Sensors and Materials, Vol. 31, No. 12 (2019) 4113–4133
MYU Tokyo

S & M 2074

Robot Collision Detection and Distinction Based on Convolution Filtering Dynamic Model

Zhijing Li, Jinhua Ye, and Haibin Wu*

School of Mechanical Engineering and Automation, Fuzhou University,
2 Xue Yuan Road, University Town, Fuzhou 350116, China

(Received April 12, 2019; accepted August 1, 2019)

Keywords: robot safety, force sensing, convolution filtering, collision detection, distinction method

With the increasing application of human–robot interaction, collision detection between robot and unknown environments, along with further distinction from the intentional contact between human and robot, have become urgent problems to be solved. In this paper, a new collision detection algorithm is proposed, and a collision distinction method is further designed on the basis of this algorithm. The generalized momentum and the convolution method are used to develop the robot convolution filtering dynamic model. Then, the force-sensing observer that uses only proprioceptive sensors is designed to observe the torque deviation of the joint online to realize robot collision detection. At the same time, the performance of the force-sensing observer is improved by compensating for joint friction. The proposed algorithm does not need any external sensors; it overcomes the disadvantage of calculation errors owing to the acquisition of joint acceleration information. The filter can be flexibly selected in the algorithm according to the actual application of the robot. Moreover, two force-sensing observers are adopted in the collision distinction method. The contact or collision between a human and a robot can be further distinguished after setting the appropriate thresholds and filtering parameters. The collision detection algorithm can be easily adapted to different types of robot to ensure human safety, and the proposed collision distinction method can be used to improve the work efficiency of the robot. External force sensing experiments show that the low-pass and bandpass observers work well and different force signals can be observed. The collision detection and human–robot interaction experiments are performed to verify that the collision detection algorithm and collision distinction method are reasonable and effective.

1. Introduction

In recent years, robot applications have expanded from traditional industrial production to medical care, service and education fields, and so forth. Human–robot interaction and cooperative work based on force sensing have become hot topics in robotics research. Safety is the most important feature for robots that share a limited working space with humans.[1] At the same time, the robot needs to sense the intentional contact force of a human to perform a

*Corresponding author: e-mail: wuhb@fzu.edu.cn
https://doi.org/10.18494/SAM.2019.2402

ISSN 0914-4935

specified action or task during the process of cooperation. The manner in which contact or collision between a human and a robot should be distinguished has become an urgent problem in robotics.[2]

According to existing research, when a robot is working in an open environment, it may collide with a human body or objects in the working space. The safest way for the robot to work in an open environment is to avoid direct contact with a human. When a human works closely with moving robot, the robot may accidentally collide with the human owing to the uncertainty of the human movement. Safety is a primary problem in robot research. Researchers have proposed many feasible methods to cope with this problem. Some measures are taken to prevent the collision a robot with a human or an object.[3] Video detection, noncontact sensing, and path planning technology have been used in those methods.[4,5] Improving the safety of robots in the collision phase is also an important direction in robot research. On the one hand, a much safer and more reliable robot body structure can be designed. For example, some new materials are used to reduce the inertia of the robot body, a new flexible joint design improves the flexibility of the robot, wrapping with a viscoelastic material buffers the impact force, and new robotic skins are designed to sense and buffer collision force.[6–8] These methods can increase the safety of the robot to a certain extent, but they also make the system more complicated and increase costs. The joint torque sensor and six-axis force/torque sensor are typically adopted to achieve collision detection between a robot and an unstructured environment. The disadvantages are that the torque sensor is expensive, and the strain characteristics of the force sensor generate noise and reduce the bandwidth of the control system.[9] Sensing a change in joint motor current is an effective method of achieving collision detection. For instance, Li *et al.* studied the method of sensing a current change to achieve robot collision detection.[10] Indri *et al.* also designed a virtual impact sensor by sensing changes in joint currents.[11] The disadvantage of the current sensing method is that the collision detection threshold setting process is complicated. The dynamic model-based control method can be used for robot collision detection. Huang *et al.* proposed the adaptive impedance control algorithm for robot collision detection.[12] Lim and Tanie designed an observer-based collision detection and safety control method.[13] Its inverse dynamic calculation increased the amount of calculation of the algorithm. The dynamic model should obtain joint acceleration information which is, however, complicated to calculate or measure. To solve it, Luca *et al.* proposed a robot collision detection algorithm according to the generalized momentum, which reduced the amount of computation and improved the collision detection efficiency.[3]

The main challenge about physical human–robot interaction is to simultaneously ensure the safety of humans and the friendly interaction of humans with robots.[14] These above-mentioned collision detection methods can only sense the magnitude of the external force. The process of setting the threshold in the collision detection method cannot effectively distinguish the contact and collision during human and robot interaction. As far as we know, there is little research on distinguishing contact and collision during human–robot interaction. In most cases, the contact is controlled intentionally by the human interacting with a robot. The contact force signal changes slowly. However, the collision duration is very short, and the collision force will generate a larger jump than contact force. It can be seen that the signals of contact force and

collision force have different distribution characteristics in the frequency domain. Geravand *et al.* attempted to distinguish the contact from collision during the interaction between a human and a robot by filtering the current of the joint motor.[15] A collision discrimination method based on machine learning was presented in Ref. 16. This method requires the offline training of contact models and the use of multiple contact features. Recently, Kouris *et al.* have proposed a novel collision distinction algorithm based on Fourier transform for analyzing the frequency change of a force signal to improve the collision distinction speed.[17] Moreover, Dixon *et al.* developed a fault detection algorithm for robot manipulators based on the convolution method.[18] Damme *et al.* also used this method for force sensing and estimation in the end effector of a robot manipulator.[19] However, these methods are not focused on distinguishing collision or contact to improve the performance of human–robot interaction.

From the perspective of force sensing and using existing information on conventional industrial robots, a new collision detection algorithm is proposed in this work. The robot generalized momentum and convolution method are used to design the robot convolution filtering dynamic model to remove the acceleration information. Then, a force-sensing observer is designed to realize robot collision detection by observing the robot joint torque deviation. At the same time, the performance of the collision detection algorithm is improved by identifying and compensating for joint friction. The proposed force-sensing observer only uses robot proprioceptive sensors and does not need any external sensors. Moreover, by filtering the joint torque signals through low-pass and bandpass filters in the convolution filter dynamic model simultaneously, a collision distinction method is designed to distinguish the contact and collision during human–robot interaction. It is significant for ensuring the safety of humans and improving the robot efficiency.

This paper is organized as follows. In Sect. 2, we outline the dynamics and useful properties of robots. In Sect. 3, we present the robot convolution filtering dynamic modelling. In Sect. 4, we describe the collision detection and distinction for human–robot interaction in detail. The experimental results and analysis are presented in Sect. 5. The conclusions are given in Sect. 6.

