2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Model Predictive Position and Force Trajectory Tracking Control for Robot-Environment Interaction

Tobias Gold, Andreas Völz, and Knut Graichen

Abstract— **The development of modern sensitive lightweight robots allows the use of robot arms in numerous new scenarios. Especially in applications where interaction between the robot and an object is desired, e.g. in assembly, conventional purely position-controlled robots fail. Former research has focused, among others, on control methods that center on robot-environment interaction. However, these methods often consider only separate scenarios, as for example a pure force control scenario. The present paper aims to address this drawback and proposes a control framework for robot-environment interaction that allows a wide range of possible interaction types. At the same time, the approach can be used for setpoint generation of position-controlled robot arms, where no interaction takes place. Thus, switching between different controller types for specific interaction kinds is not necessary. This versatility is achieved by a model predictive control-based framework which allows trajectory following control of joint or end-effector position as well as of forces for compliant or rigid robot-environment interactions. For this purpose, the robot motion is predicted by an approximated dynamic model and the force behavior by an interaction model. The characteristics of the approach are discussed on the basis of two scenarios on a lightweight robot.**

I. INTRODUCTION

Industrial robots have been automating processes in industry for many years. However, the robots mainly act in a position-controlled manner, so that a physical robot-environment interaction is not the primary focus. This is now being changed by the use of modern sensitive lightweight robots. Already the construction of these robots is designed for a possible interaction, e.g. with humans. The lightweight construction of the robots reduces, for example, the energy transferred during collisions. In addition, special control techniques have been developed to provide compliance in case of interaction [1]. Furthermore, sensitive manipulation is possible by means of force control methods [2]. These methods have already been known since the 1980s, see e.g. early works on hybrid position/force control [3] and impedance control [4].

A more recent field for dealing with this requirement is the use of model predictive control (MPC). This method is supposed to explicitly comply with force constraints or to enforce a defined behavior in case of contact loss. In [5], an approach for model predictive admittance control is presented, which allows an explicit consideration of force constraints. In this context, the use of MPC for interaction control is discussed in general. The work of [6] combines compliance control with model predictive path-following control by considering an additional admittance dynamic. A similar approach is followed by the work of [7], which combines model predictive path-following control with force control. The author of [8] transforms occurring contact forces into an evasive velocity for a model predictive pose trajectory control. Similar to admittance control, this leads to a new trajectory, which differs from the original desired trajectory, but minimizes contact forces. In [9], the interaction force is introduced as a separate state of the prediction model to realize a parallel position/force control. The previous work of the authors [10] catches up the idea of formulating the force as a state variable and presents a universal framework based on MPC for interactions with elastic environments.

For the prediction of the interaction forces in the internal optimization problem, a spring model which considers the environmental stiffness is often assumed. To this end, the stiffness of the environment must be estimated either before an interaction in an offline procedure or during the interaction with an online method. Three different methods for stiffness estimation are discussed, for example, in [11] in the context of classical impedance and force control.

This paper presents an approach for model predictive control of robot-environment interaction, which is able to handle interaction with both elastic and rigid environmental behaviors. If only the environmental stiffness is considered in the prediction model, a limitation of compliant robot-environment interaction is required. This is because in a rigid interaction, the stiffness of the environment becomes infinite, which leads to numerical instabilities. The presented approach addresses this by explicitly considering the stiffness of the underlying position controller.

The MPC acts as a setpoint generator for the robot internal position control. The dynamics of this underlying control is therefore also considered in the joint space in order to apply the approach for position-controlled robots. Due to the position control, the robot behavior can be approximated as a linear spring-damper system. Furthermore, an interaction dynamic is taken into account in the MPC framework. Since the behavior of the environment is described by a spring model, the interaction between robot and environment can be regarded mathematically as a parallel connection of position controller and environmental stiffness [12]. This has the consequence that the lower stiffness dominates the interaction model. Thus, an ideal rigid contact results in an interaction stiffness equal to the controller stiffness of the internal position controller. The advantage of this approach is a universal usability for lightweight robots. At the same

The authors are with the Chair of Automatic Control, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91058 Erlangen, Germany
{tobias.gold, andreas.voelz, knut.graichen}@fau.de

time, the MPC can be used to achieve a desired impedance behavior, even though this may be a compliant or rigid behavior. Furthermore, the approach can be extended to a position and force trajectory tracking control. This leads to a powerful framework for the control of robot-environment interactions, since in addition kinematic and force constraints can be explicitly considered by the MPC. The properties of the proposed methods are discussed on the basis of two control scenarios with a 7-DOF lightweight robot.

II. MODELING

In this section the model of robot-environment interaction is discussed. First, the controlled robot system is derived and then the interaction is addressed.

