A Transferable Force Controller based on Prescribed Performance for Contact Establishment in Robotic Assembly Tasks*

Lorenz Halt¹, Fengjunjie Pan¹, Philipp Tenbrock¹, Andreas Pott² and Thomas Seel³

Abstract—In industrial robotics, controller parameters for force control must be adjusted to the specific robot that performs a task and they must be re-adjusted when the same task is to be performed by another robot. We address this challenge by proposing a transferable force controller for contact establishment between robot and surface. The controller is implemented based on task frame formalism. The proposed controller is based on prescribed performance control (PPC) and does not rely on a dynamic model of the environment. Due to the inherent robustness of PPC, it can be used to ensure similar performance for the same task across different robots and environments. The proposed controller is validated experimentally in a simple contact establishment task performed by three different robots (Universal Robots UR5, Franka Emika Panda, Denso Wave VS087) and three different board materials providing different stiffness (steel, aluminum, PVC). The PPC is found to yield an up to two orders of magnitude smaller variance of closed-loop settling time across all robots and materials than a conventional impedance controller.

I. INTRODUCTION

The number of industrial robots in manufacturing is continuously increasing. In 2017, approximately 381 thousand industrial robots were sold. This number has increased by 30% compared to 2016 [1]. According to the estimation of the International Federation of Robotics, sales of industrial robots will increase by 10% in 2018 and by 14% in 2019.

The integration of different robots in various tasks remains a challenge. Recently the research focus is shifting towards rapid set-up of robotic applications [2]. In our previous research [3][4][5], a skill-based framework peresc has been developed, which focuses on the planning of robotic assembly tasks and allows the user to reuse different applications and to transfer them to different robots.

Contact establishment is an important process in force-based assembly tasks, which involves impact and contact control. In the current contribution, we focus on the contact control. More specifically, we consider the task that begins with the impact between end-effector and surface and ends with the contact force reaching its set-point, as further explained in Fig. 1 and Fig. 2. Obviously the dynamics of this contact control phase depends on the employed robot and the environmental stiffness. It is therefore desirable to design controllers that yield similar performance across different environmental stiffnesses (usability) and across different robots (transferability).

The reuse and transfer of task-specific controllers are supported by the architecture of the framework. In peresc, skills are complete and executable (sub)tasks. By combining such tasks, complex applications such as riveting, screwing and clipping can be achieved. During the execution of a skill an associated controller is instantiated. Peresc uses transformations between the robot-specific joint space and the task-specific operational space, called task space [6][7]. The controller is designed and acts within the task space only. This eliminates the need for modeling the (robot-specific) kinematics and thus enhances transferability between robots. Because peresc has no direct access to joint torques or motor currents, the force control loop is closed around motion/velocity control of the robot, i.e. implicit force control is used [8].

Many control theories exist in literature for contact establishment with manipulators. In [9][10], proportional control, stiffness control, damping control and impedance control are introduced for robotic contact establishment. These methods can be designed to achieve a very high quality for specific

*This work has received funding from the European Union’s Horizon 2020 Research and Innovation Programme under grant agreement n 688642 (RAMPup)

¹L. Halt, F. Pan and P. Tenbrock are with the Fraunhofer Institute for Manufacturing Engineering and Automation IPA, 70569 Stuttgart, Germany {ith, ith-fp, pgt}@ipa.fraunhofer.de

²Andreas Pott is with the Institute for Control Engineering of Machine Tools and Manufacturing Units, 70174 Stuttgart, Germany andreas.pott@isw.uni-stuttgart.de

³Thomas Seel is with the Department of Electrical Engineering and Computer Science, Technical University Berlin, 10623 Berlin, Germany seel@control.tu-berlin.de

References


tasks and environment. However, they are not flexible enough to deal with a much broader range of the aforementioned task-dependent uncertainties. To address this problem, adaptive control based on online estimation of environment stiffness is proposed in [11][12]. This approach requires an explicit model of the contact dynamics and large computation efforts. In [13], prescribed performance control (PPC) is proposed to overcome the need for modeling and to handle different environment stiffnesses.

In the present paper we adjust PPC to a task-function-approach-based environment. Further, PPC is extended to cope with practical challenges, such as slow systems or delays. A reusable and transferable controller is developed, implemented and tested within the pitasc framework. We analyze its stability analytically and present an experimental verification of the achieved reusability and transferability.

