

Virtual semi-active damping learning control for robot manipulators interacting with unknown environment

Journal of Vibration and Control
2020, Vol. 0(0) 1–12
© The Author(s) 2020
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/1077546320966430
journals.sagepub.com/home/jvc


Wenrui Wang^{1,2}, Ang Li^{1,2}, Qinwen Li^{1,2}, Jinlin Gu^{1,2},
Qi Huo¹, Mingchao Zhu¹, Yanhui Li¹ and Hairong Chu¹

Abstract

Position controllers are used for free motion, whereas force controllers are used for constrained motion of robotic manipulators. The hybrid controller switches between position and force control modes depending on whether the manipulator is in contact with the environment. To improve production efficiency, the velocity of contact between the manipulator and environment is not set to zero. However, the high impact force due to the nonzero contact velocity might damage the environment surface or manipulators. In this article, we propose a virtual semi-active damping learning method to suppress force overshoot without decreasing the contact velocity. Virtual semi-active damping is adjusted according to the manipulator position in force control. The limited-memory BFGS method is used to obtain the ideal impedance model for the unknown environment. By minimizing the defined cost function, we get the desired interaction performance. The correctness and effectiveness of the proposed method are verified by conducting simulations and experiments.

Keywords

Hybrid control, semi-active damping, impedance learning, limited-memory BFGS method, unknown environment, cost function

1. Introduction

Robots are required to interact within different environments as they are being more widely deployed in social applications, such as elderly care, medical rehabilitation, and human–robot cooperation. In these applications, although the environment is usually unknown, the responsiveness of the process is required to complete contact with the environment and achieve the desired contact performance.

In some situations, only one degree of freedom (DOF) of the manipulator end effector is affected in contact with the environment. For this contact DOF, a hybrid control strategy is used to perform the task of switching motion and force tracking. One way to avoid bouncing and damaging the environment surface is to command the robot to slow down as it approaches a surface (Carloni et al., 2007; Markus et al., 2016), resulting in the velocity tending to zero as contact occurs. Another way is to redesign a compliant wrist (Heck et al., 2016; Mohammad et al., 2018). However, these strategies either suffer from the drawback of a slow response and long transient time, which negatively affect the production efficiency, or require a special terminal structure of the manipulator, which is not universally

applicable. Although nonlinear damping control schemes can be used in robotic force control to reduce the force overshoot (Lai et al., 2012; Lai, 2014), the accuracy of position tracking is rarely considered for force control methods especially in free motion. A motion and force hybrid controller is adopted in this article. Furthermore, virtual semi-active damping is added to the controller to ensure a smooth transition from trajectory tracking to force tracking without compromising on velocity.

An impedance model is usually used to describe the interaction model, and the robustness and feasibility of impedance control are also guaranteed (Li et al., 2018; Nozaki et al., 2018; Sharifi et al., 2018). For impedance control, obtaining optimal environmental parameters is

¹Changchun Institute of Optics Fine Mechanics and Physics, Chinese Academy of Sciences, China

²University of Chinese Academy of Sciences, China

Received: 26 February 2020; accepted: 24 September 2020

Corresponding author:

Mingchao Zhu and Yanhui Li, Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China.
Email: zhumingchao@ciomp.ac.cn, liyanhui-1986@163.com

a challenge but important in ensuring high contact performance. To compensate for environment uncertainties, an adaptive control algorithm is applied to tune the impedance parameters (Arefinia et al., 2020; Li et al., 2017). Iterative learning impedance methods are also adopted to obtain the desired impedance model with better performance (Fernando et al., 2019; Li et al., 2018). Impedance parameters are modulated by learning neural networks, although this requires expensive data preprocessing to train samples (He and Dong, 2017). Reinforcement learning methods, such as the PI2 (policy improvement with path integrals) algorithm (Buchli et al., 2011) and iterative linear-quadratic-Gaussian (Luo et al., 2019), make it possible to execute various contact tasks in varying environments, by applying impedance control to robots. The limited-memory BFGS (L-BFGS) method, used in this article, is a quasi-Newton method that overcomes the disadvantages of Hessian matrix irreversibility in the Newton method and the difficulty of selecting the step size in the gradient descent method (Berahas and Tak, 2020; Philipp et al., 2016).

In contrast with controllers described in the articles mentioned above, the virtual semi-active damping controller not only guarantees the accuracy of switching motion and force tracking but also improves the productivity of processes without redesigning the end effector of the manipulator and minimizing the contact velocity. Moreover, different from the above learning methods, the proposed L-BFGS impedance learning method is based on a clear environmental dynamic model and has a simpler framework, allowing practical implementation. Regardless of the control strategy or the acquisition of environmental parameters, this article adopts a simpler and easier solution that is more implementable in actual production and life.

Conceptually, the semi-active damping method requires additional elements in mechanical design but has considerably higher bandwidth and spends less energy than active damping (Matteo et al., 2010). Virtual semi-active damping without mechanical elements is different from actual physical damping. It adjusts the damping according to the deviation between the actual position and desired position in constrained motion. In contact, the desired contact force corresponds to the virtual desired position owing to the effects of environmental impedance parameters. We therefore acquire the environmental impedance parameters and adjust the virtual damping synchronously by using a learning method based on L-BFGS. The damping is at maximum, when the position control is changed to force control, that is, when the manipulator just touches the environment surface. Moreover, it is possible to suppress bounce and prevent damage to the environment surface by modulating the semi-active damping properly.

The remainder of this article is organized as follows. In Section 2, we describe the system modeling and controller design, offering a model of contact between the manipulator and environment and designing a switching motion-force

controller. In Section 3, we present the method of virtual semi-active damping control introducing variable damping to the force controller to suppress chattering and force overshoot. We discuss details of the proposed impedance learning based on L-BFGS in Section 4. In Sections 5, simulation and experiments are conducted to verify the correctness and feasibility of the proposed method. We present conclusions in Section 6.

2. System modeling and controller design

In most cases, when the manipulator interacts with the environment, just one DOF of the end effector of the manipulator is affected. Considering the simplified interaction model as shown in Figure 1, the dynamics equation is expressed as

$$M\ddot{x} + b\dot{x} + F_e = F_c \quad (1)$$

where x is the manipulator position, M is the mass of the manipulator, and b is viscous friction acting in the joint (To dissipate energy, viscous friction b is generally set to zero).

The surface of the environment is at the position of $x = 0$, when the equivalent spring is at rest. Therefore, once the manipulator makes contact with the environment, the position of the manipulator is $x > 0$. The Kelvin–Voigt linear contact model is used to characterize the relationship between the penetration and reaction force (Diolaiti et al., 2005)

$$f_e = \begin{cases} 0 & \text{if } x \leq 0 \\ k_e x + b_e \dot{x} & \text{if } x > 0 \end{cases} \quad (2)$$

where k_e and b_e are, respectively, the stiffness and damping of the environment.

The aim of the present work is to control the manipulator such that it follows a desired trajectory $x_d(t)$ in free motion and to regulate a desired force profile $F_d(t)$ in the contact phase. The simplest strategy that accomplishes the described task is to switch between a position controller and

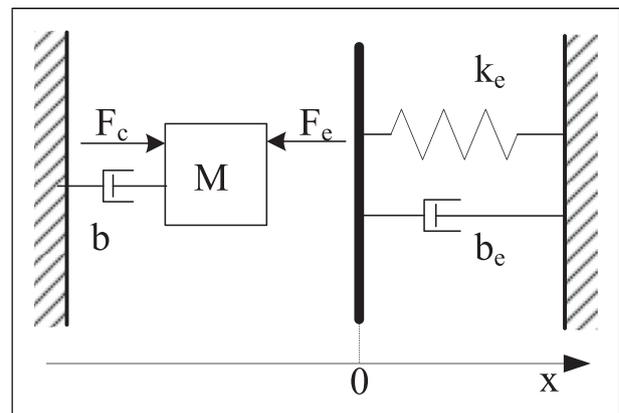


Figure 1. 1-degree-of-freedom model.

force controller. The switch of controller is realized by detecting whether the manipulator is in contact with the environment, that is, the controller is a position controller when $x < 0$ and a force controller otherwise. The two controllers are a resolved acceleration controller and proportional force controller (Heck et al., 2016)

$$F_c = \begin{cases} M\ddot{x}_d(t) + k_{p1}\Delta x + k_d\Delta\dot{x}, & \forall x < 0 \\ F_d(t) + k_{p2}\Delta F - b_f\dot{x}, & \forall x \geq 0 \end{cases} \quad (3)$$

where k_{p1} and k_{p2} are the proportional gains of the motion controller and force controller. k_d is the derivative gain of the motion controller, b_f is the damping gain, dissipating energy during the contact phase, and Δx and ΔF are, respectively, the position error and force error. The contact force F_e , position x , and velocity \dot{x} can be measured by sensors.

