348–353, 2000.
- [2] A. T. Miller, S. Knoop, and H. I. C. P.K. Allen, "Automatic grasp planning using shape primitives," in *In Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1824– 1829, 2003.
- [3] N. S. Pollard, "Closure and quality equivalence for efficient synthesis of grasps from examples," *International Journal of Robotic Research*, vol. 23, no. 6, pp. 595–613, 2004.

- [4] A. Morales, E. Chinellato, A. H. Fagg, and A. del Pobil, "Using experience for assessing grasp reliability," *International Journal of Humanoid Robotics*, vol. 1, no. 4, pp. 671–691, 2004.
- [5] R. Platt Jr, A. H. Fagg, and R. A. Grupen, "Extending fingertip grasping to whole body grasping," in *Proc. of the Intl. Conference* on Robotics and Automation, 2003.
- [6] H. Friedrich, R. Dillmann, and O. Rogalla, "Interactive robot programming based on human demonstration and advice," in *Chris*tensen et al (eds.):Sensor Based Intelligent Robots, LNAI1724, pp. 96–119, 1999.
- [7] J. Aleotti, S. Caselli, and M. Reggiani, "Leveraging on a virtual environment for robot programming by demonstration," in *Robotics* and Autonomous Systems, Special issue: Robot Learning from Demonstration, vol. 47, pp. 153–161, 2004.
- [8] M. Massimino and T. Sheridan, "Variable force and visual feedback effects on teleoperator man/machine performance," in *Proc. of NASA Conference on Space Telerobotics*, 1989.
- [9] S. Ekvall and D. Kragic, "Learning task models from multiple human demonstration," in *IEEE International Symposium on Robot* and Human Interactive Communication, RO-MAN'06, 2006.
- [10] S. Ekvall and D. Kragic, "Grasp recognition for programming by demonstration," in *IEEE/RSJ IROS*, 2005.
 [11] W. Townsend, "The barretthand grasper – programmably flexible
- [11] W. Townsend, "The barretthand grasper programmably flexible part handling and assembly," *Industrial Robot: An International Journal*, vol. 27, no. 3, pp. 181–188, 2000.
- [12] D. Ding, Y.-H. Liu, and S. Wang, "Computing 3-d optimal formclosure grasps," in *In Proc. of the 2000 IEEE International Conference* on Robotics and Automation, pp. 3573 – 3578, 2000.
- [13] N. S. Pollard, "Parallel methods for synthesizing whole-hand grasps from generalized prototypes," 1994.
- [14] C. Borst, M. Fischer, S. Haidacher, H. Liu, and G. Hirzinger, "DLR hand II: Experiments and experiences with an antropomorphic hand," in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, pp. 702–707, Sept. 2003.
- [15] A. Namiki, Y. Imai, M. Ishikawa, and M. Kaneko, "Development of a high-speed multifingered hand system and its application to catching," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 3, pp. 2666–2671, Oct. 2003.
- [16] R. J. Platt, Learning and Generalizing Control-based Grasping and Manipulation Skills. PhD thesis, 2006.
- [17] A. T. Miller and P. Allen, "Graspit!: A versatile simulator for grasping analysis," in *Proceedings of the of the 2000 ASME International Mechanical Engineering Congress and Exposition*, 2000.
- [18] S. Ekvall and D. Kragic, "Receptive field cooccurrence histograms for object detection," in *IEEE/RSJ IROS*, 2005.
- [19] S. Ekvall, D.Kragic, and F. Hoffmann, "Object recognition and pose estimation using color cooccurrence histograms and geometric modeling," *Image and Vision Computing*, vol. 23, pp. 943–955, 2005.
- [20] J. Napier, "The prehensile movements of the human hand," in Journal of Bone and Joint Surgery, vol. 38B(4), pp. 902–913, 1956.
- [21] M. Cutkosky, "On grasp choice, grasp models and the desing of hands for manufacturing tasks," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 269–279, 1989.
- [22] T. Iberall, "Human prehension and dextrous robot hands," *The Int. J. of Robotics Research*, vol. 16, no. 3, 1997.
- [23] D. Kragic, S. Crinier, D. Brunn, and H. I. Christensen, "Vision and tactile sensing for real world tasks," *Proceedings IEEE International Conference on Robotics and Automation, ICRA'03*, vol. 1, pp. 1545–1550, September 2003.
- [24] R. D. Howe, "Tactile sensing and control of robotic manipulation," Advanced Robotics, vol. 8, no. 3, pp. 245–261, 1994.
- [25] M. H. Lee and H. R. Nicholls, "Tactile sensing for mechatronics - a state of the art survey," *Mechatronics*, vol. 9, pp. 1–31, Oct. 1999.
- [26] J. Tegin and J. Wikander, "Tactile sensing in intelligent robotic manipulation – a review," *Industrial Robot*, vol. 32, no. 1, pp. 64– 70, 2004.
- [27] R. Volpe and P. Khosla, "A theorethical and experimental investigation of explicit force control strategies for manipulators," *IEEE Trans. Automatic Control*, vol. 38, pp. 1634–1650, Nov. 1993.
- [28] J. Tegin and J. Wikander, "A framework for grasp simulation and control in domestic environments," in *IFAC-Symp. on Mechatronic Syst.*, (Heidelberg, Germany), Sept. 2006.

- [29] F. Bley, V. Schmirgel, and K. Kraiss, "Mobile manipulation based on generic object knowledge," in *IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN'06*, 2006.
 [30] G. Ferretti, G. Magnani, P. Rocco, and L. Viganò, "Modelling and simulation of a gripper with dymola," *Mathematical and Computer Modelling of Dynamical Systems*, 2006.