

Motion Planning for Redundant Manipulators in Uncertain Environments based on Tactile Feedback

Christoph Schuetz, Julian Pfaff, Felix Sygulla, Daniel Rixen and Heinz Ulbrich

Abstract—The exploitation of new fields of application in addition to traditional industrial production for robot manipulators (e.g. agriculture, human areas) requires extensions to the sensor as well as to the planning capabilities. Motion planning solely based on visual information performs poorly in cluttered environments since contacts with obstacles might be inevitable and thus a distinction between hard and soft objects has to be made. In our contribution we present a novel intrinsic tactile sensing module mounted on a multipurpose 9 DOF agricultural manipulator. With its innovative sensor arrangement we consider it to be a low-cost, easily manageable and efficient solution with a reasonable abstraction layer in comparison to complex torque sensing or tactile skins. The sensor provides information about the resulting force and torque. In the second part of our paper, the tactile information is used for minimizing contact forces while pursuing the end-effector tasks as long as reasonable. Hence, we present robust and efficient extensions to Resolved Motion Rate Control for real-time application. We introduce a general formulation providing control inputs in task-space, joint-space and nullspace. Thus, we design a suitable controller by feedback linearization and feed-forward terms. Results from real-world experiments show the potential of our approach. A discussion of the different control schemes completes the paper.

I. INTRODUCTION

The application of robot systems for handling, pick-and-place, welding and similar tasks has widely penetrated large-volume manufacturing processes in industry. The tasks are characterized by repetitive sequences in well-defined environments executed by heavy and fixed industrial manipulators. In contrast, agricultural tasks such as selective harvesting of single crops in greenhouses or orchards are still done by human workers. Such tasks put high demands on the robot system: robots have to cope with dense clutter, tasks are not repetitive and cannot be defined with the same accuracy as in industry. Common approaches and assumptions from robotics are not suited to the diverse requirements in this field: information about the environment is usually gathered by vision systems. Obstacles are handled as rigid bodies without considering their mechanical properties (stiffness, mass). This assumption obviously fails in soft and cluttered environments like greenhouses: in order to reach its goal, the manipulator is mostly forced to push soft plant parts such as leaves, stems or fruits. Additionally, common vision based sensors are not able to distinguish between soft and hard objects and obstacles might be occluded.

Christoph Schuetz, Julian Pfaff, Felix Sygulla, Daniel Rixen and Heinz Ulbrich are with the Institute of Applied Mechanics, Technische Universität München, Boltzmannstr. 15, 85748 Garching, Germany {christoph.schuetz, julianpfaff, felix.sygulla, rixen, ulbrich}@tum.de

Tactile information can cope with these challenges: in biology this sensing ability is considered to be vital for fine manipulation tasks, exploration of unknown regions and as a response for own actions [1]. Taking advantage of this sense by robot manipulators requires two essential components: firstly, a sensing module is needed to measure external forces on the robot. Secondly, the planning algorithms of the manipulator system must be able to process this information and compute suitable reactions. In our contribution, we present a simple and efficient tactile sensor module which has been applied to a 9-DOF multipurpose agricultural manipulator [2]. Furthermore, we develop two approaches for a reactive inverse kinematics planning algorithm and evaluate them in suitable real-world experiments.

II. RELATED WORK

While being in focus of research for many years, earlier concepts in the field of whole-arm tactile sensing have not reached a wide-spread commercial application in robotics. In this section, we present several concepts for sensing hardware and planning approaches and put them in the context to our contribution.

A. Tactile Sensing

In industrial production, robot systems operate in separated and secured areas and the use of force sensing devices is mostly limited to a force-torque sensor at the manipulator wrist as proposed by [15] for force-controlled assembly tasks. In contrast, detecting collisions between arm and environment requires whole-arm tactile sensing abilities. These can be subdivided into intrinsic and extrinsic sensor concepts. The intrinsic sensing was proposed by [16]. It describes a method to calculate an external force and its contact location on a known structure by interpreting the information of a force torque sensor which carries the structure. Compared to that, the extrinsic sensing uses an array of sensors on the surface of a structure to determine the area in which the contact occurs [17], [18], [19]. The intrinsic approach was investigated by [20], [21] with force-torque-sensors as well as by [22] who first used joint torque sensors to evaluate the contact forces on a manipulator. The sensor arrays of extrinsic sensors are not only used to determine a precise contact area but also to measure characteristics like temperature [23] or proximity and acceleration [24] to equip robots with a human like skin. An overview of intrinsic and extrinsic sensors is given in [25]. Aiming at keeping the hardware sensor setup as simple as possible, we abstract the tactile information to *one* resulting force vector per link.

This assumption seems to be admissible for most applications in cluttered environments. However we emphasize, that our concept can deal with multi-contacts along the arm (Sec. IV-F), since the sensing modules are not coupled.

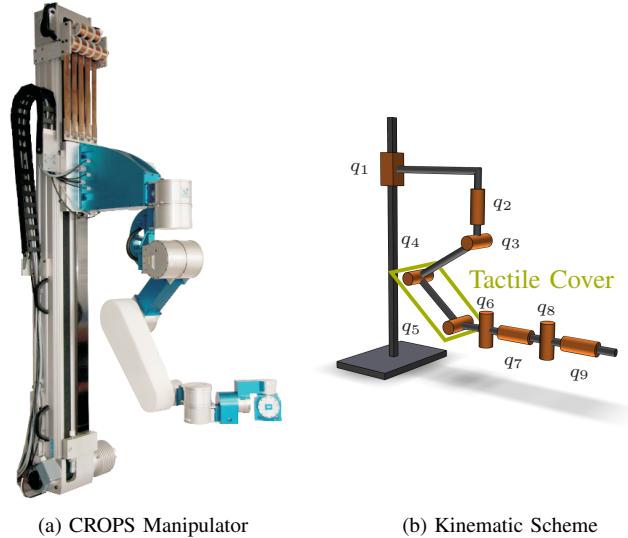
Since intrinsic and extrinsic sensors are both well suited for this (c.f. [26]) and we want an easy and robust system, we decided to use an intrinsic sensor. Methods which use joint torques to determine collisions [27] presume a very precise model of the manipulator and cannot resolve multi-contact scenarios along the arm. Furthermore it is necessary to know the mass of handled objects, which in our case is unfeasible. Commercially available force-torque-sensors as used in [20] are either too big to be applied to all joints or, if smaller, have an insufficient measurement range.

