Towards Shared Autonomy Applications using Whole-body Control Formulations of Locomanipulation

Wolfgang Merkt, Vladimir Ivan, Yiming Yang, and Sethu Vijayakumar

***Abstract*— While widely studied in robotics for decades, mobile manipulation has recently seen a surge in interest for industrial applications due to increasing demands on flexibility and agility alongside productivity, particularly in small and medium enterprises. However, most mobile manipulation solutions frequently decouple the navigation from the manipulation problem effectively performing fixed-base manipulation using a repositionable manipulator. This is not only inefficient, but moreover limits the range of applications and disregards the inherent redundancy of a mobile manipulation system.**

In this work, we introduce a high-performance omnidirectional mobile manipulation platform with integrated whole-body control, real-time collision-free whole-body motion planning, and perception. We demonstrate its capability along with application scenarios on technical demonstrators involving moving manipulation targets as well as whole-body manipulation in simulation and hardware experiments. Finally, we deploy and evaluate our solution in field trials on an industrial oil and gas training facility on a sensor placement and manipulation task.

I. Introduction

Traditional industrial automation achieves productivity gains through fast and precisely repeated pre-programmed motions in controlled environments. Here, the robots are firmly mounted to the ground allowing high-speed movement, enclosed with security fencing, and attached to unlimited shore power. These systems are custom designed at the beginning of a product life cycle and amortize costs over a large production volume with low individual variability (mass manufacturing). The recent trend for customization, however, is dominated by small batch sizes and short cycle times with frequent reconfiguration of work cells. This requires flexibility and agility to respond quickly to changes in demand, and are a particular challenge for the competitiveness of small and medium enterprises (SMEs).

As a solution, the integration of light-weight, collaborative robots (cobots) into shared human-robot-assembly lines is now widely being adopted. Cobots are safe to operate near humans without extensive safety systems and can be programmed/taught by demonstration [1]. However, as they are mounted with a fixed base they are limited in their workspace reachability often requiring a large number of cobots per plant and posing a challenge for redeployability and versatility.

This research is supported by the Engineering and Physical Sciences Research Council (EPSRC, grant reference EP/L016834/1) and the UK-India Education and Research Initiative (UKIERI, grant reference UKIERI-DST 2016-17-0152). The work has been performed at the University of Edinburgh under the Centre for Doctoral Training in Robotics and Autonomous Systems program and with support from the EPSRC UK RAI Hub for Offshore Robotics for Certification of Assets (ORCA, grant reference EP/R026173/1).

All authors are with the Institute for Perception, Action, and Behaviour, School of Informatics, The University of Edinburgh (Informatics Forum, 10 Crichton Street, Edinburgh, EH8 9AB, United Kingdom).

Email: `wolfgang.merkt@ed.ac.uk`.

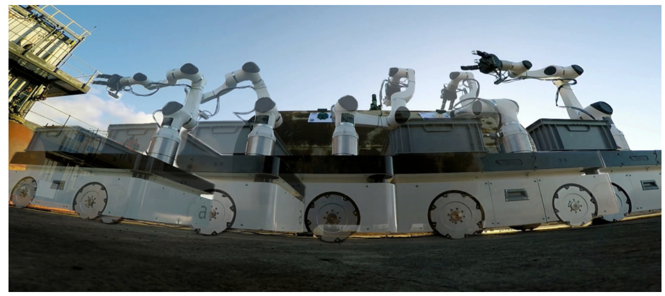

Fig. 1: Continuous manipulation using whole-body control: an industrial IoT monitoring device is placed using live sensor feedback.

This motivates the use of mobile collaborative robots, which, while opening up versatile deployment opportunities, require new considerations. While some manufacturers propose mobile manipulation as the repositioning of flexible workstations that automatically plug themselves to shore power (e.g., FANUC Robotics[1]), many integrators have combined autonomous ground vehicles (AGVs) with differential or omnidirectional drive and collaborative robots into commercially available mobile manipulators. This is of interest to both traditional workshops looking to automate tasks, e.g., machine tending in existing factory floors, as well as in the development of flexible manufacturing concepts such as matrix production.