A. Robot System

The dynamic model of a robot arm with n joints

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_{\mathrm{J}} - \boldsymbol{\tau}_{\mathrm{ext}}, \tag{1}$$

describes the relationship between the resulting motion of the generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$ and their time derivatives due to the applied generalized forces $\boldsymbol{\tau}_{\mathrm{J}} \in \mathbb{R}^n$. The external torques $\boldsymbol{\tau}_{\mathrm{ext}} \in \mathbb{R}^n$ are the result of an external force on the robot structure. In the case of an external force at the end-effector, the external torques $\boldsymbol{\tau}_{\mathrm{ext}} = \boldsymbol{J}(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\mathcal{F}}_{\mathrm{ext}}$ are obtained from the (6×1) wrench vector $\boldsymbol{\mathcal{F}}_{\mathrm{ext}} = [\boldsymbol{F}_{\mathrm{ext}}^{\mathrm{T}}, \boldsymbol{m}_{\mathrm{ext}}^{\mathrm{T}}]^{\mathrm{T}}$, composed by the forces and torques at the end-effector, using the transposed end-effector Jacobian $\boldsymbol{J}(\boldsymbol{q})$.

An underlying position control realizes the stabilization of a desired position $\boldsymbol{q}_d$ for which the corresponding joint torques $\boldsymbol{\tau}_{\mathrm{J}} = \boldsymbol{u}$ are calculated. For this purpose, the nonlinear system (1) is first transferred to a linear decoupled system using the approach of exact linearization

$$\boldsymbol{u} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{y} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \tag{2}$$

via state feedback [13]. The nonlinear term $\boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q})$ and the mass matrix $\boldsymbol{M}(\boldsymbol{q})$ are feedforwarded to compensate for them exactly. The resulting new input $\boldsymbol{y}$ can be chosen as a linear and decoupled PD control

$$\boldsymbol{y} = \ddot{\boldsymbol{q}} = \boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}\dot{\boldsymbol{q}} \tag{3}$$

to shape the dynamics of the controlled system according to

$$\ddot{\boldsymbol{q}} = \boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}\dot{\boldsymbol{q}} - \boldsymbol{J}(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\mathcal{F}}_{\mathrm{ext}}\,. \tag{4}$$

The $(n \times n)$ gain matrices are selected as diagonal matrices, e.g. $\boldsymbol{K}_P = \mathrm{diag}([k_{P,1}, \ldots, k_{P,n}])$, with the gain factors of the individual joint variables. Instead of (2), the PD control with gravitational compensation

$$\boldsymbol{u} = \boldsymbol{\tau}_{\mathrm{J}} = \boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) \tag{5}$$

is a popular control method in robotics due to its simplicity, but also due to the fact that it provides asymptotic stability [13]. The closed loop of the PD controlled robot

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}\dot{\boldsymbol{q}} - \boldsymbol{J}(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\mathcal{F}}_{\mathrm{ext}} \tag{6}$$

is obtained by substituting (5) in (1). During interaction, typically only low velocities are realized. Therefore, dynamic effects such as inertia, Coriolis and centripetal effects can be neglected ($\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} \approx \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} \approx \boldsymbol{0}$). Such an approximation may also be applied to the inverse dynamics approach (4), where $\ddot{\boldsymbol{q}} \approx \boldsymbol{0}$ is assumed. This simplifies the dynamics of the controlled robot in both cases to a first-order system

$$\dot{\boldsymbol{q}} = \boldsymbol{D}^{-1}\boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}^{-1}\boldsymbol{J}(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\mathcal{F}}_{\mathrm{ext}}\,. \tag{7}$$

The model describes the controlled robot system during free motion ($\boldsymbol{\mathcal{F}}_{\mathrm{ext}} = \boldsymbol{0}$) as well as under the influence of external forces ($\boldsymbol{\mathcal{F}}_{\mathrm{ext}} \neq \boldsymbol{0}$) during an interaction. On this basis, the contact situation is modelled subsequently.

B. Robot-Environment Interaction

In principle, two basic types of contact can be distinguished in an interaction situation between robot and environment. On the one hand, the interaction with a rigid body causes a constraint on the permitted robot motion. The second case involves dynamic interaction with a body. Fundamental mechanical effects such as inertia, dissipation, or elasticity thereby describe the interaction dynamics. The presented approach should be able to handle both cases of interaction. For this purpose, the interaction model in the case of a purely linear elastic interaction is derived first. It is assumed that no dissipative and inertial effects occur in the direction of the interaction motion. In addition, a purely rigid robot system is supposed by localizing all intrinsic passive robot compliance in the environment. Thus the interaction can be described according to a mechanical spring [12], [14]

$${}^{o}\boldsymbol{\mathcal{F}}_{\mathrm{env}} = {}^{o}\boldsymbol{K}_{\mathrm{e}}\mathrm{d}\boldsymbol{p}_{o,e}\,, \tag{8}$$

where the penetration $\mathrm{d}\boldsymbol{p}_{o,e} = \boldsymbol{p} - \boldsymbol{p}_o$ results from the displacement of the end-effector frame Σ_e with respect to the task frame Σ_o located on the object surface. Note that the force ${}^{o}\boldsymbol{\mathcal{F}}_{\mathrm{env}}$ is referred to the task frame Σ_o. In addition, the pose vector

$$\boldsymbol{p}(\boldsymbol{q}) = [\boldsymbol{t}(\boldsymbol{q})^{\mathrm{T}}, \boldsymbol{\phi}(\boldsymbol{q})^{\mathrm{T}}]^{\mathrm{T}} = {}^{b}\boldsymbol{T}_e(\boldsymbol{q}) \tag{9}$$