B. Tactile Planning

For the classic approach of collision avoidance the working area of the robot is classified into free space C_{free} and obstructed areas C_{obs} . There exist many well-established and efficient motion planning algorithms and frameworks for finding feasible paths in C_{free} without accounting for mechanical obstacle properties or dynamic environments [3]–[5]. Some related planners are able to exploit the mobility of obstacles by pushing them [6], [7]. However, they are designed as open-loop and require the pre-knowledge of objects' properties. Others are able to deal with deformable obstacles based on their deformation models [8], [9]. These approaches are not suited for the application in cluttered and biological environments: it is very likely, that no feasible path in C_{free} exists. Obstacles have a wide range of mechanical properties, from very low to high stiffness, masses and mobility. Usually, no precise model is available, thus, the integration of tactile feedback into motion planning is needed.

In robot control, the use of force-feedback has been extensively studied. Motivated by highly precise assembly tasks in stiff environments, compliant behavior is required at the end-effector. Besides passive solutions, wrist-mounted force-torque sensors allow for active compliant control schemes such as impedance and admittance control or hybrid force-motion control approaches. For an overview and further references see [10]. A framework considering multiple contacts at the robot arm has been presented in [11]. While the principal task is to control external forces, desired task-space motion is projected into the nullspace. An optimization based approach in a similar problem to ours has been presented in [12]. While considering task-space tasks, force is being minimized by solving a QP respecting force limits, kinematic joint limits and desired goal positions. Optimization is done in a MPC-like scheme with a horizon of only one timestep, thus, the solution is considered to be only locally optimal. Since solving a QP requires a considerable computational effort, reported sampling rates are between 10-20 ms.

In our contribution we propose extensions to the Resolved Motion Rate Control [13] as well as Resolved Acceleration Control [14] schemes based on nonlinear control theory. Unlike traditional force-torque control schemes, our approach



(a) CROPS Manipulator

(b) Kinematic Scheme

Fig. 1: CROPS multipurpose 9-DOF manipulator with a tactile sensing prototype module at link 4 (*left*) and its kinematics (*right*).

allows for a selective modification of the nullspace and/or taskspace of the manipulator depending on the contact forces. Additionally, instantaneous inverse kinematic schemes are considered to be locally optimal w.r.t. the cost functions, but with a much lower computational effort. Thus, we achieve sampling rates of 1 ms using these advanced approaches obtaining easy tunable and robust behavior of the manipulator.

III. HARDWARE PLATFORM

The design and capabilities of the planning framework are based on the available hardware concept. The hardware design defines the processable tactile information, thresholds, limits and resolution. In this chapter, we present our manipulator, the tactile sensing module and communication as well as software architecture.

A. CROPS Agricultural Manipulator

Our experiments are based on the CROPS agricultural manipulator (Generation 2) that has been developed at our institute within the EU-project CROPS¹ for versatile tasks in agriculture such as harvesting sweet-pepper, apples and grapes or precision spot spraying of diseased areas of grapes [2]. The manipulator in its 9-DOF configuration as well as its kinematic scheme are depicted in Fig. 1. The tactile sensing module prototype is designed for link 4 of the manipulator. Due to the manipulator design based on simple geometric primitives, other links can be covered with similar modules.

B. Tactile Sensing Module

The tactile sensing module consists of a rigid PA cover, which is fixed to a sensor frame that is mounted on the robot link. Fig. 2 depicts this setup as well as the interface

¹CROPS: Clever Robots for Crops, European Project within the 7th Framework Program, 10/2010–09/2014, GA 246252.

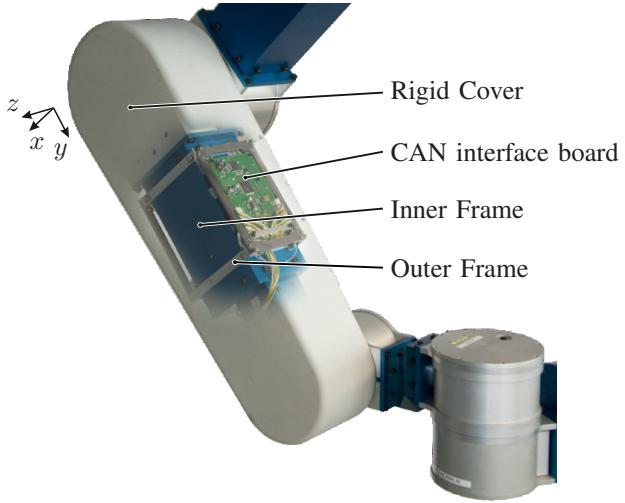


Fig. 2: Tactile Sensing Module at link 4 with its inner workings.

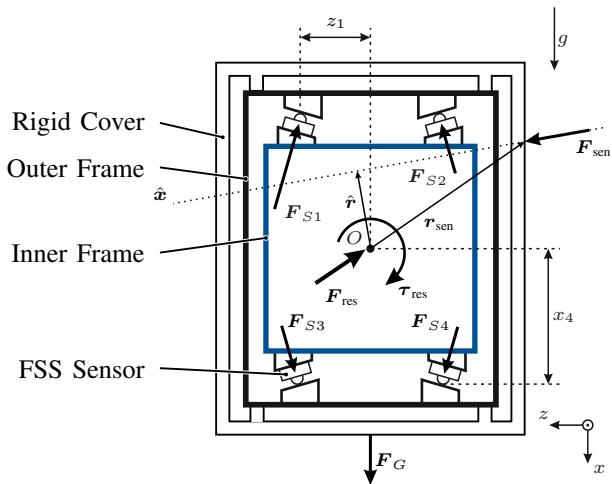


Fig. 3: Schematic sectional view of the tactile sensing module.

board, which is used to forward the sensor signals to the manipulator CAN-Bus (refer to Sec. III-D).

