Modeling and Control of Multi-Arm and Multi-Leg Robots: Compensating for Object Dynamics during Grasping

Modeling and Control of Multi-Arm and Multi-Leg Robots: Compensating for Object Dynamics during Grasping

Niels Dehio[1,2], Joshua Smith[2], Dennis Leroy Wigand[3], Guiyang Xin[2,4], Hsiu-Chin Lin[2], Jochen J. Steil[1] and Michael Mistry[2]

***Abstract*— We consider a virtual manipulator in grasping scenarios which allows us to capture the effect of the object dynamics. This modeling approach turns a multi-arm robot into an underactuated system. We observe that controlling floating-base multi-leg robots is fundamentally similar. The Projected Inverse Dynamics Control approach is employed for decoupling contact consistent motion generation and controlling contact wrenches. The proposed framework for underactuated robots has been evaluated on an enormous robot hand composed of four KUKA LWR IV+ representing fingers cooperatively manipulating a 9kg box with total 28 actuated DOF and six virtual DOF representing the object as additional free-floating robot link. Finally, we validate the same approach on ANYmal, a floating-base quadruped with 12 actuated DOF. Experiments are performed both in simulation and real world.**

I. Introduction

Multi-objective motion generation with underactuated robots was studied intensively during the last decade. Resolving underactuation is elementary to control quadrupeds and humanoids, because legged robots are typically represented with a floating base consisting of six virtual joints describing the position and orientation of the torso. In multi-contact situations, such as double-support stance or bimanual grasping, end-effectors are *virtually linked* to each other [1], resulting in closed kinematic trees. Controllers specifically addressing constrained motion generation were derived in [2]–[4]. Further works additionally consider the control of contact wrenches [1], [5]–[7].

Contact situations with multi-leg robots are fundamentally similar to grasping situations [8]–[10]: In both scenarios the robot tries to achieve a desired wrench via the contact points while the relative transformation between contacts stays constant. However, grasping situations are typically treated as fixed-base multi-arm robot systems without virtual joints, e.g. [11]–[15]. Inspired by virtual model control [16] and the idea of additional virtual contacts in the grasp matrix [17], we realize that treating the object as an additional link virtually attached to the robot, turns the multi-arm robot into an underactuated system. This allows to incorporate the object dynamics into the controller, which is a key feature to perform manipulation tasks accurately [18]. Furthermore we show that modeling and controlling multi-leg robots is equivalent to the proposed approach for multi-arm robots.

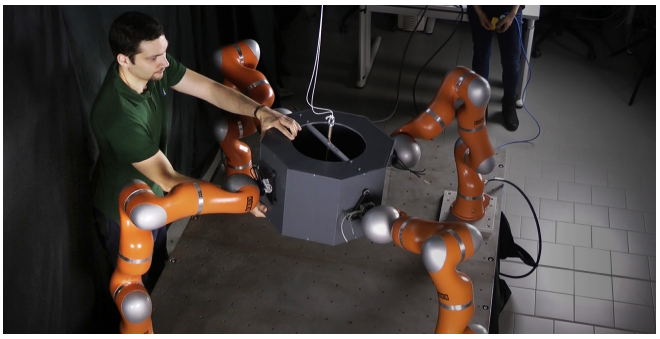

Fig. 1: A enormous robot hand with four fingers manipulates a 9.2kg object. Compensating for object dynamics enables to provide an impedance-based human-robot interaction mode.

The Projected Inverse Dynamics Control (PIDC) approach [19] is a control framework for robots subject to contact constraints. Robot control is decoupled into two orthogonal subspaces, resulting in different and independent control laws for constraint consistent motion generation and realizing desired contact wrenches. The PIDC approach was validated with a torque-controlled manipulator for wiping a board [20] and dual-arm manipulating an object [21]. [22], [23] extend the framework to the underactuated case employing constrained optimization to maintain the contact. We instead aim for an analytic solution to project the desired contact wrench control torques onto the active joints.

We demonstrate impedance-based manipulation of heavy objects modeled by six virtual joints with a robot setup consisting of four cooperative KUKA LWR IV+ manipulators with a total of 28 actuated joints (see Fig. 1). Furthermore we run the same controller on the floating-base quadruped ANYmal with 12 actuated DOF standing on shaped ground.

After related works have been presented in Sec. II, we present our approach in Sec. III. In Sec. IV and V experiments with the multi-arm and multi-leg robots are described. Sec. VI concludes the paper.

II. Background

Related work can be separated into two distinct research topics: defining grasp contact constraints and PIDC.

A. Contact Constraints in Grasping Situations

In the following we consider a robot system with a total of D joints composed of $B > 1$ manipulators manipulating a

[1] Research Institute for Robotics and Process Control (IRP), Technical University Braunschweig, Germany, www.rob.cs.tu-bs.de,
e-mail: ndehio@rob.cs.tu-bs.de
[2] School of Informatics, University of Edinburgh, United Kingdom, www.ed.ac.uk/informatics
[3] Faculty of Technology, Bielefeld University, Germany, www.cor-lab.de
[4] School of Mechanical and Electrical Engineering, Central South University, China, www.csu.edu.cn

rigid object via a rigid grasp with all end-effectors in contact. Hence, end-effectors are *virtually linked* to each other [1], i.e. the relative transformation between them stays constant, representing a closed kinematic chain (simple case) or a closed kinematic tree structure (general case).

The *partial grasp matrix* respect to a global coordinate frame $\mathbf{G}_i \in \mathbb{R}^{6\times6}$ of the i^{th} arm in a multi-arm system is defined by the mapping between the object twist to the contact twists [9]:

$$\mathbf{G}_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{0} \\ \mathbb{S}(\boldsymbol{r}_i) & \mathbf{R}_i \end{bmatrix} , \tag{1}$$

where $\mathbf{R}_i$ represents the rotation matrix of the i^{th} contact frame, $\boldsymbol{r}_i$ is the distance between the i^{th} contact position to the object center-of-mass and $\mathbb{S}(\boldsymbol{r}) \in \mathbb{R}^{3\times3}$ is the skew-symmetric matrix that performs the cross product

$$\mathbb{S}(\boldsymbol{r}) = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} . \tag{2}$$

The *complete grasp matrix* $\mathbf{G} \in \mathbb{R}^{6\times6\,B}$ is the horizontal concatenation of all B partial grasp matrices, representing the relative transformations between all end-effectors

$$\mathbf{G} = [\mathbf{G}_1, ..., \mathbf{G}_B] . \tag{3}$$

The nullspace projection of the complete grasp maptrix $\mathbf{I} - \mathbf{G}^T(\mathbf{G}^+)^T$ projects any arbitrary vector onto the nullspace of the grasp matrix. The resulting contact wrench is usually referred as the internal force, since it produces no net wrench, i.e., $\mathbf{G}\boldsymbol{\lambda}_c = \mathbf{0}$.