composes the end-effector position $\boldsymbol{t}$ and the orientation, denoted by the Euler angles $\boldsymbol{\phi}$, which is calculated by the coordinate transformation of the joint position to the Cartesian base frame Σ_b. For the sake of clarity, no superscript indices are added for quantities defined in the base coordinate system and the location and rotation of the frames in the base frame is determined by the respective pose. The elastic wrench $\boldsymbol{\mathcal{F}}_{\mathrm{env}}$ depends, as pictured in Figure 1, on the penetration of the robot end-effector into the object. Thereby, the resistance to penetration is defined by the (6×6) symmetric positive (semi)-definite stiffness matrix ${}^{o}\boldsymbol{K}_{\mathrm{e}}$. Under the neglection of the coupling stiffness between translational and rotational stiffness, the environmental stiffness ${}^{o}\boldsymbol{K}_{\mathrm{e}} = \mathrm{diag}({}^{o}\boldsymbol{K}_{\mathrm{trans}}, {}^{o}\boldsymbol{K}_{\mathrm{tors}})$ is composed by the (3×3) translational and rotational stiffness matrices ${}^{o}\boldsymbol{K}_{\mathrm{trans}}$ and ${}^{o}\boldsymbol{K}_{\mathrm{rot}}$ [14]. Due to the location of the task frame Σ_o on the object surface, the stiffness matrix ${}^{o}\boldsymbol{K}_{\mathrm{e}}$ can be selected according to the restricted directions. In the example of Figure 2, only the normal of the surface is motion constrained. Therefore the stiffness matrix has only entries for the respective direction.

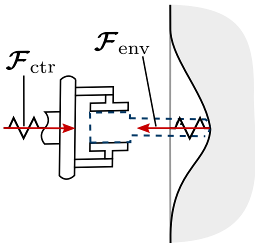

Fig. 1. PD-controlled robot in interaction with an environment.

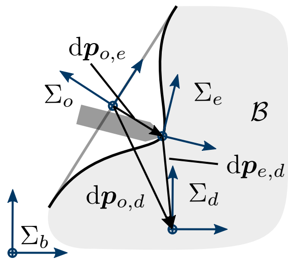

Fig. 2. Frames in robot-environment interaction.

In contrast, the PD-controlled robot reacts according to the approximated spring-damper system (7). The stiffness of this system is defined by the joint space controller stiffness $\boldsymbol{K}_{\text{P}}$. The force of the controller acting in the static case

$$\mathcal{F}_{\text{ctr}} = \boldsymbol{K}_{\text{P, cart}} \mathrm{d}\boldsymbol{p}_{e,d} \tag{10}$$

results from the weighted error of desired and current end-effector position. Note that the (6×6) Cartesian controller stiffness matrix results from the locally valid transformation

$$\boldsymbol{K}_{\text{P, cart}} = (\boldsymbol{J}(\boldsymbol{q})^{\text{T}})^{\#} \boldsymbol{K}_{\text{P}} (\boldsymbol{J}(\boldsymbol{q}))^{\#} \tag{11}$$

of the joint space stiffness matrix to the Cartesian space with the matrix pseudo inverse $(\cdot)^{\#}$ [15] [16].

If the equality $\mathrm{d}\boldsymbol{p}_{o,e} = \mathrm{d}\boldsymbol{p}_{o,d} - \mathrm{d}\boldsymbol{p}_{e,d}$ and $\mathcal{F}_{\text{ctr}} = \mathcal{F}_{\text{env}}$ [17] according to Figure 1 and 2 is considered, the resulting external wrench on the end-effector is the equivalent spring

$$\begin{aligned} {}^{o}\mathcal{F}_{\text{ext}} &= ({}^{o}\boldsymbol{K}_{\text{e}} + \boldsymbol{K}_{\text{P, cart}})^{-1}\, {}^{o}\boldsymbol{K}_{\text{e}}\, \boldsymbol{K}_{\text{P, cart}} \mathrm{d}\boldsymbol{p}_{o,d} \\ &= {}^{o}\bar{\boldsymbol{K}} \mathrm{d}\boldsymbol{p}_{o,d} \,. \end{aligned} \tag{12}$$

The overall stiffness ${}^{o}\bar{\boldsymbol{K}}$ thus results mathematically from the parallel connection of the individual stiffnesses [12]. In order to express the force in the base frame, the wrench ${}^{o}\mathcal{F}_{\text{ext}}$ in the object frame has to be rotated using the rotation matrix ${}^{b}\boldsymbol{R}_o = \boldsymbol{R}_b{}^{\text{T}} \boldsymbol{R}_o$. This results in the elastic wrench in the base frame

$$\mathcal{F}_{\text{ext}} = \begin{bmatrix} {}^{b}\boldsymbol{R}_o{}^{\text{T}} & \boldsymbol{0} \\ \boldsymbol{0} & {}^{b}\boldsymbol{R}_o{}^{\text{T}} \end{bmatrix} {}^{o}\bar{\boldsymbol{K}} \mathrm{d}\boldsymbol{p}_{o,d} = \bar{\boldsymbol{K}} \mathrm{d}\boldsymbol{p}_{o,d} \,. \tag{13}$$