The sensing device itself consists of two nested frames. The outer frame is mounted on the inner one by eight one-axis force sensors, arranged in the corners of the cuboids. The FSS020WNGXS sensors from HONEYWELL have a sensing range from 0 to 20 N. The holders of the piezoresistive elements are tapered in y - and rotated in x -direction so that all movements of the outer frame are restricted. In Fig. 3 a simplified sketch of one half of the sensor is shown. It presents a section in the x,z -plane with four sensors, an external force \mathbf{F}_{sen} , the gravitational force \mathbf{F}_G and the resulting forces on the single sensors $\mathbf{F}_{S,i}$ as well as the calculated force \mathbf{F}_{res} and torque τ_{res} of the whole sensor.

In order to avoid shearing forces that would falsify the measurement, the application of the forces is in line with the sensing direction. With this setup it is possible to determine

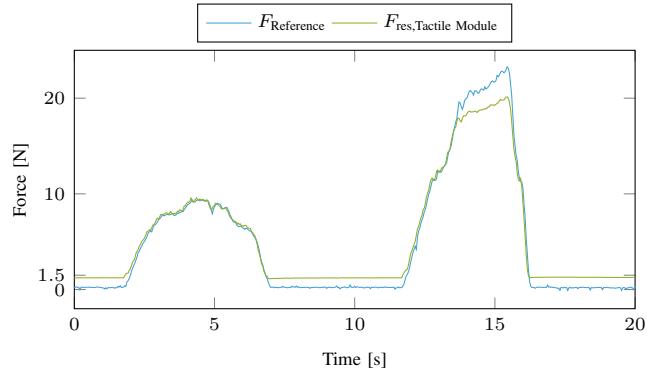


Fig. 4: Validation of the tactile sensor. External forces have been applied and measured using a hand-held force gauge.

one resulting force and torque on the cover. The calculation of the force is described in section III-C.

C. Force Detection

To determine the external force, the equilibria of force and torque are considered.

$$\mathbf{F}_{\text{res}} = \sum_{i=0}^k \mathbf{F}_{S,i} \quad (1)$$

$$\boldsymbol{\tau}_{\text{res}} = \sum_{i=0}^k \mathbf{r}_i \times \mathbf{F}_{S,i} \quad (2)$$

$$\sum \mathbf{F} = \mathbf{F}_{\text{sen}} + \mathbf{F}_G + \mathbf{F}_{\text{res}} = 0 \quad (3)$$

$$\sum \boldsymbol{\tau} = \boldsymbol{\tau}_p + \boldsymbol{\tau}_{\text{res}} = \mathbf{r}_{\text{sen}} \times \mathbf{F}_{\text{sen}} + \boldsymbol{\tau}_{\text{res}} = 0 \quad (4)$$

With Eq. (1) and (3) as well as the knowledge of the gravitational force \mathbf{F}_G , the resulting external force \mathbf{F}_{sen} can be calculated from the sensor forces $\mathbf{F}_{S,i}$. The torque $\boldsymbol{\tau}_p$ about the origin O , which results from \mathbf{F}_{sen} can be estimated with Eq. (2), (4) and the positions of the sensors $\mathbf{r}_i = (x_i, y_i, z_i)^T$ related to O . As the point of application of force one arbitrary point along the line of action of \mathbf{F}_{sen} (dotted line in Fig. 3) can be used. The simplest choice is to identify the shortest distance \hat{r} .

$$\hat{r} = \frac{\mathbf{F}_{\text{sen}} \times \boldsymbol{\tau}_p}{|\mathbf{F}_{\text{sen}} \times \boldsymbol{\tau}_p|} \cdot \frac{|\boldsymbol{\tau}_p|}{|\mathbf{F}_{\text{sen}}|} \quad (5)$$

$$(6)$$

For later purposes the real contact point on the cover can be calculated with raytracing methods and the wrench axis

$$\hat{\mathbf{x}} = \hat{\mathbf{r}} + s \mathbf{F}_{\text{sen}}. \quad (7)$$

In the current implementation of our motion planning framework, we use the center point of link 4 as the contact point \mathbf{r}_p . Fig. 4 shows a comparison between the force applied and measured with a digital force gauge and the force measured with the tactile sensor. Unstressed, there is a small offset in the signal caused by mechanical hysteresis

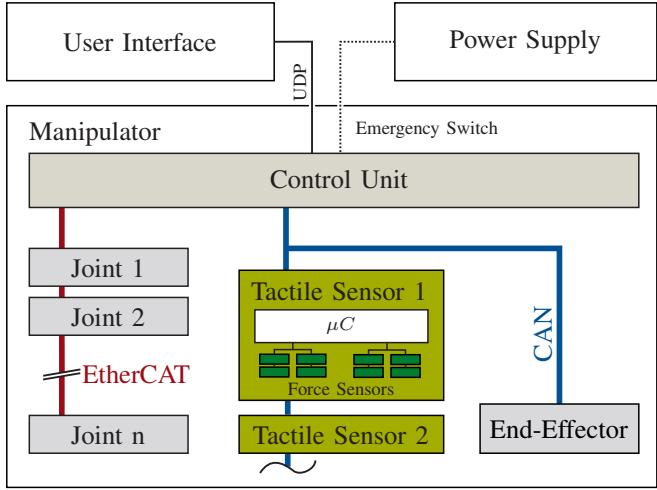


Fig. 5: Communication Architecture for the CROPS Manipulator with Tactile Sensing Module.

effects. While each single sensor has been calibrated and gravitational forces are compensated, this offset leads to an orientation bias which has a decreasing influence at higher loads. We consider this bias as negligible since the behavior of the robot is good-natured regarding the accuracy of force estimation and its direction. Higher forces, which exceed the effective range for the motion planning, lead to a saturation effect in the tactile sensor.

