The Next Step in Robot Commissioning: Autonomous Picking & Palletizing

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

***Abstract*—So far, autonomous order picking (commissioning) systems have not been able to meet the stringent demands regarding speed, safety and accuracy of real-world warehouse automation, resulting in reliance on human workers. In this work we target the next step in autonomous robot commissioning: automatizing the currently manual order picking procedure. To this end, we investigate the use case of autonomous picking and palletizing with a dedicated research platform and discuss lessons learned during testing in simplified warehouse settings. The main theoretical contribution is a novel grasp representation scheme which allows for redundancy in the gripper pose placement. This redundancy is exploited by a local, prioritized kinematic controller which generates reactive manipulator motions on-the-fly. We validated our grasping approach by means of a large set of experiments, which yielded an average grasp acquisition time of $23.5\,$s at a success rate of $94.7\,\%$. Our system is able to autonomously carry out simple order picking tasks in a human-safe manner, and as such serves as an initial step towards future commercial-scale in-house logistics automation solutions.**

***Index Terms*—Logistics, Grasping, Autonomous Vehicle Navigation, Robot Safety, Mobile Manipulation;**

I. Introduction

THE increasing need for rapid order fulfillment and accuracy in supply chain processes has created a substantial interest for autonomous robotic solutions. Of special interest in logistics is autonomous commissioning (*i. e.*, order picking and collection of goods from storage compartments in warehouses). One of the main arguments for automating this task is that the dull and strenuous nature of commissioning can cause mental and physical illness in human workers. As a result, in order to increase the humanization and efficiency of workstations, the determination within the logistics sector to invest in this area is high [1].

There exist partial solutions for the automated commissioning problem in controlled environments such as the system described in [2]. It coordinates a fleet of Autonomous Ground Vehicles (AGVs) which transport shelves filled with goods to a human worker who picks the corresponding objects to complete the order. The envisioned next step will be to allow autonomous robots to cooperate with humans in solving the commissioning task and let simple picking procedures be handled automatically.

In this work, we conducted a use case evaluation of the following important sub-task chain, which occurs during commissioning in prototypical warehouses: autonomous picking of goods from a storage location, subsequent placement on a

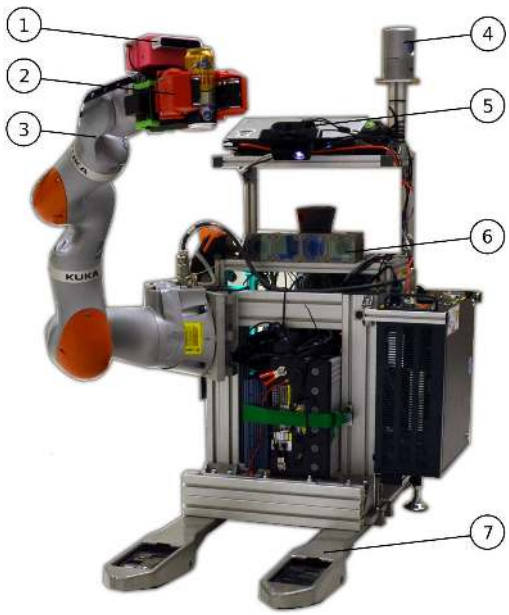


Fig. 1. *The APPLE Platform:* A retrofitted Linde CitiTruck[1] forklift AGV (7) comprises the mobile base. The nonholonomic AGV is able to detect and pick up pallets in designated loading zones using an Asus Xtion Pro Live[2] structured light camera (not depicted). For localization, a Velodyne HDL-32E[3] Lidar (4) is employed. On-board safety systems include a back-mounted SICK S300[4] laser scanner, a dedicated camera system (6) to track human workforce wearing reflective safety clothing and a projector which visually indicates the AGV's intended path (5). A lightweight KUKA LBR iiwa[5] manipulator (3), fitted with an under-actuated gripper (2) with conveyor belts on the inside of each finger, is used for robust grasping and object manipulation. Object detection is done with a Structure IO[6] device (1) which is mounted on the gripper's palm.

Manuscript received: August, 31, 2015; Revised November, 27, 2015; Accepted January, 8, 2016.

This paper was recommended for publication by Editor Antonio Bicchi upon evaluation of the Associate Editor and Reviewers' comments. The presented research has been supported by the EU's Seventh Framework Programme (FP7/2007-2013) under agreement ICT-270350 (RobLog) and the Swedish Knowledge Foundation under contract nr. 20140220 (AIR). The authors thank Antonio Bicchi for comments, as well as Per Sporrong, Bo-Lennart Silfverdal, Bengt Åsberg and Joakim Larsson for engineering support.

*R. Krug, T. Stoyanov, H. Andreasson, R. Mosberger and A. J. Lilienthal are with the AASS Research Center, Örebro University, Sweden. robert.krug@oru.se

‡V. Tincani and G. Fantoni are with the Interdepartmental Research Center "E. Piaggio", Pisa University, Italy. vinicio.tincani@for.unipi.it

Digital Object Identifier (DOI): see top of this page.

[1] http://www.citi-truck.com
[2] http://www.asus.com
[3] http://www.velodynelidar.com/hdl-32e.html
[4] http://www.sick.com/group/EN/home/products
[5] http://www.kuka-lbr-iiwa.com
[6] http://www.structure.io

standard EUR pallet and transport of the filled pallet to a target location. Our previous research efforts in logistics robotics [3], [4] indicated that human-safe AGV navigation, as well as the autonomous grasping/manipulation of unstructured goods at satisfactory cycle times are the key obstacles to overcome. This led to the development of the Autonomous Picking & Palletizing (APPLE) research platform shown in Fig. 1, which integrates our solutions to autonomous commissioning in environments shared with human workforce.