The aim of the presented approach is to handle the force as a state variable in the MPC. To this end, the dynamic behavior of the force has to be considered for the assumption of a constant task frame on the object surface. Thus, the time derivative of the force is determined according to

$$\dot{\mathcal{F}}_{\text{ext}} = \bar{\boldsymbol{K}} \boldsymbol{J}(\boldsymbol{q}) \dot{\boldsymbol{q}}_{\text{d}} \,. \tag{14}$$

The authors are aware that (14) only holds locally. This is due to the position dependency of $\boldsymbol{K}_{\text{P, cart}}$ and $\bar{\boldsymbol{K}}$ from (11). However, this can be neglected in the MPC sense if the prediction horizon is sufficiently small and the velocity in the direction of interaction is assumed to be low. Basically, the model (14) shows how the forces increase or decrease in the individual Cartesian directions during a particular desired motion $\dot{\boldsymbol{p}}_{\text{d}} = \boldsymbol{J}(\boldsymbol{q}) \dot{\boldsymbol{q}}_{\text{d}}$. Note, that the Cartesian pose motion

$$\dot{\boldsymbol{p}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = [\boldsymbol{v}(\boldsymbol{q}, \dot{\boldsymbol{q}})^{\text{T}}, \boldsymbol{\omega}(\boldsymbol{q}, \dot{\boldsymbol{q}})^{\text{T}}]^{\text{T}} = \boldsymbol{J}(\boldsymbol{q}) \dot{\boldsymbol{q}} \,, \tag{15}$$

described by the translational and the angular velocities of the end-effector $\boldsymbol{v}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and $\boldsymbol{\omega}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, is determined by the product of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$ and the joint velocity $\dot{\boldsymbol{q}}$.

However, the equations (8), (12), and (14) only hold in the case of a contact. Therefore, it is necessary to distinguish between free motion and contact. One possibility is the evaluation of the end-effector forces by a threshold value

$${}^{o}K_{\text{e},i}(t) \begin{cases} = 0 & \text{if } {}^{o}F_{\text{ext},i} < \epsilon_i \text{ or } {}^{o}m_{\text{ext},i} < \epsilon_i \\ > 0 & \text{else} \,. \end{cases} \tag{16}$$

If this threshold ϵ_i is exceeded in a certain Cartesian direction i, the end-effector is in contact in this direction. Another possibility is to create a model with geometry and mechanical features of the environment using, for example, camera data or information from previous contact situations. This information is generally used to model the object surface and the stiffness ${}^{o}\boldsymbol{K}_{\text{e}}$. In this case, ${}^{o}\boldsymbol{K}_{\text{e}}$ is selected in dependence on the environment model

$${}^{o}K_{\text{e},i}(t) \begin{cases} > 0 & \text{if } {}^{o}p_i \in \mathcal{B} \\ = 0 & \text{else} \,, \end{cases} \tag{17}$$

whereby it is analyzed whether the end-effector has penetrated the object $\mathcal{B}$. From this point on, the respective object stiffness is assigned to the prediction model. Due to the model predictive character of the presented approach, this can be used to react prior to contact. Thus, the motion can already be slowed down when predicting the contact in order to reduce occurring force peaks.

However, there remains the question of how to handle a rigid contact with the approach. It can be seen from (12) that due to the parallel connection of the stiffnesses, the total stiffness $\bar{\boldsymbol{K}}$ is always smaller than the smallest stiffness. For the case of ${}^{o}K_{\text{e},i} \to \infty$, the overall stiffness ${}^{o}\bar{K}_i = {}^{o}K_{\text{P,cart},i}$ thus results in the Cartesian controller stiffness. Thus, the interaction stiffness is of a convenient order of magnitude. The simultaneous consideration of controller and environmental stiffness yields a numerically more stable formulation of the interaction model.

III. Model Predictive Control Formulation

Model predictive control is based on the iterative (suboptimal) solution of a dynamic optimization problem

$$\min_{\boldsymbol{u}} \quad J(\boldsymbol{u}; \boldsymbol{x}_k) = V(\boldsymbol{x}(T)) + \int_0^T l(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau))\, \mathrm{d}\tau \tag{18a}$$

$$\text{s.t.} \quad \dot{\boldsymbol{x}}(\tau) = \boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau)) \,, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_k \tag{18b}$$

$$\boldsymbol{x}(\tau) \in \mathcal{X} \,,\; \boldsymbol{u}(\tau) \in \mathcal{U} \tag{18c}$$

over a moving horizon $\tau \in [0, T]$ with length T. The characteristic of MPC is that the optimization problem is initialized with the current measured or estimated value $\boldsymbol{x}_k$ of the state variables in each sampling step and the initial solution of the control variable is taken from the optimization of the last time step. Subsequently, the first part of the optimal solution is applied to the controlled system up to the next time step.