2. Robot Dynamics Modelling

By considering a robot with n degrees of freedom as open kinematic chains of rigid bodies and using the Euler–Lagrange method in the joint space coordinates, we express the robot dynamic equation as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}}) = \boldsymbol{\tau}, \tag{1}$$

where $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and $\ddot{\boldsymbol{q}}$ are the angles, angular velocities, and angular acceleration vectors of the joint, respectively; $\boldsymbol{M}(\boldsymbol{q})$ is a symmetric and positive definite inertial matrix; $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ contains Coriolis and centrifugal terms; $\boldsymbol{G}(\boldsymbol{q})$ is the gravity matrix; $\boldsymbol{F}(\dot{\boldsymbol{q}})$ is the friction of the joint; and $\boldsymbol{\tau}$ is the control torque of the joint. Since conventional industrial robots generally do not have a joint torque sensor installed, the joints are usually composed of a servo motor, a transmission mechanism, and an output link. To obtain the joint control torque in the dynamic Eq. (1), the

method of measuring the joint motor current is used in this study. The driving current of the motor is converted accordingly, and the equation for calculating the joint driving torque is $\boldsymbol{\tau} = n_{har}K_{dri}\boldsymbol{i}_m$, where $\boldsymbol{i}_m$ is the current of the motor, K_{dri} is the correlation coefficient of the current into the torque, and n_{har} is the transmission ratio of the joint motor to the output link.

From the skew symmetry of the matrix $\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})$,[15] it follows that

$$\dot{\boldsymbol{M}}(\boldsymbol{q}) = \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{C}^T(\boldsymbol{q},\dot{\boldsymbol{q}}). \tag{2}$$

Generally, the dynamic parameters of the robot can be obtained through the computer-aided design (CAD) model. Thus, the robot dynamic equation can be reexpressed as

$$\hat{\boldsymbol{M}}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \hat{\boldsymbol{C}}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \hat{\boldsymbol{G}}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}}) = \hat{\boldsymbol{\tau}}, \tag{3}$$

where $\hat{\boldsymbol{M}}(\boldsymbol{q})$, $\hat{\boldsymbol{C}}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$, $\hat{\boldsymbol{G}}(\boldsymbol{q})$, and $\hat{\boldsymbol{\tau}}$ are the CAD model parameters corresponding to Eq. (1). $\boldsymbol{F}(\dot{\boldsymbol{q}})$ generally needs to be obtained through experiments. According to Ref. 18, the dynamic equation for the robots can be linearly parameterized as

$$\boldsymbol{\tau} = \boldsymbol{Y}(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}})\boldsymbol{\theta}, \tag{4}$$

where $\boldsymbol{Y}(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}})$ is an $n \times l$ dimensional function, which is called the regression factor, and $\boldsymbol{\theta}$ is an l dimensional parameter vector.

3. Robot Convolution Filtering Dynamic Modelling

3.1 Convolution filtering dynamic model

The acceleration information of the joint is required in Eq. (1). However, it is difficult to obtain robot joint acceleration in practical applications. To eliminate the need to estimate joint accelerations and to use only proprioceptive robot sensors, the convolution method is adopted to eliminate the acceleration measurement or calculation in this paper. The generalized momentum of a robot is defined as

$$\boldsymbol{P} = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}. \tag{5}$$

The time derivation of generalized momentum (5) can be expressed as

$$\dot{\boldsymbol{P}} = \frac{d}{dt}(\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}) = \dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}. \tag{6}$$

Thus, the robot dynamic Eq. (1) can be rewritten as

$$\boldsymbol{\tau} = \dot{\boldsymbol{P}} + \boldsymbol{H}, \tag{7}$$

where

$$\boldsymbol{H} = -\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}}) . \tag{8}$$

From Eq. (7), it can be seen that $\ddot{\boldsymbol{q}}$ has been separated in a way that allows it to be filtered out. A suitable and stable filter is used to perform the convolution operation on both sides of Eq. (7),

$$\tau_f = f * \tau = f * \dot{\boldsymbol{P}} + f * \boldsymbol{H} , \tag{9}$$

where f is the impulse response of the filter and "$*$" is the convolution operator symbol. The properties of the convolution operation are as described in Ref. 18,

$$f * \dot{\boldsymbol{P}} = \dot{f} * \boldsymbol{P} + f(0) * \boldsymbol{P} - f * \boldsymbol{P}(0) . \tag{10}$$

By combining Eqs. (9) and (10), we can obtain the convolution filtering dynamic equation as

$$\tau_f = \dot{f} * \boldsymbol{P} + f(0) * \boldsymbol{P} - f * \boldsymbol{P}(0) + f * \boldsymbol{H} . \tag{11}$$

Substituting Eqs. (5) and (8) into Eq. (11) yields

$$\tau_f = \dot{f} * [\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}] + f(0)\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} - f\boldsymbol{M}(\boldsymbol{q}(0))\dot{\boldsymbol{q}}(0) + f * [-\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}})]. \tag{12}$$

The initial generalized momentum of the robot is zero, i.e., $\boldsymbol{P}(0) = 0$. Then, substituting Eq. (2) into Eq. (12) yields

$$\tau_f = \dot{f} * [\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}] + f(0)\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + f * [\boldsymbol{C}^{\mathrm{T}}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}})]. \tag{13}$$

Thus, Eq. (13) can be further expressed as

$$\tau_f = \int_0^t \dot{f}(t-r)\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}dr + f(0)\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + \int_0^t f(t-r)[\boldsymbol{C}^{\mathrm{T}}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}})]dr, \tag{14}$$

where t is the time and r is the integral variable. Therefore, from Eq. (13), one can get its equivalent linear expression as

$$\boldsymbol{\tau_f} = \boldsymbol{Y_f}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{\theta} . \tag{15}$$

By comparing Eqs. (15) and (4), we know that the generalized momentum of the robot can be used to separate the acceleration parameters in the dynamic equations. Then, the convolution operation removes the joint acceleration, which avoids complicated acceleration calculations and facilitates the implementation of advanced robot control algorithms.

