

# ALMA - Articulated Locomotion and Manipulation for a Torque-Controllable Robot

C. Dario Bellicoso, Koen Krämer, Markus Stäuble, Dhionis Sako,  
Fabian Jenelten, Marko Bjelonic, Marco Hutter

**Abstract**—The task of robotic mobile manipulation poses several scientific challenges that need to be addressed to execute complex manipulation tasks in unstructured environments, in which collaboration with humans might be required. Therefore, we present ALMA, a motion planning and control framework for a torque-controlled quadrupedal robot equipped with a six degrees of freedom robotic arm capable of performing dynamic locomotion while executing manipulation tasks. The online motion planning framework, together with a whole-body controller based on a hierarchical optimization algorithm, enables the system to walk, trot and pace while executing operational space end-effector control, reactive human-robot collaboration and torso posture optimization to increase the arm’s workspace. The torque control of the whole system enables the implementation of compliant behavior, allowing a user to safely interact with the robot. We verify our framework on the real robot by performing tasks such as opening a door and carrying a payload together with a human.

## I. INTRODUCTION

Legged robots have significant advantages over their wheeled or tracked counterparts. They are capable of traversing challenging terrain and environments designed for human use (e.g., steps and stairs). Walking robots need to break contact and modulate the ground reaction forces to propel themselves through the environment. This modulation not only allows the robot to retain balance but also enables compliant behavior. In particular, quadrupedal robots typically exhibit a larger support area than bipedal systems, easing the design of robust motions.

Typical missions for a quadrupedal robot include exploration, mapping, navigating through challenging terrain, and inspecting scenarios which are undesirable for humans to be in [1]. Direct interaction with the environment, however, has been limited to the contacts used for locomotion, with little to no flexibility in the manipulation capabilities. Few robots use their legs for manipulation, and the possible tasks using the available feet [2] or a gripper tool attached to the feet [3] remain limited and renders simultaneous locomotion and manipulation hard or impossible. Equipping a multi-legged robot with an additional limb that is dedicated to manipulation tasks, greatly extends the possible real-world

This work was supported in part by the Swiss National Science Foundation (SNF) through the National Centre of Competence in Research Robotics (NCCR Robotics) and Digital Fabrication (NCCR dfab).

This work has been conducted as part of ANYmal Research, a community to advance legged robotics.

All authors are with the Robotic Systems Lab, ETH Zurich, Switzerland, {bellicoso, koen.kraemer, markus.stäuble, dhionis.sako, fabian.jenelten, marko.bjelonic, marco.hutter}@mavt.ethz.ch



Fig. 1. The quadrupedal robot ANYmal equipped with a six DOF robotic arm. The system is fully torque-controlled, enabling compliant behavior and safe interaction.

deployment. Such a robot will be able to carry and move objects, help a human to deliver a payload, open doors and interact with its surroundings in ways that were precluded before.

Similar solutions have been explored over the past few years [4]. The quadrupedal robot *HyQ* [5] is equipped with a six DOF arm and demonstrates a static walking gait while tracking motions of the arm. The authors propose a controller that takes into account internal and external disturbances created by the dynamics of the arm by optimizing for the ground reaction forces. Impressive results have been achieved by Boston Dynamics’ quadrupedal robots *Spot* [6] and *SpotMini* [7]. *SpotMini*, equipped with a five DOF arm, shows manipulation tasks while walking, e.g., opening a door and carrying a payload. So far, none of the details on the methods and approaches used to control these robots have been made available. In an older work of Boston Dynamics, the quadrupedal robot *BigDog* [8] demonstrates a throwing maneuver with a robotic arm while trotting in place.

Controlling such a system comes with several challenges. It requires robust and fast motion planning and control to enable simultaneous locomotion and manipulation in challenging environments while being able to cope with external disturbances. Such dynamic interaction with the environment through legs and arms of a walking robot requires taking into account the full system dynamics as well as the contact forces at the robot’s end-effectors. Optimal contact

force distribution for torque-controllable quadrupedal robots was demonstrated in experiments while taking into account equality [9] and inequality [10] constraints. Optimization algorithms to solve the contact force distribution based on the full rigid-body dynamics are shown in [11] and [12]. In these approaches, inequality constraints on the direction and magnitude of the linear contact forces are prescribed, leaving the exact contact force distribution to follow from the other whole-body controller tasks. However, when actively interacting with the environment using an arm (e.g., opening a door), it may be desirable to explicitly prescribe linear contact forces as well as contact torques between the gripper and environment. This increases the complexity of the contact force distribution problem for the entire system.

In this paper, we present ALMA (Articulated Locomotion and Manipulation for ANYmal, see Fig. 1), a planning and control framework for a fully torque-controlled quadrupedal manipulator capable of performing dynamic gaits while executing manipulation tasks. The framework allows the robot to compliantly react to external forces and to maintain balance while executing locomotion and manipulation tasks. To the best of our knowledge, this is the first time that such a system is shown performing coordination between dynamic locomotion and manipulation.