The paper is structured as follows: Section II introduces the basic principle of PPC. In Section III, we design the controller and analyze its stability. The implementation of force control in pitasc is discussed in Section IV. The experimental setup and the results of conducted tests are presented in Section V. Finally, a conclusion is given in Section VI.

II. PPC METHODOLOGY

A controller with prescribed performance assures that the tracking error is limited to a predefined region and converges to the equilibrium at least with a predefined rate. In this section, the basic idea of prescribed performance control is introduced.

Let $f$ represent the measured force and $f_d$ be the constant desired contact force. The tracking error at time $t$ is

$$e_f(t) = f(t) - f_d.$$  \hspace{1cm} (1)

Considering that, in assembly tasks, error overshoot may cause the loss of contact or the damage of components, the following performance bounds are proposed:

$$-D < e_f(t) < \rho(t), \text{ if } e_f(0) \geq 0,$$

$$-\rho(t) < e_f(t) < D, \text{ if } e_f(0) < 0,$$  \hspace{1cm} (2)

where, as in Fig. 3, $D$ is a positive constant representing error tolerance in steady state. This tolerance is at least as large as the noise amplitude of the given force sensor. The performance function $\rho(t)$ is a continuously differentiable, bounded, strictly positive and decreasing function of time [13]. A commonly used performance function is:

$$\rho(t) = (\rho_0 - D)e^{-lt} + D,$$  \hspace{1cm} (3)

where $\rho_0$ is a positive constant satisfying $\rho_0 > |e_f(0)|$, $l$ is a positive constant representing the decreasing rate of $\rho(t)$. The larger $l$, the faster $\rho$ decreases. In the following description we will omit time argument $(t)$ for the sake of brevity.

The following error transformation is introduced:

$$\varepsilon := T(e_f, \rho),$$  \hspace{1cm} (4)

in which $T$ is a smooth function. Consider for example

$$\varepsilon(e_f, \rho) = \begin{cases} \ln\left(\frac{D + e_f}{\rho - e_f}\right), & \text{if } e_f(0) \geq 0, \\ \ln\left(\frac{\rho + e_f}{D - e_f}\right), & \text{if } e_f(0) < 0. \end{cases}$$  \hspace{1cm} (5)

In order to relate (5) to the common definition in [14], the following coordinate transformation is applied:

$$\tilde{\rho} = \rho + D, \tilde{e} = e_f + D \text{ if } e_f(0) \geq 0,$$

$$\tilde{\rho} = \rho + D, \tilde{e} = e_f - D \text{ if } e_f(0) < 0.$$

\hspace{1cm} (6)
Further, we define the normalized error:

\[ \xi := \frac{\varepsilon}{\hat{\rho}}. \]

(7)

With (7) and (5) the error transformation can be written as a function depending only on the normalized error \( \xi \):

\[ \varepsilon(\xi) = \begin{cases} 
\ln \left( \frac{\xi}{1 - \xi} \right) & \text{if } e_f(0) \geq 0, \\
\ln \left( \frac{1 + \xi}{-\xi} \right) & \text{if } e_f(0) < 0.
\end{cases} \]

(8)

\( \varepsilon(\xi) \) is a smooth, strictly increasing function shown in Fig. 4. Let \( \Omega_\xi \) be the values that \( \xi \) takes. Note that, if \( e_f \) stays within performance bounds, \( \Omega_\xi = \{ \xi : \xi \in (0, 1) \} \) when \( e_f(0) \geq 0 \), \( \Omega_\xi = \{ \xi : \xi \in (-1, 0) \} \) when \( e_f(0) < 0 \). The transformation \( T \) is a bijective mapping from \( \Omega_\xi \) to \( \mathbb{R} \) that assigns arbitrarily large values of \( \varepsilon \) to errors that are arbitrarily close to the performance bound (normalized error arbitrarily close to 1, cf. Fig. 4 and [13]). Furthermore, note that the derivative

\[ \psi := \frac{\partial \varepsilon}{\partial \xi} = \frac{d_\xi}{d\xi} \frac{1}{\hat{\rho}}, \]

(9)

is bounded, i.e., there exists a constant \( \lambda_\psi \) such that \( \psi \geq \lambda_\psi > 0 \) if \( e_f \) stays inside the performance bounds.

III. CONTROLLER DESIGN & STABILITY

In this section we will design a PPC-based contact force controller using the aforementioned definitions. As before, time arguments are omitted for the sake of brevity.

