An Admittance Controller with a Jerk Limiter for Position-Controlled Robots

Ryusei Mae and Ryo Kikuuwe

Machinery Dynaics Laboratory, Hiroshima University, 1-4-1 Kagamiyama, Higashi-Hiroshima, Hiroshima 739-8527, Japan

This paper proposes an admittance control scheme for robots equipped with joint-level position controllers involving deadtime. Its main feature is an elaborate discrete-time jerk limiter, which limits the third derivative of the position command sent to the controller. The jerk limiter is designed to suppress undesirable oscillation especially when the robot is in contact with stiff environments. The controller is designed as a differential inclusion involving normal cones in the continuous-time domain, and its discrete-time algorithm is derived by the implicit Euler discretization. The presented controller was validated with experiments using a collaborative robot UR3e of Universal Robots, which has the deadtime of 6 ms in the velocity-command mode.

Keywords: Admittance control, jerk limiter, vibration suppression, normal cone, differential inclusion, UR3e

1. Introduction

Robots subject to physical contact with external environments need appropriate controllers to regulate the contact forces. Most industrial robots are controlled by dedicated position controllers that force the robots to track position commands from upper-level controllers. To enable such a robot to respond to external forces, an additional controller is needed to modify the position commands according to the contact forces.

Admittance control is a control scheme that is suited for position-controlled robot manipulators. It is often referred to as position-based impedance control, and it can be implemented as an outer feedback loop attached to a position-controlled system. In an admittance controller, a virtual object having desired inertia, viscosity, and stiffness is considered, and its motion is simulated according to the external force that is obtained by a force sensor or some estimation methods. The robot is then position-controlled to track the motion of the virtual object.

One problem of admittance control is instability. The robot may become unstable, especially when it is in contact with a stiff environment and when the time delay in the controller is large [1]. It is known [1] that admittance control can be stabilized by setting high values to the viscosity and inertia of the virtual object, but it would deteriorate the responsiveness against the external forces. Some strategies have been proposed to adaptively vary the viscosity and inertia parameters of admittance control [2–11]. The trade-off between stability and responsiveness, however, is still inevitable in these methods. Current velocity feedback [12], which injects additional damping to the system, has been shown to be effective to enhance the stablity, but it would also deteriorate the responsiveness.

It has been known [13–15] that the feedforward of the target acceleration in the position controller enhances the stability. It is however not applicable to most commercially-available manipulators because they usually do not provide access to the internal algorithms of their position controllers. Some researchers [16–18] employed fractional-order dynamics for admittance controller. It has been reported [16] that it contributes to better stability in comparison to the integer-order counterparts. Its efficacy, however, is still theoretically unclear and its physical interpretation is complicated, possibly leading to difficulties in the parameter tuning.

There have also been approaches to reduce the risk of instability and vibration by limiting the actuator torque [14] and the commanded acceleration [13–15]. They are also strongly coupled with the internal position controllers, and are not very straightforward to use with commercial position-controlled robots. An approach to alter the commanded acceleration to enhance the stability has been studied [19], but it is also coupled with the internal position controller and its effectiveness in the presence of the time delay is unclear. Imposing limits to the commanded velocity and acceleration has also been studied in [20], but it is primarily intended for enhancing safety, not for the stability.

This paper proposes an admittance control scheme with a jerk limiter for position-controlled robots. The novelty of the proposed admittance controller is that it has a jerk limiter, which limits the third derivative of the position command sent to the robot. The jerk limiter is shown to be effective in suppressing the vibration that happens when the robot is in contact with stiff environments. It has a structure that adjusts the jerk limit according to the velocity and the acceleration in order not to sacrifice the responsiveness to external forces. The vibration caused by the instability is suppressed by limiting the first, second, and third derivatives of the position command. One of its practical benefits is that it suppresses the vibrations

corresponding author: Ryo Kikuuwe (kikuuwe@ieee.org)

even if the virtual viscosity and virtual inertia are set low. The effectiveness of the proposed method is verified by some experiments using a collaborative robot UR3e.

The remainder of this paper is organized as follows. Section 2 provides some mathematical preliminaries. Section 3 details the proposed admittance controller. Section 4 presents the experimental results obtained by implementing the proposed controller on a UR3e robot. Section 5 concludes this paper.

2. Mathematical Preliminaries

Let a, b, and x are real numbers and $a \leq b$. This paper uses the following functions:

$$\mathrm{proj}_{[a,b]}(x) \triangleq \begin{cases} b & \text{if} \quad b < x \\ x & \text{if} \quad a \leq x \leq b \\ a & \text{if} \quad x < a \end{cases} \tag{1}$$

$$\mathrm{dzn}_{[a,b]}(x) \triangleq x - \mathrm{proj}_{[a,b]}(x) \tag{2}$$

$$\mathcal{N}_{[a,b]}(x) \triangleq \begin{cases} (-\infty, 0] & \text{if} \quad x = a \\ 0 & \text{if} \quad a < x < b \\ [0, \infty) & \text{if} \quad x = b \\ \emptyset & \text{if} \quad x > b \vee x < a. \end{cases} \tag{3}$$

The functions proj and dzn can be referred to as the projection and deadzone functions, respectively. The function $\mathcal{N}$ is called the normal cone [21]. The following relation exists between the projection function and the normal cone:

$$y \in x - \mathcal{N}_{[a,b]}(y) \iff y = \mathrm{proj}_{[a,b]}(x), \tag{4}$$

which has been shown in previous publications (e.g., [22, Proposition 2], [23, Section A.3], and [24, Proposition 6.47]).

3. proposed method

3.1. CONVENTIONAL Admittance Controller

Fig. 1(a) is an example of the structure of an admittance controller. The controller comprises a virtual object, often referred to as a *proxy*, and the robot is position-controlled (or velocity-controlled) to track the proxy position. The proxy moves according to a predetermined equation of motion, of which a typical example can be described as follows:

$$M(\ddot{q} - \ddot{p}_d) + B(\dot{q} - \dot{p}_d) + K(q - p_d) = f + f_d. \tag{5}$$

Here, q is the proxy position and f is the force acting on the robot, which is obtained by a force sensor or other means. The quantities p_d and f_d are the desired position and the desired force, respectively, given as inputs to the controller, and M, B, and K are positive constants representing the inertia, viscosity, and stiffness, respectively. As long as the position controller is accurate enough, the robot ’ s response to the external force f is close to the proxy dynamics described by (5).