## II. MODEL FORMULATION

We formulate the model of a walking robot equipped with a robotic arm as a free-floating base  $B$  to which limbs are attached. The motion of the entire system can be described with respect to (w.r.t.) a fixed inertial frame  $I$ . The position of the Base w.r.t. the inertial frame, expressed in the inertial frame, is written as  ${}^I\mathbf{r}_{IB} \in \mathbb{R}^3$ . The orientation of the Base w.r.t. the inertial frame is parametrized using a Hamiltonian unit quaternion  $\mathbf{q}_{IB}$ . The limb joint angles are stacked in the vector  $\mathbf{q}_j \in \mathbb{R}^{n_j}$ , where  $n_j = 18$ . We write the generalized coordinate vector  $\mathbf{q}$  and the generalized velocity vector  $\mathbf{u}$  as

$$\mathbf{q} = \begin{bmatrix} {}^I\mathbf{r}_{IB} \\ \mathbf{q}_{IB} \\ \mathbf{q}_j \end{bmatrix} \in SE(3) \times \mathbb{R}^{n_j}, \quad \mathbf{u} = \begin{bmatrix} {}^I\mathbf{v}_B \\ {}_B\boldsymbol{\omega}_{IB} \\ \dot{\mathbf{q}}_j \end{bmatrix} \in \mathbb{R}^{n_u}, \quad (1)$$

where  $n_u = 6 + n_j$ ,  ${}^I\mathbf{v}_B \in \mathbb{R}^3$  and  ${}_B\boldsymbol{\omega}_{IB} \in \mathbb{R}^3$  are the linear and angular velocity of the Base w.r.t. the inertial frame expressed respectively in the  $I$  and  $B$  frame. The robot (see Fig. 1) has  $n_u = 24$ , with six, twelve, and six DOF describing the floating base, legs, and the arm, respectively. The equations of motion of a floating base system that interacts with the environment are written as  $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \mathbf{u}) = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}_s^T(\mathbf{q})\boldsymbol{\lambda}$ , where  $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_u \times n_u}$  is the mass matrix and  $\mathbf{h}(\mathbf{q}, \mathbf{u}) \in \mathbb{R}^{n_u}$  is the vector of Coriolis, centrifugal and gravity terms. The selection matrix  $\mathbf{S} = [\mathbf{0}_{n_\tau \times (n_u - n_\tau)} \quad \mathbb{I}_{n_\tau \times n_\tau}]$  selects which DOF are actuated. If all limb joints are actuated, then  $n_\tau = n_j$ . The vector of contact forces and contact torques  $\boldsymbol{\lambda}$  is mapped to the joint-space torques through the support Jacobian  $\mathbf{J}_s \in \mathbb{R}^{n_s \times n_u}$ , which is obtained by stacking the Jacobians which relate generalized velocities to limb end-effector motion as  $\mathbf{J}_s = [\mathbf{J}_{C_1}^T \quad \cdots \quad \mathbf{J}_{C_{n_c}}^T]^T$ , with  $n_c$  the number of limbs

in contact and  $n_s$  the total dimensionality of all contact wrenches. For the point-feet only three dimensional linear contact forces are modeled. In contrast, the gripper exerts a six dimensional contact wrench, when rigidly gripping onto its environment.

## III. MOTION GENERATION

Thanks to the robot's high number of DOF, it is possible to simultaneously and independently control the motion of the floating base and the gripper. The software framework schematically displayed in Fig. 2 allows to send high-level operational space velocity commands in order to drive locomotion in a specified direction, and to move the gripper to the desired pose. For locomotion, these velocity commands, together with the actual robot state, are transformed to reference footholds<sup>1</sup> and motion reference trajectories for the robot's whole-body center of mass (COM). This motion generation framework is based on our previous work [13], that describes a reactive receding-horizon ZMP-based motion planner that enables the execution of dynamic gaits such as a trot, pace and running trot. Continuous online replanning of the motion references results in a reactive behavior of the robot. Hence, the system can cope with unexpected disturbances, such as unmodeled irregularities in the terrain or a push by a human, by updating the motion plans to remain balanced.

### A. Gripper Motion References

For the gripper we continuously update a desired pose  $\mathbf{p}_{IG}^{des}$  and reference twist  ${}^I\mathbf{w}_{IG}^{des}$  to be tracked by the motion controller. These desired values are computed by a gripper motion planner, or by a twist input  $\mathbf{w}$  from a user-operated joystick. In the latter case, the desired pose is updated as  $\mathbf{p}_{IG_{k+1}}^{des} = \mathbf{p}_{IG_k}^{des} + {}^*\Delta\mathbf{tw}$ , where  $\Delta t$  is the duration of the control loop,  $k$  refers to the current time step, and  $+^*$  is defined as the vector space addition operator for the translational part of  $\mathbf{p}_{IG}$  and as  $\boxplus$  [14] for the rotational part. The reference twist is updated as  ${}^I\mathbf{w}_{IG}^{des} = \mathbf{w}$ .