3.2 Filter selection and analysis

The filter in the convolution calculation of Eq. (13) is the key to realizing the robot collision detection. Presently, there are many types of filters. An advanced filter can obtain better filtering results, but its complex response will increase the number of calculations and affect the performance of the algorithm. Considering the real-time performance of the algorithm and the research purposes, the first-order low-pass and second-order bandpass filters were selected for the robot collision detection and distinction in this paper. The first-order low-pass filter transfer function is

$$f_1(s) = \frac{\alpha}{s+\beta}, \tag{16}$$

where α and β denote positive filter constants. The inverse Laplace transform of Eq. (16) is

$$f_1(t) = \alpha e^{-\beta t}. \tag{17}$$

To filter the interference of high-frequency signals, assume $\alpha = \beta = K$. Using the Laplace transform and then substituting Eq. (17) into Eq. (14) yield

$$\frac{\tau_{f1}(s)}{\tau(s)} = \frac{\hat{\tau}_{f1}(s)}{\hat{\tau}(s)} = \frac{K}{s+K}. \tag{18}$$

After deriving Eq. (17) with respect to time and performing the Laplace transform operation, one obtains

$$f_2(t) = \dot{f}_1(t) = -K^2 e^{-Kt}, \tag{19}$$

$$f_2(s) = -\frac{K^2}{s+K}. \tag{20}$$

Considering the human–robot cooperation situation, the robot must have the ability to distinguish the contact and collision between itself and a human. The bandpass filter can be further used to filter out low-frequency signals to achieve the distinction between contact and collision.[15,17] For simplification, a low-pass filter and a high-pass filter are taken in series to form a bandpass filter. The transfer function of the bandpass filter is given by

$$f_3(s) = \frac{K_1}{s+K_1} \cdot \frac{s}{s+K_2}, \tag{21}$$

where K_1 and K_2 are the low-pass and high-pass cut-off frequencies, respectively. By using

the inverse Laplace transform, we can express the equivalent function of Eq. (21) in the time domain as

$$f_3(t) = -\frac{K_1^2}{K_2 - K_1} e^{-K_1 t} + \frac{K_1 K_2}{K_2 - K_1} e^{-K_2 t}. \tag{22}$$

Its transfer function can be obtained through the Laplace transform by using Eqs. (22) and (14):

$$\frac{\tau_{f_3}(s)}{\tau(s)} = \frac{\hat{\tau}_{f_3}(s)}{\hat{\tau}(s)} = \frac{K_1 s}{s^2 + (K_1 + K_2)s + K_1 \cdot K_2}. \tag{23}$$

Similarly, the time derivation of the filter transfer equation [Eq. (22)] and its corresponding transfer function in the Laplace domain are as follows:

$$f_4(t) = \dot{f}_3(t) = -\frac{K_1^3}{K_1 - K_2} e^{-K_1 t} + \frac{K_1 K_2^2}{K_1 - K_2} e^{-K_2 t}, \tag{24}$$

$$f_4(s) = \frac{-(K_1^2 + K_1 K_2)s - K_1^2 K_2}{s^2 + (K_1 + K_2)s + K_1 K_2}. \tag{25}$$

4. Robot Collision Detection and Distinction

4.1 Robot collision detection

From Sect. 3.1, it can be seen that less parameter information is required for the robot dynamic equation after the convolution operation. In this section, we extend the robot convolution filtering dynamics model to robot collision detection. When the robot collides with the external environment, the output torque of the robot joint is

$$\boldsymbol{\tau} = \boldsymbol{\tau_d} + \boldsymbol{\tau_c}, \tag{26}$$

where $\boldsymbol{\tau}_d$ is the ideal control torque without collision and $\boldsymbol{\tau}_c$ is the equivalent torque generated by the collision force.

$$\tau_c = \boldsymbol{J}(\boldsymbol{q})^T \boldsymbol{F}_c, \tag{27}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the Jacobian matrix and $\boldsymbol{F}_c$ is the external collision force. The convolution operation for joint output torque can be obtained as

$$\tau_f = f * (\tau_d + \tau_c) = \tau_{df} + \tau_{cf}. \tag{28}$$

The estimated generalized momentum can be obtained from the CAD model of the robot using Eq. (5).

$$\hat{\boldsymbol{P}} = \hat{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{29}$$

Similarly, the convolution calculation equation for the estimated torque can be obtained from Eqs. (3) and (13).

$$\hat{\tau}_f = \dot{f} * [\hat{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}}] + f(0)\hat{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + f * [\hat{\boldsymbol{C}}^{\mathrm{T}}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \hat{\boldsymbol{G}}(q) + \hat{\boldsymbol{F}}(\dot{\boldsymbol{q}})]. \tag{30}$$

When there is no external impact force ($\boldsymbol{\tau}_c = \boldsymbol{0}$), and $\tau_f = \tau_{df} \approx \hat{\tau}_f$, Eq. (30) can be called the torque estimator. Thus, the calculation formula of the external impact torque after convolution operation is

$$\boldsymbol{\tau}_{\boldsymbol{cf}} = \boldsymbol{\tau}_{\boldsymbol{f}} - \hat{\boldsymbol{\tau}}_{\boldsymbol{f}}\,. \tag{31}$$

Because the real dynamic parameters of the robots are generally not available in practice, it is necessary to set a reasonable threshold to avoid false collision detection. When the robot collides with an unknown environment, it will generate an equivalent torque $\boldsymbol{\tau}_c \neq 0$, which can be obtained from Eq. (31). In this case, Eq. (31) can be expressed as

$$\tau_{cf} = \left[\, \tau_{cf1} \cdots \tau_{cfi}, \tau_{cf(i+1)} \cdots \tau_{cfn} \,\right]^{\mathrm{T}}, \tag{32}$$

$$Sign = \begin{cases} \dfrac{\tau_{cfi}}{\left|\tau_{cfi}\right|}, & if \ \left|\tau_{cfi}\right| > N_i \\ 0, & else \ \left|\tau_{cfi}\right| \le N_i \end{cases}, \tag{33}$$

where i = 1, ..., n, and N_i is the torque threshold of joint i, which can be obtained from the experiment. Therefore, it can be determined by using Eqs. (32) and (33), and comparing the elements in $\boldsymbol{\tau}_{cf}$ from $\boldsymbol{\tau}_{cfn}$ to $\boldsymbol{\tau}_{cf1}$ individually that the link i has collided when $\left|\tau_{cfi}\right| > N_i$ and $\left|\tau_{cf(i+1)}\right| \le N_{i+1}$. The direction of the collision can be determined using Eq. (33).

After the robot control torque is changed from stationary to motion, the position and velocity of the joint will lag and may cause false collision detection due to inertia. The correction of $\boldsymbol{\tau}_{cf}$ is in the following equation:

$$\Delta\tau = a \cdot W + bG\,, \tag{34}$$

where $\Delta\tau$ is the corrected value for $\boldsymbol{\tau}_{cfi}$, a and b are the adjustment factors, W is the parameter related to the joint motor and servo driver, and G is the robot inertia parameter. When the robot

is changing from a static state to a moving state, $a > 0$ and $b > 0$, and other motion states, $a = 0$ and $b = 0$.

From the above analysis, it can be seen that the joint torque deviation determined using the convolution filtering dynamic model can realize the detection of the robot collision position and direction. The principle of the collision detection algorithm is shown in Fig. 1. As can be seen from Fig. 1, $\boldsymbol{\tau}_{cf}$ corresponds to the external force that the robot is subjected to, so the external force calculation method can be called a force-sensing observer.