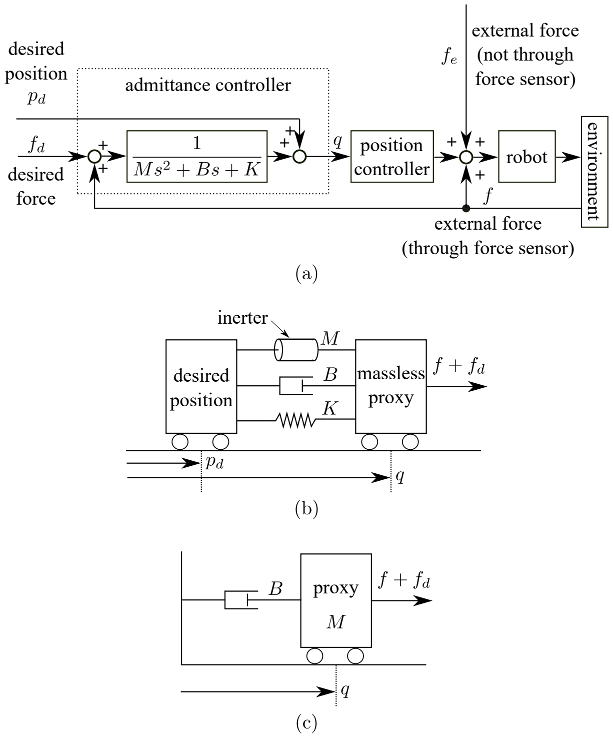


Fig. 1. System controlled with admittance controller. (a) Typical admittance controller. (b) A physical interpretation of the proxy dynamics (5). (c) A physical interpretation of the proxy dynamics (5) with $K = 0$ and $\dot{p}_d \equiv 0$.

One interpretation of the proxy dynamics (5) can be illustrated in Fig. 1(b). Here, the proxy can be considered as a massless object, and it is connected to the desired position p_d through a parallel spring-damper-inerter element[1]. The forces f and f_d act on the proxy. The proxy dynamics of this form can be used when, e.g., the robot should track the desired position p_d with a certain level of compliance against the external force f, by deviating from the desired position p_d. In such an application, the desired force f_d should usually be set to be constantly zero, and also, the parameters should be set to satisfy $B^2 > 4KM$ to prevent overshoots.

One variation of the controller can be obtained by setting $K = 0$ and $\dot{p}_d \equiv 0$ with the proxy dynamics (5). In such a case, the proxy dynamics reduces to the one illustrated in Fig. 1(c), in which the proxy is a point mass combined with a damper subject to the forces f and f_d. Such a controller can be used when, e.g., the robot should yield to the external force, such as the one applied by a human user in case of direct teaching, or when the robot should apply the force f_d to an external object in case of grinding or assembly tasks.

It is well known that admittance-controlled systems are prone to instability, especially when the robot is in contact with an external object. As has been discussed in previous work, e.g., [14, 15], the cause of the instabil-

[1] This paper uses the term inertia to mean an element that produces the force proportional to the relative acceleration between its two ends.

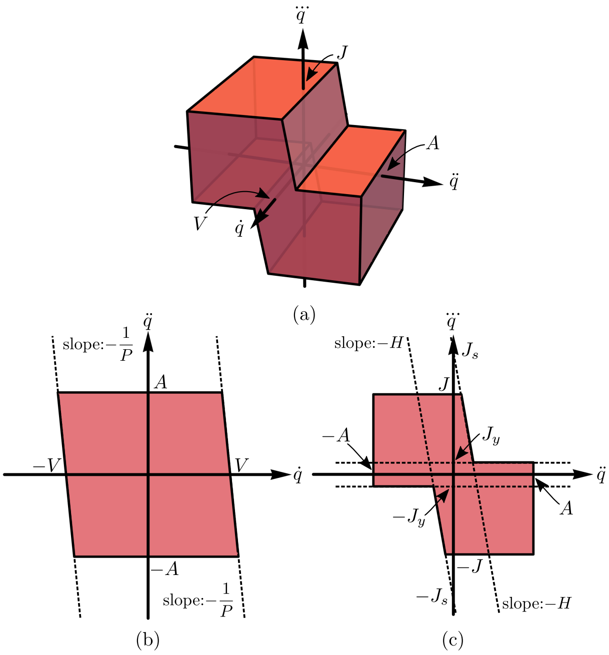


Fig. 2. Illustration of the three-dimensional region within which $[\dot q, \ddot q, \dddot q]^{\mathrm{T}}$ satisfies the constraint (8). (a) Three-dimensional plot of the region. (b) The cross-section of the region at $\dddot q = 0$. (c) The cross-section of the region at $\dot q = 0$.

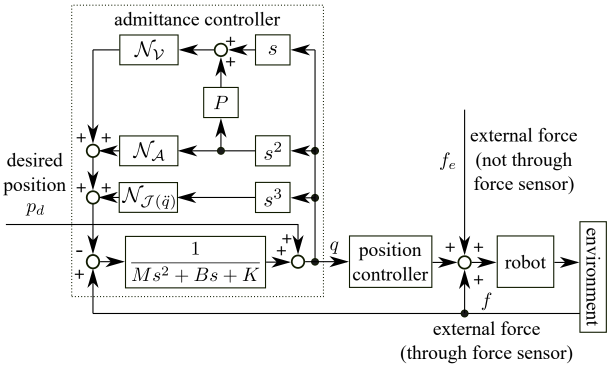


Fig. 3. System controlled with proposed admittance controller.

ity is the phase lag or the deadtime in the closed loop. One approach to compensate for the phase lag is to use phase-leading and acceleration feedforward techniques in the position controller [15]. The effect of instability can be attenuated by imposing limits on the actuator torque [14]. These approaches need to alter the structure of the position controllers and thus are not applicable to commercial robot manipulators equipped with proprietary position controllers. Moreover, it is questionable whether these methods are applicable to deadtime of more than a few milliseconds, which can exist in commercial position controllers.

3.2. Proposed Admittance controller

As an extension of the typical admittance controller (5), this paper proposes an admittance controller that can be written as follows:

$$\begin{aligned} M(\ddot q - \ddot p_d) + B(\dot q - \dot p_d) + K(q - p_d) \\ \in f - \mathcal{N}_{\mathcal{V}}(\dot q + P\ddot q) - \mathcal{N}_{\mathcal{A}}(\ddot q) - \mathcal{N}_{\mathcal{J}(\ddot q)}(\dddot q). \end{aligned} \tag{6}$$

where

$$\mathcal{V} \triangleq [-V, V] \tag{7a}$$

$$\mathcal{A} \triangleq [-A, A] \tag{7b}$$

$$\begin{aligned} \mathcal{J}(\ddot q) \triangleq [&\mathrm{proj}_{[-J,-J_y]}(-J_s - H\ddot q), \\ &\mathrm{proj}_{[J_y,J]}(J_s - H\ddot q)] \end{aligned} \tag{7c}$$

Here, P, V, A, J, J_s, H, and J_y are positive constants. The last three terms of the right-hand side of the differential inclusion (6) constrain the vector $[\dot q, \ddot q, \dddot q]^{\mathrm{T}}$ within the three-dimensional region illustrated in Fig. 2, which represents the intersection of the following three constraints:

$$\dot q + P\ddot q \in \mathcal{V},\ \ddot q \in \mathcal{A},\ \dddot q \in \mathcal{J}(\ddot q). \tag{8}$$

As long as $[\dot q, \ddot q, \dddot q]^T$ is within the region in Fig. 2, (6) reduces to a spring-damper-inerter system illustrated in Fig. 1(b). As will be detailed in the next Section 3.3, P and H should be set so that (6) has a solution in the discrete-time domain. A block diagram of the proposed controller (6) is shown in Fig. 3.