The gripper's motion references can be expressed w.r.t. to any coordinate frame, e.g., the inertial frame  $I$ , or the base frame  $B$ . Expressing the reference motion w.r.t. the inertial frame allows to drive the gripper to a desired position in the world, while the robot is still free to walk, change its posture, and retain balance if an external disturbance is acting on the system. In other situations, such as when the robot has to walk to a different location while carrying a payload, it can be more convenient to express the reference motion of the gripper w.r.t. the base frame  $B$ .

### B. Foothold Planning

Our previous foothold planning framework was based on including the inverted pendulum model [15] in an online optimization problem [13]. The latter additionally included constraints to avoid the kinematic limits of the legs. In this work, we modify this quadratic programming (QP) problem

<sup>1</sup>A *foothold* is defined as the desired contact location for a leg in swing phase.

to plan the footholds w.r.t. the position of the whole-body center of mass instead of the center of the torso. This modification is crucial since changing the arm configuration can impose a significant shift in the overall COM position.

### C. Whole-Body Center of Mass Motion Planning

The desired whole-body COM motion reference trajectory is obtained by solving an online nonlinear optimization [13] which guarantees stable locomotion by constraining the robot's Zero-Moment Point (ZMP) to always lie inside the convex hull of the contact points, i.e., the current and upcoming support polygons. The optimization takes into account different kinds of support polygons (i.e., points, lines, triangles and quadrilaterals). This flexibility makes it possible to generate motion plans for any gait that exhibits these support polygons, from a static walk to a running trot with full flight phases and a pronking gait. This motion planner reduces the model of the robot to a single point mass, being the robot's COM. It does therefore not require any adaptations to apply this planner to a quadrupedal robot equipped with one or more additional limbs.

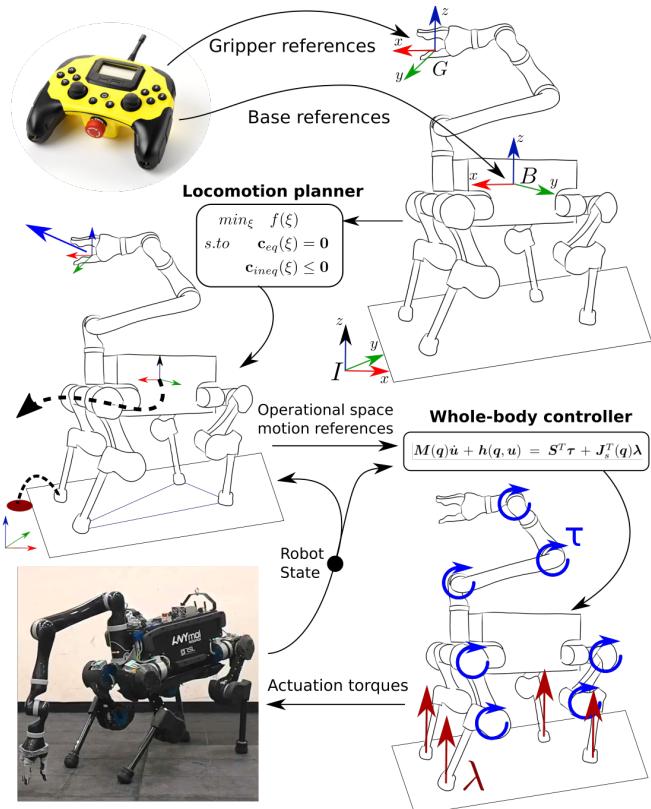


Fig. 2. The planning and control framework described in this paper. High-level velocity references are sent to the floating base or to the gripper. References for the latter are interpreted as velocity updates (w.r.t. the inertial or the floating base frame) for the gripper's desired pose. Velocity references for the base are sent to the online locomotion planner which computes COM trajectories. The operational space references are tracked by a whole-body controller algorithm based on hierarchical optimization that generates torque references for all actuated joints.

## IV. CONTROL

Extending on our previous research [16], [12], we track operational space motion and force references with a whole-body control algorithm that generates torque references for all the controllable joints by using hierarchical optimization. The controller computes optimal generalized accelerations  $\dot{u}^*$  and contact forces  $\lambda^*$  by solving a cascade of prioritized tasks which specify equality and inequality constraints as  $A\xi = b$  and  $C\xi \leq d$ , where  $\xi = [\dot{u}^T \quad \lambda^T]^T$ . The desired torques  $\tau_d$  are obtained from the optimal solution  $\dot{u}^*$  and  $\lambda^*$  as  $\tau_d = M(q)_j \dot{u}^* + h_j(q, u) - J_j(q)_s^T \lambda^*$ , where  $M_j$ ,  $h_j$  and  $J_j$  are the rows of the mass matrix, nonlinear terms and support Jacobian associated with the dynamics of the actuated degrees of freedom. Table I shows the list of prioritized tasks used throughout our experiments. The tasks with the highest priority guarantee dynamic feasibility, compliance with the robot's physical limitations, and adherence to the contact constraints [12]. The tasks at the second priority level specify the robot's desired motion. The lowest priority task removes any internal force redundancy by minimizing the norm of all contact wrenches. The implementation of the tasks dedicated to locomotion is described in [12]. Sections IV-A through IV-C describe the design of tasks tailored for manipulation and how to integrate them into our control framework to allow coordination of locomotion and manipulation.