4.2 Joint friction identification method

To reduce the interference of friction, in this work, we improve the sensitivity of the collision detection algorithm by identifying and compensating the joint friction.[20] Bittencourt and Gunnarsson pointed out that the industrial robot joint friction force usually contains the Stribeck effect.[21] Therefore, the Stribeck friction model is used to describe the conventional industrial robot joint friction, and it can be expressed as

$$F(\dot{q}) = F_c \operatorname{sgn}(\dot{q}) + F_v \dot{q} + F_s e^{-(\dot{q}/\dot{q}_s)\sigma} \operatorname{sgn}(\dot{q}), \tag{35}$$

where F_s is the Stribeck parameter, $\dot{q}_s$ is the Stribeck velocity, F_c is the Coulomb friction coefficient, $F_c \operatorname{sgn}(\dot{q})$ is the Coulomb friction, F_v is the viscous friction coefficient, $F_v \dot{q}$ is the viscous friction, and σ is the constant associated with the contact surface geometry.

To facilitate data processing with MATLAB software, Eq. (35) is simplified as

$$F(\dot{q}) = \lambda_1 + \lambda_2 \dot{q} + \lambda_3 \exp(-\lambda_4 \dot{q}), \tag{36}$$

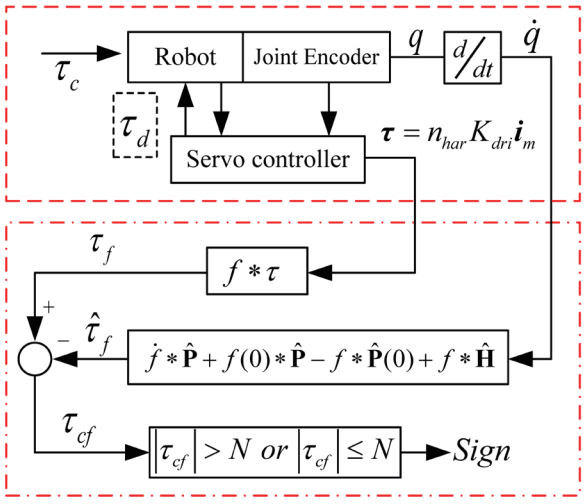

Fig. 1. (Color online) Collision detection schematic.

where λ_1, λ_2, λ_3, and λ_4 are the required parameters. Then, the parameters in Eq. (36) are estimated offline by the least-squares method.

In this study, independent experiments are performed on each joint to obtain the friction force of the joint and the parameters of the designed model. A joint of the robot is controlled to move at a constant speed $\ddot{q} = 0$, other joints are fixed, and when no external force is applied to the robot, from Eqs. (1) and (4), we have

$$g(q) + \tau_f = n_{har} K_{dri} i_m , \tag{37}$$

In Eq. (37), the joint control torque only contains gravity and friction torque. The joint of the robot is controlled to move in positive and negative directions within a given interval. At this point, Eq. (37) becomes

$$\sum_{1}^{\varphi} (g(q)^+ + \tau_f^+) = \sum_{1}^{\varphi} (n_{har} K_{dri} i_m)^+ , \tag{38}$$

$$\sum_{1}^{\varphi} (g(q)^- + \tau_f^-) = \sum_{1}^{\varphi} (n_{har} K_{dri} i_m)^- , \tag{39}$$

where the superscript symbols "+" and "–" are indicated as positive and negative directions, respectively, and φ is the amount of discrete data collected under single-velocity-value experimental conditions. When the joint is in the same position during the positive and negative motions, the gravity has $g(q)^+ = g(q)^-$ and the friction has $\tau_f^+ = -\tau_f^-$. Subtracting Eq. (38) from Eq. (39) gives

$$\tau_f = \frac{\sum_{1}^{\varphi} (n_{har} K_{dri} i_m)^+ - \sum_{1}^{\varphi} (n_{har} K_{dri} i_m)^-}{2\varphi} , \tag{40}$$

Therefore, the unknown optimized friction parameters can be obtained using the following formula:

$$\min \Delta \sum_{1}^{\Phi} \left[\tau_f - (\lambda_1 + \lambda_2 \dot{q} + \lambda_3 \exp(-\lambda_4 \dot{q})) \right]^2 , \tag{41}$$

where Φ is the amount of friction data obtained at different discrete velocities. The above experimental method can reduce the effect of the robot model errors on the joint friction measurement.

4.3 Robot collision distinction

In this paper, the collision detection algorithms with low-pass or bandpass filters can be called low-pass observers (LPOBs) and bandpass observers (BPOBs), respectively. According to the degree of danger when the robot is subjected to external forces, three different task priority levels in the control algorithm are developed to ensure the safety of a human and a robot in this work.

1) High priority: collision
2) Medium priority: contact
3) Low priority: normal work

The LPOB and BPOB working together can be used for contact and collision detection and distinction between the human and the robot. It is assumed that $\tau 1_{cf}$ and $N1$ are the observer torque and threshold of the LPOB, and $\tau 2_{cf}$ and $N2$ are the observer torque and threshold of the BPOB, respectively. The specific control process for collision detection and distinction is shown in Fig. 2.

5. Experimental Results

5.1 Robot experimental platform

The experiments were conducted with a six-DOF industrial robot in the laboratory. The robot system is mainly composed of a computer, control software, a control cabinet, and a robot body. Its overall structure is shown in Fig. 3. The control cabinet mainly includes GALIL DMC motion control cards and servo drives. The computer is used for motion planning. It uses Ethernet to communicate with the DMC motion control card. The motion control card controls the servo driver, which controls the motion of the robot. The motor servo driver and

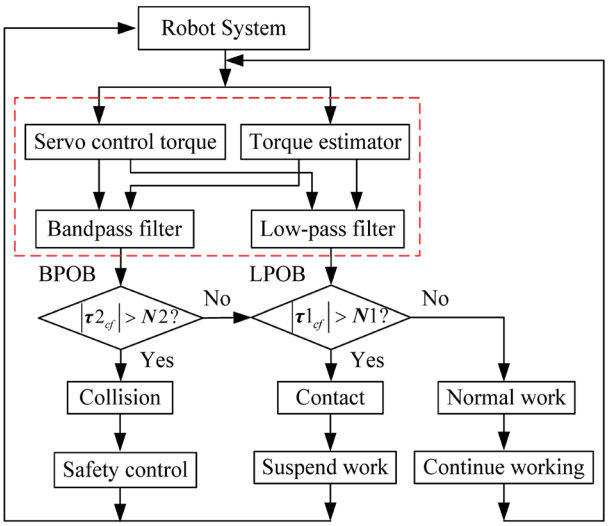

Fig. 2. (Color online) Flow chart of robot collision detection and distinction.