While some of the recent developments are spurred by the confluence of the maturity of the open source robotics ecosystem (ROS) and the availability of mature, standardized hardware platforms and software systems (e.g., ROS-Control for hardware abstraction and MoveIt! for motion planning), there is an increasing demand for industrial applications. At present, mobile robots are widely deployed for logistics and warehousing tasks such as moving shelves (e.g., Amazon/Kiva Robotics) and in-facility logistics (e.g., in hospitals and on factory floors). Manipulation tasks, on the other hand, are often limited to demonstrations of pick-and-place and material transport and are generally only applied to highly-specialized scenarios due to limitations of workspace interoperability, autonomy, battery runtime, and cost.

A. Related work

Autonomous mobile manipulation recently received a renewed focus in research, with a number of high-profile international competitions centered on service robotics (e.g.,

[1]Cf. `https://youtu.be/rQBnZuby05s`.

World Robot Challenge, RoboCup@Home, etc.), disaster recovery (e.g., DARPA Robotics Challenge), or coordinated manipulation tasks (e.g., Mohammed Bin Zayed International Robotics Competition).

However, due to the complexity of planning locomanipulation in real-time, locomotion/navigation and manipulation are often treated as separate problems and joined and coordinated by a high-level state machine [2], sequence planner, or shared autonomy control interface [3], [4]. In this case, the problems of optimal base placement [5], navigation to the base placement, and fixed-base manipulation [6], [7] are treated separately, limiting the applicability to static targets and obstacles and disregarding the inherent redundancy of high degree of freedom (DoF) systems. Yet, to achieve time- and energy-optimal solutions, locomotion and manipulation need to be considered jointly.

Early work to coordinate locomotion and manipulation introduced the Mobile Manipulator Jacobian Transpose (MMJT) [8] and demonstrated the ability to compensate vehicle motion from passive suspension to stabilize end-effector motions. [9] considered joint motion to maximize manipulability of the end-effector while following a desired or guided operational space trajectory. Similarly, [10] considered both manipulation and locomotion during motion planning in a joint optimization problem maximizing a directional manipulability metric. However, while planning jointly, in order to avoid errors from a lower-precision mobile base, the authors enforced discrete repositioning and fixed-base manipulation. [11] considered humanoid locomanipulation by planning in task-space introducing virtual joints for footsteps and an adaptive procedure to vary the number of foot placements. Recent advances in semi- and fully-constrained collision-free whole-body motion planning in time-configuration space using sampling-[12] and optimization-based [13] approaches allow synthesis of highly complex motion manipulating moving targets in dynamic environments. [14] used a Hierarchical Quadratic Programming (HQP)/Stack-of-Tasks (SoT) approach on a holonomic mobile manipulator with continuous task transition. In summary, [8], [9], [14] focused on instantaneous whole-body control to coordinate and compensate end-effector motion, while [10]–[13] focused on locomanipulation planning over longer horizons as an input into the former. In this work, we combine locomanipulation planning with coordinated whole-body control for continuous manipulation in complex environments.

Constrained optimization is a common tool for planning and controlling motion of humanoid [3] and quadruped robots [15], e.g., using trajectory optimization and task-space inverse dynamics, respectively. The task is formally described by constraints on controls, end-effector positions, and other properties such as static balance derived from inverted pendulum dynamics while control effort or other cost terms are minimized. A variety of efficient quadratic programming and nonlinear programming (NLP) solvers have been developed to solve these problems. These formulations are generic so that they can handle complex problem settings required for optimizing the motion of highly dynamical system, however, they can also be used to define tasks for non-legged robots and for trajectory optimization in complex environments instead.

B. Contribution

In this work, we introduce a high-performance manipulation system using whole-body control for continuous locomanipulation. We build on concepts and formulations for operational space and whole-body control widely used in legged robots and humanoid control and leverage it for efficient, continuous mobile manipulation which also allows whole-body visual servoing. We evaluate our proposed system by demoing a chicken-head task showcasing the decoupling of operational space manipulator motion from base motion. We highlight planning and locomanipulation capabilities in a simulated automotive fitting task and demonstrate a sensor placement task for certification of assets on an outdoor test site. Accompanying videos are available at `http://www.wolfgangmerkt.com/continuous-manipulation`.