The second term of the right-hand side of (6) is for limiting the velocity, and it is mainly for safety reasons. The term $+P\ddot q$ is for realizing the exponential convergence to the velocity limits $\pm V$. As has been pointed out in [20], a simple velocity limiter without such a term as $+P\ddot p$ would cause an abrupt change in the acceleration at the time of reaching the velocity limit, which would not be favorable.

The third and fourth terms of the right-hand side of (6) aim to reduce the amplitude of the robot's vibration by limiting the commanded jerk $\dddot q$. This is motivated by our preliminary observation that an admittance-controlled robot with a few milliseconds of deadtime generated high-frequency vibration in contact with an external object. Limiting the commanded acceleration $\ddot q$ was somewhat effective for suppressing the vibration, but it resulted in poor responsiveness of the robot to the external force. In contrast, limiting both the commanded jerk $\dddot q$ and the commanded acceleration $\ddot q$ was rather effective in realizing both smaller vibration and better responsiveness.

Our observation showed that the vibration could not be sufficiently suppressed only by the adjustment of the upperbounds A and J, which are for the commanded acceleration and the commanded jerk, respectively. Specifically, when J/A is large, high-frequency oscillations could not be eliminated, but when J/A is small, low-frequency and high-amplitude oscillations took place. The design of the acceleration-dependent jerk limits illustrated in Fig. 2 is motivated by this observation. The basic idea is that, when the acceleration is high, the jerk limit should be lowered to suppress the high-frequency vibration, but when the acceleration is low, higher jerk values should be allowed to prevent low-frequency, high-amplitude oscillation. As illustrated in Fig. 2, the parameters J_s and J_y should be chosen as $J_s < J$ and $J_y > J$. The values of J_s, J_y, and H need be chosen considering the trade-off between the vibration suppression and the responsiveness. Specifically, a higher J_y, a higher J_s, and a smaller H result in high responsiveness and larger vibration. While definitive guidelines for tuning these parameter values are currently unavailable, one can experimentally adjust them through trial and error.

3.3. Discrete-time representation

The discrete-time representation of (6) and (7) are now derived. Let T be the timestep size and k be the integer representing the discrete-time index. By using the implicit Euler discretization, (6) and (7) can be discretized as follows:

$$j_k \in j_k^* - \mathcal{N}_{\mathcal{V}}(v_k + Pa_k) - \mathcal{N}_{\mathcal{A}}(a_k) - \mathcal{N}_{\mathcal{J}(a_k)}(j_k). \quad (9)$$

where

$$v_k \triangleq (q_k - q_{k-1})/T \quad (10)$$

$$a_k \triangleq (q_k - 2q_{k-1} + q_{k-2})/T^2 \quad (11)$$

$$j_k \triangleq (q_k - 3q_{k-1} + 3q_{k-2} - q_{k-3})/T^3. \quad (12)$$

$$q_k^* \triangleq p_{d,k} + \frac{(2M+BT)(q_{k-1} - p_{d,k-1})}{M+BT+KT^2} - \frac{M(q_{k-2} - p_{d,k-2}) - T^2 f_k}{M+BT+KT^2} \quad (13)$$

$$j_k^* \triangleq (q_k^* - 3q_{k-1} + 3q_{k-2} - q_{k-3})/T^3. \quad (14)$$

Through the derivation detailed in Appendix Appendix A, (9) can be written as follows:

$$j_k \in j_k^* - \mathcal{N}_{\mathcal{L}_k}(j_k) \quad (15)$$

where

$$\bar{V}_{b,k} \triangleq \frac{-V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT} \quad (16)$$

$$\bar{V}_{t,k} \triangleq \frac{V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT} \quad (17)$$

$$\bar{A}_{b,k} \triangleq (-A - a_{k-1})/T \quad (18)$$

$$\bar{A}_{t,k} \triangleq (A - a_{k-1})/T \quad (19)$$

$$\bar{J}_{b,k} \triangleq \mathrm{proj}_{[-J,-J_y]}\Big(\frac{-Ha_{k-1} - J_s}{1+HT}\Big) \quad (20)$$

$$\bar{J}_{t,k} \triangleq \mathrm{proj}_{[J_y,J]}\Big(\frac{-Ha_{k-1} + J_s}{1+HT}\Big) \quad (21)$$

$$\mathcal{L}_k \triangleq [\max(\bar{V}_{b,k}, \bar{A}_{b,k}, \bar{J}_{b,k}), \min(\bar{V}_{t,k}, \bar{A}_{t,k}, \bar{J}_{t,k})]. \quad (22)$$

By using the relation (4), (15) can be written as follow:

$$j_k = \mathrm{proj}_{\mathcal{L}_k}(j_k^*). \quad (23)$$

In conclusion, the algorithm of the proposed controller (6) and (7) can be written as follows:

$$q_k^* := p_{d,k} + \frac{(2M+BT)(q_{k-1} - p_{d,k-1})}{M+BT+KT^2} - \frac{M(q_{k-2} - p_{d,k-2}) - T^2 f_k}{M+BT+KT^2} \quad (24a)$$

$$j_k^* := (q_k^* - 3q_{k-1} + 3q_{k-2} - q_{k-3})/T^3 \quad (24b)$$

$$a_{k-1} := (q_{k-1} - 2q_{k-2} + q_{k-3})/T^2 \quad (24c)$$

$$v_{k-1} := (q_{k-1} - q_{k-2})/T \quad (24d)$$

$$\bar{V}_{b,k} := \frac{-V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT} \quad (24e)$$

$$\bar{V}_{t,k} := \frac{V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT} \quad (24f)$$

$$\bar{A}_{b,k} := (-A - a_{k-1})/T \quad (24g)$$

$$\bar{A}_{t,k} := (A - a_{k-1})/T \quad (24h)$$

$$\bar{J}_{b,k} := \mathrm{proj}_{[-J,-J_y]}\Big(\frac{-Ha_{k-1} - J_s}{1+HT}\Big) \quad (24i)$$

$$\bar{J}_{t,k} := \mathrm{proj}_{[J_y,J]}\Big(\frac{-Ha_{k-1} + J_s}{1+HT}\Big) \quad (24j)$$

$$\mathcal{L}_k := [\max(\bar{V}_{b,k}, \bar{A}_{b,k}, \bar{J}_{b,k}), \min(\bar{V}_{t,k}, \bar{A}_{t,k}, \bar{J}_{t,k})] \quad (24k)$$

$$j_k := \mathrm{proj}_{\mathcal{L}_k}(j_k^*) \quad (24l)$$

$$q_k := 3q_{k-1} - 3q_{k-2} + q_{k-3} + T^3 j_k. \quad (24m)$$

Note that the algorithm (24) is something that can be directly implemented to control devices through translations into appropriate programming languages. The computational load of the algorithm (24) is almost negligible since it does not involve iterative computations.