This closed-loop switching system cannot prevent the bounce of the manipulator in contact with a rigid environment, unless a high damping controller is used during contact. To this end, we propose a virtual semi-active damping control method in the next section. In the force control phase, the damping term of the controller is replaced by virtual semi-active damping, which not only restrains the bounce but also saves energy.

3. Virtual semi-active damping control method

3.1. Motivation and design

In the contact phase, the contact force F satisfies the relationship described in (2) as $x > 0$. Corresponding to the desired contact force F_d , the virtual desired position x_{fd} should satisfy the relation

$$k_e x_{fd} + b_e \dot{x}_{fd} = F_d \quad (4)$$

We here only consider the simple case of a manipulator tracking a constant force. The velocity of the manipulator is $\dot{x}_{fd} = 0$ in the steady state for a constant desired force; otherwise, the system oscillates. Equation (4) is thus rewritten as

$$k_e x_{fd} = F_d \quad (5)$$

Therefore, we assume that there is also a desired x_{fd} , corresponding to the desired contact force F_d in the contact phase, where $x_{fd} = F_d/k_e$. The manipulator is stationary at position x_{fd} in the steady state as shown in Figure 2.

We consider a semi-active damper based on the position error (Stegall et al., 2017) as shown in Figure 3. The damping prevents the manipulator from moving away from the desired position, which means that a stronger damping force acts further from the desired position. We therefore define semi-active damping as

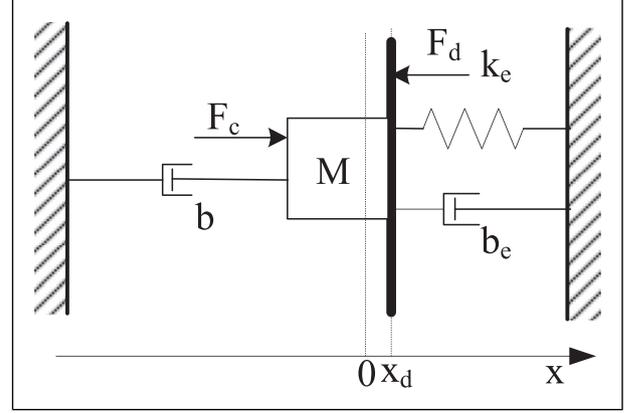


Figure 2. Steady state in contact.

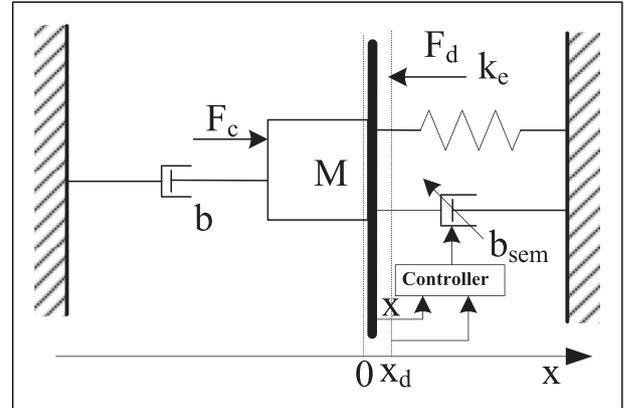


Figure 3. Contact model with virtual semi-active damping.

$$b_{sem} = b_e + b_v \quad (6)$$

where b_v is the virtual damping that we modulate actively, b_v is expressed as

$$b_v = \begin{cases} \min[\lambda|x - x_d|, b_{v,max}] & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (7)$$

$\lambda > 0$ is the gain coefficient of damping and position error, whereas $b_{v,max}$ is the saturation point for the damping coefficient. The model of contact between the manipulator and environment is rewritten as

$$M\ddot{x} + b\dot{x} + kx + b_e\dot{x} + b_v\dot{x} = F_c \quad (8)$$

b_v is introduced actively according to the position error. We can take it as the active control input, realizing semi-active damping by the switching controller

$$F_c = \begin{cases} M\ddot{x}_d + k_{p1}\Delta x + k_d\Delta\dot{x} & \forall x < 0 \\ F_d(t) + k_{p2}\Delta F - (b_f + \lambda|x - x_{fd}|)\dot{x} & \forall x \geq 0 \end{cases} \quad (9)$$

For this improved switching controller, the semi-active damping comprises b_f and b_v , where b_f is the fixed damping gain, whereas $b_v = \lambda|x - x_{fd}|$ is the active damping gain simulated by position feedback. Without considering that the contact phase x is much larger than x_{fd} , the damping is a maximum when $x = 0$, that is, when the robot has just touched the environment. Large damping could suppress force overshoot. We could change the value of λ within an allowable range according to actual needs to adjust the peak value and rate of change of damping to better suppress the overshoot.

3.2. Stability analysis

Two Lyapunov functions, corresponding to the position controller and force controller, are defined to analyze system stability. In the position control phase

$$V_p = \frac{1}{2}M\Delta x^2 + \frac{1}{2}k_{p1}\Delta x^2 \quad (10)$$

which is strictly positive in free motion with $x < 0$. In the force control case, the candidate Lyapunov function can be chosen as

$$\begin{aligned} V_f &= \frac{1}{2}M\dot{x}^2 + M\Delta x\dot{x} + \frac{1}{2}(k_{p2} + 1)\Delta x^2 \\ &\quad + \frac{1}{2}(b_f + \lambda|x - x_{fd}|)\Delta x^2 + k_{p2}\Delta F\Delta x + \frac{1}{2}k_{p2}\Delta F^2 \\ &= \frac{1}{4}M\dot{x}^2 + \frac{1}{4}M(\dot{x} + 2\Delta x)^2 + \frac{1}{2}\begin{bmatrix} \Delta x \\ \Delta F \end{bmatrix} \\ &\quad \times \begin{bmatrix} (k_{p2} + 1) + b_f + \lambda|x - x_{fd}| - 2M & k_{p2} \\ k_{p2} & k_{p2} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta F \end{bmatrix} \end{aligned} \quad (11)$$

where positive controller gains are chosen to satisfy

$$1 + b_f + \lambda|x - x_{fd}| - 2M > 0 \quad (12)$$

so that V_f is strictly positive in constrained motion with $x \geq 0$.

In the case of position control, the time derivative of (10) is

$$\dot{V}_p = -k_d\Delta \dot{x}^2 \quad (13)$$

which is negative semi-definite with $x < 0$. It is obvious that the system is in a unique asymptotically stable equilibrium if the motion is free without contact and switching.

The derivative of (11) is

$$\dot{V}_f = (M - (b_f + \lambda|x - x_{fd}|))\dot{x}^2 - \Delta \dot{x}^2 \quad (14)$$

which is negative definite if $M - (b_f + \lambda|x - x_{fd}|) \leq 0$. Hence, if the manipulator does not lose contact with the environment, then the system is in an asymptotically stable equilibrium with $x > 0$. Both position control and force

control are stable in the absence of mode switching. However, mode switching due to bouncing is inevitable in such physical systems, and we thus need to consider the overall stability of this hybrid system.