The main goal of this work is to outline possible solution strategies – developed in close collaboration with partners from industry – which will allow to overcome the current shortcomings of robotic commissioning systems in terms of speed, safety and accuracy. As most of our approaches to AGV navigation [5], [3] and human detection [6] have been published previously, here we only give a review and describe relevant extensions while focusing mainly on the manipulation aspect. The key insight is that the relatively simple structure of autonomous pick & place tasks allows fast and robust grasping by coupling compliant manipulator control strategies with a novel redundant grasp representation. This allows to substitute classical motion planning with online motion generation which omits the delays occurring in the commonly used sense-plan-act architectures. Furthermore, we propose simple compliant grasp routines that facilitate robust grasp acquisition by exploiting the structural features of our gripper as well as physical constraints imposed by the target objects. The suggested solutions are evaluated experimentally. We also report our efforts in integrating the isolated and human-safe sub-components which comprise the APPLE demonstrator and discuss our lessons learned during the Hannover Messe fair, where the platform conducted hundreds of successful commissioning tasks.

A. Positioning with Respect to Related Work

The industry standard for autonomous navigation of forklifts is to use pre-defined trajectories which are either manually defined or learned through teaching-by-demonstration from a human operator [7], [8]. Although conceptually simple, pre-defining trajectories limits pallet handling to occur only at pre-defined poses. In addition, only overly simple strategies for handling unforeseen obstacles can be applied. Compared to omni-directional mobile platforms, the fundamental difficulties of motion planning for a forklift lie in the nonholonomic constraints and the large sweep area it needs to occupy while operating in a limited work space.

In order to obtain reliable localization in large dynamic warehouses with high accuracy, it is common to mount reflectors, typically at a height which is not blocked by goods or other dynamic objects, and to use a dedicated sensing device [9]. Without additional infrastructure, navigation in large and dynamic environments remains a challenge.

Detecting pallets at unknown/uncertain poses has been studied previously [10]. However, to the best of our knowledge, no solution for a fully integrated autonomous pallet detection and picking system has been reported in the literature so far.

In current autonomous grasping systems [11], [12], [13], [4], grasp planning and manipulator motion planning are usually seen as independent sub-problems. A database storing object models together with sets of pre-computed grasps is used to find suitable gripper poses and joint configurations. In the online stage, sampling-based planners attempt to generate valid trajectories for the pre-planned grasps, which are executed in a feasible-first manner [11]. During the execution phase, such approaches necessitate many futile motion planning attempts, which often incurs significant time delays. To relax the problem, Berenson et al. [14] propose to represent equivalent grasps as a set of end-effector poses in order to exploit redundancy. They then generate manipulator trajectories by employing a randomized planner which samples the corresponding constraint manifolds. However, while being able to solve complicated planning problems if given enough time, sampling-based planners do not scale well to geometrically simple scenarios, often produce sub-optimal trajectories and are ill suited to incorporate contact events with the environment. Instead, as has been noted by Righetti et al. [15], interactions should be exploited during manipulation to facilitate dexterity. They report that a few pre-planned grasps suffice if interaction control schemes are coupled with optimization-based planning. Blurring the line between planning and control, Gienger et al. [16] advocate a holistic approach to grasp selection and online manipulator motion generation by employing an attractor-based control scheme to achieve grasps which are encoded in learned maps.

On a conceptual level, our work shares the ideas of exploiting grasp redundancy [14], [16] and online manipulator motion generation [16] to simplify/accelerate grasp acquisition, as well as using compliant interaction control to aid grasp robustness [15]. Our main contribution is a novel redundant grasp representation scheme tailored to on-the-fly manipulator motion generation. It is set apart from the conceptually similar continuous representations in [14], [16], which use box-constraints and object-specific maps respectively, in that we formulate general algebraic constraints to define valid grasp pose/configuration regions. This allows to encapsulate empirical knowledge about, $e.\,g.$, desired grasping device alignment. Such a constraint-based representation is appealing because, together with other constraints ($e.\,g.$, for obstacle and joint limits avoidance), it can directly be used in a multitude of constraint-based motion generation/planning methods ($e.\,g.$, the global trajectory optimization scheme employed in [15]).

As a second contribution we experimentally verify that a local optimization-based method to simultaneous manipulator motion generation and control, coupled with our redundant grasp representation and appropriate compliant grasp execution strategies, is sufficient to robustly solve the targeted autonomous pick & place problems without planning delays. Specifically, we use the kinematic control formulation by Kanoun et al. [17] which augments the popular Stack-of-Tasks (SoT) framework [18] with inequality tasks. Here, hierarchical tasks capturing the desired movement behavior are imposed as constraints on a series of underlying optimization problems which are solved at each time step to obtain the generalized inverse kinematics. Our compliant grasp controller which leverages underactuation and active surfaces of the utilized grasping device [19] has been presented in preliminary form

in [13] and [4]. In this article, we additionally give a statistical evaluation regarding grasp acquisition robustness based on a large set of executed grasps. Furthermore, we show how conceptually difficult cluster grasping problems can be solved reliably by exploiting compliance and physical constraints imposed by the environment [20].

II. Mobile Base

We chose a motorized forklift base (see Fig. 1), originally designed for manual operation, which provided the mechanical setup necessary for pallet picking out-of-the box. The forklift was retrofitted with a steering mechanism and a commercial AGV control system which allows manual control as well as fully autonomous employment at operational driving speeds of 1.5 m/s, which roughly corresponds to human walking speed. We review our solutions to autonomous navigation [21], [5], [3] in Section II-A and detail our measures to assure safe operation in the vicinity of human workers in Section II-B.

A. The APPLE Solution to Navigation

An industrial requirement for pallet picking is to achieve an end pose accuracy of ± 0.03 m in position and $\pm 1^\circ$ in orientation. Our navigation module ensures that these accuracy demands, as well as human-safety requirements are met. It comprises trajectory generation, tracking control, localization and a system for pallet detection.