The values for the parameters P and H should be chosen so that the algorithm (24) always has a solution. Specifically,

$$\max(\bar{V}_{b,k}, \bar{A}_{b,k}, \bar{J}_{b,k}) \leq \min(\bar{V}_{t,k}, \bar{A}_{t,k}, \bar{J}_{t,k}) \quad (25)$$

is necessary and sufficient for the existence of the solution, assuring the non-emptiness of the set $\mathcal{L}_k$ appearing in (24k). To verify the condition (25), one needs to check nine (= 3 × 3) inequalities. One can easily see that, if $[v_{k-1}, a_{k-1}, j_{k-1}]^T$ is included in the set defined by (8) and illustrated in Fig. 2, the following inequalities are satisfied:

$$\bar{A}_{b,k} \leq 0 \leq \bar{A}_{t,k},\ \bar{J}_{b,k} \leq 0 \leq \bar{J}_{t,k},\ \bar{V}_{b,k} < \bar{V}_{t,k}. \quad (26)$$

Among the nine inequalities required by (25), five are implied by (26). The remaining four inequalities are:

$$\bar{V}_{b,k} \leq \bar{A}_{t,k},\ \bar{V}_{b,k} \leq \bar{J}_{t,k},\ \bar{A}_{b,k} \leq \bar{V}_{t,k},\ \bar{J}_{b,k} \leq \bar{V}_{t,k}. \quad (27)$$

Through tedious but straightforward derivations (with the help of symbolic computation software programs such as Mathematica), one can see that the following condition is sufficient for the satisfaction of (27):

$$P \geq A/J - T \ \wedge\ H \geq 1/P. \quad (28)$$

That is, the values for the parameters P and H need to be chosen to satisfy (28) in the implementation of the algorithm (24). More specifically, P needs to be set slightly larger than $A/J - T$ because setting a larger P could lead to undesirable deceleration before reaching the velocity limits ±V. Regarding H, as mentioned in Section III.B, it should be smaller to achieve higher responsiveness, but it must be large enough not only to suppress vibrations but also to satisfy (28).

3.4. Properties of the Controller

The behavior of the robot in contact with a stiff environment is the main concern in force control applications. The proposed controller (6), of which the discrete-time representation is (24), is intended to alleviate the concern in comparison to the conventional simple controller (5), but it should be noted that the stability is not guaranteed with the proposed controller. Recall that the proposed controller (6) combines a linear controller (5) with some limiters, making the controller nonlinear. It is obvious that incorporating limiters, in principle, cannot alter an unstable system into a stable system, especially in the sense of Lyapunov stability. The following discussion shows that the limiters in the controller result in stable limit cycles in the systems when it is divergent without the limiters.

The following discussion is based on the conventional describing function method. The describing function of the proposed controller (6), or its discrete-time representation (24), cannot be analytically obtained. It, however, can be numerically obtained by providing sinusoidal inputs with different amplitudes and frequencies. Let us consider the sequence $\{f_k\}_{k\in\mathbb{Z}}$ where

$$f_k = U\cos(k\omega T), \tag{29}$$

which is a sinusoidal signal with the amplitude $U > 0$ and the frequency $\omega > 0$. Let us assume that, with the input sequence $\{f_k\}_{k\in\mathbb{Z}}$, the algorithm (24) provides an output sequence $\{q_k\}_{k\in\mathbb{Z}}$. Based on these sequences, the describing function $\Psi : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{C}$ of the algorithm (24) can be numerically obtained as follows:

$$\Psi(U,\omega) \triangleq \frac{\sum_{k=n}^{n+2\pi/\omega} q_k(\cos(k\omega T) - j\sin(k\omega T))}{U\sum_{k=n}^{n+2\pi/\omega}\cos(k\omega T)^2}. \tag{30}$$

With the algorithm (24), if the input amplitude U is small enough, the jerk j_k does not reach the upperbounds in the algorithm and thus the output $\{q_k\}_{k\in\mathbb{Z}}$ becomes equivalent to the output of the linear controller (5), which is the algorithm (24) with the limits $\{V,A,J,J_s,J_y\}$ set to be the infinity. Therefore, one can see that the function $\Psi(U,\omega)$ has the following property:

$$\lim_{U\to 0}\Psi(U,\omega) = \frac{1}{K - M\omega^2 + jB\omega}. \tag{31}$$

Here, note that the right-hand side of (31) is the transfer function of the linear controller (5) in the frequency domain. In addition, when the input amplitude U becomes larger, the signal amplitude would more frequently hit the upperbounds, and the ratio of the output amplitude against the input amplitude U would decrease. Therefore, $\Psi(U,\omega)$ also has the following property:

$$\frac{\partial|\Psi(U,\omega)|}{\partial U} < 0. \tag{32}$$

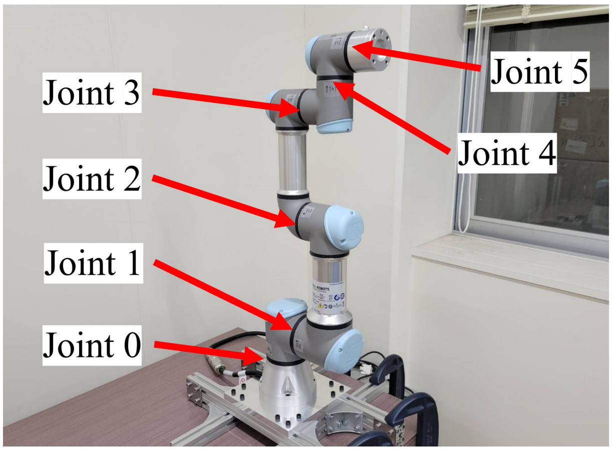


Fig. 4. Experimental setup (UR3e, Universal Robots) and its joint numbers.

Recall that the output q of the algorithm (24) is sent to the robot controller as the position command. Let $P(s)$ be the transfer function from the position command q to the robot position p. Let K_e be the stiffness of the environment in contact with the robot's end-effector. Then, the force f, which is used as the input to the algorithm (24), satisfies $f = -K_e p$, and the open-loop transfer function of the system in the frequency domain is $-K_eP(j\omega)\Psi(U,\omega)$. By using this, we can see that the oscillation of the force f with the angular frequency ω and the amplitude U persists if the following harmonic balance equation [25] is satisfied:

$$K_eP(j\omega)\Psi(U,\omega) = -1 + 0j. \tag{33}$$

In addition, because of the property (32), if the magnitude of the oscillation is larger or smaller than U satisfying (33), it would decrease or increase, respectively, until it reaches U. It means that the limit cycle with the angular frequency ω and the amplitude U satisfying (33) is stable. The existence of the limit cycle can also be explained in a similar light as in [26, Section 5.4.3].