Switching between the two controllers occurs in the state space (Zoe, and Iliadis, 2005)

$$\begin{aligned} \mathbf{S}_{p,f} &= \mathbf{S}_{k(k=2i+1)}(i = 0, 1, 2, \dots) \\ &= \{(x, \dot{x}) \in \mathfrak{R}^2 : x = 0 \text{ and } \dot{x} \geq 0\} \end{aligned} \quad (15)$$

and

$$\begin{aligned} \mathbf{S}_{f,p} &= \mathbf{S}_{k(k=2i)}(i = 1, 2, 3, \dots) \\ &= \{(x, \dot{x}) \in \mathfrak{R}^2 : x = 0 \text{ and } \dot{x} \leq 0\} \end{aligned} \quad (16)$$

where k is the number of switches, $\mathbf{S}_{p,f}$ means switching from the position controller to force controller, and $\mathbf{S}_{f,p}$ means switching from the force controller to position controller.

To conclude the closed-loop stability of system (9) with switching sequences \mathbf{S} , necessary and sufficient conditions must be imposed on stability analysis (Branicky, 1998; Pettersson and Lennartson, 1996), as given in Theorem 1.

Theorem 1. For the hybrid system (9), if there exist continuous scalar functions with continuous partial derivatives V_k , then there are class K functions $\alpha > 0$ and $\beta > 0$ such that

1. $\alpha(\|x\|) \leq V_k(x) \leq \beta(\|x\|)$
2. $\dot{V}_k \leq 0$
3. $V_p(x) \leq V_f(x)$ at $S_{f,p}$
4. $V_k(x(t_{k+1})) \leq V_k(x(t_k))$

where (3) indicates that the equilibrium point $x = 0$ of (9) is stable in the sense of Lyapunov, (4) concerns the value of V_k , which is less at switching points (at $S_{f,p}$ or $S_{p,f}$) than the last time a switch was made to the same control mode, and t_k is the switching time. Items (3) and (4) are given in Appendix 1 and are obviously true for the hybrid system (9). Items (1) and (2) are, respectively, proven in position control and force control modes. Therefore, irrespective of whether there are an infinite number of switches or a finite number of switches, the Lyapunov functions of the system are strictly decreasing each time they are switched in, and the hybrid system is therefore asymptotically stable.

4. Impedance learning

4.1. Impedance learning based on the BFGS method.

Only when the manipulator makes contact with the environment will the environment impedance work. The main control target during contact is the contact force. Therefore,

to develop impedance learning, we define a cost function with which to measure contact performance

$$J = (F - \tilde{k}x - \tilde{b}\dot{x})^2 \quad (17)$$

where \tilde{k} and \tilde{b} are, respectively, the estimated values of environmental stiffness and impedance, F , whereas x and \dot{x} represent the force, position, and velocity measured by sensors. We need to solve $\min J(\tilde{k}, \tilde{b})$ to achieve better contact performance. Unconstrained optimization variable metric algorithms are adopted for such iterative problems. Letting $\mathbf{x} = [x \ \dot{x}]$ and $\mathbf{z} = [\tilde{k} \ \tilde{b}]^T$, the cost function is rewritten as

$$J = (F - \mathbf{z}\mathbf{x})^2 \quad (18)$$

The position and velocity of the manipulator are bounded, and we thus assume that J is strongly convex in the iteration interval. Adopting the Taylor expansion of the function at \mathbf{z}_{k+1} and ignoring quadratic and higher terms, we get

$$J(\mathbf{z}) = J(\mathbf{z}_{k+1}) + \nabla J(\mathbf{z}_{k+1})^T (\mathbf{z} - \mathbf{z}_{k+1}) + \frac{1}{2} (\mathbf{z} - \mathbf{z}_{k+1})^T \nabla^2 J(\mathbf{z}_{k+1}) (\mathbf{z} - \mathbf{z}_{k+1}) \quad (19)$$

where

$$\nabla J(\mathbf{z}_{k+1}) = \begin{bmatrix} \frac{\partial J}{\partial k} \\ \frac{\partial J}{\partial b} \end{bmatrix} = \begin{bmatrix} -2x_{k+1}(f - kx_{k+1} - b\dot{x}_{k+1}) \\ -2\dot{x}_{k+1}(f - kx_{k+1} - b\dot{x}_{k+1}) \end{bmatrix} \quad (20)$$

and

$$\nabla^2 J(\mathbf{z}_{k+1}) = \begin{bmatrix} \frac{\partial^2 J}{\partial k^2} & \frac{\partial^2 J}{\partial k \partial b} \\ \frac{\partial^2 J}{\partial k \partial b} & \frac{\partial^2 J}{\partial b^2} \end{bmatrix} = \begin{bmatrix} -2x_{k+1}^2 & -2x_{k+1}\dot{x}_{k+1} \\ -2x_{k+1}\dot{x}_{k+1} & -2\dot{x}_{k+1}^2 \end{bmatrix} \quad (21)$$

Letting $\mathbf{z} = \mathbf{z}_{k+1}$, we get the gradient for (19)

$$\nabla J(\mathbf{z}_{k+1}) - \nabla J(\mathbf{z}_k) = \nabla^2 J(\mathbf{z}_{k+1}) (\mathbf{z}_{k+1} - \mathbf{z}_k) \quad (22)$$

when $\nabla J(\mathbf{z}_k) = 0$

$$\mathbf{z}_{k+1} = \mathbf{z}_k + (\nabla^2 J(\mathbf{z}_{k+1}))^{-1} \nabla J(\mathbf{z}_{k+1}) \quad (23)$$

where $\nabla^2 J(\mathbf{z}_{k+1})$ is the Hessian matrix \mathbf{H} . \mathbf{H} is iteratively updated based on the data measured by sensors. However, the speed must equal to zero during the motion, resulting in \mathbf{H} being irreversible, and preventing us from adopting the Newton method. We introduce the BFGS algorithm, which is a typical implementation of the quasi-Newton method.

We abbreviate (20) as

$$\mathbf{y}_k = \mathbf{H}_{k+1} \mathbf{s}_k \quad (24)$$

where $\mathbf{y}_k = \nabla J(\mathbf{z}_{k+1}) - \nabla J(\mathbf{z}_k)$ and $\mathbf{s}_k = \mathbf{z}_{k+1} - \mathbf{z}_k$. In this case, \mathbf{s}_k can also be expressed as $\mathbf{s}_k = \mathbf{H}_{k+1}^{-1} \mathbf{y}_k$. We construct an approximation matrix for the matrix \mathbf{H}

$$\mathbf{B}_k \approx \mathbf{H}_k \quad (25)$$

where \mathbf{B}_k is updated according to

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \Delta \mathbf{B}_k \quad (26)$$

The initial value of the matrix \mathbf{B}_0 is the identity matrix \mathbf{I} . The problem that we need to solve is to modify the construction of the matrix $\Delta \mathbf{B}_k$ in each iteration. The calculation formula is

$$\Delta \mathbf{B}_k = \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} - \frac{\mathbf{B}_k \mathbf{s}_k \mathbf{s}_k^T \mathbf{B}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k} \quad (27)$$

The next iteration \mathbf{z}_{k+1} is

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \gamma_k \mathbf{d}_k \quad (28)$$

where $\mathbf{d}_k = -\mathbf{B}_k^{-1} \nabla J(\mathbf{z}_k)$ is the search direction and $\gamma_k > 0$ is the step length. Applying the Sherman–Morrison equation, we get the relationship between \mathbf{B}_{k+1}^{-1} and \mathbf{B}_k^{-1} as

$$\begin{aligned} \mathbf{B}_{k+1}^{-1} &= \left(\mathbf{I} - \frac{\mathbf{s}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} \right) \mathbf{B}_k^{-1} \left(\mathbf{I} - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} \right) + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} \\ &= \mathbf{B}_k^{-1} + \left(\frac{1}{\mathbf{s}_k^T \mathbf{y}_k} + \frac{\mathbf{y}_k^T \mathbf{B}_k^{-1} \mathbf{y}_k}{(\mathbf{s}_k^T \mathbf{y}_k)^2} \right) \mathbf{s}_k \mathbf{s}_k^T \\ &\quad - \frac{1}{\mathbf{s}_k^T \mathbf{y}_k} (\mathbf{B}_k^{-1} \mathbf{y}_k \mathbf{s}_k^T + \mathbf{s}_k \mathbf{y}_k^T \mathbf{B}_k^{-1}) \end{aligned} \quad (29)$$