The manipulation tasks are focused on controlling the gripper's motion and interaction forces with the environment, and increasing the arm's kinematic reachability by adjusting the orientation of the torso. Furthermore, we discuss the adaptation of the motion tracking tasks in order to deal with control instabilities that occur at kinematically singular configurations.

### A. Gripper Motion and Contact Wrenches

In order to track translational and rotational motion references, the spatial motion tracking task for the gripper is written as

$$[J_G \in \mathbb{R}^{6 \times n_u} \quad \mathbf{0}_{6 \times n_s}] \xi = \ddot{x}_{ref} - \dot{J}_G u, \quad (2)$$

TABLE I

THE TASKS USED IN THE EXPERIMENTS. EACH TASK IS ASSOCIATED WITH A PRIORITY (1 IS THE HIGHEST). OF THE TASKS MARKED WITH AN ASTERISK, ONLY ONE IS ACTIVE AT A TIME.

Priority	Task
1	Equations of Motion Torque limits Friction cone limits No contact motion
2	Center of Mass horizontal motion tracking Torso height motion tracking Torso angular motion tracking Swing foot linear motion tracking Torso orientation adaptation Gripper spatial motion tracking* Gripper contact wrench* Contact wrench minimization
3	

where  $\mathbf{J}_G$  is the matrix that maps the robot's generalized velocities to the translational and rotational velocities of the gripper.  $\dot{\mathbf{J}}_G$  is the Jacobian's time derivative, and  $\ddot{\mathbf{x}}_{ref}$  is the operational space accelerations reference for the gripper.

When the gripper is rigidly in contact with the environment, we can remove the motion tracking task and instead command an interaction wrench. Since the contact forces and torques at the gripper appear explicitly in the optimization vector  $\xi$ , the required task can be written as  $[\mathbf{0}_{6 \times n_u} \quad \mathbf{S}_G \mathbf{6} \times n_s] \xi = \mathbf{w}_{ref}$ , where  $\mathbf{S}_G$  is the selection matrix that selects the contact forces and contact torques belonging to the gripper, and  $\mathbf{w}_{ref} \in \mathbb{R}^6$  is the contact wrench reference. When commanding a gripper contact wrench, the support Jacobian  $\mathbf{J}_s$  needs to be updated to include the arm. As a result, the arm is included in the "No contact motion" task.

### B. Torso Orientation Adaptation

A typical task to execute for a mobile manipulator is to reach for and grasp an object. The latter might be, however, out of the kinematic reach (e.g., when on the ground or on a high shelf). This limitation is addressed by exploiting the kinematic redundancy introduced by the floating base through adaptation of the torso orientation without interfering with the reference positions of the COM (to avoid interfering with the stability criterion in Section III-C) and the gripper (see Fig. 3). By adapting the torso's orientation appropriately, the kinematic reach of the gripper is significantly improved.

A possible approach for achieving this whole-body reaching behavior is by exploiting the hierarchical setup of the whole-body controller tasks. This approach requires to have different priority levels for the torso angular motion tracking task and the COM linear motion tracking task. Such a hierarchy setup may significantly degrade the execution of

walking gaits such as a trot, during which the robot is underactuated when only two point-feet are in contact with the environment.

For this reason, we propose the addition of a control task which explicitly prescribes the desired angular motion of the torso to increase the gripper's reachable workspace, instead of depending on the hierarchical nature of the motion controller. First, we define a desired linear velocity  ${}_B v_{IS}^{des}$  for the shoulder (i.e., the mounting point of the arm), expressed in the Base frame. We define  ${}_B v_{IS}^{des}$  as  ${}_B v_{IS}^{des} = k_p {}_B \hat{r}_{SG}$ , with  $k_p$  a scalar which multiplies the unit vector  ${}_B \hat{r}_{SG}$  that points along the direction from shoulder to the gripper. Rotational velocities of the base along its vertical  $z$  and frontal  $x$  axes produce an instantaneous velocity at the shoulder in the same direction. For this reason, the desired shoulder velocity  ${}_B v_{IS}^{des}$  cannot be mapped directly to desired angular base velocities. Instead, we project it to a desired angular velocity for the floating base only around its  $x$  and  $y$  axes as  ${}_B \omega_{IB_{xy}}^{des} = S({}_B r_{BS})^\dagger {}_B v_{IS}^{des}$ , where  $S({}_B r_{BS})^\dagger$  denotes the pseudo-inverse of the skew-symmetric matrix, computed such that  $S({}_B r_{BS}) {}_B v_{IS}^{des} = {}_B r_{BS} \times {}_B v_{IS}^{des}$ , with  ${}_B r_{BS}$  the position of the shoulder w.r.t. the base. Subsequently,  ${}_B \omega_{IB_z}^{des}$  is set proportional to  ${}_B \omega_{IB_x}^{des}$  because their relation to the direction of the velocity of the shoulder is identical. The resulting task is integrated into the task hierarchy as

