The Autonomous Picking & Palletizing (APPLE) Robot: A Research Platform for Intralogistics Applications

Robert Krug*, Todor Stoyanov*, Vinicio Tincani‡, Henrik Andreasson*, Rafael Mosberger*, Gualtiero Fantoni‡, Achim J. Lilienthal* and Antonio Bicchi‡

***Abstract*— Todo ...**

I. Introduction

[1](KIVA) [2](Logistics)

The increasing need for fast and flexible commissioning (i. e., order picking and collection of unstructured goods from storage compartments in warehouses) in logistic scenarios has created substantial interest for autonomous robotic solutions. This was also evidenced by a recent BBC investigation into a UK-based Amazon warehouse, which highlighted that the dull and strenuous nature of commissioning could cause mental and physical illness in human workers. Amazon themselves took action by organizing their first Picking Challenge at ICRA 2015.

The key obstacle for many application scenarios is the autonomous grasping in uncertain real-world environments. Currently, despite of a large research effort, no commercially viable solution is available for this problem. State of the art autonomous grasping systems [3], [4], [5] commonly employ sampling based planners [6] to generate online reach-to-grasp motion plans for offline planned grasps which are stored in a database. During the execution phase, such approaches necessitate many futile motion planning attempts which often incurs significant time delays mainly due to the frequent colli- sion checks which are necessary to avoid the robot coming in contact with itself or the environment. For APPLE, we adopted a real-time reactive control approach for manipulator mo- tion generation which allows to exploit redundancy, opposed to the commonly used sense-plan-act architectures which constrain all manipulator DoF. The main idea is to formulate a hierarchical set of tasks [7] such as move end-effector on this plane or avoid joint limits and to compute controls such that tasks of lower priorities are executed in the null-space of higher ranked tasks [8], [9].

The aim is to reduce the dependence on classical, sampling based motion planning and to move towards reactive feedback control to generate and execute complex motion behaviors of a robot. Here, only high-level behavioral goals (e. g., go to this region or stay above obstacle plane) are specified in form of task functions [7]. An intelligent control algorithm, which is based on embedded optimization of these task functions, then handles the details and synthesizes appropriate motions automatically in an online fashion. Opposed to classical sense-plan-act architectures, in this paradigm only task-relevant Degrees of Freedom (DoF) need to be constrained, which allows to exploit kinematic redundancies, e. g., for a manipulator to avoid unexpected obstacles. Regarding grasp planning, we follow the general tenet and will extract redundant representations in form of constrained pose intervals instead of discrete poses

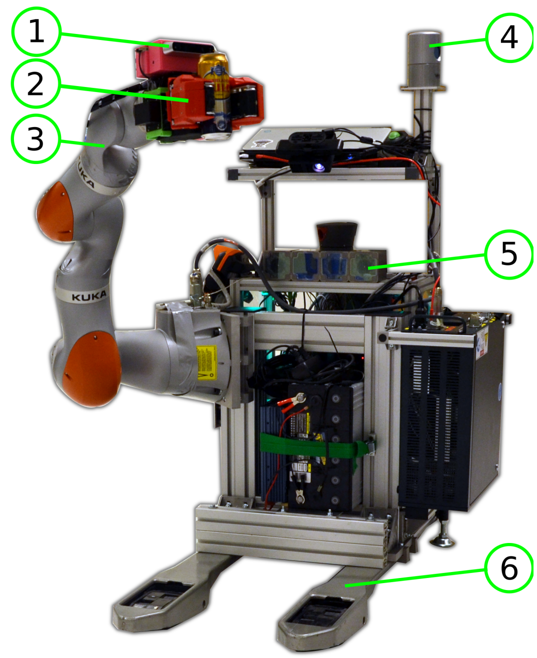


Fig. 1. *The APPLE demonstrator:* A KUKA LBR iiwa (3) is mounted on a CitiTruck AGV (6). A Velodyne Lidar (4) is used for localization, human worker detection is carried out with the RefleX camera system(5). The depicted Velvet Fingers gripper (2) is a further developed and smaller version of the gripper used in the FP7 IP project RobLog. Each of the grippers two fingers has a planar RR manipulator structure with two rotary joints and active surfaces which are implemented by conveyor belts on the inside of the two phalanges. The mechanical structure of each finger is underactuated and comprises one actuated Degree of Freedom (DoF) for opening and closing and two DoF for the belt movements. If, during grasping, the proximal phalanges are blocked by an object, the grippers distal phalanges continue to wrap around and envelope it in a firm grasp. Object and pallet detection is done with an ASUS Xtion Pro camera (1) which is mounted on the gripper's palm.