The obtained conclusion is that the admittance controller without limiters may be unstable and divergent, while the one combined with the limiters results in stable limit cycles and does not diverge. Whether the limit cycle is practically acceptable or not would depend on its amplitude and frequency and also on the application. The parameters should be chosen so that at least the amplitude of the limit cycle is small enough, although tuning guidelines of parameters are still unclear. Experiments in the next section empirically show that the oscillation can be made acceptably small with a robot with 6-ms deadtime by careful tuning of parameters.

4. Experiment

4.1. Experimental Setup

The six-DOF collaborative robot UR3e (Universal Robots) shown in Fig. 4 was used in the experiments. Its payload capacity is 3 kg, and its maximum reach is 500 mm from the center of Joint 0. Universal Robots provides an API named Real-Time Data Exchange (RTDE) to allow communication between the dedicated robot controller and user programs running on a PC through TCP/IP at the cycle of $T = 0.002$ s. This study used the RTDE function `speedJ` to send the velocity command v $(= \dot{q})$ to the robot. (The second argument A^*, the maximum acceleration, of `speedJ` was set as $A^* = 5$ rad/s^2.) We did not use the position command q directly because the function `servoJ`, which is to send position commands, caused significant latency between the sent command q and the measured position p_s, which resulted in low stability of admittance control.

We estimated the external force f, which is necessary for admittance controllers, using the motor current and precalibrated weight distribution of the robot. Specifically, we computed the input force f in the following procedure:

$$\hat{f}_c := \hat{k}_c c \tag{34a}$$

$$f := \mathrm{dzn}_{[-F,F]}(-\hat{f}_c + f_g(p_s)). \tag{34b}$$

Here, $\hat{f}_c$ is the estimated torque generated by the motor, $\hat{k}_c$ is the estimated torque constant of the motor, c is the motor torque obtained by the function `getActualCurrent`, p_s is the joint angle obtained by the function `getActualQ`, and $f_g(p)$ is the torque caused by the gravity. The use of the deadzone function in (34) is to attenuate the influence of the noise and the inertia. The parameters needed to calculate $f_g(p)$ were chosen through some preliminary experiments. The torque constants $\hat{k}_t$ were estimated as $\{\hat{k}_{t0}, \hat{k}_{t1}, \hat{k}_{t2}, \hat{k}_{t3}, \hat{k}_{t4}, \hat{k}_{t5}\} = \{18, 18, 9.5, 4, 4, 4\}$ Nm/A.

For the purpose of comparison, we used the following two controllers:

- cP: Proposed controller (24).
- cN: Controller cP with a simple jerk limiter, $\mathcal{J} = [-J, J]$.

Unless otherwise specified, the parameters were set as shown in Table 1.

4.2. Preliminary: Identification of Deadtime

We performed a set of preliminary experiments to identify the deadtime in the robot controller. The robot controller receives the velocity command v from the PC through TCP/IP through an RTDE function `speedJ`. The function receives another argument, which is an acceleration limit A^*. It then produces the motor current c, which can be monitored by the function `getActualCurrent`, and the resultant joint velocity is obtained by the function `getActualQd`. In addition, there is a function `getTargetQd` to monitor a “ target velocity, ” which we hereafter refer to as v_r. It is presumably a modified velocity command used within the controller. The relations among v, v_r, c, and v_s are inferred to be as illustrated in Fig. 5.

Table 1. Parameters used in the experiments.

symbols	physical meaning	value and units
M	the virtual inertia	0.0625 kg·m^2 $(= B^2/(4K))$
B	the virtual viscosity	0.5 Ns/rad
K	the virtual stiffness	1 N/rad
V	the velocity limit	1.5 rad/s
A	the acceleration limit	5 rad/s^2
J	the jerk limit	300 rad/s^3
P	the time constant for the velocity limit	0.015 s
H	the slope of the jerk limit shown in Fig. 2(c)	3000 s^{-1}
J_s	the intercept of the jerk limit shown in Fig. 2(c)	500 rad/s^3
J_y	the jerk limit at high acceleration	30 rad/s^3

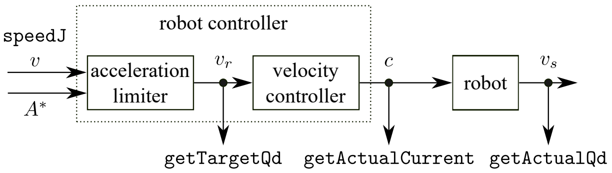


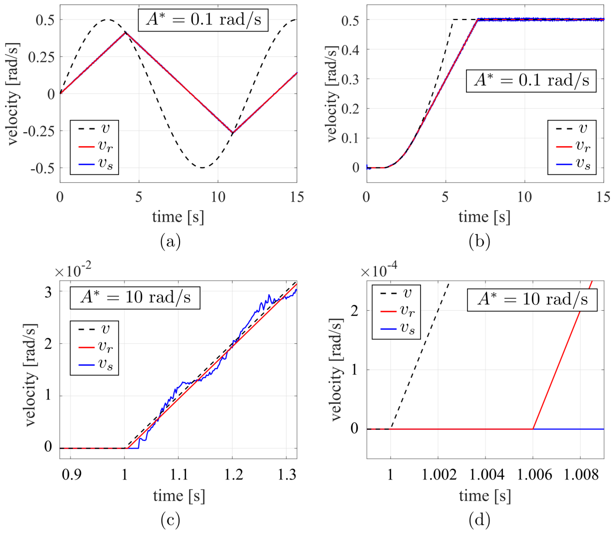
Fig. 5. Relation among the velocity command v, the modified ’target velocity’ v_r subject to the acceleration limit A^*, the motor current c, and the measured velocity v_s in the experimental setup. Relevant RTDE functions are also shown.

Fig. 6 shows the results of some preliminary experiments. Fig. 6(a) shows the results of a trial where v was a sinusoidal wave with $A^* = 0.1$ rad/s^2 and Fig. 6(b) shows the results of a trial where $v = \min(0.5, (t-1)^2/40)$ rad/s with $A^* = 0.1$ rad/s^2. It can be seen that the measured joint velocity v_s follows the modified velocity v_r, not the original velocity command v, and that v_r tracks v as $|\dot{v}|$ is smaller than A^*. Figs. 6(c) and (d) show the result of trials with $A^* = 10$ rad/s plotted in different time scales. They show that there is a time delay of approximately 6 ms between v and v_r. It means that the deadtime of approximately 6 ms exists in the robot controller.

4.3. Experiment I: moved by hand

In the first set of experiments, the experimenter moved the robot by pushing the end-effector by hand. The controllers cN and cP were implemented to Joint 0. The parameter F was set as $F = 28$ Nm. The desired position was set as $p_d = 0$ throughout the experiments.