D. Communication Architecture

The manipulator is controlled by a Simulink Real-Time Target control unit with a base sample time of 1 ms. A ROS-based user interface is provided via UDP connection, the power supply can be interrupted by the control unit in case of incident failures. The robot joints are connected to the control unit by an EtherCAT bus system. The end-effector can be connected via CAN bus as well as the tactile sensor. As described in Sec. III-B, the force is detected by 8 sensors per tactile sensor module. The sensors are evaluated by 2 AD-converters connected to an Atmel AT90CAN microcontroller. This controller publishes CAN messages (cycle time < 4 ms). The force signal is low-pass filtered at the control unit (Sec. IV-E.5). The communication scheme is shown in Fig. 5.

IV. MOTION PLANNING WITH TACTILE FEEDBACK

Using tactile feedback for intelligent collision avoidance in cluttered environment requires a robust and efficient online planning module. Based on an initial motion plan, the planner must react to external incidental disturbances by appropriate modifications of its initial plan. In the following section we consider one resultant external force per robot link \mathbf{F}_p that acts frictionless in normal direction at the contact point r_p . Although the algorithms of our planning framework are described for one robot link they can be easily applied to multiple links (Sec. IV-F).

A. Planning Framework

Our manipulator has 9 actuated joints, the joint-space \mathcal{C} is therefore $\mathbf{q} \in \mathbb{R}^9$. We defined the task-space \mathcal{W} without loss of generality as the end-effector position $\mathbf{w} := [x, y, z]^T \in \mathbb{R}^3$. Since $\dim(\mathcal{W}) < \dim(\mathcal{C})$, the manipulator is kinematically redundant. Its inverse kinematics is calculated by the Resolved Motion Rate Control scheme as proposed by [13]:

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{J}_w^\# \dot{\mathbf{w}}_{d,\text{eff}} + \alpha \mathbf{N}_w \left(-\frac{\partial H}{\partial \mathbf{q}} \right)^T \\ \mathbf{J}_w^\# &= \mathbf{W}^{-1} \mathbf{J}^T \left(\mathbf{J} \mathbf{W}^{-1} \mathbf{J}^T \right)^{-1} \\ \mathbf{N}_w &= \mathbf{W}^{-1} \left(\mathbf{I} - \mathbf{J}_w^\# \mathbf{J} \right) \\ \dot{\mathbf{w}}_{d,\text{eff}} &= \dot{\mathbf{w}}_d + \beta (\mathbf{w}_d - \mathbf{w})\end{aligned}\quad (8)$$

with the null-space projection matrix \mathbf{N} , the gain α , the Jacobian $\mathbf{J} := \frac{\partial \mathbf{w}}{\partial \mathbf{q}}$, the weighted pseudoinverse $\mathbf{J}_w^\#$ and the weighting matrix \mathbf{W} . As task-space velocity $\dot{\mathbf{w}}_{d,\text{eff}}$ with the desired task-space position \mathbf{w}_d and the scalar weight β is used instead of $\dot{\mathbf{w}}_d$ to compensate the numerical drift. We use the cost function $H(\mathbf{q})$ for secondary objectives projected in the robot's null-space by \mathbf{N}_w . H includes terms for joint limit and collision avoidance $l_{\text{lim}}, l_{\text{coll}}$ weighted by the respective factors ε

$$H = \varepsilon_{\text{lim}} l_{\text{lim}} + \varepsilon_{\text{coll}} l_{\text{coll}} \quad (9)$$

For details on the cost functions of joint limit and collision avoidance refer to [28].

B. Gradient based approach

For the gradient based approach we extend the cost function H by adding the repellent elastic potential of the environment H_e weighted by ε_e :

$$H_{\text{new}} = H + \varepsilon_e H_e \quad (10)$$

According to Eq. (8) a local minimization of H_e is achieved by descending along the gradient $\frac{\partial H_e}{\partial \mathbf{q}}$. In the following, we choose H_e to be a linear-elastic potential following Hooke's law with a constant stiffness c and only dependent on translational displacement $\Delta \mathbf{r}_p$ at the contact point with the unstressed position $\mathbf{r}_{p,0}$:

$$\Delta \mathbf{r}_p = \mathbf{r}_p - \mathbf{r}_{p,0} \quad (11)$$

$$H_e = \frac{1}{2} c \Delta \mathbf{r}_p^T \Delta \mathbf{r}_p \quad (12)$$

$$\frac{\partial H_e}{\partial \mathbf{q}} = c \underbrace{\Delta \mathbf{r}_p^T}_{-\mathbf{F}_{rp}^T} \cdot \underbrace{\frac{\partial \mathbf{r}_p}{\partial \mathbf{q}}}_{\mathbf{J}_p} = -\mathbf{F}_p^T \mathbf{J}_p \quad (13)$$

Thus, we obtain the spring force \mathbf{F}_p with its effective direction $\frac{\partial \mathbf{r}_p}{\partial \mathbf{q}}$ in joint-space corresponding to the Jacobian \mathbf{J}_p at the current contact point \mathbf{r}_p (\mathbf{r}_p is assumed to be fixed respective to the surface of the deformed object). Hence, the nullspace gradient for Eq. (8) is given by

$$\left(\frac{\partial H_{\text{new}}}{\partial \mathbf{q}} \right)^T = \left(\frac{\partial H}{\partial \mathbf{q}} \right)^T - \varepsilon_e \mathbf{J}_p^T \mathbf{F}_p. \quad (14)$$

C. Advanced Control Framework

For a more sophisticated controller design, we extend the inverse kinematics equations on velocity level by system inputs in joint-space $\mathbf{u}_{j,q}$, task-space $\mathbf{u}_{t,w}$ and nullspace $\mathbf{u}_{n,q}$ to get

$$\dot{\mathbf{q}} = \mathbf{J}_w^\# (\dot{\mathbf{w}}_{d,\text{eff}} - \mathbf{u}_{t,w}) - \mathbf{u}_{j,q} - \mathbf{N}_w \left(\mathbf{u}_{n,q} + \alpha \left(\frac{\partial H}{\partial \mathbf{q}} \right)^T \right). \quad (15)$$