II. Autonomous Forklift

Introduce the CitiTruck + RefleX camera system

A. Navigation

This module ensures that the AGV is capable to move autonomously and safely through the workspace environ-

* AASS Research Center; Örebro University; Studentgatan 1, 70182 Örebro, Sweden.
‡ Interdepart. Research Center "E. Piaggio"; University of Pisa, Via Diotisalvi 2, 56100 Pisa, Italy.

ment. In order to achieve this task, we use components of a navigation system previously developed in the context of our KKS-funded Safe Autonomous Vehicles (SAUNA) project. We construct a 3D map of the static parts of the environment (using [10]) and use it to localize the vehicle in the presence of dynamic entities (using [11]). For motion planning and control of the non-holonomic AGV, we will our lattice planner [12] and a model-predictive tracking controller. The complete navigation system has been implemented, extensively tested and successfully integrated on the APPLE demonstrator, a detailed description can be found in [13].

B. People Detection

As the envisioned mobile manipulation system will operate in environments shared with human workers, people detection and human safety are important issues. In APPLE we address the problem by using the RefleX system we recently developed [14]. RefleX is a camera-based on-board safety system for industrial vehicles and machinery for detection of human workers wearing reflective vests worn as per safety regulations. The system was designed with industrial safety standards in mind and is currently being tested as an industrial prototype.

III. Grasping and Manipulation

A. Grasp Planning

Introduce LBR iiwa + Velvet Fingers 2 Gripper [15](data driven grasping) (SotA autonomous grasping systems) [16] cylinder shell fitting

The key challenge for many applications of robotic mobile manipulation is autonomous grasping in uncertain real-world environments. Finding collision-free trajec- tories leading the gripper-arm chain from a given initial to a reachable final state (grasp planning together with vehicle and manipulator motion planning) constitutes the fundament for any robot manipulation system. Currently, despite of a large research effort, no commercially viable solution is available for this problem. In todays state-of-the-art autonomous grasping systems [3], [4], [5], grasp planning and motion planning are usually seen as independent sub-problems [17]. A database storing object models together with pre-computed grasps is used to relax the need to find suitable gripper poses/configurations [18], [19], [5]. In the online stage, sampling based planners [6] attempt to generate valid trajectories for the pre-planned grasps, which are executed in a feasible-first manner [3]. During the execution phase, such approaches necessitate many futile motion planning attempts, which often incurs significant time delays since sampling based planners suffer from the curse of dimensionality. Also, while being able to solve complicated planning problems if given enough time, these planners dont scale well to geometrically simple scenarios [20] and they are ill suited to incorporate contact events with the environment.

Instead of representing grasps as discrete gripper wrist poses and joint configurations, we use grasp interval regions as depicted in Fig. 2. These grasp intervals can easily be transcribed as target tasks for the manipulator motion control and allow for redundancy in the manipulator wrist positioning which eases reach-to-grasp acquisition. Grasp interval formulation depends on the specific target object and has to be verified experimentally. For now, we constrain ourselves to cylindrical objects as shown in Fig. 2. We then rely on the inherent capabilities of the grasping device and the compliance of the system for successful grasp execution as stated below.

[21](Identifying grasp principles from humans) [22], [23], [24](Task maps with RRT)

B. Manipulator Motion Generation

[7](task functions) [8], [9](Redundant motion generation)

We lean on the notation in [25] and define the manipulator joint configuration by the vector $\boldsymbol{q}$ and the control inputs as corresponding joint velocities $\dot{\boldsymbol{q}}$. A task function is any derivable function of $\boldsymbol{e}$. To give an example, a task with the purpose of bringing an end-effector point $\boldsymbol{p}(\boldsymbol{q})$ onto a plane described by unit normal $\boldsymbol{n}$ and offset b can be transcribed by the task function $\boldsymbol{e} = \boldsymbol{n}^T \boldsymbol{p}(\boldsymbol{q}) - d$, which formulates the projection residual between the plane and $\boldsymbol{p}(\boldsymbol{q})$. The task evolution is given by $\boldsymbol{J}\dot{\boldsymbol{q}} = \dot{\boldsymbol{e}}$ with task jacobian $\boldsymbol{J} = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{q}}$.