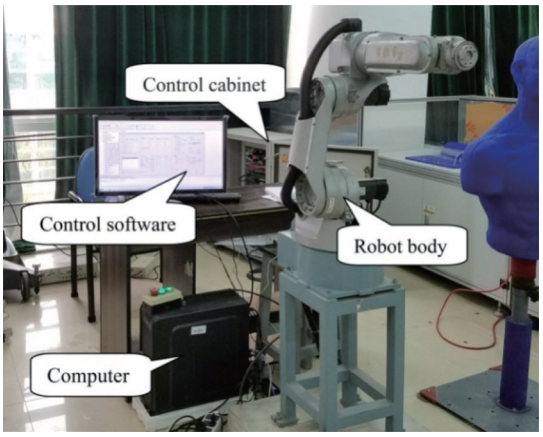

Fig. 3. (Color online) Robot experimental platform.

the computer communicate through the serial port. The computer can directly set the status of the servo driver through the serial port and read the relevant parameters of the servo driver. The control software is implemented in Microsoft VC++ 6.0 using C++ and runs in a Windows environment. The algorithm proposed in this paper is directly embedded in the control software. The specific implementation process of the proposed algorithm in the robot system is described in Ref. 22. To simplify the calculation, links 2 and 3 were used for the experiments without loss of generality.

5.2 Joint friction identification and compensation experiment

In this section, the method described in Sect. 4.2 is used to identify and compensate for the joint friction. In the experiment, the second and third joints of the robot are used. The robot joint is controlled to move at different speeds in the range [−60°, 30°]. At the same time, the output torque of the joint is collected during the joint motion, and then the friction torque of the joint is calculated using Eq. (40). After filtering the singular points of the collected data, 40 sets of effective data in the speed range of 0–20° · s^{-1} are obtained. The fitting results obtained using Eq. (41) are shown in Fig. 4. The fitting parameters are listed in Table 1. It can be seen from Fig. 4 that the Stribeck model can effectively describe and compensate the joint friction. Similarly, the friction parameters of other joints can be obtained through experiments.

After its identification, the joint friction is further compensated to improve the performance of the collision detection algorithm. According to Refs. 15–17, the cut-off frequency of the low-pass filter is set to 19.5 Hz, which can filter out high-frequency signals in the robot system. The second and third joints of the robot are controlled to move and run the algorithm described in Sect. 4 in real time without collision. The observation results of the LPOB are shown in Fig. 5. We can see that the observation bias is significantly reduced by friction compensation. Simultaneously, the observation value has a larger mutation in the movement of the robot from stationary to motion. To eliminate the interference in robot collision detection, observation

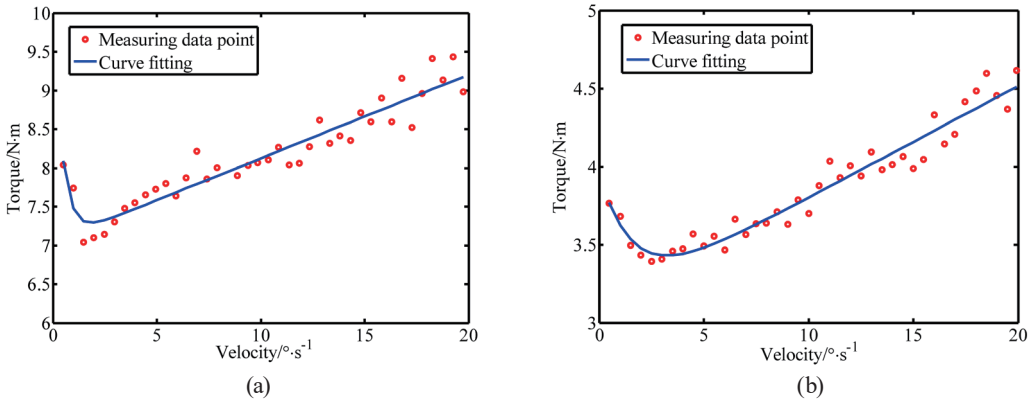

Fig. 4. (Color online) Friction data and curve fitting results: (a) robot joint 2 and (b) robot joint 3.

Table 1
Friction model curve fitting coefficient.

Parameters	λ_1	λ_2	λ_3	λ_4	Resnorm
Joint 2	7.0461	0.1078	3.0224	2.2674	1.4155
Joint 3	3.0848	0.0715	0.8797	0.6312	0.2783

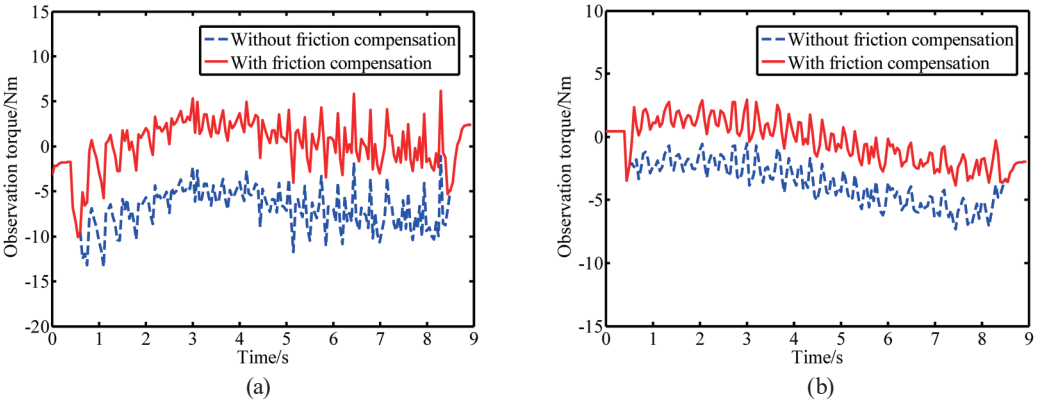

Fig. 5. (Color online) Observation results of LPOB without external forces: (a) robot joint 2 and (b) robot joint 3.

compensation can be calculated using Eq. (34). At the same time, other errors are unavoidable. To reduce the effects of the above interference factors on the observer and to prevent false detections, the thresholds of joints 2 and 3 in the LPOB are 8 and 5 Nm, respectively.

After determining the collision detection threshold of the algorithm of this paper, the observation effect of the external force is further compared through experiments. When the external force acts on link 3 of the robot, the observation results of the LPOB are as shown in Fig. 6. It can be seen from Fig. 6 that the sensitivity of the collision detection can be improved by friction compensation, and the purpose of friction compensation is achieved.