II. System Overview

Our system consists out of a high-performance ($1.5\,\mathrm{m\,s^{-1}}$) and high-payload ($500\,\mathrm{kg}$ to $800\,\mathrm{kg}$) omnidirectional mobile platform with a 6 degrees of freedom (DoF) collaborative robot for a total of 9 DoF (Adabotics Ada500). The system features a built-in $1\,\mathrm{kHz}$ whole-body control layer based on ROS-Control, with the individual system components shown in Figure 2. The platform uses two horizontal laser rangefinders as well as an Intel Realsense D-435 RGB-D sensor mounted on the wrist. It further contains two on-board computers with one dedicated to control, motion planning, and safety features and the other performing perception tasks such as mapping and object identification and tracking. The system further comes with a remote control user interface available from any phone or tablet computer and is equipped with battery capacity for a full-shift autonomous operation ($8\,\mathrm{h}$ to $10\,\mathrm{h}$). In order to maximize operation with limited on-board power, we consider energy efficiency in our motion optimization and target continuous, non-stop manipulation through execution of whole-body trajectories using coordinated, whole-body control. We continuously monitor the environment for conflicting changes and respond using a combination of real-time planning and reverting to operator input via shared autonomy.

III. Problem Formulation

The configuration for a robot manipulator with N DoF is commonly defined as $\mathbf{q}_{manipulator} \in \mathbb{R}^N$. The state $\mathbf{x}_t = (\mathbf{q}_t, \dot{\mathbf{q}}_t)$ is directly and accurately measured, and it can be directly controlled via position control, velocity control, admittance control, impedance control, or torque control. Furthermore, the state and the controls $\mathbf{u}_t$ are usually bounded (e.g., by joint position, velocity, acceleration, current, or torque limits) which limits the scope of motion planning and the working envelope in which we may want to avoid collisions or seek contacts. On the other hand, the state of

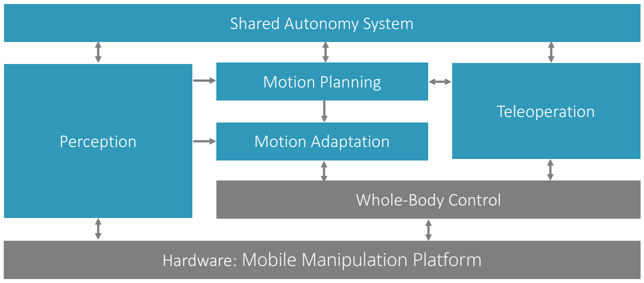

Fig. 2: Overview of the deployed system: All components are modular and can be replaced due to the specification of commonly adopted interfaces, e.g., ROS-Control. In this work, we heavily rely on the whole-body control and perception systems.

a mobile platform moving on a surface is often defined as $\mathbf{x}_{base} \in SE(2)$. This space has the topology of the *Special Euclidean group* and it is unbounded, i.e., it has no translation and rotation limits. Additionally, some mobile platform designs are non-holonomic, which means that the state cannot be directly controlled in all directions. However, we will not address this issue in this paper and utilize a holonomic hardware platform instead.

Despite these differences, our goal is to express the state of the whole robot as $\mathbf{x} = [\mathbf{x}_{base}; \mathbf{x}_{manipulator}]$, where $\mathbf{x}$ describes the state of a system with $3 + N$ DoF. This choice has significant impact on the design of the controller. We will now formulate the combined locomanipulation problem as a whole-body control problem.

A. Whole-body control

We formulate the whole-body control problem as the one-step look-ahead minimization of an optimization problem subject to all bound, linear and nonlinear inequality and equality constraints and account for modeled actuation delays:

$$\arg\min_{\mathbf{x},\mathbf{u}} \; f(\mathbf{x}, \mathbf{u}) \tag{1}$$

$$\text{s.t.: } h(\mathbf{x}, \mathbf{u}) = 0 \tag{2}$$

$$g(\mathbf{x}, \mathbf{u}) \leq 0 \tag{3}$$

$$c_{lb} \leq A\mathbf{x} + B\mathbf{u} \leq c_{ub} \tag{4}$$

Here, the upper and lower bounds of the decision variables are updated based on current state, timestep, and proximity to higher-order limits through integration similar to [16]. The equality constraints $h(\mathbf{x}, \mathbf{u}) = 0$ and inequality constraints $g(\mathbf{x}, \mathbf{u}) \leq 0$ are a set of nonlinear functions of state and/or the control. Linear equality, inequality, and bound constraints (e.g., joint position and velocity limit) are encoded using A and B. We use a different set of constraints for each control mode. For example, we use general equality constraints on the end-effector position to trace a path with the tool, and we use general inequality constraints for limiting the tool speed. We can formalize a large variety of control modes using the same generic framework by combining different sets of constraints. We do the same with the optimality criteria $f(\mathbf{x}, \mathbf{u})$, which often takes the form a weighted sum of squared error terms.