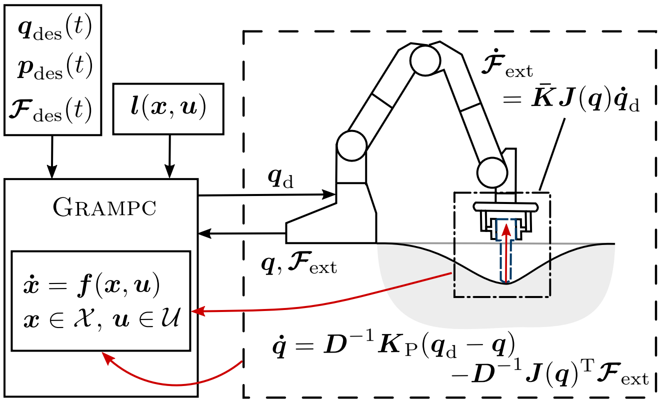

Fig. 3. System layout of the model predictive interaction control for generalized robot-environment interaction.

The entire system of the model predictive interaction control is schematically shown in Figure 3. The MPC, which is solved with the GRAMPC toolbox [18], considers the dynamics of the controlled robot and the interaction, marked with dashed boxes. The degree of freedom in the choice of cost functions and reference values is problem-specific and varies according to the application. In the following, the individual parts of the problem (18) are introduced.

A. System Dynamics

The robot-environment interaction can be described by the set of states $\boldsymbol{x} = \begin{bmatrix}\boldsymbol{q} & \boldsymbol{q}_{\mathrm{d}} & \boldsymbol{\mathcal{F}}_{\mathrm{ext}} & \boldsymbol{p}\end{bmatrix}^{\mathrm{T}}$. The dynamics of the robots joint position in free motion as well as during interaction is directly derived from (7). The desired joint position $\boldsymbol{q}_{\mathrm{d}}$ actuates the robot system dynamics (7). As pictured in Figure 3, this is forwarded to the robot system as control variable. However, to avoid steps in the control trajectory, the first derivative of the desired position $\boldsymbol{u} = \dot{\boldsymbol{q}}_d$ is internally handled as MPC control variable. Since the external force is considered as a state variable, its dynamics must also be considered in the optimization problem. This is done via equation (14) and the conditions (16) or (17). For reasons of computational efficiency, the end-effector pose is also chosen as a state, whereby its dynamics can be described using (15). This leads to the overall nonlinear dynamics (18b) during a contact situation according to

$$\begin{bmatrix}\dot{\boldsymbol{q}} \\ \dot{\boldsymbol{q}}_{\mathrm{d}} \\ \dot{\boldsymbol{\mathcal{F}}}_{\mathrm{ext}} \\ \dot{\boldsymbol{p}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{D}^{-1}\boldsymbol{K}_{\mathrm{P}}(\boldsymbol{q}_d - \boldsymbol{q}) - \boldsymbol{D}^{-1}\boldsymbol{J}(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\mathcal{F}}_{\mathrm{ext}} \\ \boldsymbol{u} \\ \bar{\boldsymbol{K}}\boldsymbol{J}(\boldsymbol{q})\boldsymbol{u} \\ \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}\end{bmatrix}. \tag{19}$$

The initial state value $\boldsymbol{x}(0) = \boldsymbol{x}_k$ is thereby initialized by the measurements of the respective state in step k.

B. Cost Function Design

The control target of MPC is defined in the cost function (18a). With the presented method, an offline calculated trajectory $\boldsymbol{x}_{\mathrm{des}}(t)$ of the state variable has to be stabilized optimally. By penalizing the deviation from a desired trajectory, the optimization problem tries to minimize these deviations. Therefore, the cost function (18a) may be based for example solely on the integral cost term

$$l(\boldsymbol{x},\boldsymbol{u}) = l_{\mathrm{pos}}(\boldsymbol{q}) + l_{\mathrm{pose}}(\boldsymbol{p}) + l_{\mathrm{force}}(\boldsymbol{\mathcal{F}}_{\mathrm{ext}}) + l_{\mathrm{ctr}}(\boldsymbol{u})\,. \tag{20}$$

The problem of trajectory tracking can be handled by penalizing the deviation of a quantity $\tilde{\boldsymbol{x}}(\tau) = \boldsymbol{x}(\tau) - \boldsymbol{x}_{\mathrm{des}}(\tau)\,, \tau \in [0,T]$ to the desired trajectory $\boldsymbol{x}_{\mathrm{des}}(t)$ by using the weighted norm $\|\tilde{\boldsymbol{x}}\|^2_{\boldsymbol{A}} = \frac{1}{2}\tilde{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{A}\tilde{\boldsymbol{x}}$ with the positive (semi)-definite weighting matrix $\boldsymbol{A}$. Note that the choice of the individual penalty terms depends on the respective application and can also be set to zero. Some example applications are presented in Section IV and in the previous work of the authors [10].

C. System Constraints

In addition, the MPC scheme allows to consider constraints (18c) systematically. In the case of interaction control, system-related constraints, such as limited joint position

$$\boldsymbol{q} \in [\boldsymbol{q}^-, \boldsymbol{q}^+]\,, \tag{21a}$$

as well as task-related constraints, such as a limited workspace or limited interaction forces

$$\boldsymbol{p} \in [\boldsymbol{p}^-, \boldsymbol{p}^+] \tag{22a}$$

$$\boldsymbol{\mathcal{F}}_{\mathrm{ext}} \in [\boldsymbol{\mathcal{F}}^-_{\mathrm{ext}}, \boldsymbol{\mathcal{F}}^+_{\mathrm{ext}}] \tag{22b}$$

can be taken into account. The potential constraints are not limited to box constraints of the state variables as shown.