During grasping, the constraints should enforce a firm grasp without interfering the motion task. To satisfy this behavior, one should control the contact wrenches such that only the internal wrench is allowed [14] [21]. For this reason, the constraint Jacobian is

$$\mathbf{J}_c \in \mathbb{R}^{6B\times D} = \left(\mathbf{I} - \mathbf{G}^T\left(\mathbf{G}^+\right)^T\right)\mathbf{J}_{ee} , \tag{4}$$

where $\mathbf{J}_{ee}$ denotes the block-diagonal combination of manipulator Jacobian

$$\mathbf{J}_{ee} = \text{blockdiag}\left[\mathbf{J}_1, ..., \mathbf{J}_B\right] \tag{5}$$

The Jacobian associated with the object center-of-mass $\mathbf{J}_o \in \mathbb{R}^{6\times D}$ is then defined by

$$\mathbf{J}_o = \left(\mathbf{G}^+\right)^T\mathbf{J}_{ee}. \tag{6}$$

B. Projected Inverse Dynamics Control

A common theme among many works dealing with floating-base legged robots is that the control schemes are designed to remain dynamically consistent with respect to the contact constraints. Considering the constraints in the primary level of a task hierarchy and employing the inertia matrix in the nullspace projection the motion controllers contribute no acceleration at the constraint locations. Constraints, by definition, are indeed already dynamic consistent, because they do not generate acceleration. Consequently, constraints are able to apply necessary wrenches to maintain their own consistency, and control schemes do not need to address the dynamic consistency property explicitly.

The Projected Inverse Dynamics Control (PIDC) approach [19] assumes bilateral constraints with zero Cartesian velocities and accelerations at the B contact points. The theoretical derivation of the control scheme is based on the static contact assumption with a rigid object. The contact constraint is modeled as

$$\mathbf{J}_c\,\dot{\mathbf{q}} = \mathbf{0} \text{ and } \mathbf{J}_c\,\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c\,\dot{\mathbf{q}} = \mathbf{0} , \tag{7}$$

where $\mathbf{J}_c \in \mathbb{R}^{6\,B\times D}$ describes the constrained Jacobian associated with the B contact points and $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}} \in \mathbb{R}^D$ are generalized coordinates.

Motion generation is not allowed to effect contact constraints by projecting motion tasks $\boldsymbol{\tau}_{\text{motion}} \in \mathbb{R}^D$ onto the nullspace $\mathbf{P} \in \mathbb{R}^{D\times D}$ of the constraint

$$\mathbf{P} = \mathbf{I} - \mathbf{J}_c^T\left(\mathbf{J}_c^+\right)^T , \tag{8}$$

where $\mathbf{J}_c^+$ is the Moore-Penrose inverse. In comparison, the orthogonal subspace $(\mathbf{I} - \mathbf{P})$ is employed for contact wrench control. The total torque command $\boldsymbol{\tau}_{\text{cmd}}$ becomes

$$\boldsymbol{\tau}_{\text{cmd}} = \mathbf{P}\boldsymbol{\tau}_{\text{motion}} + (\mathbf{I} - \mathbf{P})\left(\mathbf{J}_c^T\boldsymbol{\lambda}_c + \boldsymbol{\epsilon} + \mathbf{h}\right) , \tag{9}$$

with

$$\boldsymbol{\epsilon} = \mathbf{M}\,\mathbf{M}_c^{-1}\left(\mathbf{P}\boldsymbol{\tau}_{\text{motion}} - \mathbf{P}\mathbf{h} + \dot{\mathbf{P}}\dot{\mathbf{q}}\right) , \tag{10}$$

and the so-called (invertible) constrained inertia matrix

$$\mathbf{M}_c = \mathbf{P}\mathbf{M} + \mathbf{I} - \mathbf{P}. \tag{11}$$

In (9), $\mathbf{h} \in \mathbb{R}^D$ compensates for gravity and Coriolis effects, $\mathbf{M} \in \mathbb{R}^{D\times D}$ is the inertia matrix, $\boldsymbol{\lambda}_c \in \mathbb{R}^{6\,B}$ is the vertical concatenation of all B desired contact wrenches, and $\boldsymbol{\epsilon} \in \mathbb{R}^D$ enables constraint consistent motion control. For more details, including a detailed derivation, we refer the interested reader to [19]. Within PIDC, motion control $\boldsymbol{\tau}_{\text{motion}}$ may be performed in joint-space ($\boldsymbol{\tau}_{\text{motion}} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{h}$) as shown in [19], or in operational space ($\boldsymbol{\tau}_{\text{motion}} = \mathbf{J}^T\boldsymbol{\lambda}$) as proposed in [4], or may consider multiple prioritized objectives as demonstrated in [20]. Note that force-torque feedback provided by additional sensors is not required. Instead the control law is based only on joint angle and velocity readings.

Note that $\mathbf{P}$ is an orthogonal projection (with $\mathbf{P} = \mathbf{P}\mathbf{P}$ and $\mathbf{P} = \mathbf{P}^T$), because no specific weighting is imposed. This is different from dynamically consistent strict nullspace hierarchies, which employ the inertia matrix as weighting [24]. It is noteworthy that the orthogonal projection is derived only from kinematic parameters, which are easier to accurately determine than inertial parameters [25]. Also, real-robot experiments proved that incorporating the inertia matrix in practice does not significantly show the theoretical conceptual superiority. The authors in [24] realize only minor improvements in tracking performance.