Each of these system inputs can be utilized for integrating tactile feedback into the motion planning equations. The resulting controllers have different characteristics related to the respective nature of the system input space (nullspace, task-space or joint-space). Before applying the commonly known feedback linearization control theory, we first transform equation (15) into the cartesian space at the point of contact \mathbf{r}_p and then in the direction of the contact force. This reduces the order of the problem from $\dim(\mathcal{C})$, over the three-dimensional cartesian space, to the one-dimensional space along the contact force (*contact space*)

$$\hat{\mathbf{F}}_p = \frac{\mathbf{F}_p}{\|\mathbf{F}_p\|}. \quad (16)$$

This approach greatly reduces the complexity of the controller design problem and makes the dynamic behavior of the feedback loop more transparent for parameter tuning. The kinematic relation

$$\dot{\mathbf{r}}_p = \frac{\partial \mathbf{r}_p}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{J}_p \dot{\mathbf{q}} \quad (17)$$

gives the velocity of the contact point \mathbf{r}_p as a function of inputs $\mathbf{u}_{n,q}$ and $\mathbf{u}_{j,q}$ in their joint-space representation. To transform the system inputs into the cartesian system at the point of contact we use the transposed Jacobian as motivated in section IV-B to get

$$\mathbf{u}_{n,q} = \mathbf{J}_p^T \mathbf{u}_n, \quad (18)$$

$$\mathbf{u}_{j,q} = \mathbf{J}_p^T \mathbf{u}_j. \quad (19)$$

Furthermore, a transformation of \mathbf{u}_t from the cartesian space of the point of contact into the task-space representation $\mathbf{u}_{t,w}$ is needed. This is achieved by first transforming into the joint-space with the transposed Jacobian and then transforming into the task-space with the Jacobian of the end effector position \mathbf{J}_w and yields the relation

$$\mathbf{u}_{t,w} = \mathbf{J}_w \mathbf{J}_p^T \mathbf{u}_t. \quad (20)$$

Combining equations (15), (17), (18) and (19) leads to the motion equations for the point of contact in cartesian form:

$$\begin{aligned} \dot{\mathbf{r}}_p = & \mathbf{J}_p \mathbf{J}_w^\# (\dot{\mathbf{w}}_{d,\text{eff}} - \mathbf{J}_w \mathbf{J}_p^T \mathbf{u}_t) - \mathbf{J}_p \mathbf{J}_p^T \mathbf{u}_j \\ & - \mathbf{J}_p \mathbf{N}_w \mathbf{J}_p^T \mathbf{u}_n. \end{aligned} \quad (21)$$

Note that we do not consider the additional nullspace gradients in equation (15) for the controller design, instead these are considered as superposed dynamics, which are not directly influenced by the contact force. Multiplication of

equation (21) with the transposed force $\hat{\mathbf{F}}_p$ and transformation of the system inputs $\mathbf{u}_{j,n,t}$ to scalar inputs $u_{j,n,t}$ with

$$\mathbf{u}_{j,n,t} = \hat{\mathbf{F}}_p u_{j,n,t} \quad (22)$$

yields the scalar motion equation in contact space

$$\dot{x}_p = \dot{x}_{p,d,\text{eff}} - k_t(t)u_t - k_j(t)u_j - k_n(t)u_n. \quad (23)$$

with

$$\dot{x}_{p,d,\text{eff}} := \hat{\mathbf{F}}_p^T \mathbf{J}_p \mathbf{J}_w^\# \dot{\mathbf{w}}_{d,\text{eff}} \quad (24)$$

$$k_n(t) := \hat{\mathbf{F}}_p^T \mathbf{J}_p \mathbf{N}_w \mathbf{J}_p^T \hat{\mathbf{F}}_p \quad (25)$$

$$k_t(t) := \hat{\mathbf{F}}_p^T \mathbf{J}_p \mathbf{J}_w^\# \mathbf{J}_w \mathbf{J}_p^T \hat{\mathbf{F}}_p \quad (26)$$

$$k_j(t) := \hat{\mathbf{F}}_p^T \mathbf{J}_p \mathbf{J}_p^T \hat{\mathbf{F}}_p. \quad (27)$$

If we assume a linear elastic material and a contact without friction, the relation between the scalar motion along the contact force can be described as

$$\Delta F_p = -c \Delta x_p \quad (28)$$

with a constant spring-coefficient c denoting the environment stiffness. c is either considered as a design parameter for control or can be estimated online. The direction of the contact force on the tactile cover and direction of the movement are opposite, requiring a minus in the equation. Differentiation of (28) and insertion into (23) yields the differential equation for the contact force

$$\dot{F}_p = -c \dot{x}_{p,d,\text{eff}} + ck_t(t)u_t + ck_j(t)u_j + ck_n(t)u_n, \quad (29)$$

which is the basis for the following controller design. Because of the discrete nature of contact / non-contact state, this dynamic is only valid during contact with the environment. We utilize feedback linearization to design controllers with the target system dynamics Ξ . The controller can be designed for task-space, joint-space or nullspace, by setting the remaining system inputs to zero and equalizing the system dynamics (29) with $\Xi(t)$. Although we use Resolved Motion Rate Control for the motion equations in this section, the presented framework works analogously with an inverse kinematics at acceleration level, commonly known as Resolved Acceleration Control, [14].

D. Utilization of the Control Framework

The described control framework allows several variants for tactile feedback controllers with different properties. One possibility are controllers, which do exclusively operate in nullspace, task-space or joint-space. While nullspace controllers are limited to the space of feasible directions in nullspace, with resulting constraints on the controller performance, task- and joint-space controllers lead to deviations from the desired task-space trajectory². In addition to standalone controllers operating only in one space, a combination of nullspace- and task- / joint-space controllers is possible.

²However, the drift-compensation, which is usually used to compensate for integral drift, limits the deviations from the task-space trajectory whenever nullspace movements are possible.