$$[\mathbf{J}_{B_r} \mathbf{6} \times n_u \quad \mathbf{0}_{3 \times n_s}] \xi = k_d ({}_B \omega_{IB_{xyz}}^{des} - {}_B \omega_{IB_{xyz}}) - \dot{\mathbf{J}}_{B_r} \mathbf{u}, \quad (3)$$

where  $\mathbf{J}_{B_r}$  is the Jacobian of rotational motions of the base,  $k_d$  is a scalar derivative gain,  ${}_B \omega_{IB_{xyz}}^{des}$  and  ${}_B \omega_{IB_{xyz}}$  are the reference and measured angular velocity respectively, and  $\dot{\mathbf{J}}_{B_r}$  is the time derivative of  $\mathbf{J}_{B_r}$ .

### C. Kinematic Singularity Robustness

While executing manipulation tasks, the reference motion for the end-effector can drive a limb to a kinematically singular joint-space configuration. This situation is likely to occur in the form of a full knee or elbow extension, for example when trying to pick up an object that is too far away.

When a limb is in a kinematically singular configuration, it loses one controllable operational space DOF of the end-effector. This situation is characterized by one of the Jacobian's singular values approaching zero, and the controller computing infeasibly high joint torque references, leading to control instabilities. For traditional analytic inverse kinematics and inverse dynamics control approaches, this problem is typically addressed through the use of the damped pseudo-inverse [17]. This approach, however, is not relevant for the presented control framework which relies on numerical optimization to solve for the task hierarchy. Therefore, we address the issue of kinematic singularities by setting a lower bound on the Jacobian's singular values, as mentioned in [18]. The singular value decomposition of the limb's Jacobian that is used, e.g., inside the motion control task in (2) is performed by setting a minimum non-zero value to the computed singular values and by using the latter to

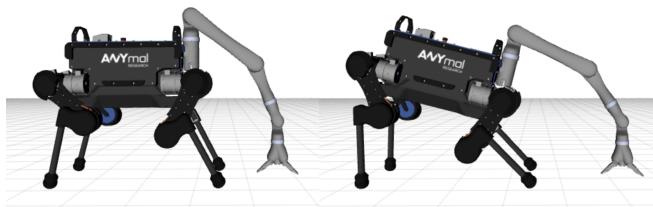


Fig. 3. Reaching on the ground with no torso orientation adaptation (left) results in a limited kinematic reach compared to taking into account the configuration of the arm (right). By including the torso orientation adaption task, the reach in the depicted situation is increased by 15 cm.

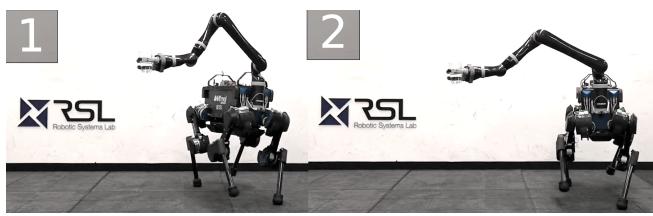


Fig. 4. By combining a ZMP-based locomotion planner with hierarchical whole-body control, the robot is able to keep a glass of water at a fixed pose while the torso is commanded to walk in the desired direction.

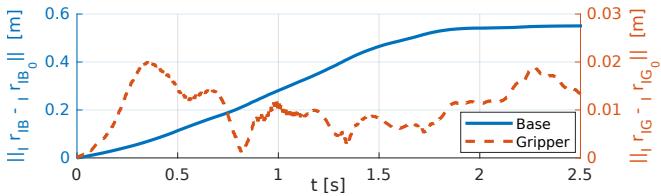


Fig. 5. The motion of the torso and gripper positions  $r_{IB}$  and  $r_{IG}$  w.r.t. the initial positions  $r_{IB_0}$  and  $r_{IG_0}$  when the gripper is controlled to remain at a fixed position during locomotion. While the torso travels 55 cm, the gripper's deviation from its initial position is at most 2 cm.

recompute the Jacobian. This adaptation results in the ability to drive the robot into kinematically singular configurations without any arising motion instabilities. We apply this singular value adjustment for all of the robot's motion tasks: "No contact motion", "Swing foot linear motion tracking", and "Gripper spatial motion tracking".