Consider the allowance set \( \Omega_f = \{ f \in \mathbb{R} : \tilde{f} \geq f \geq \tilde{f} \} \) of the force \( f \), where \( \tilde{f} \) is the maximum admissible force and \( \tilde{f} \) is the minimum force that guarantees contact. Then the performance bounds should fulfill the following conditions:

\[
\begin{aligned}
\text{if } e_f(0) \geq 0, & \quad \sup_{t \geq 0} \{ \rho + f_d \} < \tilde{f}, -D + f_d \geq \tilde{f}; \\
\text{if } e_f(0) < 0, & \quad \inf_{t \geq 0} \{ -\rho + f_d \} \geq \tilde{f}.
\end{aligned}
\]

(10)

(11)

This implies \( f \in \Omega_f \) for all \( e_f \) inside the bounds (2). Recall that \( f \) is to be controlled by manipulating \( v_r \) (cf. Fig. 2). The following control law is proposed:

\[ v_r = \begin{cases} 
-a\varepsilon - b(b\tilde{\rho} + \hat{\rho}) & \text{if } e_f(0) \geq 0, \\
-a\varepsilon - b(b\tilde{\rho} + \hat{\rho}) & \text{if } e_f(0) < 0,
\end{cases} \]

(12)

with constant control gains \( a > 0 \) and \( b \geq 0 \), transformed error \( \varepsilon \), and with \( \tilde{f} \) and \( \hat{\rho} \) being the derivative of force and performance function with respect to time.

In the following, we analyze the stability of the proposed control loop. The controlled force is assumed to be well described by the elastic force. We employ the force model used in [14]:

\[ f = F(\chi), \forall \chi \geq 0, F : [0, +\infty) \to [0, +\infty), \]

(13)

where \( F \) is a positive, strictly increasing and continuously differentiable function of the deformation \( \chi \), for example, the Hertz model \( F(\chi) = k\chi^2 \) with \( k > 0 \).

Differentiating \( f \) with respect to time yields

\[ \dot{f} = F'(\chi)v_t, \]

(14)

with \( F'(\chi) := \frac{\partial f(\chi)}{\partial \chi} \) and \( v_t \) being the current tip velocity (cf. Fig. 2). If \( f \in \Omega_f \) for all \( t > 0 \), then there exists bounds \( \hat{f}', \tilde{f}' \) such that \( 0 < \hat{f}' < F'(\chi) < \tilde{f}' \), \( \forall t > 0 \).

Let \( e_v := v_t - v_r \) describe the error of the velocity controller. Then substituting (12) in (14) yields

\[ \dot{f} = \begin{cases} 
K(-a\varepsilon + \frac{b}{2}\hat{\rho} + e_v) & \text{if } e_f(0) \geq 0, \\
K(-a\varepsilon - \frac{b}{2}\hat{\rho} + e_v) & \text{if } e_f(0) < 0,
\end{cases} \]

(15)

with \( K = \frac{F'(\chi)}{1+b\hat{\rho}} \) being positive and bounded from below and above by \( \frac{F'}{1+b\hat{\rho}} \). If \( f \in \Omega_f \),

To analyze the stability of the force control loop, we introduce the Lyapunov candidate function:

\[ V = \frac{1}{2}\varepsilon^2. \]

(16)

Using (7), (9), (6), (1), (15), in this order, we find that

\[ \dot{V} = \varepsilon \frac{d\varepsilon}{d\xi} \varepsilon = \varepsilon \frac{d\varepsilon}{d\xi} \frac{d\hat{\rho}}{d\hat{\rho}} = \varepsilon(-A\varepsilon^2 + B\varepsilon), \]

(17)

where \( A = a\hat{K} \), \( B = \hat{K}(\frac{\hat{b}}{2} + e_v) - \hat{f}_d - \hat{\rho} \).