Given start and goal poses, the first step in the online trajectory generation is to generate a discretized path, feasible for the nonholonomic forklift kinematics, using a lattice planner [22]. Then, we provide this path as an initial guess to a trajectory optimizer which generates a smooth, collision-free and continuous reference trajectory [5]. A model-predictive tracking controller allows to follow the generated trajectory with high accuracy. For localization, a Velodyne HDL-32E Lidar is utilized to construct a 3D map of the static parts of the environment [23]. The map and odometry information is then used to localize the vehicle in the presence of dynamic entities using a dual timescale approach [21].

The current system for pallet detection and pickup requires a rough estimate of the location of the pallet ($i.e.$, a predefined pickup zone). In order to compute the final pose based on sensory data from an Asus Xtion Pro Live mounted on the AGV, a Signed Distance Function (SDF) tracker [24] is used with a given SDF model of the pallet. The tracking is done while driving towards the pickup zone. Once the pallet pose is obtained with sufficient confidence, the driving trajectory is re-computed on the fly in order to pick the pallet.

B. AGV Safety

To assure safe operation, the vehicle conforms with the standard regulation for AGV systems according to EN 1525 and is equipped with a safety laser scanner which can directly brake the AGV independently of the navigation controller. In addition, the vehicle is equipped with an industrial prototype system for detecting human workforce [6]. Robust human detection is crucial to ensure a safe work environment. We address this problem by using a camera-based safety system that detects and tracks human workers wearing off-the-shelf reflective clothing as shown in Fig. 2.

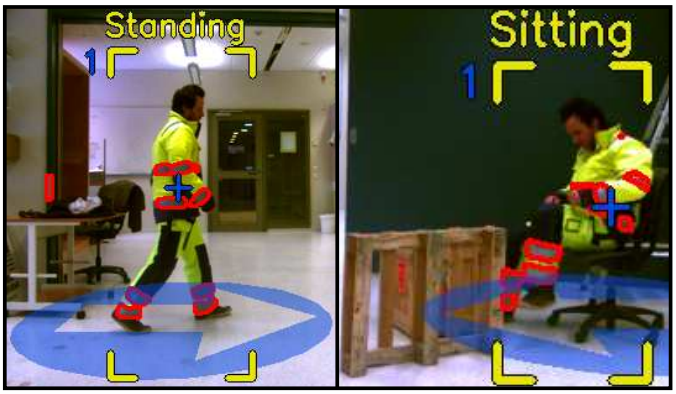


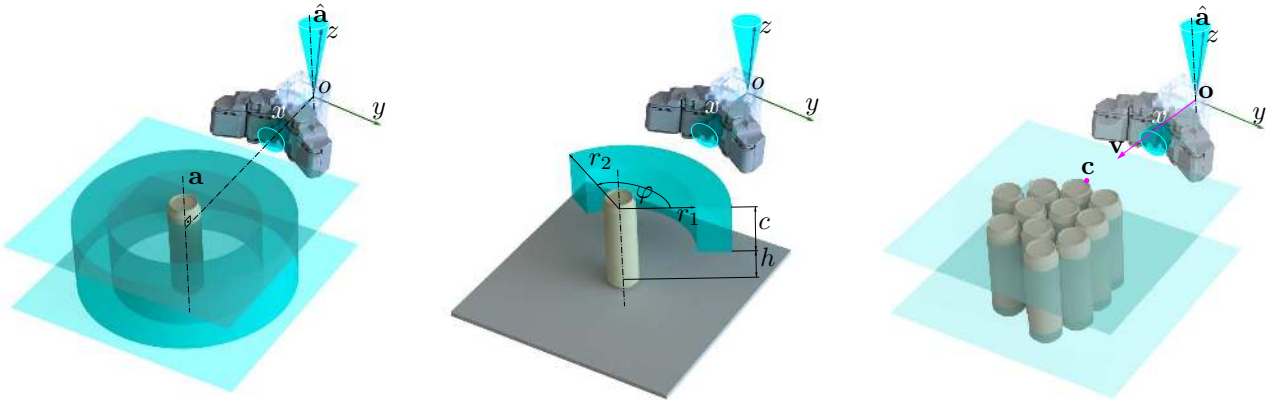
Fig. 2. *Human Detection:* Reflective markers on safety garments are detected and located at a coverage angle of 100 degree, in a range of 1–20 m. This enables the robot to detect and track humans in the close neighborhood for safe navigation. The system can estimate human positions/velocities and infer body pose information. Active near-infrared stereo vision in combination with reflective safety clothing ensures robust performance nearly independent of the illumination conditions (see [6] for details).

III. Grasping & Manipulation System

For an autonomous grasping system to be viable in logistics scenarios it needs to approach human working speed – a requirement which, based on our previous experience [13], [4], is not achievable with traditional sense-plan-act architectures. Also, the targeted use-case requires sharing the work environment with humans, which excludes stiffly position/velocity controlled manipulators as they might exert large forces to the environment. In this section, we address these issues with a novel integrative approach for grasp representation/planning and manipulator motion generation. The main idea is to provide a functional representation of grasps as intervals in task space, which allows redundancy in the gripper pose prior to executing the grasp as discussed in Section III-A below. In Section III-B we leverage the obtained redundancy to generate reactive motions by using the control framework in [17] instead of classical motion planning. Our approach relies on robust grasp execution strategies which we present in Section III-C before discussing safety aspects in Section III-D.