Goal is to compute joint velocities such that the task evolution follows a desired reference profile $\dot{\boldsymbol{e}}^*$ (often chosen as exponential decay $\boldsymbol{e}^* = -\lambda \boldsymbol{e}$, with $\lambda \in \mathbb{R}_+$). For a single equality task, this necessitates to solve the following least squares Quadratic Program (QP):

$$\dot{q}^* \in \arg\min_{\dot{\boldsymbol{q}}} ||\boldsymbol{J}\dot{\boldsymbol{q}} - \dot{\boldsymbol{e}}^*||$$

In order to allow for inequality tasks, we henceforth use a general task formulation with upper bounds $\boldsymbol{J}\boldsymbol{q} \dot{\leq} \boldsymbol{e}^*$ as in [25]. This allows to transcribe lower bounds $\boldsymbol{J}\boldsymbol{q} \dot{\geq} \boldsymbol{e}^*$, double bounds $\underline{\dot{\boldsymbol{e}}}^* \leq \boldsymbol{J} q \leq \dot{\bar{\boldsymbol{e}}}^*$ and equalities $\boldsymbol{J}\boldsymbol{q} \dot{=} \boldsymbol{e}^*$ by reformulating the task respectively as $-\boldsymbol{J}\boldsymbol{q} \dot{\leq} -\boldsymbol{e}^*$, $\begin{bmatrix} -\boldsymbol{J} \\ \boldsymbol{J} \end{bmatrix} \dot{\boldsymbol{q}} \leq \begin{bmatrix} -\underline{\dot{\boldsymbol{e}}}^* \\ \dot{\bar{\boldsymbol{e}}}^* \end{bmatrix}$ and $\begin{bmatrix} -\boldsymbol{J} \\ \boldsymbol{J} \end{bmatrix} \dot{\boldsymbol{q}} \leq \begin{bmatrix} -\dot{\boldsymbol{e}}^* \\ \dot{\boldsymbol{e}}^* \end{bmatrix}$.

If the constraint $\boldsymbol{J}\boldsymbol{q} \dot{\leq} \boldsymbol{e}^*$ is infeasible, a least squares solution for $\boldsymbol{q}^*$ can be found by introducing the slack variable $\boldsymbol{w}$ in the decision variables

$$\begin{aligned} \min_{\dot{\boldsymbol{q}},\, \boldsymbol{w}} \quad & ||\boldsymbol{w}|| \\ \text{subject to} \quad & \boldsymbol{J}\dot{\boldsymbol{q}} \geq \boldsymbol{e}^* + \boldsymbol{w} \end{aligned} \tag{1}$$

For reactive on-the-fly motion generation we formulate a stack of hierarchical tasks and use the recently developed method by Kanoun et al. [26], which allows to account for inequality tasks and solves a sequence of convex optimization problems at each time step to obtain appropriate joint velocity commands (the method also can be used to directly generate torque commands while accounting for the robot dynamics [27]).

Obstacle avoidance is also achieved on a control-level, by formulating tasks which maintain minimum distances between simple geometric primitives such as spheres, planes,