As we are solving a limited size problem initialized from the present state, we obtain fast convergence for control using a nonlinear optimization problem. While traditionally quadratic programming-based formulations are chosen for whole-body control in legged platforms, e.g. in [17], the comparatively small size of a mobile manipulation problem (9-17 DoF) allows us to leverage nonlinear optimization to include much more expressive cost and constraint terms. An overview of how this solver is embedded into the full framework is shown in Figure 3.

B. State estimation

The state estimation module is based on sensor fusion of the wheel odometry, the on-board inertial measurement unit (IMU), and exteroceptive sensors (e.g., visual odometry/SLAM from monocular, stereo vision, or RGB-D sensors, laser localization, GPS, etc.). Here, we use the Unscented Kalman Filter (UKF) with the sensing modalities configured in two stages: local odometry frame and global frame. The odometry frame is a smooth signal updated at a high frequency and with high accuracy over short periods of time. However, this state estimate accumulates error over time. The global frame stays consistent over long periods of time but its updates are less frequent, often more costly, they are less accurate, and the global frame states are not smooth over time.

We use the the odometry frame estimates for control due to its smoothness and local consistency. We then use the global frame estimate for planning and for slow corrections of trajectories over time.

C. Whole-body locomanipulation planning

In order to achieve fast motion synthesis for longer horizon planning which includes locomotion and manipulation in the presence of moving targets and obstacles, we formulate a trajectory optimization problem in time-configuration space. Each timestep hereby preserves the same formulation and expressiveness in cost and constraints as described in Section III-A for the one timestep look-ahead control, with further constraints introduced for dynamic consistency and smooth transition between states, similar to [18], [19]. However, due to the inherent non-convexity of nonlinear optimization, solvers are not guaranteed to converge to a valid solution in a given time budget – or at all – unless provided with a suitable initialization seed. This is especially the case when considering collision avoidance in complex, unknown environments. In known environments, suitable warm-start solutions can be encoded in a trajectory library [20]. In order to operate in unseen environments, we employ fast, global sampling-based planners to provide a feasible initialization to the trajectory optimization. Random sampling-based planners, however, are not suited to satisfy general constraints.[2] As thus, we use constraint relaxation as well as a framework to solve constrained time-configuration space problems by decomposition [12] for initialization. We formulate both the real-time control optimization as well as the nonlinear locomanipulation problem in the open source Extensible Optimization Toolset EXOTica [22].

[2]Readers are referred to [21] for a review of approaches for sampling-based planning in presence of constraints.

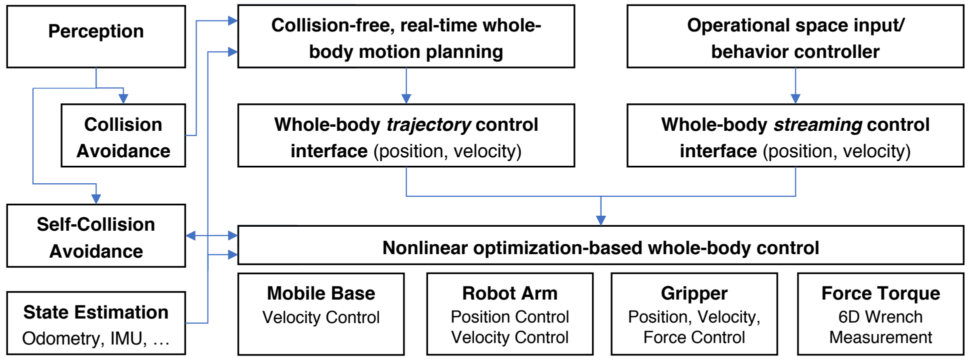

Fig. 3: Overview of the nonlinear optimization-based whole-body control framework: Commands can be issued either as whole-body trajectories or from operational space targets in a streaming mode, where individual commands/targets are sent to the controller at every time step. The controller satisfies all applicable bound constraints as well as general nonlinear safety constraints (e.g., self-collision avoidance).