A. Grasp Representation and Planning

Grasps planned in simulation might fail in practice due to the inevitable uncertainties in modeling, perception and positioning, which make it impossible to precisely reproduce pre-planned hand poses/configurations and contact locations. This holds especially true, when considering an underactuated grasping device as we do in the presented work. In this case, the gripper's joint configuration depends on the interaction with the environment and is difficult or impossible to determine at planning time. Therefore, we represent grasps as pre-defined pose envelopes associated with primitive object geometries as exemplary shown in Fig. 3(a) for side grasps on cylindrical objects. Note that analogous envelopes can also be formed to enable, $e.g.$, top grasps. These envelopes



(a) Grasp Envelope (b) Truncated Grasp Envelope (c) Cluster Grasp Envelope

Fig. 3. *Grasp Constraints:* (a) The shaded cyan regions illustrate the side grasp envelope constraints for a cylindrical object. For a successful grasp, the palm frame origin $\boldsymbol{o}$ needs to lie inside the depicted cylindrical shell, which is aligned with object axis $\boldsymbol{a}$. The cylinders height is limited by two planes which are normal to $\boldsymbol{a}$. Additionally, the gripper's vertical axis (z) is constrained to lie in a cone whose axis $\hat{\boldsymbol{a}}$ is parallel to the object axis $\boldsymbol{a}$. Furthermore, the gripper's approach axis (x) has to lie inside a cone centered on the normal which connects axis $\boldsymbol{a}$ and point $\boldsymbol{o}$. (b) During the online stage, the corresponding grasp envelope shown in Fig. 3(a) needs to be truncated (*i. e.*, parameters for r_1, r_2, c, h and φ need to be determined) to accommodate the specific target object dimensions and to account for the fact that some regions of the grasp envelope might not be feasible due to obstruction by the environment. (c) The palm frame origin $\boldsymbol{o}$ needs to lie between the cyan shaded planes, the gripper's z-axis is constrained to lie in a cone whose axis $\hat{\boldsymbol{a}}$ is vertical. Furthermore, the gripper's approach axis (x) has to lie inside a cone centered on the horizontal approach vector $\boldsymbol{v}$ which points at the closest point $\boldsymbol{c}$ on the cluster ($\boldsymbol{c}$ is evaluated once at grasp approach movement initiation).

bound the grasping device's position and orientation, but do not fully constrain its pose. In this work, for simplicity, we limit ourselves to the illustrated grasp envelope defined for cylindrical shapes. Corresponding envelopes can be defined for other shape categories such as spheres and parallelepipeds as well. Opposed to the similarly defined representations in [16], which are learned in simulation, we deliberately design grasp envelopes to incorporate prior knowledge about robust grasp poses in our representation. In a concept originating from observations of human grasping behavior, it has been shown that the grasping device should be roughly aligned with the target object's principal components to achieve robust grasps [25]. For the case depicted in Fig. 3(a), this property is achieved by imposing cone constraints.

For evaluation purposes, the parameters of the grasp envelopes such as the distance range between gripper and object were determined experimentally in this work. To ease this non-trivial requirement, we rely on a gripper that offers a low pre-grasp pose sensitivity combined with a compliant and robust grasp execution routine, as discussed in Section III-C. During operation, after the target object pose is detected, the grasp envelope needs to be adapted to the specific scene and target object dimensions as illustrated in Fig. 3(b). For the evaluation in Section IV, we pre-defined the corresponding parameters and gripper pre-grasp joint configuration, an appropriate programmatic approach is subject of ongoing work.

B. Manipulator Motion Generation

We use a control-based approach to leverage the freedom in pre-grasp position and orientation gained from the previously described grasp representation scheme. Although our representation could be used in any constraint-based motion generation framework, we employ the method developed in [17] since it offers real-time capabilities. The grasp envelope constraints, as well as additional desiderata such as joint limit avoidance, are cast in form of task functions. These are subsequently utilized to form a hierarchical SoT which is then used to compute prioritized controls during movement execution. Therefore, motions are generated instantaneously without the planning delays occurring in sense-plan-act architectures. Since we directly use the method in [17], we only briefly revise the concept in the following and refer the reader to the original work for details.

We lean on the notation in [26] and describe the manipulator joint configuration with the vector $\boldsymbol{q}$ and the control inputs as corresponding joint velocities $\dot{\boldsymbol{q}}$. A task function $\boldsymbol{e}(\boldsymbol{q})$ is any derivable function of $\boldsymbol{q}$. 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 d can be transcribed by the task function $\boldsymbol{e}(\boldsymbol{q}) = \boldsymbol{n}^T\boldsymbol{p}(\boldsymbol{q}) - d$, 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 $\boldsymbol{J} = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{q}}$. The goal is to compute joint velocities such that the task evolution follows a desired reference profile $\dot{\boldsymbol{e}}^*$ (in this work chosen as exponential decay $\dot{\boldsymbol{e}}^* = -\lambda \boldsymbol{e}$, with $\lambda \in \mathbb{R}_+$). For a single equality task, this necessitates to solve the following least squares Quadratic Program (QP)

$$\dot{\boldsymbol{q}}^* \in \arg\min_{\dot{\boldsymbol{q}}} ||\boldsymbol{J}\dot{\boldsymbol{q}} - \dot{\boldsymbol{e}}^*||. \tag{1}$$

In order to allow for inequality tasks, we henceforth use a general task formulation with upper bounds

$$\boldsymbol{J}\dot{\boldsymbol{q}} \leq \dot{\boldsymbol{e}}^*. \tag{2}$$

As stated in [26], this allows to transcribe lower bounds $\boldsymbol{J}\dot{\boldsymbol{q}} \geq \dot{\boldsymbol{e}}^*$, double bounds $\underline{\dot{\boldsymbol{e}}}^* \leq \boldsymbol{J}\dot{\boldsymbol{q}} \leq \dot{\bar{\boldsymbol{e}}}^*$ and equalities $\boldsymbol{J}\dot{\boldsymbol{q}} = \dot{\boldsymbol{e}}^*$ by reformulating the task respectively as $-\boldsymbol{J}\dot{\boldsymbol{q}} \leq -\dot{\boldsymbol{e}}^*$, $\begin{bmatrix} -\boldsymbol{J} \\ \boldsymbol{J} \end{bmatrix} \dot{\boldsymbol{q}} \leq \begin{bmatrix} -\underline{\dot{\boldsymbol{e}}}^* \\ \dot{\bar{\boldsymbol{e}}}^* \end{bmatrix}$ 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 in (2) is infeasible, a least squares solution for $\dot{\boldsymbol{q}}^*$ as in (1) can be found by introducing the slack variable