IV. EXPERIMENTAL VALIDATION

The presented approach is validated in two scenarios with a 7-DOF Franka Emika Panda robot. The external wrench at the end-effector is estimated using the build-in joint torques measurements with a generalized momentum observer [19]. Alternatively, the force can also be measured using a force/torque sensor on the end-effector. However, the approach is not limited to measured states. Estimated quantities can also be used, e.g. following the approach of [20] or [21].

In the first scenario, a pure force trajectory is applied to materials with different stiffness up to a rigid contact. The second scenario pairs the force control for a rigid contact with a motion control in the complementary directions. For this purpose, the robot arm is drawing a curve on a plane with chalk. The orientation of the object surface is set horizontally so that the orientation of the object frame and base frame is identical. Thus, no rotation of the wrench (12) is necessary.

The sampling time of the MPC is set to $10\,\mathrm{ms}$ and the underlying PD control with gravitational compensation according to (5) is running with a sampling time of $1\,\mathrm{ms}$.

A. Comparing Different Stiffness

First, the approach is demonstrated for the interaction with different object stiffness. For this purpose, an interaction setup as schematically shown in Figure 3 is assumed. The robot shall apply a defined force of $-2\,\mathrm{N}$ in z-direction to different objects. On the one hand, a foam with a stiffness of ${}^{o}k_{\mathrm{e},z} = 500\,\mathrm{N\,m^{-1}}$ and, on the other hand, the rigid table surface is chosen for the interaction. The object stiffness

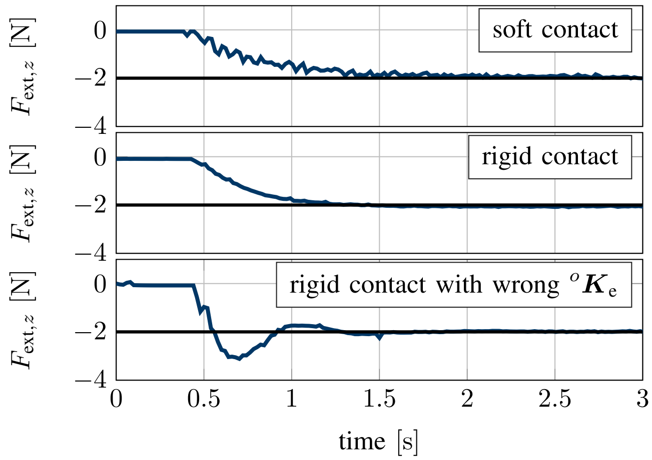

Fig. 4. Comparison of force trajectories in different interactions.

was calculated offline before interaction. However, an online stiffness estimation as presented in [10] is also possible.

Figure 4 shows the force trajectory for three different cases. The first plot shows the trajectory for the elastic interaction. Due to the low object stiffness, the overall stiffness $\bar{\boldsymbol{K}}$ is significantly reduced compared to the transformed controller stiffness $\boldsymbol{K}_{\mathrm{P,\,cart}}$. This allows the robot to react fast enough to the contact that occurs at approximately $0.5\,\mathrm{s}$. The desired reference value is thus reached after an additional time of $0.6\,\mathrm{s}$ without overshoot.

The second plot of Figure 4 shows the rigid interaction with the table surface. Due to the object stiffness tending towards infinity, the total stiffness $\bar{\boldsymbol{K}}$ is equal to the transformed controller stiffness $\boldsymbol{K}_{\mathrm{P,\,cart}}$. As with the elastic interaction, contact occurs shortly before $0.5\,\mathrm{s}$ and the setpoint is reached after another $0.6\,\mathrm{s}$. Theoretically, the object stiffness has to be defined as infinitely high. In practice, especially when using lightweight robots, the stiffness of the joints with approximately $10^4\,\mathrm{N\,rad^{-1}}$ will become noticeable. By assuming a concentration of robot intrinsic stiffness in the environment, the joint stiffness is dominating the object stiffness. This means that the joint stiffness can be used to specify the object stiffness in the rigid case.

The last plot of Figure 4 shows the behavior of the control in case of a wrong assumption of the object stiffness $^{o}\boldsymbol{K}_{\mathrm{e}}$. Again, the robot is interacting with the rigid table surface, but assumes a compliant object stiffness of $^{o}k_{\mathrm{e},z} = 500\,\mathrm{N\,m^{-1}}$. Thus, a significant overshoot can be seen in the transient zone, since the controller expects a weaker increase of the forces. In the Cartesian case, this would mean that the controller initially places the target position of the underlying PD controller further into the object.