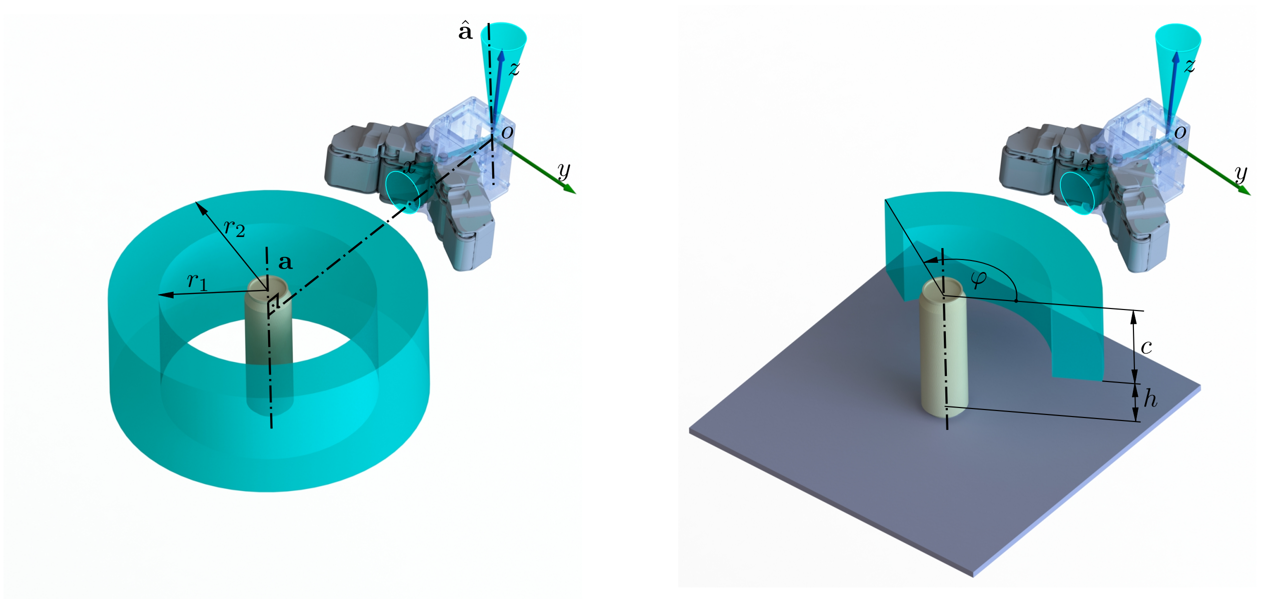


Fig. 2. *Grasp interval:* (Left) todo ... (Right) truncated grasp interval ...

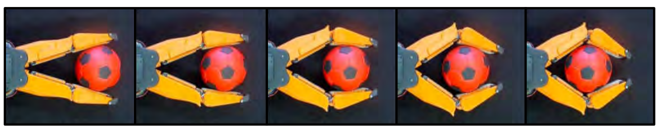


Fig. 3. *Pull-in grasping strategy:* Depicted is a sequence of intermediate grasp states where the belts of the gripper are used to pull the object towards its palm which results in a transition from a fingertip to an enveloping grasp.

points and capsules. We argue that for the considered application strict collision avoidance is neither necessary nor desired, since picking and manipulation inherently necessitates contact events between the robot and the environment. Also, in real-world applications where knowledge about the environment is available only in form of noisy sensor data, it might not be possible to avoid contacting the environment without being overly conservative. This makes the KUKA LBR iiwa with its compliant low-level control schemes and contact detection abilities an ideal platform for the tackled purpose and motion generation scheme. The relatively simple picking task in APPLE provides an ideal testbed in a real-world scenario.

[28](Task function descriptions)

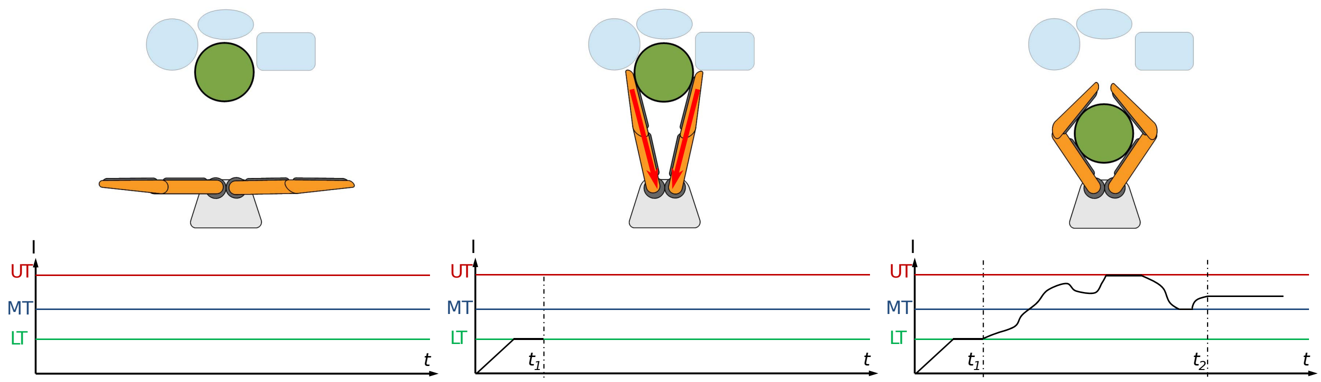
C. Robust Grasp Execution

For this component, we leverage the capabilities of the Velvet Gripper, namely underactuation and conveyor belts on the finger pads in order to achieve robust grasping behavior. Especially in cluttered scenes, a "pull-in" strategy has been shown to be especially effective to achieve stable grasps while starting from a relatively wide range of initial gripper poses with respect to the target object [5]. Here, the features of the grasping device are exploited to embrace the object in a firm envelope grasp by simultaneously squeezing it in a compliant fashion while actuating the belts inwards.