$\boldsymbol{w}$ in the decision variables

$$\begin{aligned} \min_{\dot{\boldsymbol{q}},\, \boldsymbol{w}} \quad & ||\boldsymbol{w}|| \\ \text{subject to} \quad & \boldsymbol{J}\dot{\boldsymbol{q}} \leq \dot{\boldsymbol{e}}^* + \boldsymbol{w}. \end{aligned} \tag{3}$$

To form a hierarchical SoT with $p = 1, \ldots, P$ priority levels, we stack all task jacobians in (2) with the same assigned priority in a matrix $\boldsymbol{A}_p$, and all corresponding reference velocities in a vector $\boldsymbol{b}_p$ to form one constraint of the form $\boldsymbol{A}_p\dot{\boldsymbol{q}} \leq \boldsymbol{b}_p$ for each hierarchy level. The aim is to sequentially satisfy a constraint at best in the least square sense while solving for the subsequent constraints of lower priority in the null-space of the previous constraint, such that the previous solution is left unchanged. Therefore, the following QP, where the previous slack variable solutions $\boldsymbol{w}_i^*$ are frozen between iterations, needs to be solved for $p = 1, \ldots, P$

$$\begin{aligned} \min_{\dot{\boldsymbol{q}},\, \boldsymbol{w}_p} \quad & ||\boldsymbol{w}_p|| \\ \text{subject to} \quad & \boldsymbol{A}_i\dot{\boldsymbol{q}} \leq \boldsymbol{b}_i + \boldsymbol{w}_i^*, \quad i = 1, \ldots, p-1 \\ & \boldsymbol{A}_p\dot{\boldsymbol{q}} \leq \boldsymbol{b}_p + \boldsymbol{w}_p. \end{aligned} \tag{4}$$

The control vector $\dot{\boldsymbol{q}}^*$ is then obtained from the P^{th} solution of (4). Essentially, a sequence of instantaneous optimal control problems is solved at each time-step. The method is local in that it does not account for the temporal state evolution. However, for the targeted relatively simple autonomous pick and place operations, we found the performance satisfactory as demonstrated in Section IV. Opposed to sampling-based planners, the chosen control scheme can easily incorporate qualitative requirements ($e.\,g.$, the desired gripper alignment) during motion generation and allows for redundancy exploitation via appropriate task function formulations. For the case of grasp pose and configuration control, these task functions incorporate our grasp envelopes as exemplary shown in Fig. 3.

Manipulator obstacle avoidance is also achieved on a control level, by formulating tasks which maintain minimum distances between simple shape primitives such as spheres, planes, points and capsules which are used to model the robot and environment geometries. In real-world applications, where knowledge about the environment is available only in form of noisy sensor data, it might not be possible to strictly avoid contact with the environment without being overly conservative. Instead, we argue that for the considered application strict collision avoidance is neither necessary nor desired, since picking and manipulation inherently necessitate contact events which can even be exploited to achieve robust grasping behavior as discussed in Section III-C. To this end, the APPLE platform relies on compliant low-level control schemes. Regarding the manipulator, we use a joint impedance controller to track the reference velocities obtained in each time step by solving (4). If a sufficiently accurate dynamic model were available, the scheme could easily be adapted to torque control [26]. However, we are not considering this option at present.

C. Robust Grasp Control & Execution

For this component, we exploit the capabilities of the utilized gripper [19], namely underactuation and active surfaces on the finger pads in order to achieve robust grasping behavior. Each of the grippers two fingers has a planar 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 is underactuated and comprises only one actuated Degree of Freedom (DoF) for opening and closing and two DoF per finger 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 we reported in [13] 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 which hamper the grasp approach. Therefore, we employ the "pull-in" strategy which is illustrated in Fig. 4. As demonstrated previously [13], this strategy is very effective in achieving stable grasps from a wide range of gripper pre-grasp poses.

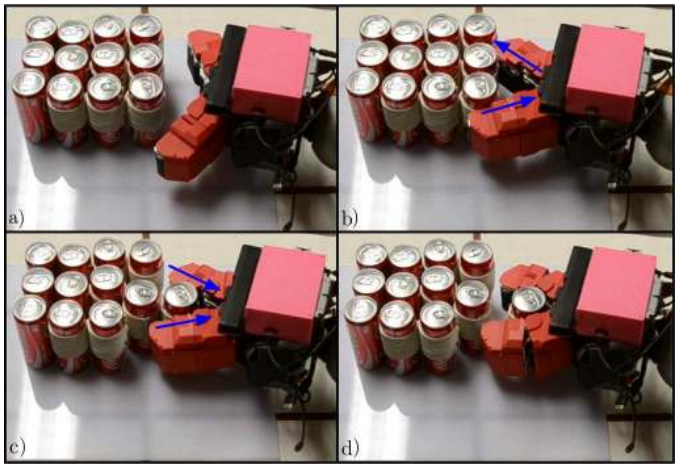


Fig. 4. *Cluster Grasping with Active Surfaces:* (a) The grasp pose is obtained by detecting the depicted cluster from a pre-defined sensing pose, forming a grasp envelope as illustrated in Fig. 3(c) and posing corresponding constraints to the manipulator control in (4), which generates the approach motion reactively. To allow passive wrist pose adaption during grasp execution, the corresponding Cartesian stiffness is set low in the horizontal plane. Finger closing is triggered by setting a relatively low current setpoint to the open/close motor controller. (b) Once the fingers come into contact with the object and the target current is reached, the motion stops which is detected by monitoring the corresponding motor encoder. If, as in the depicted case, the opening angle indicates a grasp on multiple objects, an optional separation step is triggered where the belts move in opposite directions. (c) The belts are actuated to pull in the target object. (d) Once the belts block and the phalanges have wrapped around the object an enveloping grasp is achieved, after which a higher current setpoint is given to the open/close motor controller to ensure a firm grasp.