Note that

i) \( a, b \) are positive constants by definition,
ii) \( K \) is bounded from below and above if \( f \in \Omega_f \),
iii) \( e_v \) is negligible if the velocity control is almost perfect,
iv) \( \hat{f}_d \) is zero (an upper bound is known even in the case of a slowly varying reference force \( f_d \)).

v) \( \hat{\rho} \) is limited by construction,
vii) \( \xi \leq 1 \) if \( e_f \) is within the performance bounds (2). Hence, there exists a lower bound \( \lambda_A \) with \( A \geq \lambda_A > 0 \) and an upper bound \( \lambda_B \) with \( |B| < \lambda_B \). Recall that \( \psi \geq \lambda_\psi > 0 \). Therefore, \( V < 0 \) if \( \varepsilon > \frac{\lambda_B}{A} \); \( V < 0 \) if \( \varepsilon < \frac{\lambda_A}{\lambda_B} \), which assures the following property: If the error \( e_f \) is within the performance bounds (2) initially, then the transformed error \( \varepsilon \) decreases until it reaches the set

\[ \Omega_\varepsilon := \{ \varepsilon \mid |\varepsilon| < \min(\{\varepsilon(0), \frac{\lambda_B}{\lambda_A}\}) \} \]

(18)

and remains inside that set. In particular, \( \varepsilon \) cannot become arbitrarily large.

This allows us to show, by contradiction, that \( e_f \) will not exceed the performance bounds (2) at any time if it starts within the bounds. If \( e_f \) exceeds the performance bounds at any moment in time, then, by continuity, there exist a time \( t^* > 0 \) at which \( e_f(t^*) \) reaches one of the performance bounds for the first time. Without loss of generality, consider the case \( e_f(0) \geq 0 \). Then there are two possible cases:

a) if \( e_f(t^*) = \rho(t^*) \), then \( e_f(t) < \rho(t) \forall t \in [0, t^*] \).

Furthermore, it implies that \( f(t) \in \Omega_f \) due to the condition (10) fulfilled by the performance bounds. From (5) we get \( \lim_{t \to t^*} \varepsilon(t) = +\infty \), which contradicts (18).
b) if $e_f(t^*) = -D$, then $e_f(t) \geq -D \forall t \in [0, t^*]$. Furthermore, it implies that $f(t) \in \Omega_f$ due to the condition (10) fulfilled by the performance bounds. From (5) we get $\lim_{t \to t^*} \dot{e}(t) = -\infty$, which contradicts (18).

Similar statements can be obtained for the case $e_f(0) < (0)$.

Under the used assumptions, the following stability property is obtained: If the initial error $e_f(0)$ lies within the performance bounds, then the error $e_f(t)$ will remain inside the performance bounds for all times $t > 0$. In the following, we will investigate whether this result can be confirmed experimentally.

IV. PRACTICAL IMPLEMENTATION IN PITAIC

A. Adaptation of the Controller

The proposed controller is implemented as a separate class in pitaic and is instantiated during execution of each skill. However, due to the complexity and variety of hardware and software, the desired performance boundaries may be exceeded. In particular, the controller must be implemented in discrete time with a finite sampling rate, and the underlying velocity control error might be non-negligible in practice. If a performance boundary is reached, this lead to failure of the error transformation (5) due to an invalid argument of the natural logarithm. The following additional measures are proposed to address this problem in practice:

(a) Saturate the input of the error transformation (5) near the singularities, such that the transformed error is equal to a very large value $\varepsilon$ whenever the error is extremely close to, on, or even outside the performance bounds.

(b) Whenever the performance bounds are exceeded, shift them in time such that the error lies on the bounds, and increase the controller gain $a$ by a predefined factor.

In the following, we briefly discuss the implications of the proposed input saturation of the performance function and increase of controller gain $a$. With (1), (15) the time-derivative of $e_f$ is

$$
\dot{e}_f = \dot{f} = \begin{cases} 
K(-ae_f + b\dot{\dot{e}} + e_v) & \text{if } e_f(0) \geq 0, \\
K(-ae_f - b\dot{\dot{e}} + e_v) & \text{if } e_f(0) \leq 0.
\end{cases}
$$

As before, assume that $e_v$ is at least bounded, and note that $b, \dot{\dot{e}}$ are bounded by definition. Then, by choosing $a$ large enough, one can assure $|ae_f| > |b\dot{\dot{e}} + e_v|$. Recall that $K$ is positive and use (5) to find the following relationships:

- $\dot{e}_f < 0$ if the error reaches the upper bound.
- $\dot{e}_f > 0$ if the error reaches the lower bound.

Therefore, the error $e_f$ decreases with time and, due to $\lim_{t \to +\infty} \dot{\dot{e}} = 0$ and $\dot{D} = 0$, it is eventually forced into the region between performance bounds.