This combines the benefits of nullspace movements without deviations from the task-space trajectory, and the possibility of relaxing the task-space position, e.g. when the contact forces exceeds a pre-defined limit.

In this paper, we concentrate on the design and experimental validation of a standalone nullspace controller with linear second order dynamics. The dynamics $\Xi(t)$ are given by

$$\ddot{\Xi} = -\frac{2d}{T}\dot{\Xi} - \frac{1}{T^2}\Xi \quad (30)$$

with time constant T and damping factor d . The dynamics describe a decay of the force to zero depending on the coefficients. A lower time constant leads to a faster elimination of the contact force. Integrating (30), setting $F_p = \Xi$ and equalizing with (29) yields the controller equation

$$u_n = \frac{1}{k_n(t)}\dot{x}_{p,d,\text{eff}} - \frac{2d}{Tck_n(t)}F_p - \frac{1}{T^2ck_n(t)} \int_0^t F_p d\tau. \quad (31)$$

The vectorial representation for application of the control law in the nullspace is obtained with equations (18), (22) and (24) – (27). We recognize that this controller contains the desired velocity of the point of contact p in form of a feedforward approach. This tries to eliminate all movements of p in direction of the obstacle, utilizing nullspace movements. Furthermore, the controller consists of a proportional and an integral part.

E. Extensions to the Control Laws

Several extensions to the feedback linearization control laws are described in the following.

1) *Continuous Feedforward Compensation:* The simple feedforward control is augmented by storing the force direction of the last known contact and applying the feedforward compensation also during a non-contact scenario. This leads to an intelligent behavior, as the manipulator tries to avoid movements in a direction, in which it already encountered an obstacle in the past. The stored force direction is reset when a new contact occurs. In addition, we ensure that only movements against the obstacle, i.e. $\dot{x}_{p,d,\text{eff}} < 0$, are compensated as it is possible that a demanded velocity directs away from an obstacle.

2) *Integral Part Transition:* During the change from a contact to non-contact scenario, an integral part of the controller is likely to have a non-zero value. This claims a transition strategy to transform the discrete state change into a continuous change of the controller output. We define F_{int} as the integrator value in the controller equation. At the transition point $t = t_t$ it is given by

$$F_{\text{int}}(t_t) = \int_0^{t_t} F_p d\tau. \quad (32)$$

As the contact force becomes zero after the transition, $F_{\text{int}}(t \geq t_t)$ does not change, resulting in a constant controller output. The transition handling strategy includes the ongoing calculation of the controller output according to the

corresponding equation (e.g. equation (31) in nullspace) and a first-order decrease of the integrator part:

$$\dot{F}_{\text{int}} = -\frac{1}{T_t}F_{\text{int}}. \quad (33)$$

This ensures continuous controller outputs during the contact-state change and has the advantageous effect of moving the contact point away from the obstacle after the impact, compared to staying very close to the obstacle. The time constant T_t is either selected depending on the velocity of the contact point during transition:

$$T_t = \max \left\{ \frac{d_t}{\hat{F}_p^T \mathbf{J}_p \dot{q}}; T_{t,\text{max}} \right\}, \quad (34)$$

to reach a certain distance d_t to the obstacle, or it is set to a fixed value causing time-dependent decrease.

3) *Nullspace Velocity Limitation:* The integral parts can easily lead to high controller outputs for situations in which the contact force does not decrease fast enough due to ongoing movements against the obstacle. Furthermore, the scalar factor $k_n(t)$ may get zero causing very high controller outputs. This happens when the space of feasible nullspace velocities is orthogonal to the direction of the transformed contact force. As the high resulting velocities are still executed in the nullspace, even though they have no effect in the direction of the force, we must limit the maximum velocity on joint level. To keep the direction of the controller output, the whole vector $\mathbf{u}_{n,q}$ is scaled such as to keep the highest joint-velocity in the vector below a certain limit $\dot{q}_{n,\text{max}}$. This also ensures that the feedback controllers do not override additional gradients in nullspace, which are, for example, used for joint limit- or self collision avoidance. This extension only applies to nullspace controllers.

4) *Force Threshold:* In real contact scenarios, forces below a certain threshold F_{th} are not of any interest and should not change the planned manipulator motion. Thus, we use the following relation between the internal contact force \mathbf{F}_p , which is used for calculating the controller feedback, and the sensor force \mathbf{F}_{sen} :

$$\mathbf{F}_p = \begin{cases} \mathbf{0}, & \text{if } \|\mathbf{F}_{\text{sen}}\| \leq F_{th} \\ \frac{\mathbf{F}_{\text{sen}}}{\|\mathbf{F}_{\text{sen}}\|} (\|\mathbf{F}_{\text{sen}}\| - F_{th}), & \text{if } \|\mathbf{F}_{\text{sen}}\| > F_{th} \end{cases} \quad (35)$$

5) *Filtering:* The force components from the tactile sensor module are filtered with a 10Hz digital lowpass filter (Butterworth, order 8) to reduce noise in the signal caused by the sensor data fusion and the sensors itself. In addition, the force direction is filtered separately to suppress changes in direction caused by dynamic forces on the tactile cover during manipulator motion. This direction-filter basically limits the change rate of the normalized force vector.

F. Multiple Contacts

The presented approaches can be easily extended to scenarios with multiple contacts. More than one contact force per link are aggregated to one resulting force while multiple contacts

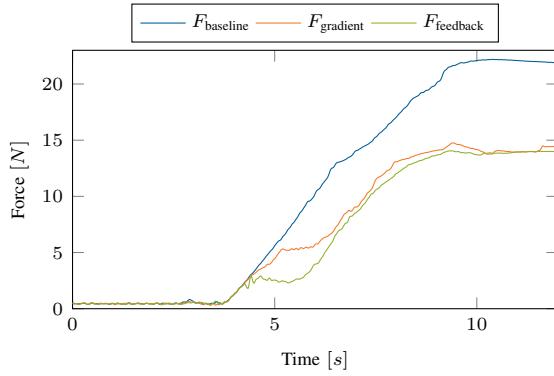


Fig. 6: Measured resulting force on the tactile sensor module with different control strategies.

on different links could be taken into account *separately*. For the gradient based approach (Sec. IV-B), the respective environmental potential of other contact points has to be added to Eq. (10). Applying the method presented in Sec. IV-C, one controller instance per robot link (assuming one resulting external force per link) has to be provided.