The grasping controller was implemented by means of low-level current control of the gripper's DC-drives. Since the gripper drives have a low transmission ratio and are easily back-driveable, current control enables a simple compliant behavior because the current absorption increases with increasing effort on the output. Thus, the motor current is proportional to the resulting grasping force. The main parameters to this routine (the current thresholds for contact and final enveloping grasp) depend on the target object properties – namely friction coefficient and mass. Here, good values for these parameters were found experimentally for a set of target objects.

TABLE I
GRASP ACQUISITION EVALUATION

Scenario	# of exp.	Success	Rate [%]	$\sigma^c_{1,2}\,[cm]$	$\lvert\sigma^d_{1,2}\rvert\,[cm]$	$l\,[rad]$	$t^p\,[s]$	$t^m\,[s]$	$\sum t\,[s]$
Beer rct.	25	22	88.0	1.1, 0.8	6.8, 5.9	5.2 ± 0.3	7.3 ± 1.0	9.0 ± 0.6	16.3 ± 0.7
Beer rd.	25	23	92.0	2.5, 1.4	6.9, 5.1	5.4 ± 0.4	3.4 ± 0.3	11.1 ± 0.6	14.5 ± 0.7
Coke rct.	25	25	100	3.0, 2.2	6.4, 5.4	5.6 ± 0.1	3.0 ± 0.2	11.2 ± 0.5	14.2 ± 0.5
Coke rd.	25	24	96.0	2.6, 1.4	7.3, 5.4	5.1 ± 0.3	3.0 ± 0.3	11.5 ± 0.6	14.5 ± 0.6
Bull rct.	25	25	100	3.2, 2.3	5.5, 4.8	5.7 ± 0.2	2.4 ± 0.2	11.1 ± 0.5	13.5 ± 0.5
Bull rd.	25	23	92.0	2.2, 1.4	6.7, 5.0	5.1 ± 0.3	2.4 ± 0.2	11.5 ± 0.5	14.0 ± 0.5
Total	150	142	94.7	3.9, 2.6	6.6, 5.3	5.3 ± 0.4	3.6 ± 1.7	10.9 ± 1.0	14.5 ± 1.0

Rather than picking a specific object, commissioning often involves the problem of picking one instance from a cluster of identical objects. Trying to solve this task by identifying and selecting an appropriate target object poses serious challenges to perception and grasp/manipulation planning. We devise a novel, robust solution to this problem by leveraging our grasp representation framework and compliant interaction control to exploit contact between grasping device and environment. To this end, we formulate the grasp envelope constraints illustrated in Fig. 3(c). In practice, this results in a substantial speed-up due to the simplified perception problem since we only detect the cluster of objects and not each separate object instance. During grasp execution, after approaching the target cluster with the gripper fully opened, we set the manipulator wrist impedance low in a horizontal plane while executing the previously described pull-in strategy. This allows the end-effector to automatically adjust its pose based on the occurring grasp forces (see Fig. 4), a strategy which proved to be very effective as shown in the evaluation in Section IV.

D. Grasping & Manipulation Safety

Specific safety standards for collaborative robots are still not fully defined, although a corresponding technical specification (ISO TS 15066) is in development. Current guidance by the National Institute of Standards and Technology (NIST) includes power and force limiting. We use a light-weight manipulator, which reduces the kinetic energy during motion, limiting potential impact forces. Also, interaction forces are limited by employing impedance controllers as discussed above. Additionally, the manipulator's joint torque sensors allow a fast emergency stop if forces anywhere on its surface exceed given thresholds. Due to current control and low gear transmission ratios, the grasping device is back-driveable. This ensures compliant behavior and allows to restrict the generated forces by limiting the actuator currents.

IV. EVALUATION

The previously described software components were implemented in the Robot Operating System (ROS)[7] framework. An off-the-shelf solver[8] is used to carry out the optimizations for the motion control according to (4). A rigorous evaluation of the full system would need to consider a staggering variety of environment and task parameters in order to produce generalizeable results. Thus, instead of attempting to holistically evaluate the performance of the APPLE system in an uncontrolled warehouse environment, it is more prudent to validate the performance of each sub-system in isolation. The successful integration between the described sub-systems then depends on the application specifics and target requirements. Of these components, only the manipulation framework has not undergone prior evaluation (for tests of the navigation system see [21], [5], [3], for the AGV safety features see [6]).

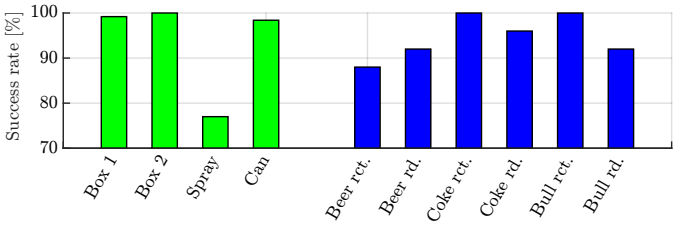


Fig. 5. *Grasp Success Rates:* (Left) Grasp control evaluation rates depicted in green; (Right) Full grasp acquisition evaluation rates shown in blue; Unsuccessful attempts were mainly due to failure in acquiring a firm envelope grasp leading to slippage in the subsequent object lifting phase. This happened when, during pull-in (*cf.* Fig. 4), the object got stuck on the hinges between proximal and distal phalanges which are not covered by the active surfaces.

The manipulation system was evaluated with a stationary base in a two-step manner: first, we characterized the robustness of our grasp control scheme in isolation. To this end, we set up a grid of initial end-effector poses and attempted grasping and lifting of four different target objects – two square boxes (146 g/45 mm and 163 g/50 mm), a spray bottle (127 g/⌀52 mm) and a can (361 g/⌀53mm). The gripper was placed within a box of 10 x 8 x 4 cm and roll/pitch angles of $\pm 10°$, resulting in 121 to 161 different grasps per object for a total of 526 grasps. A grasp was deemed successful if the object could be lifted. Figure 5 visualizes the corresponding success rates. The overall success rate was $92.4\,\%$, with the spray bottle resulting in more failures under gripper perturbation due to its irregular shape.