## V. EXPERIMENTS

Our experiments were conducted on ANYmal [19], an accurately torque-controllable quadrupedal robot, equipped with the Jaco<sup>2</sup> [20] six DOF robotic arm from Kinova. The arm is light-weight (4.4 kg) and allows for torque-control of all six actuators. The control references are generated in a 400 Hz control loop that runs on the robot's onboard computer (Intel i7-7600U, 2.7 - 3.5GHz, dual-core 64-bit) together with state estimation [21]. We use the open-source Rigid Body Dynamics Library [22] (RBDL), a C++ implementation of the algorithms described in [23], to generate the model of the kinematics and dynamics of the system. The locomotion planner uses a custom sequential quadratic programming (SQP) framework that iteratively solves a sequence of QP problems by using a custom version of the open-source QuadProg++ [24] library, a C++ implementation of the Goldfarb-Idnani active-set method [25]. The same algorithm is used to numerically solve the cascade of prioritized tasks in the whole-body controller. The following experiments are supported by the video submission<sup>2</sup>.

### A. Locomotion and End-effector Motion Control

We show the strength of the combination of our ZMP-based locomotion planning framework and whole-body control by commanding the robot to walk in various directions while commanding the gripper to stay at a fixed pose in the inertial frame (see Fig. 4). As depicted in Fig. 5, the robot can accurately track the gripper's desired pose while trotting away from it.

### B. Reactive Behavior and Posture Adaptation

The torque-controlled system and the applied motion and control framework allow to robustly deal with external disturbances. At the actuation level, the robot exhibits compliant behavior because of its torque-controlled joints and the inverse dynamics-based whole-body controller. Instead of the robot being rigidly stiff, external forces on the gripper

or the torso will result in a spring-damper type of motion. On the planning level, the system displays compliance and reactive behavior w.r.t. its environment through the locomotion planning framework that continuously replans in a receding-horizon fashion. The robot's motion references are updated accordingly when external disturbances, such as rough terrain or external forces, act on the robot. The human-robot collaboration scenario, depicted in Fig. 6, demonstrates the system's compliance and reactive behavior. The gripper is commanded to pick up a 3.3 kg box together with a human collaborator. When the box is lifted, the desired locomotion velocity is computed based on the direction of any horizontal forces acting on the gripper. By pulling the box, the human initiates locomotion in the direction of the detected forces. Despite the human pulling on the gripper, and the robot carrying an object with unmodeled weight, a collaborative payload delivery task is accomplished.

Section IV-B describes a whole-body controller task that extends the kinematic reach of the gripper by adapting the angular velocity references of the torso. Fig. 8 depicts how the torso adapts to different configurations of the arm, allowing the robot to reach the ground around itself easily.

### C. Opening a Door

To illustrate the ability of our whole-body control framework to handle commanded contact forces and torques at the gripper, while accurately controlling the robot's COM motion, we demonstrate the execution of a door opening task while using a trotting gait. To push the door and open it, desired contact forces at the gripper are commanded as described in Section IV-A. The direction and magnitude of the desired contact force at the gripper is computed at each moment in time based on the currently estimated door opening angle and the error between a desired and actual gripper velocity. Simultaneously the robot is commanded to trot forward to pass through the door. Fig. 7 shows the execution of this task for a spring-loaded door. The advantage of commanding contact forces for the gripper instead of the desired motion is that the gripper can passively follow the kinematically constrained path prescribed by the door motion, even when contact forces are commanded that are not perfectly tangential to the motion path of the door handle. The whole-body controller task setup, and most specifically the compliance with the high-priority equations of motion task, guarantees a dynamically consistent contact force distribution over the robot's limbs during the execution of this task. The force distribution that is computed while interacting with the door handle is depicted in Fig. 9.

## VI. CONCLUSIONS AND FUTURE WORK

We present results on ANYmal, a quadrupedal robot, equipped with a six DOF robotic arm, forming a fully torque-controlled mobile manipulator. This system is able to perform dynamic locomotion while executing manipulation tasks, e.g., payload delivery, human-robot collaboration, and opening doors. An online ZMP-based motion planning framework is employed on the system to enable robust

<sup>2</sup>Available at <https://youtu.be/XrcLXX4AEWE>

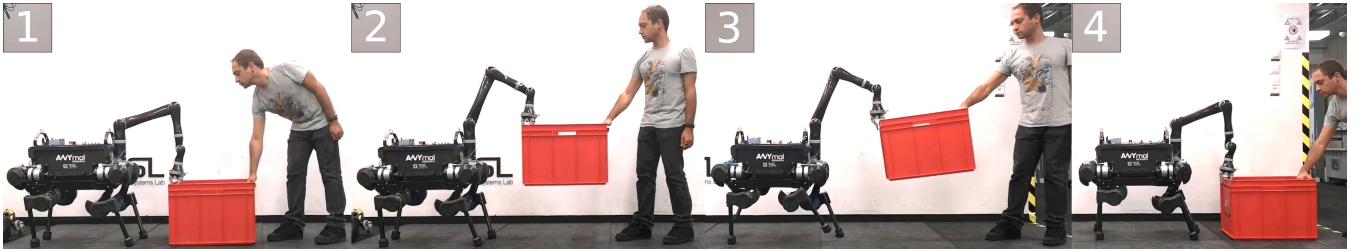


Fig. 6. The robot carrying a 3.3 kg payload together with a human collaborator. Locomotion is triggered when the force on the gripper in the  $x - y$  plane perpendicular to gravity is greater than a user-defined threshold. The online motion planner described in Section III-C computes the required motion reference trajectories for locomotion in the direction of the detected force.