In the implementations of the BFGS algorithm, $\gamma_k > 0$ satisfies the Wolfe conditions (Neculai, 2018)

$$J(\mathbf{z}_k + \gamma_k \mathbf{d}_k) \leq f(\mathbf{z}_k) + \delta_1 \gamma_k \mathbf{d}_k^T \nabla J(\mathbf{z}_k) \quad (30)$$

$$\mathbf{d}_k^T \nabla J(\mathbf{z}_k + \gamma_k \mathbf{d}_k) \geq \delta_2 \mathbf{d}_k^T \nabla J(\mathbf{z}_k) \quad (31)$$

where δ_1 and δ_2 are constants such that $0 < \delta_1 \leq \delta_2 < 1$. When the objective function $J(\mathbf{z})$ is convex and if the Wolfe conditions of an inexact linear search are satisfied, then this algorithm is globally convergent. The pseudo-code of our algorithm is presented as Algorithm 1.

Algorithm 1. L-BFGS method

Input:

Initial iterate $\mathbf{z}_0 \in R^2$, initial step length γ_0 , constants δ_1 and δ_2 , and sufficiently small value ε

Output: \mathbf{z}_k

1. while $\nabla J(\mathbf{z}) > \varepsilon$ do

2. Compute $\mathbf{d}_k = -\mathbf{B}_k^{-1}\nabla J(\mathbf{z}_k)$
3. Find γ_k satisfying the Wolfe line search conditions (30) and (31)
4. Compute $\mathbf{z}_{k+1} = \mathbf{z}_k + \gamma_k\mathbf{d}_k$
5. Compute \mathbf{B}_{k+1}^{-1} according to (29)
6. $k = k + 1$
7. end while

4.2. Convergence analysis

To prove the global convergence of the L-BFGS algorithm, we consider the following proposition. This proposition is an important tool for analyzing the L-BFGS method.

Proposition 1. The selected cost function, $J(\mathbf{z})$, satisfies the following.

1. The objective function $J(\mathbf{z})$ is twice continuously differentiable (Abdi and Shakeri, 2019).
2. The level set $\Omega = \{\mathbf{z} \in \mathbb{R}^2 : J(\mathbf{z}) \leq J(\mathbf{z}_0)\}$ is convex.
3. There exist positive constants M_1 and M_2 such that (Raghu et al., 2018)

$$\mathbf{M}_1 \mathbf{I} \leq \nabla^2 J(\mathbf{z}) \leq \mathbf{M}_2 \mathbf{I}$$

By Proposition 1(3) and $\mathbf{y}_k = \mathbf{H}_{k+1}\mathbf{s}_k$ (Liao, 1997)

$$\mathbf{M}_1 \|\mathbf{s}_k\|^2 \leq \mathbf{y}_k^T \mathbf{s}_k \leq \mathbf{M}_2 \|\mathbf{s}_k\|^2 \quad (32)$$

and

$$\mathbf{M}_1 \leq \frac{\|\mathbf{y}_k\|^2}{\mathbf{y}_k^T \mathbf{s}_k} = \frac{\mathbf{s}_k^T \mathbf{H}_{k+1}^2 \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k} \leq \mathbf{M}_2 \quad (33)$$

From (33) and (34), we estimate the trace of the Hessian approximation

$$\begin{aligned} \text{tr}(\mathbf{B}_{K+1}) &= \text{tr}(\mathbf{B}_K) - \frac{\|\mathbf{B}_k \mathbf{s}_k\|^2}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k} + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} \leq \text{tr}(\mathbf{B}_K) + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k} \\ &\leq \text{tr}(\mathbf{B}_K) + \mathbf{M}_2 \leq \text{tr}(\mathbf{B}_0) + k\mathbf{M}_2 \leq \mathbf{M}_3 \end{aligned} \quad (34)$$

\mathbf{M}_3 is a positive constant. Then, to bound the determinant, we can write

$$\begin{aligned} \det(\mathbf{B}_{K+1}) &= \det(\mathbf{B}_K) \det\left(\mathbf{I} - \frac{\mathbf{s}_k \mathbf{s}_k^T \mathbf{B}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k} + \frac{(\mathbf{B}_k)^{-1} \mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}\right) \\ &= \det(\mathbf{B}_K) \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k} = \det(\mathbf{B}_K) \frac{\mathbf{y}_k^T \mathbf{s}_k}{\|\mathbf{s}_k\|^2} \frac{\|\mathbf{s}_k\|^2}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k} \\ &\geq \det(\mathbf{B}_K) \frac{\mathbf{M}_1}{\text{tr}(\mathbf{B}_K)} \geq \det(\mathbf{B}_K) \frac{\mathbf{M}_1}{\mathbf{M}_3} \end{aligned} \quad (35)$$

From (35) and (36), we conclude that there is a constant $\delta > 0$ such that

$$\cos \theta_k \equiv \frac{\mathbf{s}_k \mathbf{B}_k \mathbf{s}_k^T}{\|\mathbf{s}_k\| \|\mathbf{B}_k \mathbf{s}_k\|} \geq \delta \quad (36)$$

Theorem 2. Let \mathbf{z}_0 and \mathbf{B}_0 be the initial iterations, such that $J(\mathbf{z})$ satisfies Proposition 1 and $\{\|\mathbf{B}_k\|\}$ is bounded. Algorithm 1 then generates a sequence $\{\mathbf{z}_k\}$ that converges to \mathbf{z}^* . Furthermore, there is a constant r $0 \leq r < 1$ such that (30)

$$J_k - J_* \leq r^k (J_0 - J_*)$$

According to the line search conditions, (30) and (31), and as Proposition 1 implied, we have (Liu and Nocedal, 1989)

$$J(\mathbf{z}_{k+1}) - J(\mathbf{z}_*) \leq (1 - \alpha \cos^2 \theta_k) (J(\mathbf{z}_k) - J(\mathbf{z}_*)) \quad (37)$$

where α is a constant $\alpha > 0$. Moreover, we have

$$\frac{1}{2} \mathbf{M}_1 \|\mathbf{z}_k - \mathbf{z}_*\|^2 \leq J_k - J_* \quad (38)$$

which combined with Theorem 2, indicates the sequence $\{\mathbf{z}_k\}$ is also R-linearly convergent.

5. Simulations and experiments

5.1. Simulations

In this section, we illustrate the function of the proposed virtual semi-active damping control algorithm through simulations. We consider a one-DOF manipulator with $M = 1$ kg interacting with an environment with $k_e = 10^4$ N/m and $b = 20$ Ns/m. The initial position of the manipulator is -1 m. The desired trajectory of the manipulator is $x_d = t^2 - 1$ in free motion, and the desired force is 10N in the contact phase where $x \geq 0$.

We first design the controller as (3) without active damping. For the controller parameters, we choose $M = 1$, $k_{p1} = 5000$, $k_d = 100$, $k_{p2} = 100$, and $b_f = 1$, satisfying (12) and (43). In this way, the manipulator tracks x_d in free motion, but owing to the nonzero velocity of contact with the stiff environment, there is a large peak force at the moment of contact (see the second plot in Figure 4). The stronger force causes backward motion and the manipulator then breaks contact from the environment at about 1.03 s (see the first plot in Figure 4). Although the manipulator recovers and maintains contact after 1.1 s, it continues to oscillate slightly at x_{fd} . Even if we do not account for the fact that strong peak forces can damage the manipulator or the surface of the environment, oscillation adversely affects force tracking.