Second, we evaluated the full grasp acquisition procedure for cluster grasping, starting from a pre-defined sensing pose. First, cluster detection, segmentation and pose estimation were carried out with standard algorithms from the Point Cloud Library[9]. Subsequently, grasp approach motions were generated as described in Section III-B, followed again by grasp execution and a brief lifting phase to determine grasp

[7] http://www.ros.org
[8] http://www.gurobi.com
[9] http://www.pointclouds.org

Fig. 6. *Testing:* (Left) The robot picks up an empty pallet in a designated zone; (Middle) The robot navigates to a loading zone where a can is detected and picked up; (Right) The loaded pallet is transported to a target location (see the video attachment to this article).

success. We formed tightly packed clusters of 3 different object types each in 2 different configurations – rectangular (as in Fig. 4) and pseudo-random for a total of 6 scenarios with 25 test runs each. The utilized objects were Beer (522 g/⦰66 mm), Coke (358 g/⦰58 mm) and Red Bull (272 g/⦰53 mm). The full results are listed in Table I. Success rates (separately visualized in Fig. 5) reveal a satisfactory outcome across all scenarios with an overall rate of $94.7\,\%$, which is comparable to the previous set of experiments. This indicates that the approach is robust to variations in perception and grasp approach. To give an idea about the horizontal variability of cluster positioning and dimensions for each scenario, Table I also lists the standard deviations of the cluster center points $\sigma^c_{1,2}$, as well as the standard deviation means of the points forming the respective clusters $|\sigma^d_{1,2}|$. Both quantities are measured along the eigenvectors of their respective covariance matrices. We also measured mean/std perception execution times t^p and grasp approach movement durations t^m which are consistent across all test runs – we attribute the seemingly longer perception times in the first scenario to the fact that computationally expensive visualization was running on the same machine for this case. Grasp execution duration is roughly constant at either $6\,s$ or $12\,s$, depending on whether the optional object separation step is triggered when multiple objects are grasped initially (see Fig. 4). Therefore, in total, the full grasp acquisition procedure takes on average $23.5\,$s. A direct comparison to previous works is difficult due to the vast differences in hardware/experimental scenario setup and the fact that few works actually report complete system run times. Compared to our own previous efforts tackling shipment container unloading (with a very different hardware setup) using a database of 500 pre-planned grasps and a sampling based planner [13], [4], the presented approach offers a speed-up factor of roughly 5. This is despite the fact that in the present work we used very conservative movement speeds and spent little effort in parameter tuning. The authors of [15] report average run times of $75.4\,$s in their grasping/manipulation experiments but the specific scenario setup is unclear. Lastly, we computed the joint travel distance summed over all manipulator joints $l = \int_0^{t^m} \|\dot{\boldsymbol{q}}(t)\|_1 dt$. It is evident, that the approach motions were consistent and repeatable to a high degree across all experiments.

Finally, to asses the feasibility of the targeted use case, we conducted test runs in a simplified commissioning scenario as depicted in Fig. 6. The corresponding flow chart in Fig. 7 outlines the temporal sequence of events in these test runs. We carried out successful trials in two sample system deployments: one in our lab, and the second at the week-long industrial fair Hannover Messe. In the second case, the APPLE system was in continuous operation inside a 3 x 5m confined area, performing hundreds of successful commissioning tasks. Run-times for the procedure illustrated in Fig. 7 were approximately 2–3 minutes, depending on driving distance, of which roughly 1 minute was spent on object detection and manipulation. In the trials the major sub-components of the APPLE system – navigation, manipulation and safety – all satisfied the posed requirements.

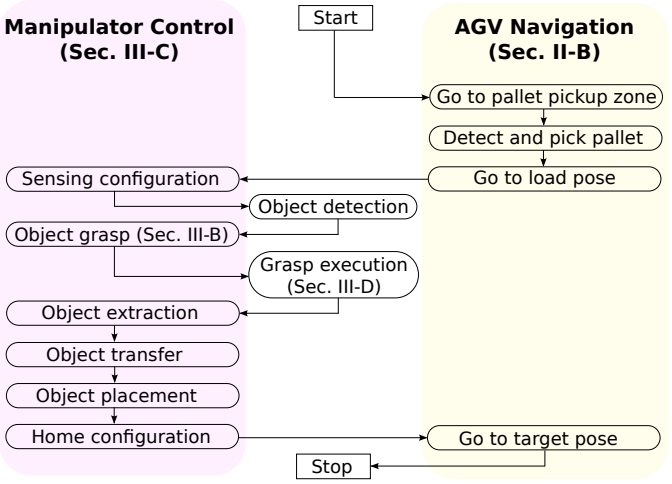


Fig. 7. *Test Run Flow Chart:* After navigating to a pickup zone, which is established in advance, a pallet is detected and picked as described in Section II-A. Subsequently, the robot drives to a given loading location and the manipulator assumes a pre-defined sensing configuration which allows to observe the target objects with the palm-mounted camera. After object detection, the pick & place procedure is sub-divided and carried out by controlling the manipulator under obstacle constraints and sequentially different constraint sets for grasping (using constraints according to either Fig. 3(a) or Fig. 3(c)), object extraction (lifting the object above a given plane), object transfer (move the object into a given cylindrical region over the pallet) and object placement (lowering the object with low cartesian wrist impedance in vertical direction while maintaining the cone constraint on the gripper's vertical axis). After moving the manipulator to its home configuration, the loaded pallet is transported to a set target location.

V. Conclusions and Lessons Learned

In this paper, we target the next step in autonomous robot commissioning: automatizing the order picking procedure. Our

work is based on the use case of autonomous picking and palletizing (APPLE) for which we developed a dedicated research platform, in order to assess the feasibility of solving the task under stringent industry requirements regarding speed, accuracy and safety. The APPLE robot carries safety systems for the detection and avoidance of human workers wearing reflective safety clothing. The developed manipulation system for loading/unloading unstructured goods from pallets operates on a novel redundant grasp representation, which allows to incorporate empirical knowledge. The key insight is that exploiting grasp redundancy and compliant interaction control schemes allows to robustly solve simple pick & place tasks by leveraging fast, local motion generation based on real-time control rather than explicit motion planning. We provide an experimental validation of the APPLE grasping/manipulation sub-system and demonstrate successfully executed trials of a simplified commissioning task (see the video attachment).