In the beginning, the robot was stationary at $p_d = 0$ as



(a) (b) (c) (d)

Fig. 6. Velocity command v sent to the robot controller, the ‘target velocity’ v_r obtained from the robot controller, and the measured joint velocity v_s. (a) A sin wave was commanded by v with the acceleration limit of `speedJ` set as $A^* = 0.1$ rad/s^2. (b) A parabola $(t-1)^2/40$ with an upper bound 0.5 rad/s was commanded by v with $A^* = 0.1$ rad/s^2. (c) The command v was changed from 0 rad/s with $A^* = 10$ rad/s^2. (d) An enlarged version of (c).

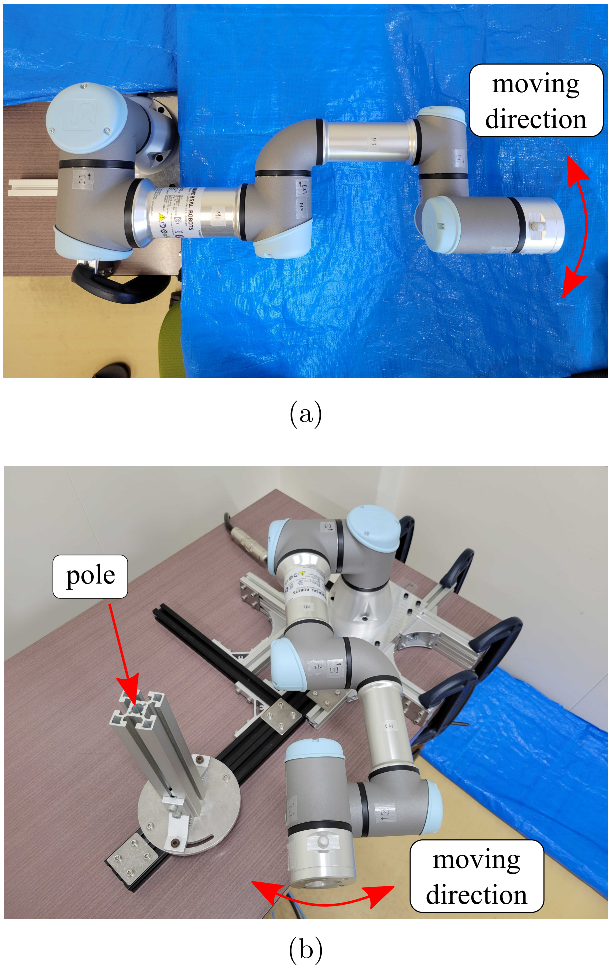
shown in Fig. 7(a). Then, the experimenter pushed the robot by hand on the end-effector for a while (Period A). After that, the experimenter stopped and firmly grasped the end-effector (Period B). Finally, the experimenter put his hand away from the end-effector.

The results are shown in Fig. 8. In Periods A, the robot was being moved by the external force, the generated torque $\hat{f}_c$ and the commanded acceleration a vibrated, but the amplitude was smaller with cP than with cN. This can be seen as the effect of cP bounding the commanded jerk j. The difference was more distinct in Periods B, in which the vibrations are much smaller with cP than with cN, especially in the commanded acceleration a. In fact, the actuator generated a noisy sound with cN but not with cP, and the experimenter did not feel the vibration with cP while he was grasping the robot. After Period B, the experimenter released the robot, and the robot position p converged to p_d smoothly with both controllers, as intended by the setting of the values of M, B, and K resulting in the critical damping.

Although it is not shown in the figures, when J was chosen larger, the vibration amplitude became larger, and the teaching pendant of UR3e displayed an error message.

4.4. Experiment II: contact with environment

The second set of experiments was conducted to investigate the performance of the proposed controller in contact with a rigid external object. As shown in Fig. 7(b), an aluminum pole was fixed to the robot ’ s base, and Joint 0 was controlled so that the end-effector gains contact with the pole. Specifically, p_d of Joint 0 was initially set as $p_d = -2.67$ rad, and from $t = 1$ s, p_d was varied with the velocity $\dot{p}_d = -0.01$ rad/s. The end-effector contacted the pole at around $t = 1.6$ s, and after that, the robot continued applying the force on the pole. The parameters were set as the same as in Experiment I, except that F was set as $F = 6$ Nm.



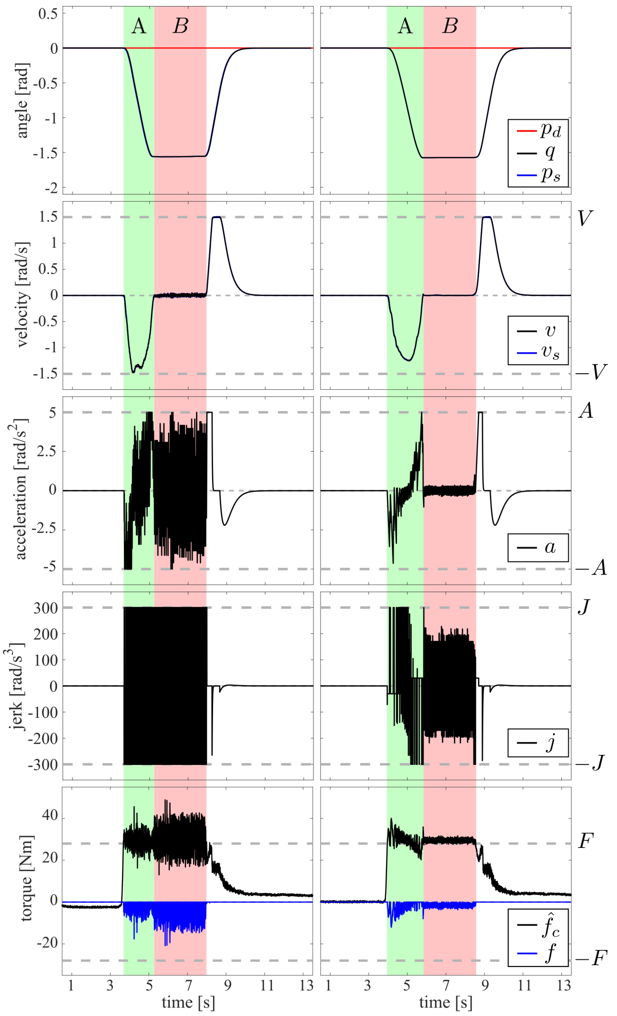
(a) (b)

Fig. 7. Configurations for (a) Experiment I and (b) Experiment II.

The results are shown in Fig. 9. The controller cN resulted in bouncing when the end-effector contacted the pole. In contrast, the controller cP maintained contact with the surface of the pole. Even with the controller cP, the force $\hat{f}_c$ vibrated with a small amplitude during the contact with the pole as shown in Fig. 9(b). No noisy sound, however, was heard from the actuator and the contact surface.