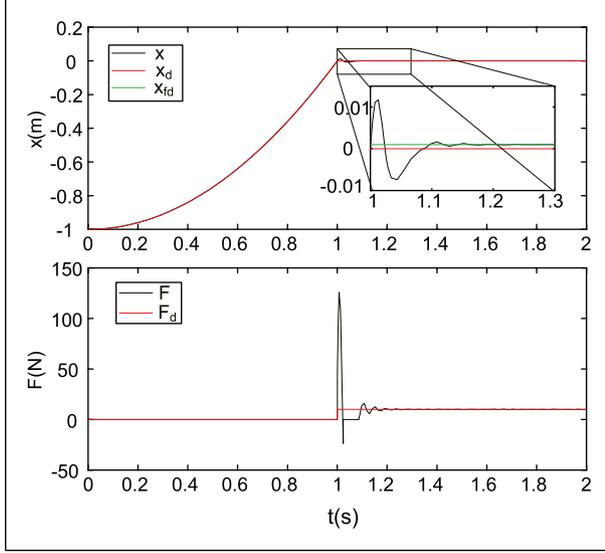


Figure 4. Simulation results of controller (3) with $b_f = 1$.

We then redesign the controller (3) on the basis of Figure 4. The problem is the lack of damping in the contact phase. Therefore, we increase damping in the force controller and set $b_f = 900$ without changing other parameters. With $b_f = 900$, the manipulator does not bounce against the environment (see the first plot in Figure 5) and the peak force decreases appreciably (see the second plot in Figure 5). However, we find that the force and trajectory converge to desired values in 1.2 s as shown in Figure 5. This not only results in greater energy consumption but is also unacceptable for high-precision tasks.

We finally design the virtual semi-active damping controller by setting $b_f = 1$ and picking an appropriate λ according to (9). In addition, the choice of λ should satisfy conditions for system stability

$$\lambda |x - x_{fd}| \geq M - b_f$$

$$\lambda x_{fd} \geq k_{p1} - (k_{p2} + 1 + b_f + 2k_{p2}k_e + k_{p2}k_e^2)$$

It is obvious that the right-hand side of the inequality is not more than zero, whereas x_{fd} and $|x - x_{fd}|$ are greater than or equal to zero. The system is stable so as long as $\lambda > 0$. Meanwhile λ should be set to a larger value to inhibit bounce and to accelerate convergence. By adjusting the value of λ , we ensure that the peak force does not exceed 40N (i.e., the peak force in Figure 5). Furthermore, the trajectory and force converge quickly to the desired position x_{fd} and the desired force F_d . In practice, the environment impedance parameters are always unknown such that x_{fd} cannot be obtained either. It is necessary to introduce the impedance learning algorithm to realize the design of a virtual semi-active damping controller.

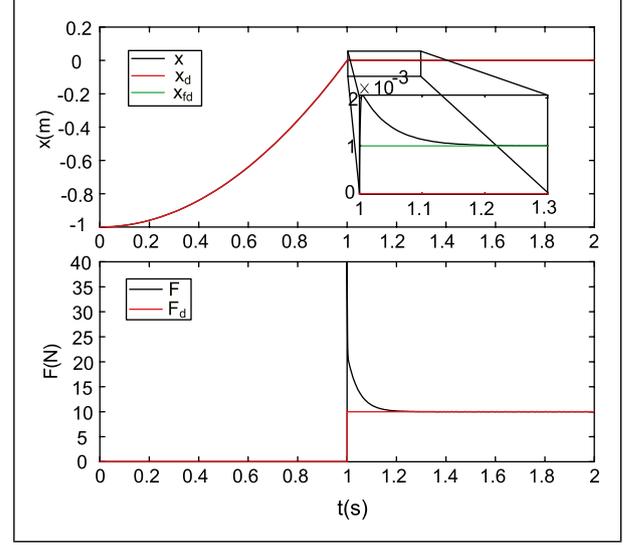


Figure 5. Simulation results of controller (3) with $b_f = 900$.

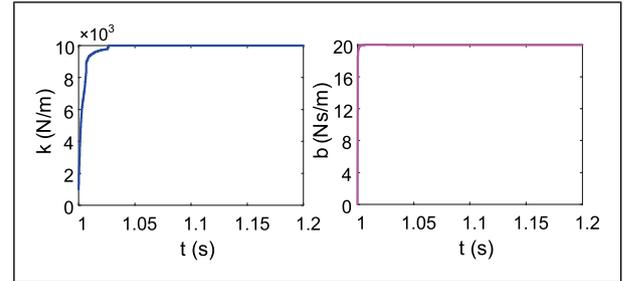


Figure 6. Update of stiffness k and damping b with the impedance learning algorithm.

To verify the validity of the impedance learning algorithm based on the BFGS method, we apply the algorithm to the virtual semi-active damping controller. Here, the desired position x_{fd} is not directly obtainable because of the unknown environment impedance parameters, and it needs to be updated with the impedance parameter. In the impedance learning algorithm, we set $z_0 = [1000 \ 0]^T$, $\delta_1 = 0.1$, $\delta_2 = 0.5$, and $\varepsilon = 10^{-3}$. The environment impedance parameters, including k and b , are updated iteratively and they converge to fixed values, $k = 10,000 \text{ N/m}$ and $b = 20 \text{ Ns/m}$, at 1.03s (see Figure 6). We find that they are the same as the environment parameters that we set. At the same time, the desired position x_{fd} converges to 0.001 at 1.03 s (see Figure 7) when calculating $x_{fd} = F_d/k$. Through the impedance learning algorithm, environment impedance parameters can be quickly and accurately estimated; thus, x_{fd} is quickly obtained. Moreover, the motion and force tracking results are almost the same as those of the virtual semi-active damping controller with the known environment (see Figure 8). Therefore, the learning algorithm not only allows

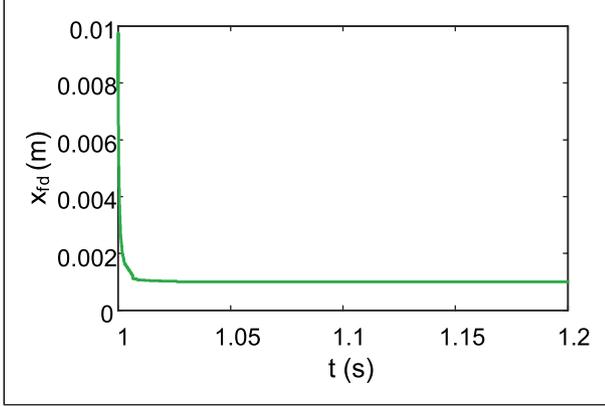


Figure 7. Modification of desired position x_{fd} with impedance parameter update.

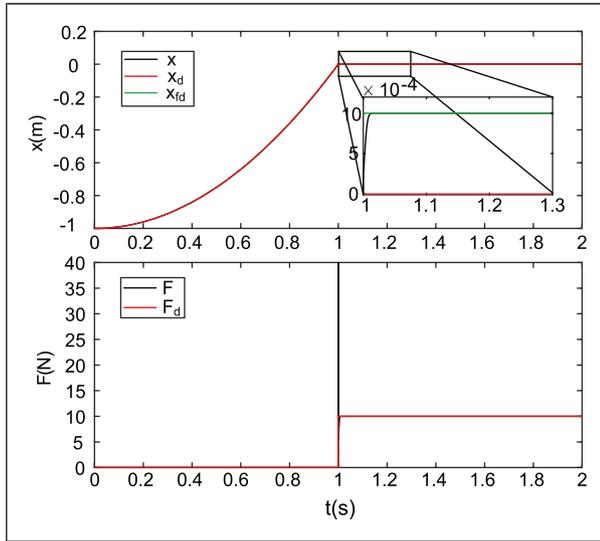


Figure 8. Simulation results of controller (9).

us to obtain an accurate impedance model but also guarantees the performance of virtual semi-damping control.