V. RESULTS

We performed several experiments based on the presented hardware setup. Since the manipulator is equipped with one tactile sensing module to date, the manipulator hits a pretensioned vertical latex band (stiffness at the contact point in horizontal direction $c_p \approx 60 \text{ N/m}$) at link 4. We tested the baseline controller (8) against the gradient based (14) and the nullspace feedback controller with feed-forward term and 2nd order dynamics (31). The manipulator had to follow a straight line \mathcal{W} -space path and hits the elastic band. The weight was chosen to $\varepsilon_e = 1$ and the force threshold to $F_{\text{th}} = 2 \text{ N}$ while nullspace velocities were limited to $|q_{\text{prism,max}}| \leq 2 \text{ m/s}$ for prismatic and $|q_{\text{rot,max}}| \leq 2 \text{ rad/s}$ for rotational joints. The transition handling parameters were $T_{t,\text{max}} = 1 \text{ s}$, $d_t = 0.05 \text{ m}$ with distance based transition. The controller was parametrized with $c = 60 \text{ N/m}$, $T = 0.1837 \text{ s}$ and $d = 0.1633$. While the baseline controller has a smooth velocity profile (Fig. 7), it tensions the band increasing the resulting force to $> 20 \text{ N}$. Considering tactile information the external force can be reduced to $< 15 \text{ N}$ by suitable nullspace motion (Fig. 6). Compared to the gradient based approach, the feedback controller manages to decrease the external force faster (Fig. 6) resulting in higher joint velocities (Fig. 7). The experiment using the baseline compared to the feedback controller is depicted in Fig. 8. Both the gradient-based approach and the feedback controller are not able to completely eliminate the contact force. This results from the limitation of the nullspace, i.e. the manipulator is not able to reach the desired position without contact to the obstacle. During the experiments, which are shown in our video attachment, it turned out that especially the filter design of the force feedback has significant influence on stability and performance of the planning framework (cf. IV-E.5).

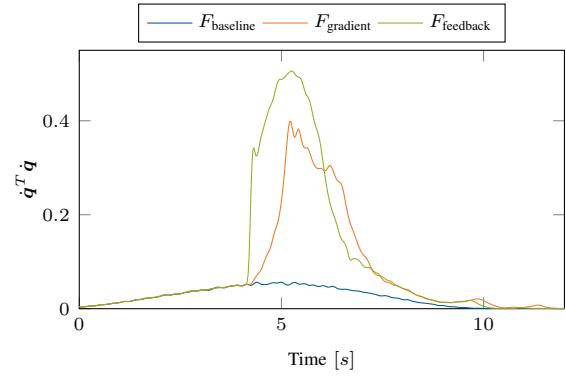


Fig. 7: Measured squared joint velocities $\dot{\mathbf{q}}^T \dot{\mathbf{q}}$ of the real-world experiment.

VI. CONCLUSION

In our contribution we presented a simple but effective rigid tactile sensing module which allows measuring the resulting force and its wrench axis. Furthermore we showed a general formulation for reactive planning with tactile feedback for redundant manipulators. By projecting the external force to its point of application we derived a one-dimensional system equation (per resulting force) and developed a suitable controller using feedback linearization with second order dynamics. Experiments with a 9-DOF manipulator and the tactile sensor module in combination with different control strategies in nullspace complete this paper. We could show the fast reaction to external forces and their significant reduction by evasive nullspace motion of the manipulator. Although our proposed framework provides controller inputs for nullspace, \mathcal{W} - and \mathcal{C} -space trajectory modifications, we limited this paper to nullspace controllers. Since it is likely that forces cannot be reduced to an acceptable level by modifying trajectories only in nullspace, we will examine combinations of nullspace and \mathcal{W} - or \mathcal{C} -space controllers in our upcoming work.

ACKNOWLEDGMENT

We want to thank the anonymous reviewers whose comments helped to strengthen our contribution considerably. Special thanks go to Georg Mayr and Sebastian Roder whose contribution was essential for the realization of the tactile sensing module.

REFERENCES

- [1] M. R. Cutkosky, R. D. Howe, and W. R. Provancher, “Force and Tactile Sensors,” in *Springer Handbook of Robotics: Chapter 19*, 2008, pp. 455–476.
- [2] C. Schuetz, J. Pfaff, J. Baur, T. Buschmann, and H. Ulbrich, “A Modular Robot System for Agricultural Applications,” in *Proceedings International Conference of Agricultural Engineering*, Zurich, 2014.
- [3] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.
- [4] I. Sucan, M. Moll, and L. Kavraki, “The Open Motion Planning Library,” *IEEE Robotics & Automation Magazine*, vol. 19, pp. 72–82, 2012.