4.5. Experiment III: moved by hand with 6-DOF control

In the third set of experiments, the controller cN and cP were implemented in all joints of the robot with the setting $K = 0$. The experimenter grasped the end-effector and moved it in a circle. Because of the setting $K = 0$, the controller was supposed to behave as a mass-damper system illustrated in Fig. 1(c). The parameter F were set

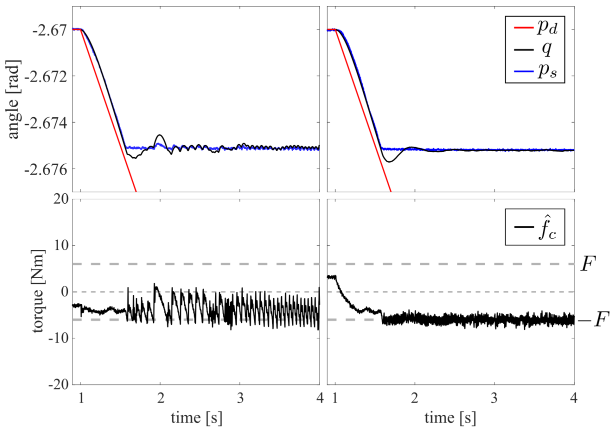


(a) controller cN (b) controller cP

Fig. 8. Results of Experiment I. In Period A, the experimenter moved the robot by hand. In Period B, the experimenter held the robot and stopped its movement. After Period B, the experimenter took his hand away from the robot.

as $\{F_0,F_1,F_2,F_3,F_4,F_5\} = \{4,6,3,1,0.5,0.4\}$ Nm. The values of F were determined so that the experimenter was able to move the robot with a light hand and also $|\hat{f}_e|$ does not exceed F during the motion as long as no external forces acted. Other parameters were set the same as in Experiment I for all joints.

The results are shown in Fig. 10. With both controllers, the experimenter was able to make the robot move along a circle. With controller cP, the magnitudes of vibration of $\hat{f}_c$ were smaller than controller cN. With controller cN, the experimenter heard a noisy sound from the actuators and felt a vibration in his hand, but with cP, he did not perceive sound or vibration.



(a) controller cN (b) controller cP

Fig. 9. Results of Experiment II. The end-effector contacted the pole at around $t = 1.6$ s and it kept pushing after that.

5. Conclusion

This paper has proposed an admittance control scheme for joint-level position-controlled robots. The controller has an elaborate discrete-time jerk limiter to limit the third derivative of the position command sent to the robot controller. It effectively suppresses the amplitude of oscillation presumably caused by the deadtime inside the robot controller, especially when the robot is in contact with external environments. The controller was validated with a UR3e robot, which has a 6 ms deadtime in the velocity-command mode.

Future work should address the extension of the controller to a task-space controller. In addition, the clarification of guidelines for tuning the parameters would also be necessary.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP21K04122.

References:

[1] D. A. Lawrence, "Impedance control stability properties in common implementation," in *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, 1988, pp. 1185–1190.

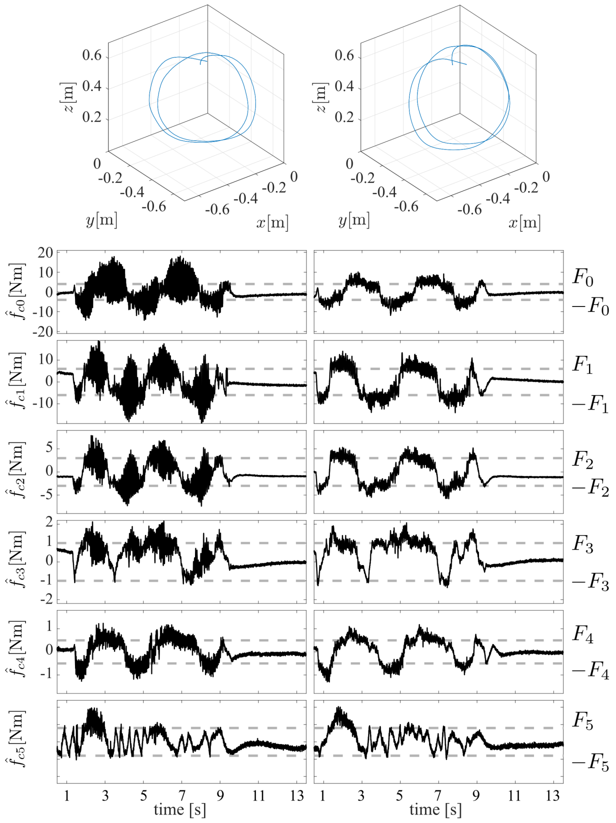
[2] C. T. Landi, F. Ferraguti, L. Sabattini, C. Secchi, M. Bonfe, and C. Fantuzzi, "Variable admittance control preventing undesired oscillating behaviors in physical human-robot interaction," in *Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 3611–3616.

[3] Y. Sun, M. Van, S. McIlvanna, S. McLoone, and D. Ceglarek, "Adaptive admittance control for safety-critical physical human robot collaboration," *arXiv preprint arXiv:2208.05061*, 2022.

[4] T. Sun, Z. Wang, C. He, and L. Yang, "Adaptive robust admittance control of robots using duality principle-based impedance selec-tion," *Applied Sciences*, vol. 12, no. 23, p. 12222, 2022.

[5] C. T. Landi, F. Ferraguti, L. Sabattini, C. Secchi, and C. Fantuzzi, "Admittance control parameter adaptation for physical human-robot interaction," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation*, 2017, pp. 2911–2916.

[6] F. Dimeas and N. Aspragathos, "Online stability in human-robot cooperation with admittance control," *IEEE Transactions on Haptics*, vol. 9, no. 2, pp. 267–278, 2016.



(a) controller cN (b) controller cP

Fig. 10. Results of Experiment III. The experimenter grasped the end-effector and moved the robot along a circle.

[7] B. Yao, Z. Zhou, L. Wang, W. Xu, Q. Liu, and A. Liu, "Sensorless and adaptive admittance control of industrial robot in physical human-robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 51, pp. 158–168, 2018.

[8] G. Kang, H. S. Oh, J. K. Seo, U. Kim, and H. R. Choi, "Variable admittance control of robot manipulators based on human intention," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 1023–1032, 2019.

[9] F. Ferraguti, C. Talignani Landi, L. Sabattini, M. Bonfe, C. Fantuzzi, and C. Secchi, "A variable admittance control strategy for stable physical human–robot interaction," *International Journal of Robotics Research*, vol. 38, no. 6, pp. 747–765, 2019.

[10] Z. Chen, Q. Guo, T. Li, Y. Yan, and D. Jiang, "Gait prediction and variable admittance control for lower limb exoskeleton with measurement delay and extended-state-observer," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[11] C. Liu, Y. He, X. Chen, and H. Cao, "Adaptive enhanced admittance force-tracking controller design for highly dynamic interactive tasks," *Industrial Robot: the international journal of robotics research and application*, 2022.

[12] K. Haninger, M. Radke, A. Vick, and J. Krüger, "Towards high-payload admittance control for manual guidance with environmental contact," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4275–4282, 2022.

[13] R. Kikuuwe, "A sliding-mode-like position controller for admittance control with bounded actuator force," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 5, pp. 1489–1500, 2014.