(the following is copy/pasted from the Gripper control workshop paper) [29] Each of the grippers two fingers has a planar manipulator structure with two joints and active surfaces which are implemented by coupled conveyor belts on the inside of the two phalanges. The mechanical structure is underactuated and comprises only one actuated Degree of Freedom (DoF) for opening and closing and two DoF for the belt movements. If, during grasping, the proximal phalanges are blocked by an object, the grippers distal phalanges continue to wrap around and envelope it in a firm grasp. The experiments reported in [5] showed, that in cluttered scenes fingertip grasps are more likely to be feasible than robust enveloping grasps, because the latter necessitate large opening angles resulting in bulky gripper silhouettes for which no collision free approach trajectories can be found. Therefore, we employ the pull-in strategy which is illustrated in Fig. **??**. Here, the underactuated nature and the conveyor belts on the grasping device are exploited to embrace the object in a firm envelope grasp by simultaneously squeezing it while actuating the belts inwards. This is achieved by compliant low-level position controllers which saturate on experimentally evaluated current thresholds. We use a simple grasping routine which is triggered after an initial fingertip grasp is achieved (see Fig. **??**). This routine consists of issuing commands to fully close the gripper while moving the belts a pre-defined offset towards the palm. Three thresholds on the current absorption of the opening motor are used: a low threshold (LT) signifies contact between the gripper and the object and a mid threshold (MT) indicates a large enough contact force to stop the closing movement. Finally, an upper threshold (UT) prevents damage to the grasping device. Once the pull-in sequence is completed, the controllers maintain the final torques to ensure a stable grasp.

IV. Evaluation

[30](Used off-the-shelf solver) [31](ROS)



(a) (b) (c)

Fig. 4. *Grasp Execution Control:* As the open griper closes in on the object (Left), the current through the opening motor is monitored. When contact is made (Middle), the actuated belts are switched on to pull in the object. The controller then strives to maintain the current in between two target thresholds by opening or closing the gripper during in-hand manipulation (Right).

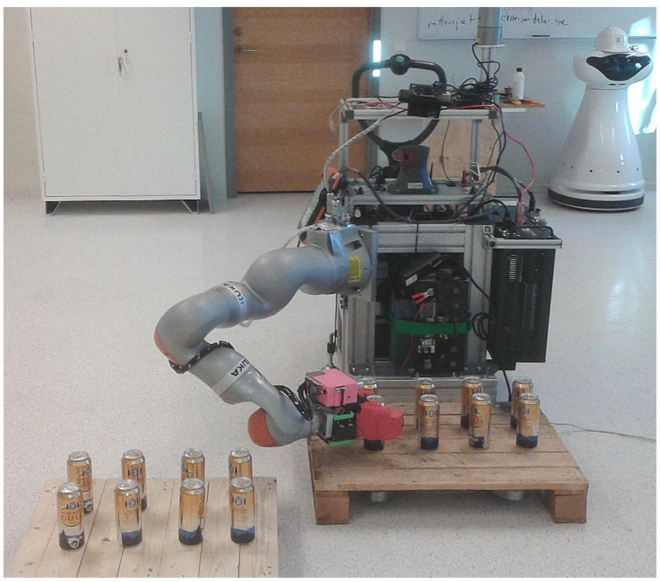


Fig. 5. *Demo setup:* The robot loads beer onto a previously picked up pallet and subsequently transports it to a predefined target location.

V. Discussion and Outlook

[32][33](optimal control for motion generation)

References

[1] P. R. Wurman, R. D'Andrea, and M. Mountz, "Coordinating hundreds of cooperative, autonomous vehicles in warehouses," *AI magazine*, vol. 29, no. 1, pp. 9–20, 2008.

[2] W. Echelmeyer, A. Kirchheim, and E. Wellbrock, "Robotics-logistics: Challenges for automation of logistic processes," in *Proc. of the IEEE International Conference on Automation and Logistics*, 2008, pp. 2099–2103.

[3] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *Proc. IEEE/RAS International Conference on Humanoid Robots*, 2007, pp. 42–48.

[4] S. Srinivasa, D. Ferguson, C. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. VandeWeghe, "Herb: A home exploring robotic butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.