IV. Evaluation

A. Whole-body control evaluation on chicken-head task

We evaluate the performance of the implemented whole-body control scheme by maintaining an operational space target for the end-effector while commanding a desired target for the base controller (commonly referred to as the *chicken-head problem*). In a laboratory setting, we increase the velocity of the base command (to track a circle on the ground) while tracking ground truth using a VICON camera system. We have formulated the tracking problem as unconstrained minimization of the end-effector position in the global frame over the base position and the arm configuration and used the Levenberg-Marquardt [23] algorithm to solve this problem. Note, this is a relaxation of Equations (1)-(4) as the manipulator may pass through singular configurations resulting in a violation of real-time requirements. The results are depicted in Figure 4 validating the relaxation to be suitable, and snapshots of an applicable real-world task experiment depicted in Figure 6. We show the end-effector error against ground truth from a VICON motion capture system in Figure 5. This task is very simplistic but this formulation enables us to handle tracking moving targets in arbitrary frames of reference, which opens up our framework to a multitude of practical applications.

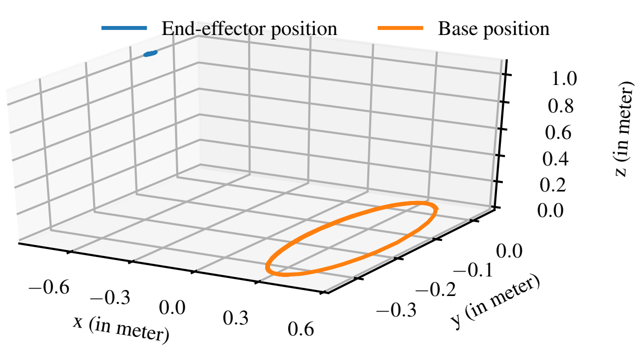

Fig. 4: Visualization of the robot internal state and ground truth for both the end-effector and base frames while carrying out the chicken-head task: This experiments demonstrates the ability of the whole-body control scheme to decouple the end-effector from the base motion and coordinate both at the same time.

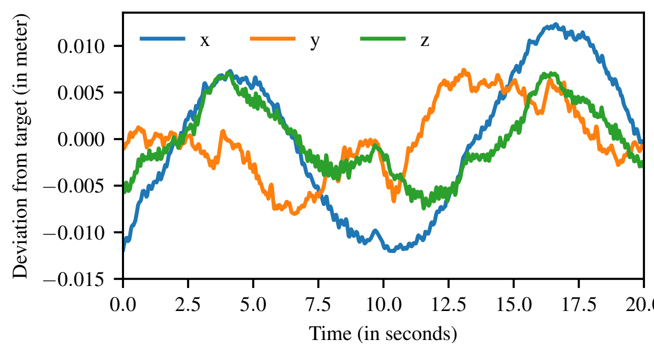

Fig. 5: Visualization of the task space error of the end-effector while the base is following a circular motion using solely odometry (no sensor fusion with IMU): both x and y drift, where the error in z is due to stiff suspension on an uneven floor.

B. Automotive assembly fitting simulation

A frequent task for the deployment of collaborative robots is the fitting of insulation, adhesives, and subassemblies in automotive manufacturing (e.g., sealants on doors). These tasks are correlated with a high risk for repetitive strain injury.

In this scenario, a mobile collaborative robot carries out manipulation tasks on a moving assembly part by coordinating whole-body motion. We formulate a whole-body constrained nonlinear optimization problem to minimize control effort in the presence of moving targets and solve it using the commercial solver SNOPT [24]. In particular, the manipulation motions (e.g., drilling and fitting trajectories) are encoded as semi-constrained end-effector paths (3-DoF position, 2-DoF axis alignment) with further constraints on continuous collision avoidance using the approach from [13].

We have used the whole-body controller in the trajectory mode for executing the motion. This is sufficient in simulation (cf. Figure 7), however, a real-world deployment requires active sensing, tracking of the assembly, and other steps correcting the synchronization of the robot motion with the environment. We address these issues in our next experiment.