To follow a desired trajectory with an operational point $\mathbf{x} \in \mathbb{R}^6$ associated to the robot by the Jacobian $\mathbf{J}_o$ and realizing an Cartesian impedance behaviour [13], [15] where the desired inertia is identical to the robot inertia, we choose

the motion control law as in our previous work for bimanual grasping (without object dynamics) [21]

$$\boldsymbol{\tau}_{\text{motion}} = \mathbf{J}_o^T \boldsymbol{\lambda}_o + \mathbf{N}_o \left(-\delta \dot{\mathbf{q}}\right) \ , \tag{12}$$

with the wrench $\boldsymbol{\lambda}_o \in \mathbb{R}^6$

$$\boldsymbol{\lambda}_o = \boldsymbol{\Lambda}_o \left[\ddot{\mathbf{x}}_{\text{ref}} + \mathbf{J}_o \mathbf{M}_c^{-1} \left(\mathbf{P}\mathbf{h} - \dot{\mathbf{P}}_c \dot{\mathbf{q}}\right) - \dot{\mathbf{J}}_o \dot{\mathbf{q}}\right] + \boldsymbol{K}\widetilde{\mathbf{x}} + \boldsymbol{D}\dot{\widetilde{\mathbf{x}}} \tag{13}$$

where $\boldsymbol{K}$ and $\boldsymbol{D}$ are damping and stiffness matrices, and employing the dynamically consistent task-space inertia matrix $\boldsymbol{\Lambda}_o \in \mathbb{R}^{6\times 6}$ [4], [24]

$$\boldsymbol{\Lambda}_o = \left(\mathbf{J}_o \, \mathbf{M}_c^{-1} \mathbf{P} \mathbf{J}_o^T\right)^{-1} . \tag{14}$$

Furthermore, to prevent drift in the nullspace $\mathbf{N}_o$ we add a dynamically consistent joint velocity damping task with gain $\delta > 0$ similar to [20].

The feedback control law to compute $\widetilde{\mathbf{x}} \in \mathbb{R}^6$ is based on the current pose and a virtual equilibrium point. Note that realizing a rotational spring in orientation space is not as straight forward as implementing a translational spring [9].

The PIDC approach can be employed for optimization of contact wrenches considering friction cones [22], [23]. We demonstrated optimization of contact wrenches for multi-arm robots in [21] without compensating for object dynamics. However, these approaches are based on prior knowledge about contact surface friction coefficients. Contact wrench optimization is beyond the scope of the current paper. We instead employ heuristics to verify that the chosen wrenches maintain contact to simplify matters. In grasping situations we choose contact forces such that they point towards the center of all contact points. Combining [21] with the approach for object dynamics compensation presented in this paper remains future work.

III. Approach

PIDC was extended in [22] and [23] to the underactuated case. These approaches employ quadratic programming to project desired contact wrenches control torques onto the active joints. We here instead analytically extend the PIDC formalism for robots with passive degree of freedom without need of applying optimization techniques, following and extending [4]. Note that we do not treat the problem of friction cones (inequalities) and rather focus on the equality constraint imposed by underactuated systems. Furthermore, we propose a virtual manipulator with a virtual contact to account for object dynamics in grasping situations and highlight similarities to modeling and control of legged robots. The overall control architecture is depicted in Fig. 2.

A. PIDC for Underactuated Robots

For a robot system with passive degree of freedom, the following equality constraint always must be satisfied [4]:

$$\boldsymbol{\tau} = \mathbf{S}\boldsymbol{\tau} \ , \tag{15}$$

with a diagonal matrix $\mathbf{S} \in \mathbb{R}^{D\times D}$ selecting actuated joints.

In the case of decoupled motion and contact wrench control, and assuming that motion control requires all actuated DOF of the robot, we may still be able to satisfy (15) by adding constraint wrenches to resolve underactuation, without inducing any additional motion. In

$$\boldsymbol{\tau} = \mathbf{P}\boldsymbol{\tau}_{\text{M}} + \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}} + \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{U}} \ , \tag{16}$$

$\boldsymbol{\tau}_{\text{M}}$ performs motion control, $\boldsymbol{\tau}_{\text{C}}$ applies desired contact wrenches and $\boldsymbol{\tau}_{\text{U}}$ are torques that induce only contact wrenches to resolve underactuation.

Inserting (16) into (15), we obtain

$$\left(\mathbf{I} - \mathbf{S}\right)\left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{U}} = \left(\mathbf{I} - \mathbf{S}\right)\left[-\mathbf{P}\boldsymbol{\tau}_{\text{M}} - \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}\right]. \tag{17}$$

We can then solve for $\boldsymbol{\tau}_{\text{U}}$ employing the Moore-Penrose inverse realizing the minimum possible $\|\boldsymbol{\tau}\|$ assuming that all robot actuators are rotary joints

$$\boldsymbol{\tau}_{\text{U}} = \left[\left(\mathbf{I}-\mathbf{S}\right)\left(\mathbf{I}-\mathbf{P}\right)\right]^{+}\left(\mathbf{I}-\mathbf{S}\right)\left[-\mathbf{P}\boldsymbol{\tau}_{\text{M}} - \left(\mathbf{I}-\mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}\right]. \tag{18}$$

Provided (18) has at least one valid solution for $\boldsymbol{\tau}_{\text{U}}$, we can substitute it into (16) and write the control equation as:

$$\begin{aligned}\boldsymbol{\tau} = {} & \mathbf{P}\boldsymbol{\tau}_{\text{M}} + \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}} + \left(\mathbf{I} - \mathbf{P}\right) \\ & \left[\left(\mathbf{I} - \mathbf{S}\right)\left(\mathbf{I} - \mathbf{P}\right)\right]^{+}\left(\mathbf{I} - \mathbf{S}\right)\left[-\mathbf{P}\boldsymbol{\tau}_{\text{M}} - \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}\right].\end{aligned} \tag{19}$$

Because $\left(\mathbf{I} - \mathbf{S}\right)$ and $\left(\mathbf{I} - \mathbf{P}\right)$ are both orthogonal projections, the equation simplifies to[1]

$$\begin{aligned}\boldsymbol{\tau} &= \mathbf{P}\boldsymbol{\tau}_{\text{M}} + \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}} + \\ &\qquad \left[\left(\mathbf{I} - \mathbf{S}\right)\left(\mathbf{I} - \mathbf{P}\right)\right]^{+}\left[-\mathbf{P}\boldsymbol{\tau}_{\text{M}} - \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}\right] \\ &= \left[\mathbf{I} - \left[\left(\mathbf{I} - \mathbf{S}\right)\left(\mathbf{I} - \mathbf{P}\right)\right]^{+}\right]\left[\mathbf{P}\boldsymbol{\tau}_{\text{M}} + \left(\mathbf{I} - \mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}\right].\end{aligned} \tag{20}$$