[5] R. Krug, T. Stoyanov, M. Bonilla, V. Tincani, N. Vaskevicius, G. Fantoni, A. Birk, A. J. Lilienthal, and A. Bicchi, "Velvet fingers: Grasp planning and execution for an underactuated gripper with active surfaces," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2014, pp. 3669–3675.

[6] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, 2006.

[7] C. Samson, B. Espiau, and M. L. Borgne, *Robot control: the task function approach*. Oxford University Press, 1991.

[8] B. Siciliano and J.-J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. of the International Conference on Advanced Robotics*. IEEE, 1991, pp. 1211–1216.

[9] L. Sentis, J. Park, and O. Khatib, "Compliant control of multicontact and center-of-mass behaviors in humanoid robots," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 483–501, 2010.

[10] T. Stoyanov, J. P. Saarinen, H. Andreasson, and A. J. Lilienthal, "Normal distributions transform occupancy map fusion: Simultaneous mapping and tracking in large scale dynamic environments," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 4702–4708.

[11] R. Valencia, J. Saarinen, H. Andreasson, J. Vallve, J. Andrade-Cetto, and A. Lilienthal, "Localization in highly dynamic environments using dual-timescale ndt-mcl," in *Proc. of the IEEE International Conference on Robotics and Automation*, May 2014, pp. 3956–3962.

[12] M. Cirillo, T. Uras, S. Koenig, H. Andreasson, and F. Pecora, "Integrated motion planning and coordination for industrial vehicles," in *Proc. of the International Conference on Automated Planning and Scheduling*, 2014.

[13] H. Andreasson, J. Saarinen, M. Cirillo, T. Stoyanov, and A. J. Lilienthal, "Fast, continuous state path smoothing to improve navigation accuracy," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2015, to appear.

[14] R. Mosberger, H. Andreasson, and A. J. Lilienthal, "A customized vision system for tracking humans wearing reflective safety clothing from industrial vehicles and machinery," *Sensors*, vol. 14, no. 10, pp. 17 952–17 980, 2014.

[15] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesisa survey," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[16] A. t. Pas and R. Platt, "Localizing grasp affordances in 3-d points clouds using taubin quadric fitting," *arXiv preprint arXiv:1311.3192*, 2013.

[17] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Carnegie Mellon University, Robotics Institute, 2010.

[18] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics and Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[19] C. Goldfeder and P. K. Allen, "Data-driven grasping," *Autonomous Robots*, vol. 31, no. 1, pp. 1–20, 2011.

[20] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *Proc.*

of the IEEE International Conference on Robotics and Automation, 2009, pp. 489–494.

[21] R. Balasubramanian, L. Xu, P. Brook, J. Smith, and Y. Matsuoka, "Physical human interactive guidance: Identifying grasping principles from human-planned grasps," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 899–910, 2012.

[22] M. Gienger, M. Toussaint, and C. Goerick, "Task maps in humanoid robot manipulation," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 2758–2764.

[23] M. Gienger, M. Toussaint, N. Jetchev, A. Bendig, and C. Goerick, "Optimization of fluent approach and grasp motions," in *Proc. IEEE/RAS International Conference on Humanoid Robots*, 2008, pp. 111–117.

[24] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, "Integrated grasp and motion planning," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2010, pp. 2883–2888.

[25] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *International Journal of Robotics Research*, 2014.

[26] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 785–792, 2011.

[27] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 346–362, 2013.

[28] O. Kanoun, "Contribution à la planification de mouvement pour robots humanoïdes," Ph.D. dissertation, l'Université Toulouse III - Paul Sabatier, 2009.

[29] R. Krug, T. Stoyanov, M. Bonilla, V. Tincani, N. Vaskevicius, G. Fantoni, A. Birk, A. J. Lilienthal, and A. Bicchi, "Improving grasp robustness via in-hand manipulation with active surfaces," in *IEEE International Conference on Robotics and Automation - Workshop on Autonomous Grasping and Manipulation: An Open Challenge*, 2014.

[30] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2015. [Online]. Available: http://www.gurobi.com

[31] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source robot operating system," in *IEEE International Conference on Robotics and Automation - Workshop on open source software*, vol. 3, no. 3.2, 2009, p. 5.

[32] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 4906–4913.

[33] V. Kumar, Z. Xu, and E. Todorov, "Fast, strong and compliant pneumatic actuation for dexterous tendon-driven hands," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2013, pp. 1512–1519.