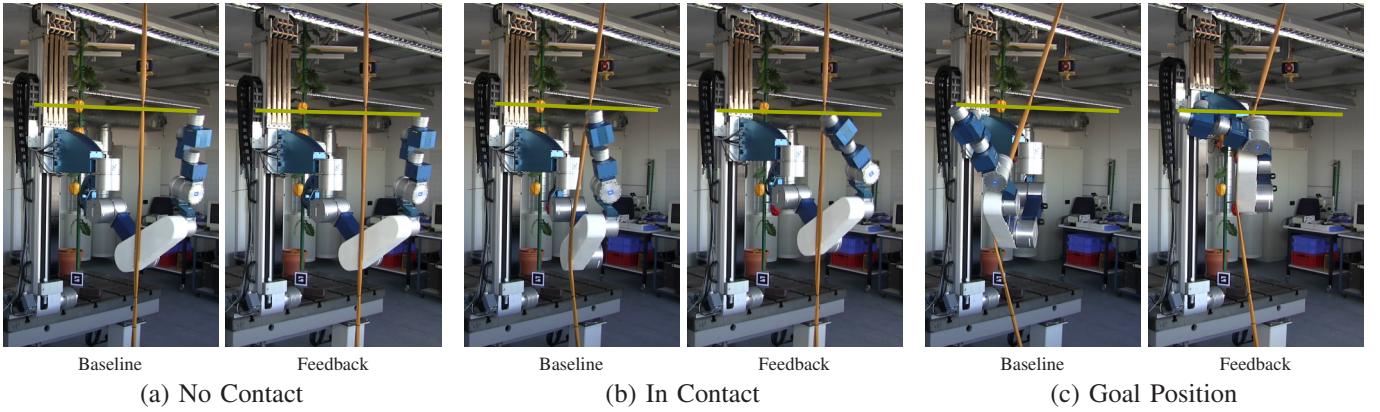


Fig. 8: The 9-DOF manipulator hits an elastic vertical band with the tactile sensing module, the given end-effector path is drawn as a green line. In comparison, the behavior of the baseline controller (without tactile feedback) and the feedback controller are shown at different time instances. It can be seen, that the manipulator tries to avoid the strip with tactile feedback while respecting the \mathcal{W} -space path.

- [5] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, “Finding Locally Optimal, Collision-Free Trajectories with Sequential Convex Optimization,” in *Proceedings of Robotics: Science and Systems*, 2013.
- [6] M. Dogar and S. Srinivasa, “A Framework for Push-Grasping in Clutter,” in *Robotics: Science and Systems VII*, 2011.
- [7] M. Stilman, J. U. Schamburek, J. Kuffner, and T. Asfour, “Manipulation planning among movable obstacles,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2007, pp. 3327–3332.
- [8] B. Frank, C. Stachniss, N. Abdo, and W. Burgard, “Efficient motion planning for manipulation robots in environments with deformable objects,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Sept. 2011, pp. 2180–2185.
- [9] S. Patil, J. van den Berg, and R. Alterovitz, “Motion planning under uncertainty in highly deformable environments,” in *Robotics: Science and Systems*, Los Angeles, CA, USA, 2011.
- [10] J. Villani, Luigi. Schutter De, “Force Control,” in *Springer Handbook of Robotics: Chapter 7*, 2008, pp. 161–185.
- [11] J. Park and O. Khatib, “Robot multiple contact control,” in *Robotica*, vol. 26, 2008.
- [12] A. Jain, M. D. Killpack, A. Edsinger, and C. C. Kemp, “Reaching in clutter with whole-arm tactile sensing,” *The International Journal of Robotics Research*, vol. 32, no. 4, pp. 458–482, Mar. 2013.
- [13] A. Liègeois, “Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 7, no. 12, pp. 868–871, 1977.
- [14] J. Luh, M. Walker, and R. Paul, “Resolved-acceleration control of mechanical manipulators,” *IEEE Transactions on Automatic Control*, vol. 25, pp. 468–474, 1980.
- [15] J. L. Nevins and D. E. Whitney, “The Force Vector Assembler Concept,” in *On Theory and Practice of Robots and Manipulators SE - 19*, ser. International Centre for Mechanical Sciences. Springer Berlin Heidelberg, 1972, pp. 273–288.
- [16] J. K. Salisbury, “Interpretation of contact geometries from force measurements,” *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 240–247, 1984.
- [17] M. Hakozaki and H. Shinoda, “Digital tactile sensing elements communicating through conductive skin layers,” *IEEE International Conference on Robotics and Automation*, vol. 4, no. May, pp. 2–6, 2002.
- [18] Y. Ohmura, Y. Kuniyoshi, and A. Nagakubo, “Conformable and scalable tactile sensor skin for a curved surfaces,” in *IEEE International Conference on Robotics and Automation*, vol. 2006, May 2006, pp. 1348–1353.
- [19] M. Shimojo, T. Araki, S. Teshigawara, A. Ming, and M. Ishikawa, “A net-structure tactile sensor covering free-form surface and ensuring high-speed response,” in *IEEE International Conference on Intelligent Robots and Systems*. Ieee, Oct. 2007, pp. 670–675.
- [20] H. Iwata and S. Sugano, “Whole-body covering tactile interface for human robot coordination,” in *IEEE International Conference on Robotics and Automation*, 2002.
- [21] A. Bicchi, J. K. Salisbury, and D. L. Brock, “Contact sensing from force measurements,” *The International Journal of Robotics Research*, vol. 12, pp. 249–262, 1993.
- [22] B. S. Eberman and J. K. Salisbury, “Determination of manipulator contact information from joint torque measurements,” in *Lecture Notes in Control and Information Sciences*. Springer Science + Business Media, 1990, pp. 463–473.
- [23] F. Castelli, “An integrated tactile-thermal robot sensor with capacitive tactile array,” *IEEE Transactions on Industry Applications*, vol. 38, no. 1, pp. 85–90, 2002.
- [24] P. Mittendorfer and G. Cheng, “Humanoid multimodal tactile-sensing modules,” *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 401–410, 2011.
- [25] M. Lee and H. Nicholls, “Review Article Tactile sensing for mechatronics state of the art survey,” *Mechatronics*, vol. 9, no. 1, pp. 1–31, Feb. 1999.
- [26] J. S. Son, M. R. Cutkosky, and R. D. Howe, “Comparison of contact sensor localization abilities during manipulation,” in *International Conference on Intelligent Robots and Systems*. Institute of Electrical & Electronics Engineers (IEEE), 1995.
- [27] A. De Luca and R. Mattone, “Sensorless robot collision detection and hybrid force/motion control,” in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2005, 2005, pp. 999–1004.
- [28] C. Schuetz, T. Buschmann, J. Baur, J. Pfaff, and H. Ulbrich, “Predictive Online Inverse Kinematics for Redundant Manipulators,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Hong Kong, 2014, pp. 5056–5061.