However, the previous tests only considered a constant setpoint. As the presented methodology also allows time-variant setpoints, a force trajectory following control is examined in the following for a soft contact interaction. The target trajectory is chosen to

$$F_{\mathrm{ext,\,z,\,des}}(t) = 5\mathrm{N}\sin(\pi t + 0.1) - 7\mathrm{N}\,. \tag{23}$$

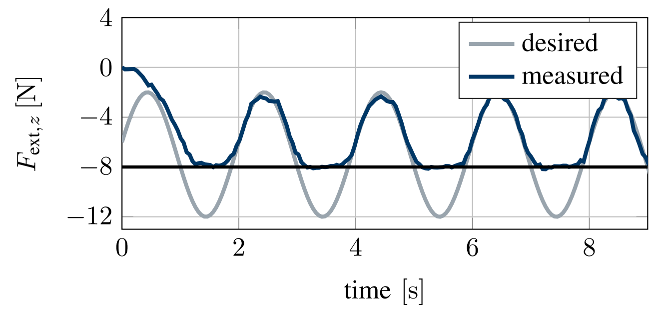

Fig. 5. Oscillating force trajectory applied on a compliant surface.

In addition, the absolute force in z-direction is constrained to a maximum of $8\,\mathrm{N}$. The Figure 5 shows a continuous interaction with an oscillating force trajectory from the beginning. The constraint at $-8\,\mathrm{N}$ is continuously satisfied during the subsequent trajectory course. In case of a setpoint within the permissible range, the trajectory is followed sufficiently well.

B. Writing with Chalk

The second application aims to demonstrate the applicability of the methodology as hybrid force/motion control for rigid contact. For this purpose, the robot has to draw a path on a blackboard with a piece of chalk. The degrees of freedom of the end-effector are divided for this purpose into position and force controlled subspaces. In the following example, the x-direction is supposed to be stabilized to a constant value of $5\,\mathrm{N}$, while the y- and z-direction shall follow a trajectory according to a Lissajous figure

$$t_{x,\mathrm{des}}(t) = 0.1\sin(0.15\cdot 2\pi t) \tag{24a}$$
$$t_{y,\mathrm{des}}(t) = 0.1\sin(0.075\cdot 2\pi t) + 0.9\,, \tag{24b}$$

whereby the orientation is kept to a constant value. The constraints (21a) are considered in the optimization problem.

The path followed by the end-effector is shown in Figure 6 for two Lissajous repetitions, whereas the corresponding trajectories are given in Figure 7. The motion starts with an initial offset from the desired trajectory that is corrected immediately. When considering the MPC parameters in Table I, it is noticeable that especially the deviation to the desired trajectory is highly weighted in order to achieve a good trajectory tracking. In contrast, the deviation of the force to the reference value is less penalized, which results in a worse force tracking accuracy. It can be seen that the maximum values in the periodic force trajectory correlate

TABLE I

MPC PARAMETERS FOR THE HYBRID FORCE/MOTION CONTROL.

prediction horizon	T	0.3 s
sampling points	N_{hor}	30
max. gradient iterations	i_{grad}	4
max. multiplier iterations	i_{mult}	1
position weights	$\boldsymbol{Q}_p$	$\mathrm{diag}([0, 10^4, 10^4, 10, 10, 10])$
interaction force weights	$\boldsymbol{Q}_{\mathcal{F}}$	$\mathrm{diag}([0.5, 0, 0, 0, 0, 0])$
control weights	$\boldsymbol{R}$	$\mathrm{diag}([0.1, \ldots, 0.1])$

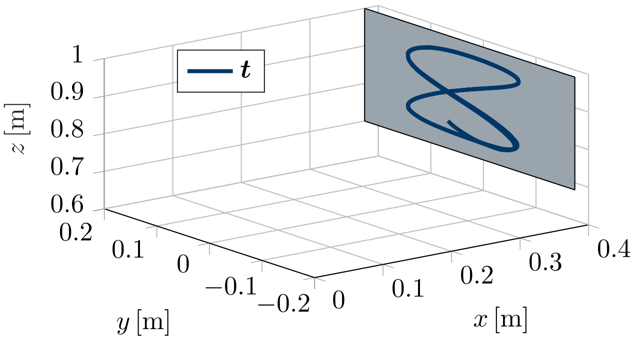

Fig. 6. Path of the end-effector while writing with chalk.

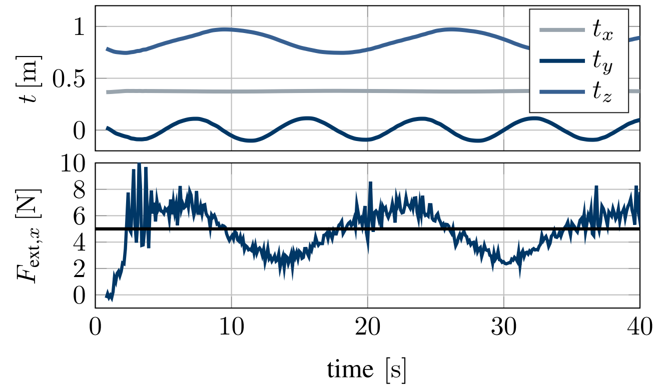

Fig. 7. End-effector and force trajectory while writing with chalk.

with the writing direction. If the chalk is rather pushed, larger forces result. On the other hand, the forces decrease as soon as the chalk is pulled. However, if Figure 7 is considered with the background of a moving interaction under the influence of friction, the force trajectory still represents a good compromise. The experiments have revealed that with a higher weighting of the force deviation, the force tracking becomes significantly better. However, the quality of the end-effector trajectory deteriorates as a result. In addition, a reduction of the end-effector trajectory velocity also improves force tracking noticeably.