C. Sensor placement

Off-shore assets such as oil and gas platforms are structures that are designed to operate for decades in harsh environments. Sea water, wind, and temperature changes cause material

Fig. 6: The chicken-head task on a rough outdoor surface carrying out a proposed scenario: manipulation of a static end-effector affordance while responding to disturbance with the redundancy of the omnidirectional base following a high velocity figure-eight target.

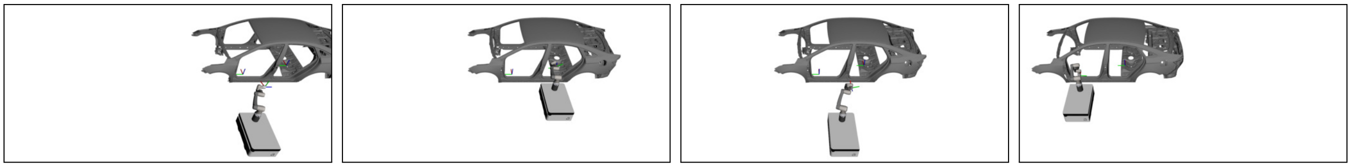

Fig. 7: Assembly tasks on a car body structure using a mobile manipulator: While the assembly line is moving at $0.5\,\mathrm{m\,s^{-1}}$, the manipulator carries out two collision-free manipulation actions for $4\,\mathrm{s}$ each while following the moving target.

deterioration and failure that can be prevented by regular maintenance and monitoring. Our industrial partners[3] have identified the need for automating these tasks. They are currently executed by humans, which is both costly and potentially dangerous for the workers. A large amount of monitoring can be done remotely, as long as the monitored structure can be equipped with sensors. The Limpet sensor [25] was designed for these applications in harsh environments, limited power, remote operation with long distance communication, and real-time monitoring capability. We have applied our whole-body control framework to place these sensors semi-autonomously. The user remotely specifies the sensor location while the planning and control framework ensures accurate placement of the sensor. This process requires minimal data bandwidth and it is therefore suitable for applications where teleoperation is not possible due to communication quality.

In our experiment, we placed a container with the Limpet sensors on top of the robot. The user then specifies the target location. For repeatability and easy visual confirmation, we mark and track the target locations with AprilTag [26] fiducial markers. However, the target detection and tracking can be done using any combination of visual and depth features, such as in [27]. The execution then used a finite state machine to switch between sensor pick-up, sensor placement using visual servoing, and arm parking motion. The sensor placement was triggered by the sensing module detecting the target marker.

In the first phase, we have constructed a motion planning problem for computing a collision-free pick-up trajectory for the robot arm using the RRT-Connect [28]. The trajectory was executed using our controller in the trajectory mode. Once the target was detected the tracker provides updates of the target position. These were fed into the controller in the streaming mode. The controller solves the inverse kinematics formulated as an unconstrained NLP problem (see Section III-A) using the Levenberg-Marquardt [23] algorithm. The solution was then used to command the arm position in real-time to compensate for the relative target motion. We have also superimposed a short place, hold, and release trajectory over the target position. This ensures that the gripper has enough time open. The parking motion is then planned using RRT-Connect and executed in the trajectory mode, same as the sensor pick-up motion in the first phase.

[3]Through the ORCA Hub, we engage with a variety of industrial partners such as Total, BP, and their sub-contractors, cf. `https://orcahub.org`.

This experiment was executed both in the lab environment (see Fig. 8) and in an outdoors mock oil rig designed for fire fighter training (see Fig. 1). The task can be easily modified for similar scenarios by modifying the state machine or changing any of the sub-problems to fit the needs. The advantage of using the whole-body controller in this scenario is that the framework can handle all the different operation modes, which allowed us to execute the whole task continuously, without stopping. This is possible due to the inherent synchronization of the base and arm movement.

V. Discussion

We have presented an architecture for whole-body control and planning of collaborative mobile manipulators. This system exploits a generic formulation of the task as a constrained nonlinear program and it integrates inputs from state estimation, perception, and the user to generate complex collision-free motion plans. The control architecture then minimizes the tracking error while satisfying the task specific constraints. The formulation of the problem allows us to formulate a wide variety of motion planning tasks and match them with customized controllers.