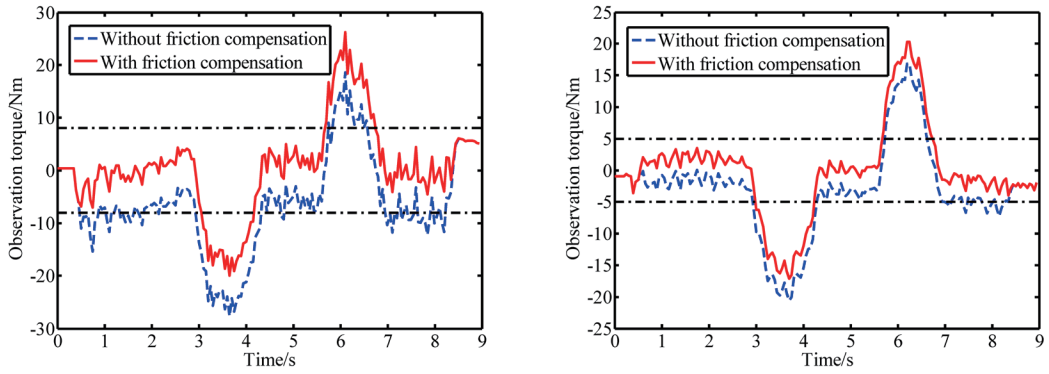

Fig. 6. (Color online) Observation results of LPOB with external forces: (a) robot joint 2 and (b) robot joint 3.

5.3 Observer performance comparison experiment

According to Refs. 15–17, the human contact force signal mainly distributes in the range of 0–4.6 Hz. Therefore, the cut-off frequencies of the bandpass filters can be chosen to be 4.6 and 19.5 Hz. Another experiment was conducted to verify the performance of the LPOB and BPOB. Controlling joints 2 and 3 of the robot moves from the start position [60°, −60°] to the end position [120°, 0°], and during the robot movement, the contact forces are applied by the human to the positive and negative directions of robot link 2 randomly. The collision force is applied by hitting robot link 2 with the rubber hammer to prevent harming the human body caused by direct collision. Similarly, the robot is controlled to move in the opposite direction, and the contact force and collision are also applied to link 3. When robot link 2 or 3 is subjected to external forces, the observation results of the LPOB and BPOB are as shown in Figs. 7 and 8, respectively.

It can be seen from Figs. 7(a) and 8(a) that the LPOB can observe different external forces, including contact force and collision force. However, it cannot distinguish the two by a single threshold. As shown in Figs. 7(b) and 8(b), the BPOB can filter out force signals with frequencies below 4.6 Hz and observe the force of high-frequency signals. It is suitable for observing external collisions of robots. In practical applications, the intentional contact force applied to the robot should be relatively low. Therefore, two force-sensing observers can be used to detect and distinguish contact and collision after setting the appropriate thresholds and filtering parameters. Furthermore, the LPOB and BPOB can select a more advanced filter to improve the filtering performance according to the actual application of the robot. From the observations in Figs. 7 and 8, when link 2 is subjected to an external force, joint 2 is subjected to a corresponding force and joint 3 is not affected by the external force. At this point, it can be seen from the observer results of the LPOB and BPOB that the observations of joint 2 will considerably change, while the observations of joint 3 will maintain a stationary value close to zero. However, when link 3 is subjected to an external force, both joints 2 and 3 are subjected to an external force, and both the observations of joints 2 and 3 will change. Therefore, according

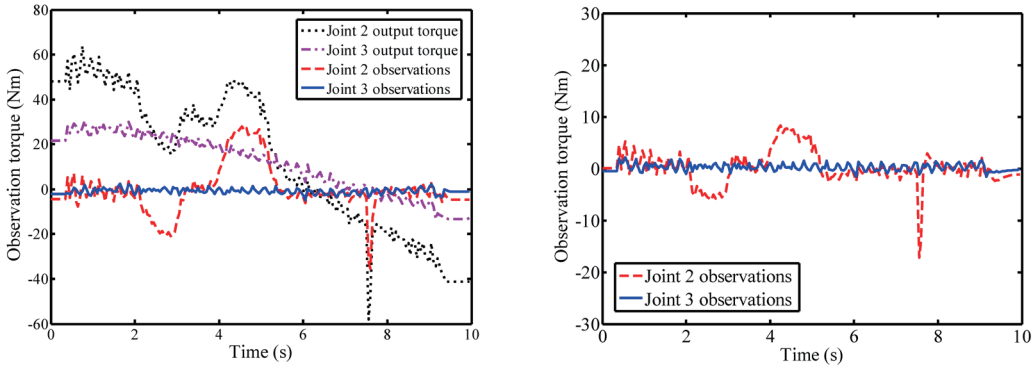

Fig. 7. (Color online) Observations when link 2 is subjected to external forces: (a) LPOB and (b) BPOB.

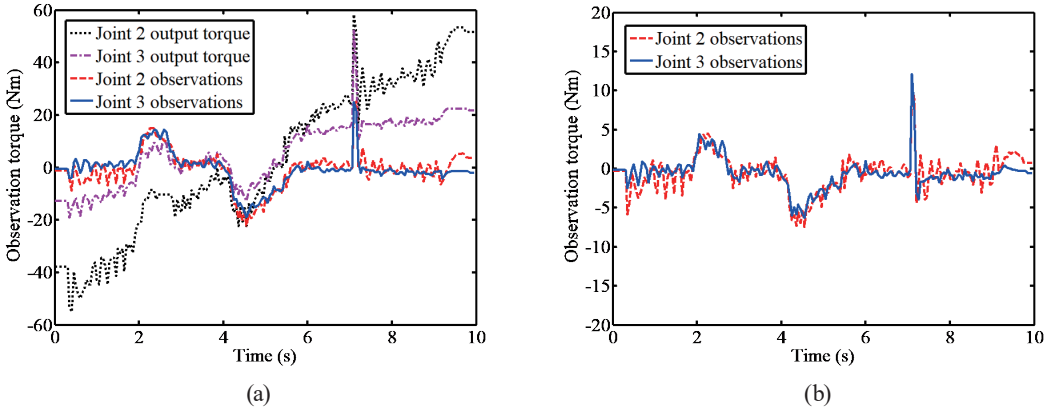

Fig. 8. (Color online) Observations when link 3 is subjected to external forces: (a) LPOB and (b) BPOB.

to Eqs. (32) and (33), the force-sensing observer designed in this study can detect external forces and identify the collided link.

5.4 Robot collision detection experiment

To verify the effectiveness of the collision detection algorithm proposed in this paper, the collision detection experiment based on the LPOB is designed. A screenshot of the experimental process is shown in Fig. 9. The robot works in an open environment as shown in Fig. 9(a). Figure 9(b) shows that the robot works according to the planned path without collision. Once the robot collides with the external environment, as shown in Figs. 9(c) and 9(d), the control system immediately takes safety measures to control the movement of the collided joint in the opposite direction of the collision to ensure the safety of the robot.