A comparison of the above three simulation results reveals that the force overshoot in Figure 4 is much greater than those in the other two simulations when contact is made with the environment at the same speed; however, the response speed in Figure 5 is much higher than that in Figure 8. In short, in the case of obtaining environmental parameters adopting the L-BFGS impedance learning method, the virtual semi-active damping algorithm proposed in this article ensures a fast response of the system and suppresses the force overshoot without reducing the contact speed.

5.2. Experiments

In this section, the virtual semi-active damping learning control is further validated using a UR10 that makes contact

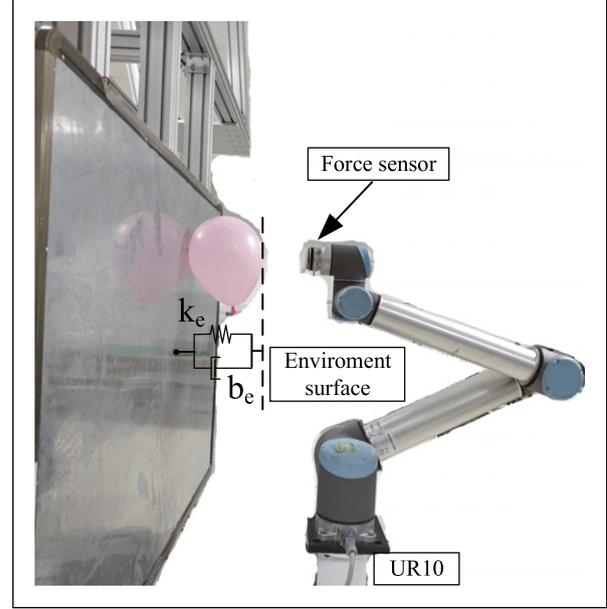


Figure 9. Experimental setup of contact with a balloon.

with environments having different impedance parameters. A force sensor is installed at the end of the UR10 to measure contact forces with the environment. The type of the force sensor is an OptoForce HEX-70-XE-200N having a force acquisition period of 0.01 s. The control period cannot be shorter than the force acquisition period, and we adopt a control period of 0.01 s. In our experiments, the UR10 robot is required to move at a speed of 0.03 m/s and switch to the force control mode after contact with the environment. The desired forces are set differently in environments having different impedance parameters to clarify the experimental results.

In the first experiment, the UR10 robot is required to make contact with a balloon (see Figure 9), where the desired contact force is set to $F_d = 10\text{N}$ and the position of the environment surface is set to $x = 0$. For the UR10 robot, we adopt the speed control mode based on the “speed” instruction. In this mode, the acceleration is calculated as $\ddot{x} = (F_c - F_e)/M$ and the velocity is obtained in the control period. To suppress higher acceleration in the contact phase, we select the value of M to be greater than the actual mass of the force sensor, that is, we assume that the robot end is equipped with a larger mass of the end effector. In this way, we choose $M = 10$, $k_{p1} = 1000$, $k_d = 10$, and $k_{p2} = 10$. As for b_f and λ , the controller is (3) if we set $\lambda = 0$ and (9) otherwise. We carry out the experiments by selecting different values of b_f and λ . The experimental results are shown in Figure 10. Note that the position and interaction forces when $\lambda = 0$ and $b_f = 50$ in Figure 10 have not converged at the observation time; therefore, in practical application, we need to select a larger intrinsic damping gain to accelerate the convergence as the results when $b_f = 500$. It is also found that the

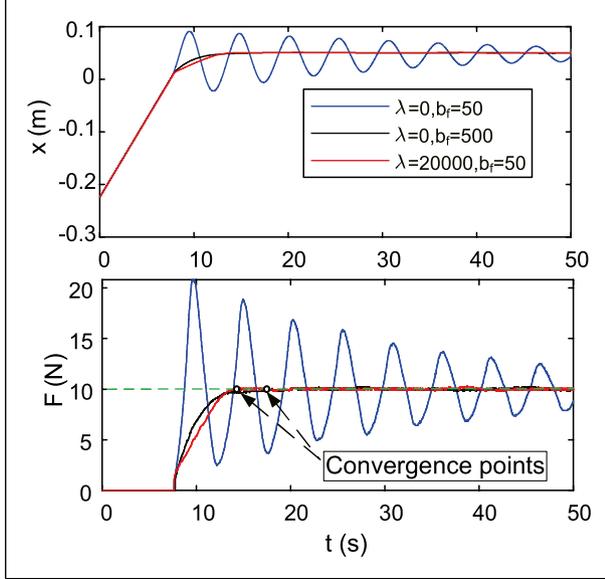


Figure 10. Experiment results of contact with a balloon.

force converges to the expected value earlier when $\lambda = 20,000$ and $b_f = 50$ through the comparison of convergence points, and we consider that the virtual semi-damped controller has faster convergence than the large damped controller in practical applications, which are consistent with the results of simulation. x_{fd} is calculated as $x_{fd} = F_d/k_e$, and the environment parameter k_e is obtained adopting the impedance learning algorithm based on the L-BFGS method. For the impedance learning algorithm, the input parameters are set as $z_0 = [50 \ 0]^T$, $\delta_1 = 0.1$, $\delta_2 = 0.5$, and $\varepsilon = 0.1$. Because the environment damping b_e is not used in the present controller design, we focus only on the environment stiffness k_e and the desired position x_{fd} in the contact phase (see Figure 11). We find that x_{fd} , corresponding to the real environment stiffness, is obtained before the convergence point by comparing with the second plot in Figure 10.

In the second experiment, the UR10 robot is required to make contact with an aluminum plate (see Figure 12), where the desired contact force is set to $F_d = 50N$. In this experiment, except for b_f and λ , the selected control parameters are consistent with parameters in the first experiment; however, b_f needs to be selected as a larger value to ensure and accelerate the system convergence. The experimental results for different values of b_f and λ are shown in Figure 13. The force overshoots are the same for $\lambda = 0$, $b_f = 1800$ and $\lambda = 500,000$, $b_f = 400$, but the virtual semi-active controller has faster convergence although the difference is obvious (see Figure 13). Even if the robot is in contact with such a high stiffness environment, the learning algorithm designed in this article ensures that the environment stiffness can be obtained accurately before the force control convergence (see Figure 14). Without changing other

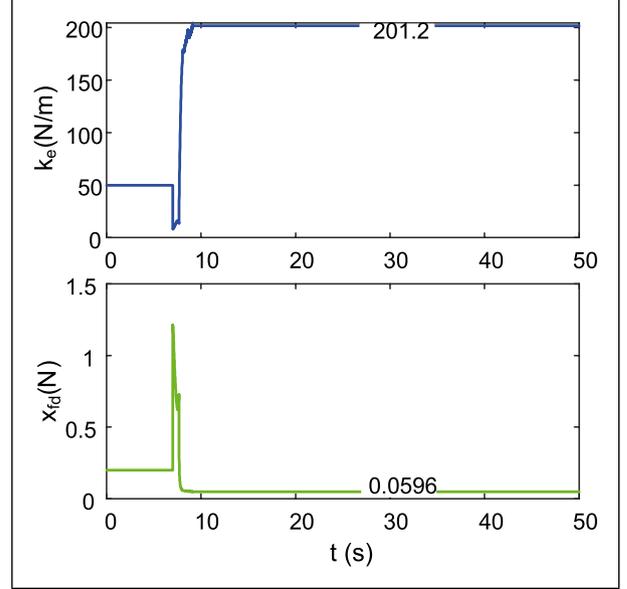


Figure 11. Impedance learning results and x_{fd} profiles with environment stiffness in the first experiment.