Our evaluation on the chicken-head task validates the architecture. The tracking results then show the overall performance of the system on our hardware platform. The results demonstrate the performance of the controller and of the platform itself in a controlled environment as well as in an outdoors trial. Using the controller implementation in EXOTica, we achieve a $100\,\mathrm{Hz}$ control rate on an Intel i7-7567U CPU with peak performance at $500\,\mathrm{Hz}$. The bottleneck of the controller is the state estimation. The accuracy of the end-effector tracking depends largely on the quality of the state estimate that is used for closing the control loop. Drift, delays, and position error all contribute to this issue. Delays can be computed and accounted for, e.g., using model predictive control (MPC), as for instance in [29]. Drift can be eliminated by exploiting exteroceptive sensors and computing the global reference as described in Section III-B.

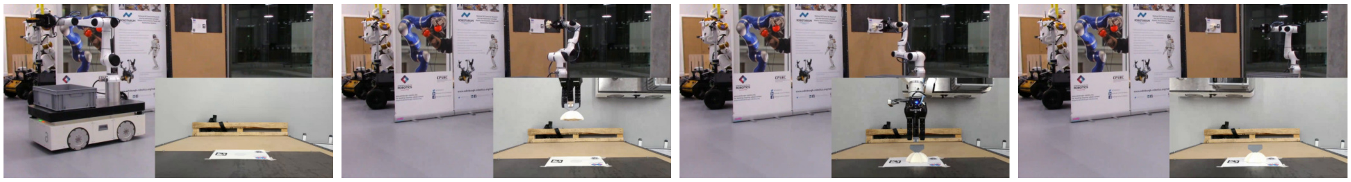

Fig. 8: Sensor deployment trials in a laboratory: The robot proceeds to the next placement task in the stack and after acquiring the target carries out a whole-body visual servoing task without stopping the navigation/base motion. The accuracy of the sensor placement task with respect to the target can be seen from the over-head camera.

We have also used the generic problem formulation for solving a motion planning problem in the automotive industry. The versatility of this formulation allowed us to do trajectory optimization with collision avoidance. This is a notoriously difficult problem due to the non-convexity and nonlinearity of the collision constraint. We have exploited the state-of-the-art collision term formulation presented in [13]. The whole-body paradigm then allowed us to use a relatively small robot and extend its range without sacrificing the optimality of the task. This experiment did not consider any control errors nor any control delays since a simulator was used.

In our last experiment, we deployed our system on the real robot to perform a non-stop pick-and-place task. While we used a visual marker to track the target location, the perception method can be easily swapped for a more advanced technique that does not require any fiducial information. We have also relied on the soft housing of the sensor when making contact during the placement. If the sensor did not provide a soft interface between the robot and the solid wall structure, we would consider using compliant control using a force torque sensor at the end-effector.

Each industrial application requires a specific set of sensing, planning, and control solutions. The architecture we proposed opens possibilities for designing these techniques using well defined building blocks. Such an approach can rapidly accelerate the development and deployment of robotic systems in automotive manufacturing, off-shore asset maintenance, and many other fields.

REFERENCES

[1] Y. Wang, Y. Jiao *et al.*, "MASD: A Multimodal Assembly Skill Decoding System for Robot Programming by Demonstration," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 4, pp. 1722–1734, Oct 2018.

[2] J. Carius, M. Wermelinger *et al.*, "Autonomous mission with a mobile manipulator–a solution to the mbzirc," in *Field and Service Robotics*. Springer, 2018, pp. 559–573.

[3] M. Fallon, S. Kuindersma *et al.*, "An architecture for online affordance-based perception and whole-body planning," *Journal of Field Robotics*, vol. 32, no. 2, pp. 229–254, 2015.

[4] W. Merkt, Y. Yang *et al.*, "Robust shared autonomy for mobile manipulation with continuous scene monitoring," in *Proc. IEEE Conf. Autom. Sci. Eng. (CASE)*, Aug 2017, pp. 130–137.

[5] Y. Yang, W. Merkt *et al.*, "Efficient humanoid motion planning on uneven terrain using paired forward-inverse dynamic reachability maps," *IEEE Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2279–2286, Oct 2017.