Based on our experience with AGV navigation, trajectory generation and tracking need to be treated in a holistic way to achieve the required navigation accuracy. To this end, we leverage trajectory optimization, which allows for a better subsequent tracking performance due to the limited amount of control actions required (one of the optimization criteria). In turn, smooth execution with few changes in the control commands leads to higher end pose accuracies and less jerky movements.

Regarding grasping, compared to our previous experience with sense-plan-act architectures [13], [4], the presented reactive manipulator motion generation scheme resulted in a tremendous speed-up, while producing more natural movements at the same time. Also, exploiting compliance in the gripper/manipulator, while relying on grasp execution schemes with low pre-grasp wrist pose sensitivity, proved to be more reliable than using sampling-based planning with strict obstacle avoidance (which, due to sensor noise, is precarious anyway). The chosen local motion control method proved to be sufficient for the relatively simple pick and place operations in this work, albeit it required to split the object manipulation task in pre-defined discrete phases (see Fig. 7).

While the developed grasping approach showed good success rates, an industry-grade solution needs additional measures to avoid grasp failure. We plan to address this by evaluating grasp success online using sensory feedback after the gripper contacts the object. This will allow to initiate early countermeasures such as adjusting the hand pose and/or contact locations. Also, future work will aim at reducing the local nature of the manipulator motion generation and control scheme by investigating optimal control approaches able to take future state evaluations into account.

REFERENCES

[1] W. Echelmeyer, A. Kirchheim, and E. Wellbrock, "Robotics-logistics: Challenges for automation of logistic processes," in *Proc. IEEE ICAL*, 2008, pp. 2099–2103.

[2] 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.

[3] H. Andreasson, A. Bouguerra, M. Cirillo, D. Dimitrov, D. Driankov, L. Karlsson, A. Lilienthal, F. Pecora, J. Saarinen, A. Sherikov, and T. Stoyanov, "Autonomous transport vehicles: Where we are and what is missing," *IEEE RAM*, vol. 22, no. 1, pp. 64–75, 2015.

[4] T. Stoyanov, N. Vaskevicius, C. Müller, *et al.*, "No more heavy lifting: Robotic solutions to the container unloading problem," *IEEE RAM*, 2016, to appear.

[5] H. Andreasson, J. Saarinen, M. Cirillo, T. Stoyanov, and A. J. Lilienthal, "Fast, continuous state path smoothing to improve navigation accuracy," in *Proc. IEEE ICRA*, 2015, pp. 662–669.

[6] 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.

[7] T. Hellström and O. Ringdahl, "Follow the past: a path-tracking algorithm for autonomous vehicles," *IJVAS*, vol. 4, no. 2, pp. 216–224, 2006.

[8] J. Marshall, T. Barfoot, and J. Larsson, "Autonomous underground tramming for center-articulated vehicles," *JFR*, vol. 25, no. 6-7, pp. 400–421, 2008.

[9] K. Hyyppa, "Method of navigating an automated guided vehicle," 1989, uS Patent 4,811,228.

[10] R. Varga and S. Nedevschi, "Vision-based autonomous load handling for automated guided vehicles," in *Proc. IEEE ICCP*, 2014, pp. 239–244.

[11] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *Proc. IEEE/RAS Humanoids*, 2007, pp. 42–48.

[12] 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," *AURO*, vol. 28, no. 1, pp. 5–20, 2010.

[13] 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. IEEE ICRA*, 2014, pp. 3669–3675.

[14] D. Berenson, S. Srinivasa, and J. Kuffner, "Task space regions: A framework for pose-constrained manipulation planning," *IJRR*, vol. 30, no. 12, pp. 1435–1460, 2011.

[15] L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. C. Voorhies, G. S. Sukhatme, and S. Schaal, "An autonomous manipulation system based on force control and optimization," *AURO*, vol. 36, no. 1-2, pp. 11–30, 2014.

[16] M. Gienger, M. Toussaint, and C. Goerick, "Task maps in humanoid robot manipulation," in *Proc. IEEE/RSJ IROS*, 2008, pp. 2758–2764.

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

[18] N. Mansard, O. Stasse, P. Evrard, and A. Kheddar, "A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks," in *Proc. ICAR*, 2009, pp. 1–6.

[19] V. Tincani, M. G. Catalano, E. Farnioli, M. Garabini, G. Grioli, G. Fantoni, and A. Bicchi, "Velvet fingers: A dexterous gripper with active surfaces," in *Proc. IEEE/RSJ IROS*, 2012, pp. 1257–1263.

[20] C. Eppner, R. Deimel, J. Alvarez-Ruiz, M. Maertens, and O. Brock, "Exploitation of environmental constraints in human and robotic grasping," *IJRR*, vol. 34, no. 7, pp. 1021–1038, 2015.

[21] 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. IEEE ICRA*, May 2014, pp. 3956–3962.

[22] M. Cirillo, T. Uras, S. Koenig, H. Andreasson, and F. Pecora, "Integrated motion planning and coordination for industrial vehicles," in *Proc. ICAPS*, 2014.

[23] 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. IEEE/RSJ IROS*, 2013, pp. 4702–4708.

[24] D. R. Canelhas, T. Stoyanov, and A. J. Lilienthal, "SDF tracker: A parallel algorithm for on-line pose estimation and scene reconstruction from depth images." in *Proc. IEEE/RSJ IROS*. IEEE, 2013, pp. 3671–3676.

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

[26] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *IJRR*, vol. 23, no. 7, pp. 1006–1028, 2014.