B. Application Setup

The contact establishment process in pitaic is divided into four steps:

Step 1. move the robot end effector to a vertical position with respect of the surface

Step 2. move the end effector towards the surface with constant speed until the measured force exceeds a predefined value.

Step 3. hold the current position long enough to assure accurate measurement of the current contact force

Step 4. activate the proposed PPC-based controller

V. EXPERIMENT & DISCUSSION

In this section, we will introduce the experimental setup and describe the experiments that are conducted to validate the transferability and reusability of the proposed controller.

A. Experimental Setup

The implementation described in Section IV is used to test the proposed contact force controller in environments with different stiffnesses and robots (Fig.1). The environmental stiffness is altered by using three different materials: steel, aluminum and PVC, as they are the most commonly used materials in industry. All materials are presented as thin boards with dimensions $100 \times 65 \times 1$ [mm] and are supported such that an elastic deformation takes place when a force acts on the center of the sheet (cf. Fig.1). The stiffnesses of the boards are, respectively, 54600 N/m, 18200 N/m, 260 N/m. The robots are lightweight manipulators from different manufactures:

- UR5 from Universal Robots A/S, Denmark is a 6-axis manipulator. Additionally, an external KMS40 force/torque sensor and an electrical two-finger gripper WSG 50 from Weiss Robotic are attached to the end effector for force-activated assembly tasks.
- Panda robot from Franka Emika GmbH in Munich, Germany is 7-axis manipulator. It is equipped with an internal force sensor and a two-finger gripper by factory default.
- Denso VS087 from Denso, Japan is a 6-axis robot. An external force sensor DynPick from Wacoh Tech and a gripper WSG50 from Schunk are equipped for assembly.

A qualitative relationship between their stiffnesses can be described as Denso > UR5 > Panda.

We compare the proposed PPC-based controller with a first-order-impedance controller based on stiffness control with low pass filters [10]. The filter is implemented as a first-order IIR filter. Two parameters are needed for the configuration of this controller: compliance of the environment, or rather the inverse value of stiffness, and cutoff frequency of the filter. They are chosen from exact/estimated environment model.

B. Parameterization of Performance Bounds

In this section, we briefly study the closed-loop dynamics of the PPC-based controller for two differently parameterized performance bounds. For sake of brevity, we only present results for the UR5 robot establishing a contact force of 45 N with the steel board. We consistently choose controller parameters $a = 0.0002$, $b = 0.00002$, and $D = 3$ [N].
Fig. 5 compares experimental results of both performance bounds. In the upper plot, the performance function is set to $\rho(t) = (10 + |e_f(0)|)e^{-3t} + 3$. The error decreases as quickly as pre-defined by $\rho(t)$. In contrast, the performance function is set to $\rho(t) = (0.00001 + |e_f(0)|)e^{-3t} + 3$ in the lower plot. It is observed that this desired decrease is too fast to be realized by the robot. Hence the adaptations described in Section IV-A are applied. The performance function is shifted in time, as illustrated in the plot. Despite the performance bound violations, the tracking error decays slightly faster than in the upper plot.

TABLE I. Settling time achieved by the proposed PPC-based controller for different robots and environments.

<table>
<thead>
<tr>
<th></th>
<th>Steel</th>
<th>Aluminum</th>
<th>PVC</th>
<th>variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>UR5</td>
<td>1.0 s</td>
<td>1.1 s</td>
<td>2.2 s</td>
<td>0.3 s²</td>
</tr>
<tr>
<td>Panda</td>
<td>2.3 s</td>
<td>2.5 s</td>
<td>2.9 s</td>
<td>0.1 s²</td>
</tr>
<tr>
<td>Denso</td>
<td>0.7 s</td>
<td>0.9 s</td>
<td>1.9 s</td>
<td>0.3 s²</td>
</tr>
<tr>
<td>variance</td>
<td>0.5 s²</td>
<td>0.5 s²</td>
<td>0.2 s²</td>
<td></td>
</tr>
</tbody>
</table>

TABLE II. Settling time achieved by the conventional impedance controller for different robots and environments.