The solution can be further simplified

$$\boldsymbol{\tau} = \left[\mathbf{P}\mathbf{S}\right]^{+}\mathbf{P}\boldsymbol{\tau}_{\text{M}} + \left[\mathbf{I} - \left[\left(\mathbf{I}-\mathbf{S}\right)\left(\mathbf{I}-\mathbf{P}\right)\right]^{+}\right]\left(\mathbf{I}-\mathbf{P}\right)\boldsymbol{\tau}_{\text{C}}. \tag{21}$$

The nullspace projection $\mathbf{P}$ can be computed based on a QR decomposition of the associated Jacobian transpose $\mathbf{J}_c^T = \left[\mathbf{Q}_1\mathbf{Q}_2\right]\left[\mathbf{R}^T\mathbf{0}\right]^T$, resulting in a formulation without explicit inverse operator [26]

$$\mathbf{P} = \mathbf{I} - \mathbf{Q}_1\mathbf{Q}_1^T = \mathbf{Q}_2\mathbf{Q}_2^T. \tag{22}$$

We realize, that $\left[\mathbf{P}\mathbf{S}\right]^{+}\mathbf{P} = \left[\mathbf{Q}_2^T\mathbf{S}\right]^{+}\mathbf{Q}_2^T$. Consequently, (21) is equivalent to the orthogonal projection approach derived in [3], except that our formulation additionally provides contact wrench control. Further note that (21) reduces to the formulation for a fully actuated robot system with $\mathbf{S} = \mathbf{I}$.

B. Accounting for Object Dynamics with Multi-Arm Robots

The external wrench $\boldsymbol{\lambda}_{ext} \in \mathbb{R}^6$ acting on the object, assuming no human interaction, is given in compact form in the world frame by

$$\boldsymbol{\lambda}_{ext} = \mathbf{M}_o\ddot{\mathbf{x}}_o + \mathbf{h}_o \ , \tag{23}$$

with $\mathbf{M}_o \in \mathbb{R}^{6\times 6}$ as the object inertia tensor, $\mathbf{h}_o \in \mathbb{R}^6$ containing the gravitational and Coriolis effects of the motion

[1] According to [4, Appendix C], for two orthogonal projection operators $\mathbf{X}$ and $\mathbf{Y}$ holds $\mathbf{X}[\mathbf{Y}\mathbf{X}]^{+}\mathbf{Y} = [\mathbf{Y}\mathbf{X}]^{+}$

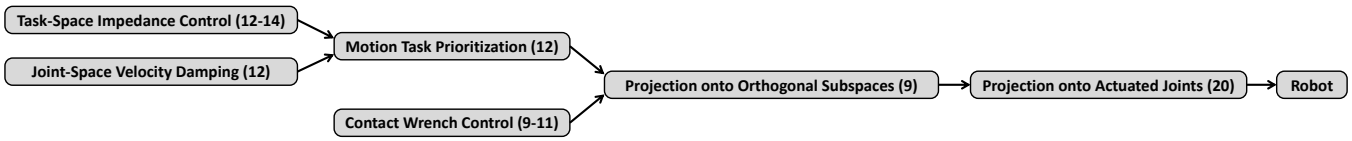

Fig. 2: Block diagram describing the overall control architecture. The numbers refer to the corresponding equations.

and $\ddot{\mathbf{x}}_o \in \mathbb{R}^6$ expressing translational and angular accelerations of the object

$$\mathbf{M}_o = \begin{bmatrix} m_o \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_o \end{bmatrix} \text{ and } \mathbf{h}_o = \begin{bmatrix} m_o \boldsymbol{g} \\ \mathbb{S}(\boldsymbol{\omega}_o)\boldsymbol{I}_o\boldsymbol{\omega}_o \end{bmatrix} , \quad (24)$$

where $m_o \in \mathbb{R}^D$ is the total mass of the object, $\boldsymbol{I}_o \in \mathbb{R}^{3\times3}$ is the symmetric inertia tensor, $\boldsymbol{\omega}_o \in \mathbb{R}^3$ represents angular velocities of the object, $\boldsymbol{g} = [0, 0, -9.81]^T$ is the gravity vector and choosing the object frame such that it coincides with the objects center of mass.

Virtual model control includes virtual components in the control law to move the robot as if these simulated virtual components actually exist. We model the external wrench as a *virtual manipulator* with six DOF acting directly on the objects center of mass. Accordingly, the generalized coordinates are extended with the object pose $\mathbf{x}_o$ containing translation and orientation

$$\mathbf{q} = \left[\mathbf{q}_1^T, ..., \mathbf{q}_B^T, \mathbf{x}_o^T\right]^T, \quad (25)$$

a *virtual contact* represented by $\mathbf{G}_o$ is added to (3)

$$\mathbf{G} = [\mathbf{G}_1, ..., \mathbf{G}_B, \mathbf{G}_o] , \quad (26)$$

and (5) is modified to

$$\mathbf{J}_{ee} = \text{blockdiag}\left[\mathbf{J}_1, ..., \mathbf{J}_B, \mathbf{I}_{6\times6}\right] , \quad (27)$$

Furthermore the robot dynamics given by $\mathbf{M}$ and $\mathbf{h}$ are extended with the object dynamics $\mathbf{M}_o$ and $\mathbf{h}_o$. With this formulation the object can be seen as a free-floating robot link being connected by six virtual joints. The robot system becomes underactuated and cannot be controlled with (9). Instead, the proposed control scheme for the underactuated case (21) can be employed, introducing a diagonal matrix $\mathbf{S} \in \mathbb{R}^{(D+6)\times(D+6)}$ which selects active joints

$$\mathbf{S} = \begin{bmatrix} \mathbf{I}_{D\times D} & \mathbf{0}_{D\times6} \\ \mathbf{0}_{6\times D} & \mathbf{0}_{6\times6} \end{bmatrix}. \quad (28)$$

Without knowledge about object dynamics, one can easily set $m_o = 0$ and $\boldsymbol{I}_o = \mathbf{0}_{3\times3}$, which then corresponds to the fixed-base PIDC approach[2]. However, it is easy to update the object model at runtime, e.g. during a water-bottle pouring tasks.