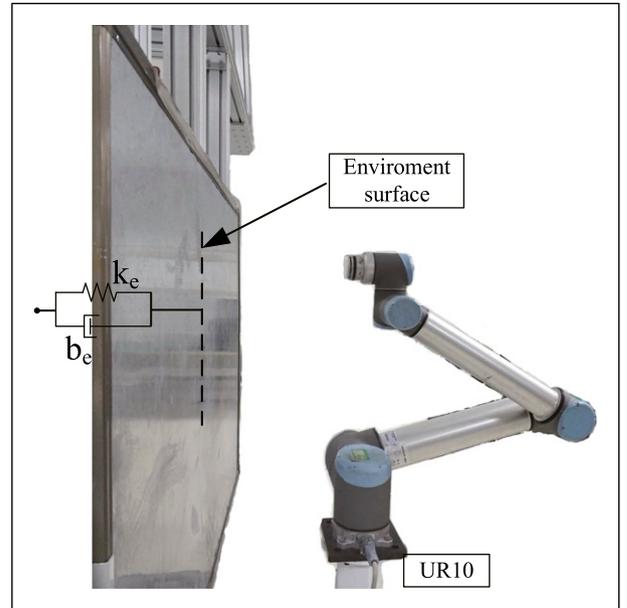


Figure 12. Experimental setup of contact with an aluminum plate.

parameters of the learning algorithm, we set $\varepsilon = 1$ to increase tolerance for error because it is difficult for $\Delta J(\mathbf{z})$ to converge to a small value in the learning algorithm.

It is not difficult to find that in the experiment of the robot in contact with two different environments, we take different values for λ and b_f to correspond to the three situations of the simulation described above. The experimental results show that without reducing the contact

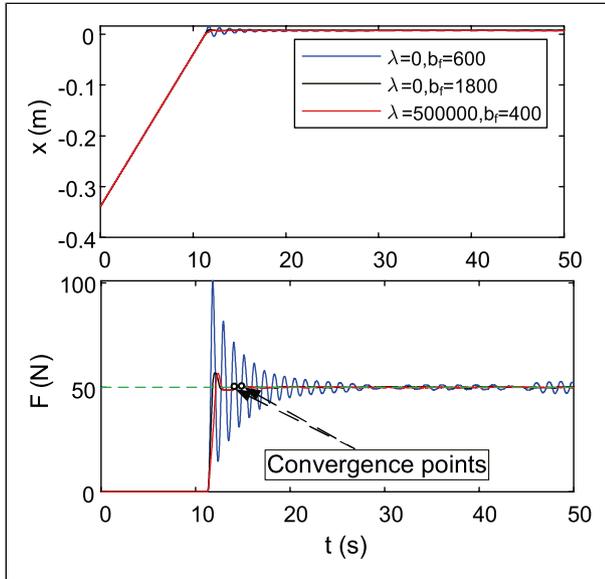


Figure 13. Experiment results of contact with an aluminum plate.

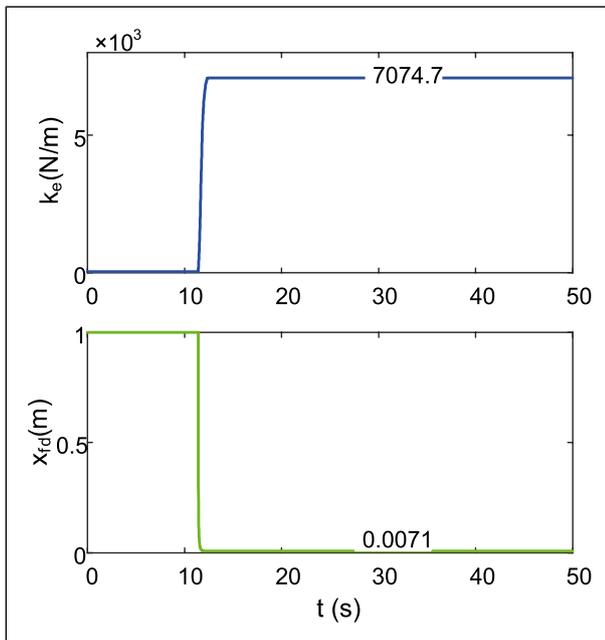


Figure 14. Impedance learning results and x_{fd} profiles with environment stiffness in the second experiment.

speed, both the large damping controller and the virtual semi-active damping controller proposed in this article suppress force overshoot. Meanwhile, the virtual semi-active damping controller has faster convergence, which is conducive to saving energy, and this advantage is more apparent for contact with a low stiffness environment as in the first experiment. By applying the impedance learning algorithm, the parameters of the environmental impedance are obtained before the force control converges when the

robot is in contact with different environments. In this way, it is feasible to apply the environment parameters obtained in designing the controller.

6. Conclusion

We investigated the motion and force hybrid tracking task of a manipulator making contact with unknown environments, considering only a single direction of the contact interaction. The proposed virtual semi-active damping based on the position provides an effective means for the damping of the bouncing and force overshoot, which may occur at the moment that the manipulator is in contact with the environment at a nonzero velocity. The environment impedance parameters are needed to obtain the relation between the virtual damping and position. The L-BFGS impedance learning method was therefore developed to estimate stiffness and damping. A simulation and contact experiment involving a UR10 robot in different environments demonstrated that the proposed method is feasible and effective.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The authors are grateful to the Science and Technology Development Plan of Jilin Province (2018020102GX) and Jilin Province and the Chinese Academy of Sciences Cooperation in Science and Technology High-Tech Industrialization Special Funds Project (2018SYHZ0004) and the National Key Research and Development Program of China (2017YFC0822403).

ORCID iDs

Wenrui Wang <https://orcid.org/0000-0002-3023-9659>

Ang Li <https://orcid.org/0000-0002-0063-1422>

Jinlin Gu <https://orcid.org/0000-0003-2144-9897>

Hairong Chu <https://orcid.org/0000-0003-1071-195X>

References

- Abdi F and Shakeri F (2019) A globally convergent BFGS method for pseudo-monotone variational inequality problems. *Optimization Methods and Software* 34(1): 25–36.
- Arefinia E, Talebi HA and Doustmohammadi A (2020) A robust adaptive model reference impedance control of a robotic manipulator with actuator saturation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50(2): 409–420.
- Berahas AS and Tak M (2020) A robust multi-batch l-bfgs method for machine learning. *Optimization Methods and Software* 35(1): 191–219.