Fig. 7. Commanding a contact force task instead of a motion task for the gripper allows the robot to open a spring-loaded door without requiring exact knowledge of the door kinematics. The integration of this task in the whole-body controller combined with the ZMP-based locomotion controller enables the robot to execute a trotting gait while passing through the door.

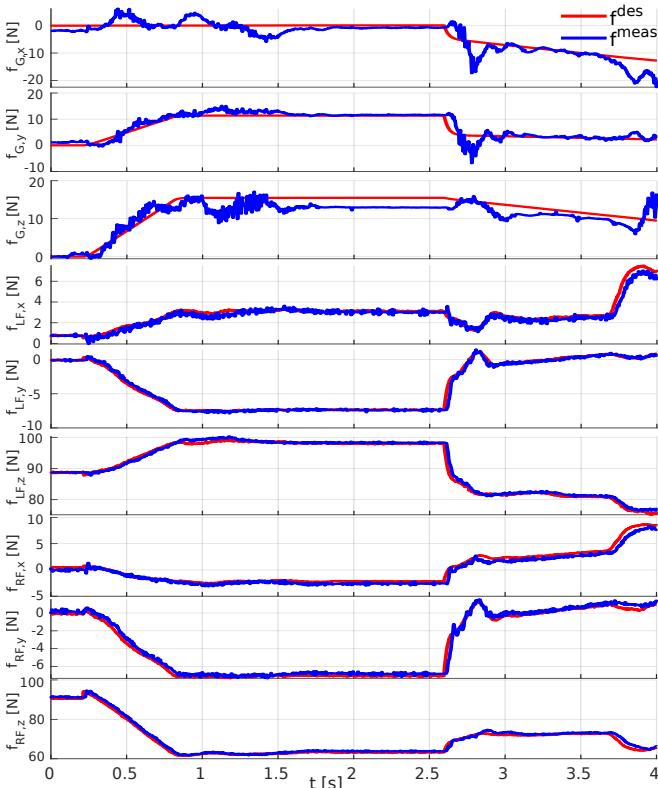


Fig. 9. The time evolution of the commanded and estimated reaction forces acting on the gripper and the left and right front foot when the robot is standing and the gripper is interacting with the handle of a door. The first three plots refer to the gripper being commanded to push both vertically and laterally on the gripper to unlock the door. The commanded gripper force is planned offline. The three middle and lower plots depict the force distribution on the left and right fore leg respectively. The actual forces are estimated based on the measured joint torques. After 2.5 s, the gripper is commanded to push against the door to open it.

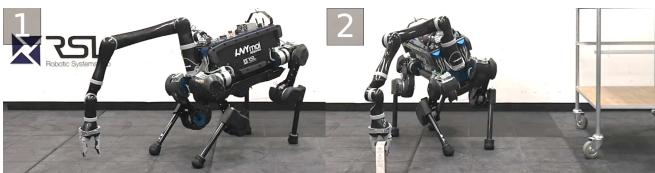


Fig. 8. The torso orientation adaptation task in the whole-body controller task hierarchy results in an emerging whole-body reaching behavior when the gripper is commanded to move to various locations. Frame 1 shows a robot configuration where the torso's pitch contributes to the arm's reach, whereas in frame 2 it is the torso's roll and yaw that contribute significantly.

and reactive locomotion while executing manipulation tasks. Joint torque references are generated by a whole-body controller which takes into account the dynamics of the whole system, in contrast to other works in this field where the arm is seen as a disturbance that needs to be compensated for. The torque control approach allows a compliant and safe interaction with the environment.

Future work will focus on extending the motion planning framework to take into account the contact locations of the hand and the forces that it produces on the environment. Taking these quantities into account will produce motion plans that are consistent with the increased complexity of non-coplanar contact configurations. Specifically, trajectory optimization algorithms should be explored that plan for contact locations for the arm's end-effector. Adding these quantities to the planner's optimization variables allows the generation of complex maneuvers, such as the robot holding itself on a rail while walking on stairs.

## REFERENCES

- [1] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser, and M. Hutter, “Advances in real-world applications for legged robots,” *accepted for Journal of Field Robotics*, 2018.
- [2] P. Fankhauser, “Perceptive Locomotion for Legged Robots in Rough Terrain,” Ph.D. dissertation, ETH Zurich, 2018.
- [3] G. Heppner, T. Buettner, A. Roennau, and R. Dillmann, “Versatile - High Power Gripper for a Six Legged Walking Robot,” in *Mobile Service Robotics*. WORLD SCIENTIFIC, aug 2014, pp. 461–468. [Online]. Available: [http://www.worldscientific.com/doi/abs/10.1142/9789814623353\\_0054](http://www.worldscientific.com/doi/abs/10.1142/9789814623353_0054)
- [4] B. U. Rehman, D. G. Calwell, and C. Semini, “Centaur robots - a survey,” in *Human-centric Robotics-Proceedings Of The 20th International Conference Clawar 2017*. World Scientific, 2017, pp. 247–258.
- [5] B. U. Rehman, M. Focchi, J. Lee, H. Dallali, D. G. Caldwell, and C. Semini, “Towards a multi-legged mobile manipulator,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 3618–3624.