Note that multi-arm robots are subject to switching contact constraints. Three cases can be distinguished (see also [3] for floating-base multi-leg robots):

1) Multi-arm robots can be seen as fully actuated when there are as many constraints as free floating DOFs (e.g. only one arm in contact). The dynamic model can be reduced to a fixed-base model with the object rigidly attached to the end-effector. There is exactly one solution provided that the desired accelerations are constraint consistent.
2) Multi-arm robots can be underactuated when there are less constraints than free floating DOFs (e.g. loosing all contacts to the object). In this case we loose full control authority. There is at most one solution to the inverse dynamics problem: for a solution to exist, the desired accelerations must not only be constraint consistent, they moreover need to be consistent with the dynamics.
3) Multi-arm robots can be overactuated when there are more constraints than free floating DOFs (e.g. two or more end-effectors in contact) and there exists an infinite choice of possible torques to achieve the desired constraint-consistent motion.

C. Relation to Floating-Base Multi-Leg Robots

In the previous section, the world inertial frame coincides with the fixed-base of the multi-arm robot. As a result, we modeled a free-floating object, with six virtual joints, representing the position and orientation of the object's center with respect to the world inertial frame. However, this is not the only choice: One can also choose the center of the object as world inertial frame, such that the object is rigidly connected to the world. From this perspective, the previously free-floating object becomes static (fixed-base) and the previously fixed-base multi-arm robot turns into a floating-base robot, as illustrated in Fig. 3. This kind of setting is typically used for multi-leg robots such as quadrupeds or humanoids, and the object is referred to be the ground. It is noteworthy that the object dynamics in that representation do not play any role: there exist no object dynamics because the object is rigidly connected to the world. On the other hand, dynamics of the robot's base are considered, which were irrelevant before.

Observing this equivalence, one can apply the previously derived multi-arm controller also to multi-leg robots. Similar to the previous section, maintaining the contact constraint represents a closed kinematic chain for bipedal systems or a closed kinematic tree for robots with more than two legs in contact. The virtual joints representing the robot's floating-base can be treated identical to the virtual joints representing the free-floating object in the previous section.

The main advantage of the floating-base approach for multi-leg robots is that the systems are able to be modeled even when no contacts exist in the environment, e.g. when the system is in flight or in the air. This also translates to multi-arm robots as we can model objects in free space as well in a similar fashion. This formulation is advantageous

[2] In this case the inertia matrix is not positive semidefinite anymore and is not invertible in a direct way. However, one can invert the upper-left corner, which does not contain the object related rows and columns.

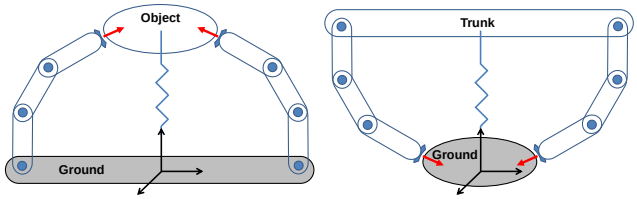

Fig. 3: Schematic view of underactuated kinematic tree structures for a multi-arm robot rigidly grasping an object (left) and a multi-leg robot (right). The contact constraints and floating-base representations are fundamentally similar.

compared to previous work where control schemes have to be switched in case of contact transitions. Note that both multi-leg and multi-arm robot systems experience discontinuities in the control law when making or breaking contacts and abruptly adapting the corresponding constraint.

D. Comparison with state-of-the-art

Early works on underactuated humanoids did not consider the control of contact wrenches [2]–[4]. This was possible, because these robots always were placed on flat, horizontal ground. In that case, gravity prevents slipping when there is enough contact friction. This is similar to a bimanual grasping situation with both end-effectors horizontally aligned below the object, where no contact wrenches are required to maintain the grasp. In that specific situation the contacts are coplanar and the grasp matrix degenerates [27].

[22] and [23] solve the equality constraint imposed by underactuation (15) for controlling contact wrenches based on quadratic programming. We instead solve it analytically. Our formalism is different from the approaches proposed in [1], [5]–[9] because motion control and contact wrench control are projected onto orthogonal subspaces providing the basis for decoupled feedback control laws.

In grasping scenarios, tracking a desired trajectory with the object will be less accurate when neglecting the object dynamics. Manipulating heavy objects is not possible without high PD feedback gains to correct for the external wrenches, which is not desired. As an illustrative example, consider statically holding an object of 9.2kg with the impedance scheme derived in (12), assuming a perfect dynamics model. The external force due to gravity in z direction is $-9.81\frac{\mathrm{m}}{\mathrm{s}^2}9.2\mathrm{kg} \approx -90\mathrm{N}$ and is treated as external disturbance by the impedance controller. As a result, to achieve tracking accuracy of $\Delta x = 0.01\mathrm{m}$ one would have to set the proportional gain to 9000. Without incorporating object dynamics in the control scheme, one cannot achieve zero steady state error. Furthermore, when increasing the objects weight incrementally, the object will always move down due to non-compensated gravity.

In our previous grasping work [21] we neglected object dynamics and, due to that reason, handled only lightweight objects similar as many other related works. In contrast, the authors in [9] report an experiment for lifting an object of 12.2kg exhibiting tracking performance issues, due to non-modeled dynamics, that lead to non-zero steady state error.

To avoid this issue one can treat the object as massless and alternatively update the dynamic model of the end-effector link as proposed in [28]. However, we believe that our strategy is more general and advantageous when switching contact constraints. Recently, [18] proposed an approach for incorporating object dynamics in the control law, utilizing the grasp matrix (3). This method however is based on a decentralized scheme, controlling each robot arm independently without explicit communication between manipulators and requires a fixed-base dynamic model. [13] demonstrates an ability to add object dynamics for fixed-base robots, requiring wrist-mounted force-torque sensors, which are not needed within our approach. Another approach to compensate for object dynamics is to add the term $\mathbf{J}_o^T \boldsymbol{\lambda}_{ext}$ to (12). This causes the corresponding nullspace to lose dynamic consistency.

The "virtual grasp" concept in [12] introduces virtual springs attached to the contact points instead of applying the grasp matrix. Therefore it is different from our approach introducing a virtual manipulator with a virtual contact.

IV. Experiments with four-fingered robot hand

We extensively tested our approach with the dynamics simulation available in Gazebo simulator[3] using ODE before conducting experiments on the real robot, reproducing the results. Due to page-limit constraints, only results obtained from the real robots are reported. A video for agile and dexterous multi-arm object manipulation is online available at `https://youtu.be/Ao-0W9chAd4`.