The controllers are computed for both scenarios on a Ubuntu 18.04 OS with Intel(R) Core(TM) i5-8250U CPU. With the MPC parameters from Table I, an average computing time of $3.0\,\mathrm{ms}$ and a worst case computing time of $4\,\mathrm{ms}$ is achieved. Thus, the computing time is well within the $10\,\mathrm{ms}$ MPC sampling time.

V. CONCLUSIONS

This contribution presents an MPC-based approach to control the robot-environment interaction. The handling of the problem using MPC results in a flexible and comprehensive framework for various application scenarios where interaction with an environment is desired including interactions with a rigid environment. The results of the experimental validation show that even small forces can be realized with a good control quality during an end-effector motion. This allows to use the approach for sequential manipulation tasks such as the classical peg-in-hole problem to avoid switching between different controller types. Further work will focus on the topic of environment modeling based on camera and tactile data. This improves the approach by predicting the contact before interaction. Further downloadable material for this article is available at http://ieeexplore.ieee.org. The material includes a video that illustrates the writing with chalk scenario.

REFERENCES

[1] A. Albu-Schaffer and G. Hirzinger, "Cartesian impedance control techniques for torque controlled light-weight robots," in *Proc. of ICRA*, 2002, pp. 657–663.

[2] G. Zeng and A. Hemami, "An overview of robot force control," *Robotica*, vol. 15, no. 5, pp. 473–482, 1997.

[3] M. Raibert and J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement and Control*, vol. 102, no. 3, pp. 126–133, 1981.

[4] N. Hogan, "Impedance control: An approach to manipulation," *Journal of Dynamic Systems, Measurement and Control*, vol. 107, no. 1, pp. 8–16, 1985.

[5] A. Wahrburg and K. Listmann, "MPC-based admittance control for robotic manipulators," in *Proc. of CDC*, 2016, pp. 7548–7554.

[6] K. J. Kazim, J. Bethge, J. Matschke, and R. Findeisen, "Combined predictive path following and admittance control," in *Proc. of ACC*, 2018, pp. 3153–3158.

[7] J. Matschke, J. Bethge, P. Zometa, , and R. Findeisen, "Force feedback and pathfollowing using predictive control: Concept and application to a lightweight robot," in *Proc. of IFAC World Congress*, 2017, pp. 9827–9832.

[8] A. Zube, J. Hofmann, and C. Frese, "Model predictive contact control for human-robot interaction," in *Proc. of ISR*, 2016, pp. 279–285.

[9] J. de la Casa Cardenas, A. Garca, S. Martnez, J. Garca, and J. Ortega, "Model predictive position/force control of an anthropomorphic robotic arm," in *Proc. of ICIT*, 2015, pp. 326–331.

[10] T. Gold, A. Völz, and K. Graichen, "Model predictive interaction control for industrial robots," in *IFAC World Congress*, 2020, accepted.

[11] D. Erickson, M. Weber, and I. Sharf, "Contact stiffness and damping estimation for robotic systems," *The International Journal of Robotics Research*, vol. 22, no. 1, pp. 41–57, 2003.

[12] S. Jung, T. C. Hsia, and R. G. Bonitz, "Force tracking impedance control for robot manipulators with an unknown environment: Theory, simulation, and experiment," *International Journal of Robotics Research*, vol. 20, no. 9, pp. 765–774, 2001.

[13] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. New York: Wiley, 2005.

[14] B. Siciliano and O. Khatib (Eds.), *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer, 2016.

[15] J. K. Salisbury, "Active stiffness control of a manipulator in Cartesian coordinates," in *Proc. of CDC*, 1980, pp. 95–100.

[16] S.-F. Chen and I. Kao, "Simulation of conservative congruence transformation. Conservative properties in the joint and cartesian spaces," in *Proc. of ICRA*, 2000, pp. 1283–1288.

[17] B. Siciliano, L. Sciavicco, L. Villani, *et al.*, *Robotics - Modelling, Planning and Control*. London: Springer, 2009.

[18] T. Englert, A. Völz, F. Mesmer, S. Rhein, and K. Graichen, "A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC)," *Optimization and Engineering*, vol. 20, no. 3, pp. 769–809, 2019.

[19] A. De Luca and R. Mattone, "Actuator fault detection and isolation using generalized momenta," in *Proc. of ICRA*, 2003, pp. 634–639.

[20] T. Gold, A. Völz, and K. Graichen, "External torque estimation for an industrial robot arm using joint torsion and motor current measurements," in *Proc. of Joint Conference on MECHATRONICS and NOLCOS*, 2019, pp. 879–884.

[21] A. Wahrburg, E. Morara, B. Cesari, G.and Matthias, and H. Ding, "Cartesian contact force estimation for robotic manipulators using kalman filters and the generalized momentum," in *Proc. of CASE*, 2015, pp. 1230–1235.