<table>
<thead>
<tr>
<th></th>
<th>Steel</th>
<th>Aluminum</th>
<th>PVC</th>
<th>variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>UR5</td>
<td>1.8 s</td>
<td>1.2 s</td>
<td>13.0 s</td>
<td>29.4 s²</td>
</tr>
<tr>
<td>Panda</td>
<td>10.0 s</td>
<td>10.4 s</td>
<td>20.1</td>
<td>21.8 s²</td>
</tr>
<tr>
<td>Denso</td>
<td>0.5 s</td>
<td>0.5 s</td>
<td>8.0 s</td>
<td>12.5 s²</td>
</tr>
<tr>
<td>variance</td>
<td>17.7 s²</td>
<td>20.3 s²</td>
<td>24.6 s²</td>
<td></td>
</tr>
</tbody>
</table>

C. Transferability & Reusability

In order to evaluate the transferability and reusability, we compare the closed-loop dynamics of the proposed controller with those of the conventional first-order impedance controller. Both controllers are tuned only once initially and then remain unchanged throughout all experiments. These initial parameterizations are chosen such that both controllers have similar dynamics for the UR5 establishing a contact force of 45 N with the aluminum environment. In the proposed controller, this leads to a performance function $\rho(t) = (0.00001 + |e_f(0)|)e^{-3t} + 3, D = 3$ and controller parameters $a = 0.0002, b = 0.00002$, with the exception that for the Denso robot $b = 0$ was chosen due to large sensor noise, which leads to inaccurate differentiation of the force measurements. The first-order impedance controller is parametrized to have a compliance of 0.000061 m/N and cutoff frequency of 6 Hz.

Fig. 6 shows the time course of the tracking error for different robots and boards for both controllers. Only the first 4 s out of the total duration of 25 s are presented for each measurement. The tracking error tolerance ($\pm 3$ N) is marked by a gray band. The smallest time after which the tracking error remains inside that band is denoted settling time.
The same reference value is used for all experiments. However, there are small differences between the presented initial errors. These are due to different dynamics and uncertainties during impact control and switching to contact control after contact perception (cf. Section IV-B). We ignore these differences, since they are negligible compared to the effects we study in the following.

Analyzing Fig. 6, we find that both controllers achieve faster convergence for the steel board (highest stiffness) than for the other environments and faster convergence in the Denso robot (highest stiffness) than in the other robots. However, a remarkable difference between both controllers is the following: The first-order impedance controller exhibits overshoots for UR5 and Denso establishing contact with the steel environment. Such overshoots are undesirable, since they may lead to damage of the workpiece in assembly tasks. In addition, the convergence speed that is achieved by the impedance controller varies largely from the Franka Panda to the UR5 and the Denso robot, at least for the steel and aluminum environment, and it is generally very slow for the PVC environment. This result implies that the impedance controller would have to be reparameterized every time the robot and the environment changes in order to avoid bad performance during industrial production.

In strong contrast, all error trajectories that are generated by the proposed controller have similar characteristics. They enter the region of tolerance within 3 s and exhibit no overshoot in any of the trials. Therefore, the proposed PPC-based controller can be transferred from one robot to another and reused for environments with different stiffnesses without any reparameterization.

The reusability and transferability are further analyzed in Tables I and II, which compare the variance of the settling time for different robots (rows) and environments (columns). We use the empirical variances to quantify the dispersion of settling time in different situations. They are calculated respectively with three time values in each row/column. Comparison of both tables reveals that the variance of the PPC-based controller is always lower than that of the impedance controller – by a factor ranging from 35 to more than 100. This quantifies how much better the reusability and transferability of the proposed controller is compared to the conventional impedance controller.

VI. CONCLUSIONS

In this work, a force controller for contact establishment in assembly tasks of industrial robots has been proposed. The controller is based on prescribed performance and does not require a model of the robot dynamics or the environments and assures similar closed-loop dynamics for different robots and environment stiffnesses.

We evaluated the performance of the proposed controller with UR5, Panda and Denso robots for contacts with steel, aluminum and PVC boards. We found that, in contrast to a conventional first-order impedance controller, the proposed controller can be transferred from one robot to the next and reused for different environment materials without any parameter adjustment. It consistently achieved similar performance characteristics across all trials, i.e. in particular no overshoot and no slow convergence.

Further research will aim at investigating the phenomenon that the original prescribed performance bounds are violated in some cases. Convergence analysis should account for non-negligible velocity control dynamics, and methods for discrete-time design of PPC should be developed, which is inline with [16]. Besides, further experiments will be conducted to evaluate the performance of the proposed controller within a benchmarking contour-following task [17].

REFERENCES

[1] IFR, "Executive Summary World Robotics 2018 Industrial Robots" International Federation of Robotics, 2018