To obtain the current object pose we do not rely on external sensors, such as vision tracking systems. Instead we infer the pose based on the assumption of a rigid grasp with constant relative transformation between end-effectors and object center. Note that one can also detect slipping (by evaluating $\mathbf{J}_c \dot{\mathbf{q}} \neq \mathbf{0}$) to update the grasp matrix if necessary.

A. Robot Hand Platform

We compose a highly redundant robot system by mounting four KUKA LWR IV+ manipulators on a table, which we treat as a single underactuated robot with 28 active joints and six virtual joints for the free-floating object. The end-effectors consist of a triangular metal plate with three small rubber feet mounted near each corner. This enables us to make stable contact also on non-planar surfaces, even though we are controlling only a single contact point per manipulator. To the authors knowledge, this is the first publication treating more than three industrial manipulators in a cooperative manner for dexterous object manipulation via a force-closed grasp. Employing the well-known grasp matrix constraint described previously this robot system represents an enormous hand with four fingers, each of approximately 1.2m length when including the end-effectors.

Throughout our grasping experiments we grasp two different solid objects, both with 0.3m height. A cylinder (mass $m = 3.0\mathrm{kg}$, radius $r = 0.15\mathrm{m}$) made out of transparent

[3] See http://gazebosim.org/

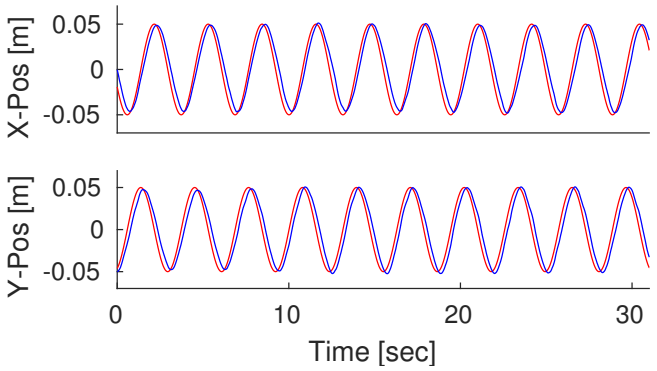

Fig. 4: Accurately tracking a circular trajectory with a diameter of 0.1 m. The red line indicates the desired object position and the blue line the estimated position.

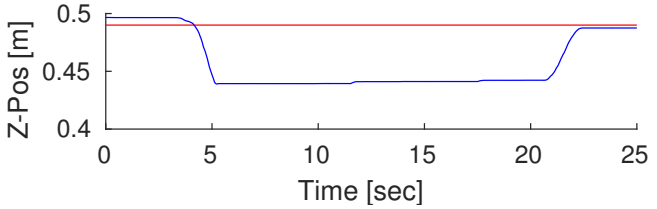

Fig. 5: When adding additional 3kg mass to the grasped cylinder-like object (total 6kg) it drops down. We then incrementally update the internal model in three steps, each of adding 1kg to the object dynamic parameters. Because of high joint friction, the object moves back to its initial pose when fully compensating for its total weight. The red line indicates the desired object position and the blue line the estimated position.

plastic and a bulky box (mass $m = 9.2$kg) with an octaedal base area (each surface width $l = 0.2$m) made out of hard plastic. We chose these two objects to demonstrate manipulation of heavy objects and grasping non-planar surfaces. For simplification, we assume that object dynamics as well as object shape with predefined desired contact points and associated contact surface normals are known to the controller. This is legitimate as other works on grasping are based on the same assumptions.

The four manipulators are provided by different labs and vary with respect to their hardware parameters: They have been used for different applications for different amounts of time, also some joints were replaced by the manufacturer. Accordingly, we experienced uneven wear and tear for each unit, resulting in different joint friction behavior. However, our current implementation is based on a single dynamic model for all four manipulators. In future we target to improve the existing model for each robot arm individually by applying learning techniques to handle non-linearities, friction and temperature effects.

Our C++ implementation is separated into multiple Orocos components as indicated in Fig. 2. For more information regarding our domain-specifc-modeling approach we refer to [29]. The final reference torques are sent to the four control units via the KUKA FRI employing the joint-space impedance control mode using the hard real-time Orocos execution environment[4] with Linux and Xenomai as in [30].

B. Incorporating Object Dynamics during Manipulation

First of all, we demonstrate the overall controller, including underactuation, with standard operational space control and no impedance behavior, for tracking a circular trajectory in the horizontal plane with the 9.2kg box object. The contact forces are heuristically chosen such that they point towards the center of all contacts. Results are shown in Fig. 4. All four robot arms cooperatively hold the object without slipping effects while following the desired trajectory precisely.

C. Online Adaptation according to varying Object Dynamics

In this scenario we aim for online adaptation of object dynamics. The 3kg cylinder is at rest and a human adds 3kg sand as illustrated in Fig. 6, to drastically double the objects total mass. Due to the non-compensated additional weight the cylinder moves down. Next we update the internal object model manually and accordingly the controller moves the object back to its desired position. Results are plotted in Fig. 5. We experience precision issues due to high joint friction with the real robot, which is not the case in simulation. In the future we would like to apply a method for online identification of object dynamics [18] [31] to deal with uncertain/unknown objects or with those incorporating changing dynamics.

D. Assisted Gravity Compensation Mode for Interaction

The Cartesian impedance formulation derived in (12) treats the object as a spring-damper system, transforming deviations from the desired pose into wrenches. This enables a chosen explicit behavior of the system with respect to an external disturbance, e.g. from human interaction with the object. Because we compensate for object dynamics, we can set the proportional feedback gain to zero (no stiffness) and command zero desired velocities and accelerations to realize an *assisted gravity compensation mode* with velocity damping. This enables the human to easily move the cumbersome object to a desired position and orientation, where the controller then performs pure gravity compensation. A snapshot from the accompanying video is shown in Fig. 1. Such a mode is highly beneficial in programming-by-demonstration applications, e.g. [32] for teaching bimanual manipulation.