[14] R. Kikuuwe, "Torque-bounded admittance control realized by a set-valued algebraic feedback," *IEEE Transactions on Robotics*, vol. 35, no. 5, pp. 1136–1149, 2019.

[15] M. T. S. Aung and R. Kikuuwe, "Stability enhancement of admittance control with acceleration feedback and friction compensation," *Mechatronics*, vol. 45, pp. 110–119, 2017.

[16] Y. Aydin, O. Tokatli, V. Patoglu, and C. Basdogan, "Stable physical human-robot interaction using fractional order admittance control," *IEEE Transactions on Haptics*, vol. 11, no. 3, pp. 464–475, 2018.

[17] Y. Aydin, O. Tokatli, V. Patoglu, and C. Basdogan, "Fractional order admittance control for physical human-robot interaction," in *2017 IEEE World Haptics Conference (WHC)*, 2017, pp. 257–262.

[18] K. Li, Y. He, K. Li, and C. Liu, "Adaptive fractional-order admittance control for force tracking in highly dynamic unknown environments," *Industrial Robot*, vol. 50, no. 3, pp. 530–541, 2023.

[19] A. Morbi and M. Ahmadi, "Safely rendering small impedance in admittance-controlled haptic devices," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1272–1280, 2016.

[20] A. Lecours and C. Gosselin, "Computed-torque control of a four-degree-of-freedom admittance controlled intelligent assist device," in *Experimental Robotics*, ser. Springer Tracts in Advanced Robotics, J. Desai, G. Dudek, O. Khatib, and V. Kumar, Eds. Springer, 2013, vol. 88, pp. 635–649.

[21] R. Kikuuwe and B. Brogliato, "A new representation of systems with frictional unilateral constraints and its Baumgarte-like relaxation," *Multibody System Dynamics*, vol. 39, no. 3, pp. 267–290, 2017.

[22] B. Brogliato, A. Daniilidis, C. Lemaréchal, and V. Acary, "On the equivalence between complementarity systems, projected systems and differential inclusions," *Systems & Control Letters*, vol. 55, no. 1, pp. 45–51, 2006.

[23] V. Acary and B. Brogliato, *Numerical Methods for Nonsmooth Dynamical Systems: Applications in Mechanics and Electronics*, ser. Lecture Notes in Applied and Computational Mechanics. Springer, 2008, vol. 35.

[24] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 2nd ed. Springer, 2016.

[25] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River: Prentice Hall, 2002.

[26] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1990.

Appendix A. Derivation from (9) to (15)

Eq. (15) can be derived from (9) as follows:

$$
\begin{aligned}
(9) \iff\ & j_k \in j_k^* - \mathcal{N}_{\mathcal{V}}(T(Tj_k + a_{k-1}) + v_{k-1} \\
& \quad + P(Tj_k + a_{k-1})) \\
& \quad - \mathcal{N}_{\mathcal{A}}(Tj_k + a_{k-1}) \\
& \quad - \mathcal{N}_{\mathfrak{J}(Tj_k + a_{k-1})}(j_k) && \text{(35a)} \\
\iff\ & j_k \in j_k^* - \mathcal{N}_{\overline{\mathcal{V}}}(j_k) - \mathcal{N}_{\overline{\mathcal{A}}}(j_k) \\
& \quad - \mathcal{N}_{\overline{\mathfrak{J}}(a_{k-1})}(j_k) && \text{(35b)} \\
\iff\ & j_k \in j_k^* - \mathcal{N}_{\overline{\mathcal{V}} \cap \overline{\mathcal{A}} \cap \overline{\mathfrak{J}}(a_{k-1})}(j_k) && \text{(35c)} \\
\iff\ & j_k \in j_k^* - \mathcal{N}_{\mathcal{L}_k}(j_k) && \text{(35d)} \\
\iff\ & (15) && \text{(35e)}
\end{aligned}
$$

where $\overline{\mathcal{V}}$, $\overline{\mathcal{A}}$, and $\overline{\mathfrak{J}}(a_{k-1})$ are closed intervals defined as

$$
\begin{aligned}
\overline{\mathcal{V}} \triangleq \Bigg[& \frac{-V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT}, \\
& \frac{V - v_{k-1} - (T+P)a_{k-1}}{T^2 + PT} \Bigg] && (36)
\end{aligned}
$$

$$
\overline{\mathcal{A}} \triangleq \left[\frac{-A - a_{k-1}}{T}, \frac{A - a_{k-1}}{T} \right] \qquad (37)
$$

$$
\begin{aligned}
\overline{\mathfrak{J}}(a_{k-1}) \triangleq \Bigg[& \mathrm{proj}_{[-J,-J_y]}\left(\frac{-Ha_{k-1} - J_s}{1 + HT} \right), \\
& \mathrm{proj}_{[J_y,J]}\left(\frac{-Ha_{k-1} + J_s}{1 + HT} \right) \Bigg]. && (38)
\end{aligned}
$$

Here, the derivation from (35a) to (35b) can be explained by the following equivalences:

$$
j_k \in \mathfrak{J}(Tj_k + a_{k-1})
$$

$$
\begin{aligned}
&\iff \mathrm{proj}_{[-J,-J_y]}(-J_s - Ha_{k-1} - HTJ_k) \\
&\qquad \le j_k \le \mathrm{proj}_{[J_y,J]}(J_s - Ha_{k-1} - HTJ_k) \\
&\iff \Big((-J \le j_k) \wedge \big((-J_y \le j_k) \\
&\qquad \vee (-J_s - Ha_{k-1} - HT\,j_k \le j_k)\big)\Big) \\
&\qquad \wedge \Big((J \ge j_k) \wedge \big((J_y \ge j_k) \\
&\qquad \vee (J_s - Ha_{k-1} - HT\,j_k \ge j_k)\big)\Big) \\
&\iff \Big((-J \le j_k) \wedge \Big((-J_y \le j_k) \\
&\qquad \vee \Big(\frac{-J_s - Ha_{k-1}}{1+HT} \le j_k\Big)\Big)\Big) \\
&\qquad \wedge \Big((J \ge j_k) \wedge \Big((J_y \ge j_k) \\
&\qquad \vee \Big(\frac{J_s - Ha_{k-1}}{1+HT} \ge j_k\Big)\Big)\Big) \\
&\iff \mathrm{proj}_{[-J,-J_y]}\Big(\frac{-J_s - Ha_{k-1}}{1+HT}\Big) \\
&\qquad \le j_k \le \mathrm{proj}_{[J_y,J]}\Big(\frac{J_s - Ha_{k-1}}{1+HT}\Big) \\
&\iff j_k \in \overline{\mathcal{J}}(a_{k-1}). \qquad (39)
\end{aligned}
$$

The equivalence between (35c) and (35d) can be easily seen by $\mathcal{L}_k = \overline{\mathcal{V}} \cap \overline{\mathcal{A}} \cap \overline{\mathcal{J}}(a_{k-1})$ by the definitions in (22), (36), (37), and (38).