- [6] *Boston Dynamics: Spot*, 2015 (accessed February 27, 2019). [Online]. Available: <https://youtu.be/M8YjvHYbZ9w>
- [7] *Boston Dynamics: SpotMini*, 2018 (accessed February 27, 2019). [Online]. Available: <https://youtu.be/fUyU3IKzio>
- [8] Y. Abe, B. Stephens, M. P. Murphy, and A. A. Rizzi, "Dynamic whole-body robotic manipulation," in *Proc. SPIE 8741, Unmanned Systems Technology*, 2013.
- [9] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, "Optimal distribution of contact forces with inverse-dynamics control," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 280–298, 2013.
- [10] M. Hutter, H. Sommer, C. Gehring, M. Hoepflinger, M. Bloesch, and R. Siegwart, "Quadrupedal locomotion using hierarchical operational space control," *The International Journal of Robotics Research (IJRR)*, vol. 33, no. 8, pp. 1062–1077, may 2014. [Online]. Available: <http://dx.doi.org/10.1177/0278364913519834>
- [11] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Autonomous Robots*, vol. 40, no. 3, pp. 473–491, 2016.
- [12] C. D. Bellicoso, F. Jenelten, P. Fankhauser, C. Gehring, J. Hwangbo, and M. Hutter, "Dynamic Locomotion and Whole-Body Control for Quadrupedal Robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017), Vancouver, Canada, September 2428, 2017*. IEEE, 2017.
- [13] C. D. Bellicoso, F. Jenelten, C. Gehring, and M. Hutter, "Dynamic locomotion through online nonlinear motion optimization for quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2261–2268, July 2018.
- [14] M. Bloesch, H. Sommer, T. Laidlow, M. Burri, G. Nützi, P. Fankhauser, D. Bellicoso, C. Gehring, S. Leutenegger, M. Hutter, and R. Siegwart, "A primer on the differential calculus of 3d orientations," *CoRR*, vol. abs/1606.05285, 2016. [Online]. Available: <http://arxiv.org/abs/1606.05285>
- [15] C. Gehring, S. Coros, M. Hutter, C. D. Bellicoso, H. Heijnen, R. Diethelm, M. Bloesch, P. Fankhauser, J. Hwangbo, M. Hoepflinger, and R. Siegwart, "Practice makes perfect: An optimization-based approach to controlling agile motions for a quadruped robot," *IEEE Robotics Automation Magazine*, vol. 23, no. 1, pp. 34–43, March 2016.
- [16] C. D. Bellicoso, C. Gehring, J. Hwangbo, P. Fankhauser, and M. Hutter, "Perception-less terrain adaptation through whole body control and hierarchical optimization," in *2016 IEEE-RAS 16th Int. Conf. on Humanoid Robots (Humanoids)*, Nov 2016, pp. 558–564.
- [17] A. S. Deo and I. D. Walker, "Overview of damped least-squares methods for inverse kinematics of robot manipulators," *Journal of Intelligent and Robotic Systems*, vol. 14, no. 1, pp. 43–68, 1995.
- [18] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, "Integration of reactive, torque-based self-collision avoidance into a task hierarchy," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1278–1293, 2012.
- [19] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, P. Fankhauser, M. Bloesch, R. Diethelm, and S. Bachmann, "ANYmal - A Highly Mobile and Dynamic Quadrupedal Robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- [20] A. Campeau-Lecours, H. Lamontagne, S. Latour, P. Fauteux, V. Maheu, F. Boucher, C. Deguire, and L.-J. C. L'Ecuyer, "Kinova modular robot arms for service robotics applications," *Int. J. Robot. Appl. Technol.*, vol. 5, no. 2, pp. 49–71, July 2017. [Online]. Available: <https://doi.org/10.4018/IJRAT.2017070104>
- [21] M. Bloesch, M. Burri, H. Sommer, R. Siegwart, and M. Hutter, "The two-state implicit filter recursive estimation for mobile robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 573–580, Jan 2018.
- [22] Martin Felis, "Rigid Body Dynamics Library," Available at <http://rbdl.bitbucket.org/>.
- [23] R. Featherstone, *Rigid Body Dynamics Algorithms*. Boston, MA: Springer US, 2008. [Online]. Available: <http://link.springer.com/10.1007/978-1-4899-7560-7>
- [24] Luca Di Gaspero, "QuadProg++," Available at <http://quadprog.sourceforge.net/>, 1998.
- [25] D. Goldfarb and A. Idnani, "A numerically stable dual method for solving strictly convex quadratic programs," *Mathematical Programming*, vol. 27, no. 1, pp. 1–33, 1983.