The real-time movement parameters of the robot during the experiment are shown in Fig. 10. The LPOB real-time observation results are shown in Fig. 11. When the robot works without collision, the observed value of the LPOB in Fig. 11 is less than the threshold. The observed value of the LPOB will increase rapidly after a collision. It can be seen from Fig. 11 that the observed values of joints 2 and 3 exceed the threshold. The observed results are consistent with

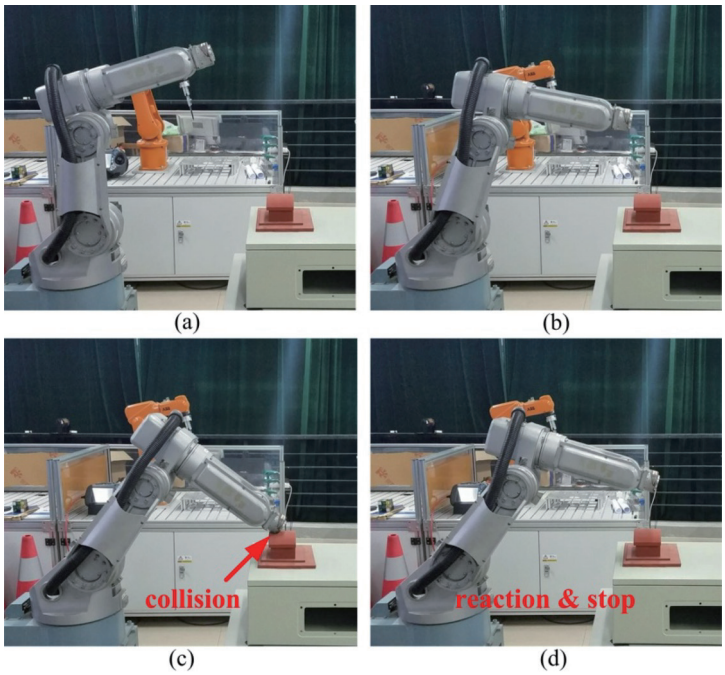

Fig. 9. (Color online) Robot collision detection experiment.

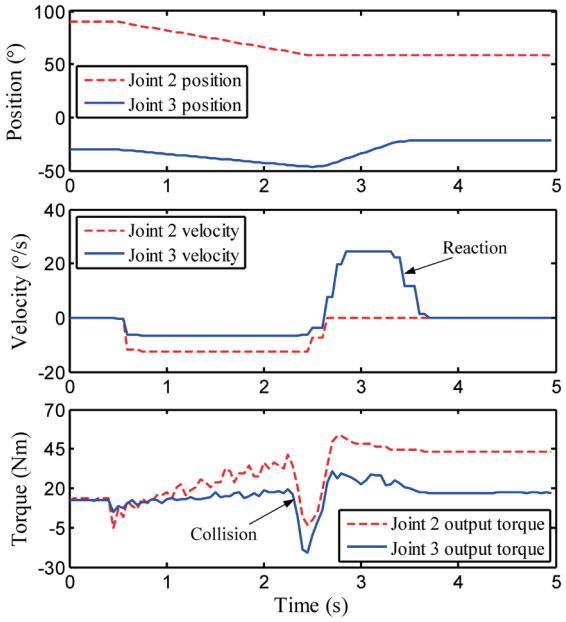

Fig. 10. (Color online) Robot motion parameters with collision.

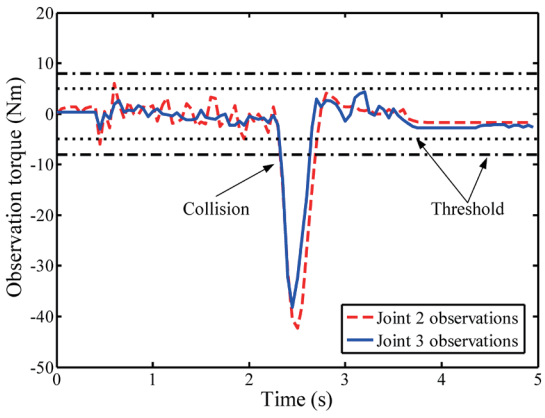

Fig. 11. (Color online) Observation results of LPOB.

the output torque of the joint, as shown in Fig. 10. According to Eq. (33), it can be determined that the collision occurred on the third link of the robot. The above experimental results show that the proposed algorithm can effectively detect the robot collision and enhance the safety of the robot.

5.5 Robot collision distinction experiment

During human–robot interaction, the force that the human body intentionally exerts on the robot may also be larger, and the threshold of the BPOB directly affects the efficiency of the algorithm. A low threshold can improve the sensitivity of the collision detection, which may also lead to a low working efficiency of the robot. A high threshold may make the robot less sensitive to collisions, causing pain, bruise body tissues, and so forth in the human body. To set a suitable threshold, five members of the laboratory intentionally push the robot 10 times. At the same time, the human contact force is observed through the BPOB. Considering the safety of the human body and the working efficiency of the robot, the threshold of 8.5 Nm is selected for the BPOB in this study. The threshold can be adjusted according to needs in practical applications.

Figure 12 shows a screenshot of the experimental results of the human–robot interaction. The initial position of the robot in the experiment is described in Fig. 12(a). Figure 12(b) shows the normal working state of the robot. At this time, the robot runs according to the planned