V. Quadruped Experiments

A. Quadruped Robot Platform

The experimental platform is the ANYbotics quadruped robot, ANYmal (cf. Fig. 7), a torque-controlled robot with 12 actuated joints [33]. A dynamic model is delivered by the manufacturer. The position and orientation of the robot torso are chosen as the operational space for the Cartesian impedance behavior. We treat the robot feet as point contacts where the positions are fixed on the ground and orientations

[4]See http://www.orocos.org/

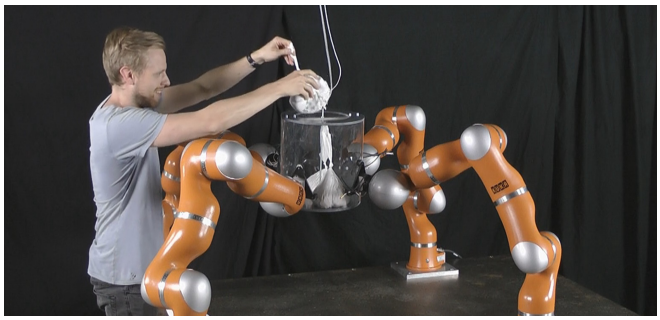

Fig. 6: When adding mass to the object, the internal model can easily be updated. The controller then compensates for the new object dynamics.

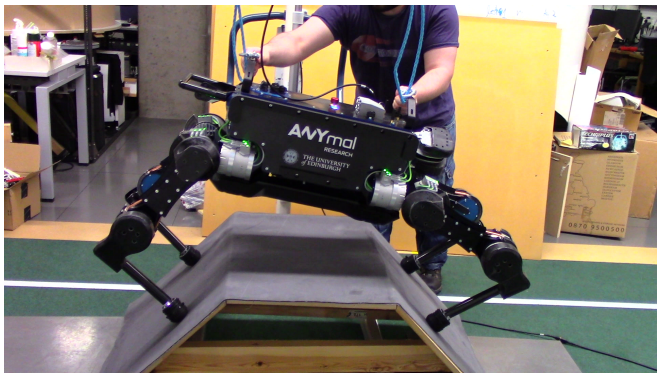

Fig. 7: ANYmal standing on an a slope ramp. Because of the imposed impedance behavior, the robot reacts to human interaction as a spring-damper system.

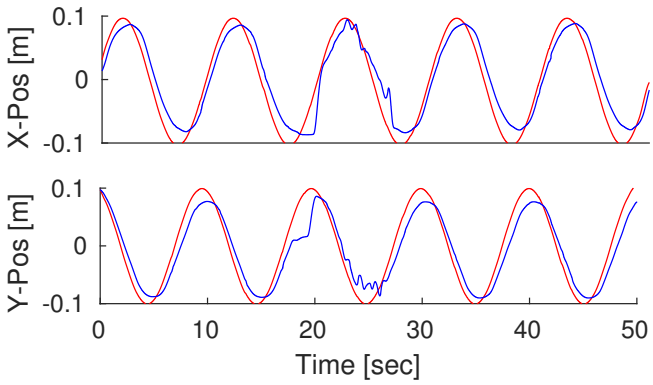

Fig. 8: The torso tracks a circular trajectory in the horizontal plane. Desired positions are red and estimated blue. After 18 seconds, the robot is perturbed for a short period.

are not considered. Accordingly, we cannot control torques at the contacts.

Note that the constraint Jacobian based on the grasp matrix constrains the end-effectors relatively (4). Instead one can impose an absolute constraint by choosing $\mathbf{J}_c = \mathbf{J}_{ee}$ which is supposed to be more robust to external disturbances, at the cost of losing redundancy in the motion control.

The supplementary video with all quadruped experiments is available at `https://youtu.be/Ao-0W9chAd4`.

B. Decoupled Motion and Contact Force Control

In the first experiment with the quadruped, we show that our controller can perform on flat terrain demonstrating the robots workspace. Similar to the previous section, the torso is commanded to follow a circular trajectory inside the x-y plane, while the end-effectors push toward the ground to avoid slipping. We set an appropriate $\boldsymbol{\lambda}_c$ in order to make the actual contact forces stay within the friction cones. The desired and estimated horizontal position of the torso is plotted in Fig. 8. The trajectory is tracked precisely, the controller also recovers from external disturbances.

C. Human Interaction while standing on a Ramp

Next we demonstrate the robot reactions to human interaction. The robot stands on a slope ramp with inclined contact surfaces of 45 degrees each. In order to show the properties of the impedance controller, we set stiffness and damping such that the robot is compliant in x-direction but stiffer in the y- and z-axis. When pushing against the torso, the robot behaves as a spring-damper system. A snapshot of the accompanying video is shown in Fig. 7.

In simulation, we also tried to place the quadruped on the ramp without controlling contact forces. The robot fails to maintain contact as expected. This shows the underlying advantage of our controller compared to [2]–[4].

VI. Conclusion and Future Work

The approach presented in this paper extends earlier work on dexterous grasping by simultaneously controlling four industrial manipulators, representing an enormous robot hand. Employing a virtual manipulator we include the object dynamics for precise manipulation, which turns the robot into an underactuated system. Furthermore, we solve the underactuation constraint for contact wrench control analytically.

PIDC decouples motion generation and contact wrench control, resulting in different and independent control laws. The unconstrained (motion) controller accomplishes a tracking task with desired impedance behavior to deal with external disturbances while the constrained component enforces the contact. The present work is evaluated on both multi-arm and multi-leg platforms, demonstrating our controllers robustness and ability to maintain a grasp, subject to unknown human interactions. The analogy between multi-arm and multi-leg robots has been recognized long ago, however, the similarities when describing the contact situations have been more recent. This paper additionally shows the equivalence between the systems under the same modeling and control framework.

Our approach, as well as discussed alternative approaches, requires knowledge about the contact situation. Here we assumed an a priori well known world and make use of simple heuristics, e.g. only end-effector is in contact. In scenarios where this is not the case, the robot has to detect contact constraints first before being able to exploit them. Recently, we did a first step in this direction, however for a very simple scenario [34]. Methods which estimate external

wrenches and their application locations on-line may also detect contacts [35], [36].

In a follow-up paper we will integrate friction cone optimization, add force-torque sensors to evaluate the accuracy of realized contact wrenches and compare the optimization-based approach with our analytic solution. Our approach is also suited for underactuated motion planning. Future research includes testing the approach with actual robot hands that contain passive elements, e.g. [37].

ACKNOWLEDGMENT

This work received partial funding by the European Community's Horizon 2020 robotics program: CogIMon ICT-23-2014 644727 and THING ICT-2017-1 780883, as well as EPSRC's: ORCA EPR026173/1 and NCNR EPR02572X/1.

REFERENCES

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

[2] L. Sentis and O. Khatib, "A Whole-Body Control Framework for Humanoids Operating in Human Environments," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2006, pp. 2641–2648.