- Branicky MS (1998) Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control* 43(4): 475–482.
- Buchli J, Stulp F, Theodorou E, et al. (2011) Learning variable impedance control. *The International Journal of Robotics Research* 30(7): 820–833.
- Carloni R, Sanfelice RG, Teel AR, et al. (2007) A hybrid control strategy for robust contact detection and force regulation. In: 2007 American control conference, NY, USA, 9–13 July 2007, pp. 1461–1466. IEEE.
- Diolaiti N, Melchiorri C and Stramigioli S (2005) Contact impedance estimation for robotic systems. *IEEE Transactions on Robotics* 21(5): 925–935.
- Fernando H, Marshall JA, Larsson J, et al. (2019) Iterative learning-based admittance control for autonomous excavation. *Journal of Intelligent & Robotic Systems* 96(3): 493–500.
- He W and Dong Y (2017) Adaptive fuzzy neural network control for a constrained robot using impedance learning. *IEEE Transactions on Neural Networks and Learning Systems* 29(4): 1174–1186.
- Heck D, Saccon A, Van de Wouw N, et al. (2016) Guaranteeing stable tracking of hybrid position-force trajectories for a robot manipulator interacting with a stiff environment. *Automatica* 63: 235–247.
- Lai CY (2014) Improving the transient performance in robotics force control using nonlinear damping. In: IEEE/ASME international conference on advanced intelligent mechatronics (AIM), Besancon, France, 8–11 July 2014, pp. 892–897. New york: IEEE.
- Lai CY, Li YP, Vuong ND, et al. (2012) Nonlinear damping for improved transient performance in robotics force control. In: 2012 IEEE/ASME international conference on advanced intelligent mechatronics (AIM), Kaohsiung, Taiwan, 11–14 July 2012, pp. 314–319. New york: IEEE.
- Li Z, Huang B, Ajoudani A, et al. (2018) Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations. *IEEE Transactions on Robotics* 34(1): 264–271.
- Li Z, Liu J, Huang Z, et al. (2017) Adaptive impedance control of human-robot cooperation using reinforcement learning. *IEEE Transactions on Industrial Electronics* 64(10): 8013–8022.
- Liao A (1997) Modifying the BFGS method. *Operations Research Letters* 20(4): 171–177.
- Liu DC and Nocedal J (1989) On the limited memory BFGS method for large scale optimization. *Mathematical Programming* 45(1): 503–528.
- Luo J, Solowjow E, Wen C, et al. (2019) Reinforcement learning on variable impedance controller for high-precision robotic assembly. In: International conference on robotics and automation (ICRA). Montreal, QC, 20–24 May 2019, pp. 3080–3087. New york: IEEE.
- Markus ED, Agee JT and Jimoh AA (2016) Flat control of industrial robotic manipulators. *Robotics and Autonomous Systems* 87: 226–236.
- Matteo L, Tsagarakis NG and Caldwell DG (2010) A variable physical damping actuator (VPDA) for compliant robotic joints. In: 2010 IEEE international conference on robotics and automation. Anchorage, Alaska, 3–7 May 2010, pp. 1668–1674. New york: IEEE.
- Mohammad AEK, Hong J and Wang D (2018) Design of a force-controlled end-effector with low-inertia effect for robotic polishing using macro-mini robot approach. *Robotics and Computer-Integrated Manufacturing* 49: 54–65.
- Neculai A (2018) An adaptive scaled BFGS method for unconstrained optimization. *Numerical Algorithms* 77(2): 413–432.
- Nozaki T, Shimizu S, Murakami T, et al. (2018) Impedance field expression of bilateral control for reducing data traffic in haptic transmission. *IEEE Transactions on Industrial Electronics* 66(2): 1142–1150.
- Petersson S and Lennartson B (1996) Stability and robustness for hybrid systems. In: Proceedings of 35th IEEE conference on decision and control. Kobe, Japan, 13 December 1996, pp. 1202–1207. New york: IEEE.
- Philipp M, Nishihara R and Jordan M (2016) A linearly-convergent stochastic L-BFGS algorithm. In: 2016 international conference on artificial intelligence and statistics, Cadiz, Spain, 9–11 May 2016, pp. 249–259. Amsterdam, the Netherlands: Elsevier.
- Raghu B, Mudigere D, Nocedal J, et al. (2018) A progressive batching L-BFGS method for machine learning. In: Proceedings of the 35th international conference on machine learning, Stockholm, Sweden, 10–15 July 2018. Chicago: ML.
- Sharifi M, Salarieh H, Behzadipour S, et al. (2018) Impedance control of non-linear multi-DOF teleoperation systems with time delay: absolute stability. *IET Control Theory & Applications* 12(12): 1722–1729.
- Stegall P, Zanotto D and Agrawal SK (2017) Variable damping force tunnel for gait training using ALEX III. *IEEE Robotics and Automation Letters* 2(3): 1495–1501.
- Zoe D and Iliadis G (2005) Contact stability analysis of a one degree-of-freedom robot using hybrid system stability theory. *Robotica* 23(5): 607–614.

Appendix I

The proofs of statements (1) and (2) are straightforward from [Section 3](#).

In proving statement (3), we let switching times $S_k = S_{f,p}$, when the arm moves from the contact phase to free motion at $t = t_{k(k=2i)}$. In this case, we have $\dot{x} > 0$, $\Delta x = -x_d$ and $\Delta F = -F_d$. After first contact occurs, the control objective of the whole system is to keep the interaction between the robot arm and environment with the desired force f_d . Therefore, the desired position can be regarded as $x_{fd} = F_d/k_e$ and the velocity is zero. We then can express $V_p(x(t_k))$ and $V_f(x(t_k))$ as

$$V_p(x(t_k)) = \frac{1}{2}M\Delta\dot{x}^2 + \frac{1}{2}k_{p1}\Delta x^2 = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}k_{p1}x_d^2 \quad (39)$$

and

$$\begin{aligned}
V_f(x(t_k)) &= \frac{1}{2}M\dot{x}^2 + M\Delta x\dot{x} + \frac{1}{2}(k_{p2} + 1)\Delta x^2 \\
&\quad + \frac{1}{2}(b_f + \lambda|x - x_{fd}|)\Delta x^2 + k_{p2}\Delta F\Delta x + \frac{1}{2}k_{p2}\Delta F^2 \\
&= \frac{1}{2}M\dot{x}^2 - Mx_{fd}\dot{x} + \frac{1}{2}(k_{p2} + 1)x_{fd}^2 \\
&\quad + \frac{1}{2}(b_f + \lambda x_{fd})x_{fd}^2 + k_{p2}k_e x_{fd}^2 + \frac{1}{2}k_{p2}k_e^2 x_{fd}^2
\end{aligned} \tag{40}$$

Therefore

$$\begin{aligned}
V_p(x(t_k)) - V_f(x(t_k)) &= \frac{1}{2}k_{p1}x_{fd}^2 + Mx_{fd}\dot{x} - \left(\frac{1}{2}(k_{p2} + 1)x_{fd}^2 + \frac{1}{2}(b_f + \lambda x_{fd})x_{fd}^2 \right) \\
&\quad - \left(k_{p2}k_e x_{fd}^2 + \frac{1}{2}k_{p2}k_e^2 x_{fd}^2 \right) \\
&\leq \frac{1}{2}k_{p1}x_{fd}^2 - \left(\frac{1}{2}(k_{p2} + 1)x_{fd}^2 + \frac{1}{2}(b_f + \lambda x_d)x_d^2 \right) \\
&\quad - \left(k_{p2}k_e x_{fd}^2 + \frac{1}{2}k_{p2}k_e^2 x_{fd}^2 \right) \\
&= \frac{1}{2}(k_{p1} - (k_{p2} + 1) - (b_f + \lambda x_{fd}) - 2k_{p2}k_e - k_{p2}k_e^2)x_{fd}^2
\end{aligned} \tag{41}$$

when

$$k_{p1} \leq (k_{p2} + 1) + (b_f + \lambda x_{fd}) + 2k_{p2}k_e + k_{p2}k_e^2 \tag{42}$$

In proving statement (4), because $V_p(x(t))$ and $V_f(x(t))$ are nonincreasing functions, we have

$$V_p(x(t_{k(k=2i)})) \geq V_p(x(t_{k(k=2i+1)})) \tag{43}$$

and

$$V_f(x(t_{k(k=2i+1)})) \geq V_f(x(t_{k(k=2i+2)})) \tag{44}$$

which imply

$$\frac{1}{2}M\dot{x}(t_{k(k=2i)})^2 \geq \frac{1}{2}M\dot{x}(t_{k(k=2i+1)})^2 \tag{45}$$

and

$$\frac{1}{2}M\dot{x}(t_{k(k=2i+1)})^2 \geq \frac{1}{2}M\dot{x}(t_{k(k=2i+2)})^2 \tag{46}$$

such that

$$|\dot{x}(t_{k(k=2i)})| \geq |\dot{x}(t_{k(k=2i+1)})| \geq |\dot{x}(t_{k(k=2i+2)})| \tag{47}$$

It is inferred from (25) that the velocity of the system is decreasing at consequent times that the controller switches to the position control mode and it is thus obvious that

$$V_p(t_{k(k=2i)}) \geq V_p(t_{k(k=2i+2)}) \tag{48}$$

Furthermore, we easily deduce that

$$V_p(t_{k(k=2i+1)}) \geq V_p(t_{k(k=2i+3)}) \tag{49}$$

It is easily proved by contradiction that there is a finite number of Lyapunov function switches according to Theorem 1(4). The energy function $V_{k=2i}$ also drops appreciably when the manipulator breaks from the surface of the environment. Therefore, the last switch brings the system to the contact phase. The force controller with the Lyapunov function is strictly decreasing in restricted motion, and the whole system is thus asymptotically stable.