[6] Y. Yang, V. Ivan *et al.*, "Scaling sampling-based motion planning to humanoid robots," in *Proc. IEEE Int. Conf. Robotics and Biomimetics (ROBIO)*, Dec 2016, pp. 1448–1454.

[7] Y. Yang, W. Merkt *et al.*, "HDRM: A Resolution Complete Dynamic Roadmap for Real-Time Motion Planning in Complex Scenes," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 551–558, Jan 2018.

[8] N. A. M. Hootsmans, "The motion control manipulators on mobile vehicles," Ph.D. dissertation, Massachusetts Institute of Tech., 1992.

[9] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," in *Proc. IEEE Conf. Decision and Control*, Dec 1992, pp. 2643–2648 vol.3.

[10] Dong Hun Shin, B. S. Hamner *et al.*, "Motion planning for a mobile manipulator with imprecise locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS)*, vol. 1, Oct 2003, pp. 847–853.

[11] D. Dang, F. Lamiraux, and J. Laumond, "A framework for manipulation and locomotion with realtime footstep replanning," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Oct 2011, pp. 676–681.

[12] Y. Yang, W. Merkt *et al.*, "Planning in time-configuration space for efficient pick-and-place in non-static environments with temporal constraints," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Nov 2018, pp. 1–9.

[13] W. Merkt, V. Ivan, and S. Vijayakumar, "Continuous-time collision avoidance for trajectory optimization in dynamic environments," in *Under review for IEEE IROS 2019*, 2019.

[14] S. Kim, K. Jang *et al.*, "Whole-body control of non-holonomic mobile manipulator based on hierarchical quadratic programming and continuous task transition," in *Proc. IEEE Int. Conf. Advanced Robotics and Mechatronics*, July 2019.

[15] G. Xin, H. Lin *et al.*, "A model-based hierarchical controller for legged systems subject to external disturbances," in *Proc. IEEE Int. Conf. Rob. Autom. (ICRA)*, May 2018, pp. 4375–4382.

[16] A. D. Prete, "Joint position and velocity bounds in discrete-time acceleration/torque control of robot manipulators," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 281–288, Jan 2018.

[17] A. Bolotnikova, K. Chappellet *et al.*, "A circuit-breaker use-case operated by a humanoid in aircraft manufacturing," in *Proc. IEEE Conf. Autom. Sci. Eng. (CASE)*, Aug 2017, pp. 15–22.

[18] J. Schulman, Y. Duan *et al.*, "Motion planning with sequential convex optimization and convex collision checking," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1251–1270, 2014.

[19] M. Toussaint, "Newton methods for k-order Markov constrained motion problems," *arXiv preprint arXiv:1407.0414*, 2014.

[20] W. Merkt, V. Ivan, and S. Vijayakumar, "Leveraging precomputation with problem encoding for warm-starting trajectory optimization in complex environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS)*, Oct 2018, pp. 5877–5884.

[21] Z. Kingston, M. Moll, and L. E. Kavraki, "Sampling-based methods for motion planning with constraints," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 159–185, 2018.

[22] V. Ivan, Y. Yang *et al.*, "EXOTica: An Extensible Optimization Toolset for Prototyping and Benchmarking Motion Planning and Control," in *Robot Operating System (ROS): The Complete Reference (Volume 3)*, A. Koubaa, Ed. Springer, 2019.

[23] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis*. Springer, 1978, pp. 105–116.

[24] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM Rev.*, 2005.

[25] M. E. Sayed, M. P. Nemitz *et al.*, "The limpet: A ros-enabled multi-sensing platform for the orca hub," *Sensors*, vol. 18, no. 10, 2018.

[26] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS)*, Oct 2016, pp. 4193–4198.

[27] K. Pauwels and D. Kragic, "Simtrack: A simulation-based framework for scalable real-time object pose detection and tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS)*, Sep. 2015, pp. 1300–1307.

[28] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Proc. IEEE Int. Conf. Rob. Autom. (ICRA)*, vol. 2, April 2000, pp. 995–1001.

[29] J. Koenemann, A. Del Prete *et al.*, "Whole-body model-predictive control applied to the HRP-2 humanoid," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS)*, Sep. 2015, pp. 3346–3351.