[3] L. Righetti, J. Buchli, M. Mistry, and S. Schaal, "Inverse dynamics control of floating-base robots with external constraints: a unified view," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2011, pp. 1085–1090.

[4] M. Mistry and L. Righetti, "Operational space control of constrained and underactuated systems," in *Proceedings of Robotics: Science and Systems VII*. MIT Press, 2011.

[5] Y. Lee, S. Hwang, and J. Park, "Balancing of humanoid robot using contact force/moment control by task-oriented whole body control framework," *Autonomous Robots*, vol. 40, no. 3, pp. 457–472, 2016.

[6] 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.

[7] M. Focchi, A. del Prete, I. Havoutis, R. Featherstone, D. G. Caldwell, and C. Semini, "High-slope terrain locomotion for torque-controlled quadruped robots," *Autonomous Robots*, vol. 41, no. 1, pp. 259–272, 2017.

[8] C. Ott, M. A. Roa, and G. Hirzinger, "Posture and balance control for biped robots based on contact force optimization," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2011, pp. 26–33.

[9] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *Int. Journal of Robotics Research*, vol. 35, no. 12, pp. 1522–1543, 2016.

[10] T. Asfour, J. Borràs, C. Mandery, P. Kaiser, E. E. Aksoy, and M. Grotz, *On the Dualities Between Grasping and Whole-Body Loco-Manipulation Tasks*. Cham: Springer International Publishing, 2018, pp. 305–322.

[11] A. M. . Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, vol. 1, 2000, pp. 255–262.

[12] T. Wimbock, C. Ott, and G. Hirzinger, "Analysis and experimental evaluation of the intrinsically passive controller (ipc) for multifingered hands," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2008, pp. 278–284.

[13] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-dof impedance control of dual-arm cooperative manipulators," *IEEE Transactions on Mechatronics*, vol. 13, no. 5, pp. 576–586, 2008.

[14] S. Erhart and S. Hirche, "Internal force analysis and load distribution for cooperative multi-robot manipulation," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1238–1243, 2015.

[15] B. Nemec, N. Likar, A. Gams, and A. Ude, "Bimanual human robot cooperation with adaptive stiffness control," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2016, pp. 607–613.

[16] J. Pratt, C.-M. Chew, A. Torres, P. Dilworth, and G. Pratt, "Virtual model control: An intuitive approach for bipedal locomotion," *Int. Journal of Robotics Research*, vol. 20, no. 2, pp. 129–143, 2001.

[17] R. Haschke, J. J. Steil, I. Steuwer, and H. Ritter, "Task-oriented quality measures for dextrous grasping," in *Int. Symposium on Computational Intelligence in Robotics and Automation*, 2005, pp. 689–694.

[18] A. Marino, G. Muscio, and F. Pierri, "Distributed cooperative object parameter estimation and manipulation without explicit communication," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2017, pp. 2110–2116.

[19] F. Aghili, "A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection operator: Applications to control and simulation," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 834–849, 2005.

[20] V. Ortenzi, M. Adjigble, J. A. Kuo, R. Stolkin, and M. Mistry, "An experimental study of robot control during environmental contacts based on projected operational space dynamics," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2014, pp. 407–412.

[21] H.-C. Lin, J. Smith, K. Kouhkiloui Babarahmati, N. Dehio, and M. Mistry, "A projected inverse dynamics approach for multi-arm cartesian impedance control," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2018.

[22] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, "Optimal distribution of contact forces with inverse-dynamics control," *Int. Journal of Robotics Research*, vol. 32, no. 3, pp. 280–298, 2013.

[23] F. Aghili and C.-Y. Su, "Control of constrained robots subject to unilateral contacts and friction cone constraints," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2016, pp. 2347–2352.

[24] A. Dietrich, C. Ott, and A. Albu-Schäffer, "An overview of null space projections for redundant, torque-controlled robots," *Int. Journal of Robotics Research*, vol. 34, no. 11, pp. 1385–1400, 2015.

[25] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *Int. Journal of Robotics Research*, vol. 27, no. 6, pp. 737–757, 2008.

[26] F. Flacco and A. De Luca, "Fast redundancy resolution for high-dimensional robots executing prioritized tasks under hard bounds in the joint space," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 2500–2506.

[27] T. Yoshikawa, "Virtual truss model for characterization of internal forces for multiple finger grasps," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 5, pp. 941–947, 1999.

[28] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *Int. Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[29] D. L. Wigand, A. Nordmann, N. Dehio, M. Mistry, and S. Wrede, "Domain-Specific Language Modularization Scheme Applied to a Multi-Arm Robotics Use-Case," *Journal of Software Engineering for Robotics*, vol. 8, no. 1, pp. 45–64, 2017.

[30] M. Liu, S. Hak, and V. Padois, "Generalized projector for task priority transitions during hierarchical control," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2015, pp. 768–773.

[31] D. Ćehajić, P. B. g. Dohmann, and S. Hirche, "Estimating unknown object dynamics in human-robot manipulation tasks," in *IEEE/RSJ Int. Conf. on Robotics and Automation*, 2017, pp. 1730–1737.

[32] J. Silvério, L. Rozo, S. Calinon, and D. G. Caldwell, "Learning bimanual end-effector poses from demonstrations using task-parameterized dynamical systems," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2015, pp. 464–470.

[33] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, R. Diethelm, S. Bachmann, A. Melzer, and M. Hoepflinger, "Anymal - a highly mobile and dynamic quadrupedal robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 38–44.

[34] V. Ortenzi, H.-C. Lin, M. Azad, R. Stolkin, J. A. Kuo, and M. Mistry, "Kinematics-based estimation of contact constraints using only proprioception," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2016, pp. 1304–1311.

[35] S. Faraji and A. J. Ijspeert, "Designing a virtual whole body tactile sensor suit for a simulated humanoid robot using inverse dynamics," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 5564–5571.

[36] F. Flacco, A. Paolillo, and A. Kheddar, "Residual-based contacts estimation for humanoid robots," in *IEEE/RAS Int. Conf. on Humanoid Robots*, 2016, pp. 409–415.

[37] L. U. Odhner and A. M. Dollar, "Stable, open-loop precision manipulation with underactuated hands," *Int. Journal of Robotics Research*, vol. 34, no. 11, pp. 1347–1360, 2015.