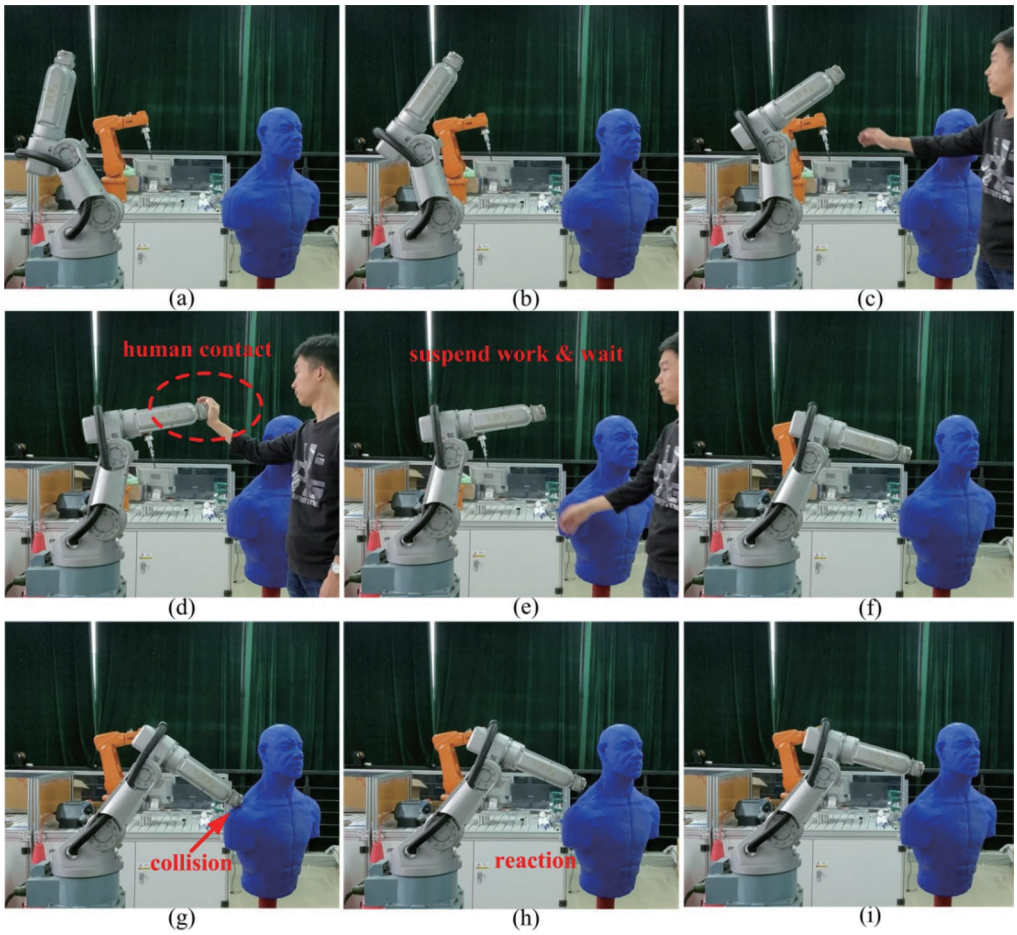

Fig. 12. (Color online) Detection and distinction of contact and collision between human and robot.

trajectory and speed. Figure 12(c) shows the human and the robot sharing a limited working space. In the human–robot interaction, it is necessary for the human to physically contact the robot. This contact force is controlled by the human body without danger, as shown in Fig. 12(d). To avoid false collision detection and ensure the safety of the human and robot, the force from the intentional contact exerted by the human body is judged whether it occurs after 0.5 s and then stops the current task. When the contact force of the human body disappears, the robot continues to wait for 2 s to ensure that the human body has moved to a safe area. Then, the robot continues to complete the interrupted task, as shown in Figs. 12(e) and 12(f). In Figs. 12(g) and 12(h), once it detects a collision with a dummy in the workspace, the robot then immediately undertakes a safe reaction strategy to ensure its safety and the safety of the human body. As shown in Fig. 12(i), when it reaches a safe area, the robot will stop running and wait for the operator to further check and eliminate hidden dangers before restarting it.

The motion parameters of the robot during the experiment are shown in Fig. 13. The observations of the BPOB and LPOB are shown in Figs. 14 and 15, respectively. When there is no external force, the BPOB and LPOB observations in both Figs. 14 and 15 are less than the threshold. When the robot is intentionally pushed, the joint output torque of the robot generates a corresponding change, as shown in Fig. 13. The comparison of Figs. 14 and 15 shows that the BPOB is insensitive to the contact force signal, which indicates that it attenuates the force signal with a frequency lower than 4.6 Hz, and that the LPOB can observe all external forces. The experimental results are consistent with the simulation results, and the robot can recognize that it has been subjected to external forces. When a collision occurs, the joint output torque of the robot abruptly changes as shown in Fig. 13. The BPOB observations in Fig. 14 quickly change

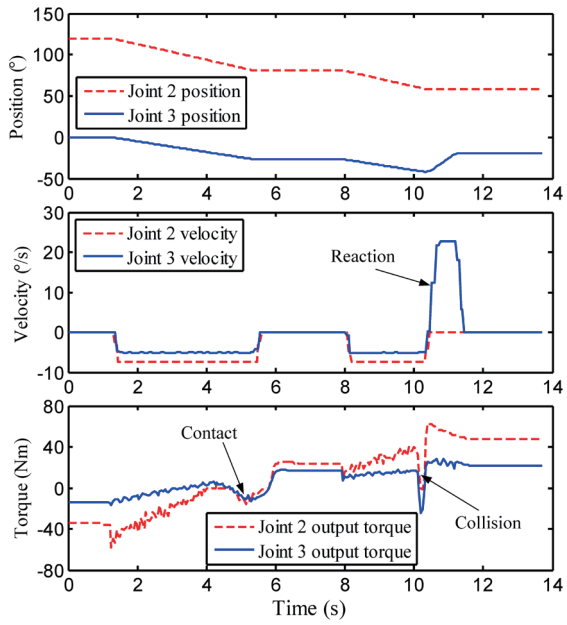

Fig. 13. (Color online) Robot motion parameters in human–robot interaction experiment.

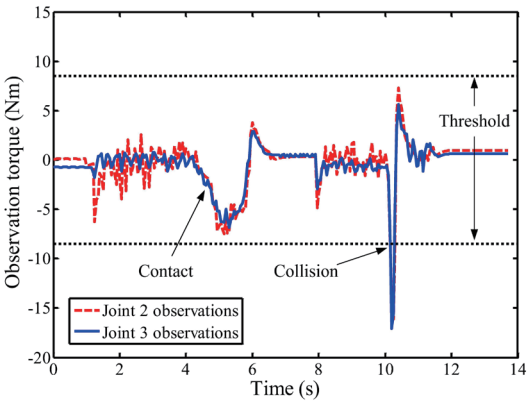

Fig. 14. (Color online) Observation results of BPOB in human–robot interaction experiment.

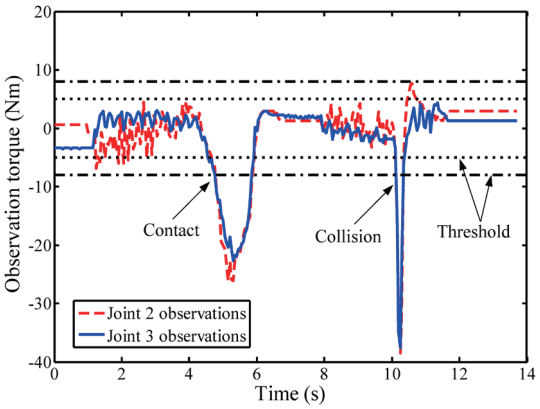

Fig. 15. (Color online) Observation results of LPOB in human-robot interaction experiment.

and increase beyond the threshold. It can be determined that the robot had an unexpected collision. Additionally, the algorithm proposed in this paper can filter out slight unexpected contact or disturbances from load changes in the process of human–robot interaction and prevents the robot from producing an overreaction, frequently abandoning the task or restarting the system.

From the above experimental results and analysis, it can be seen that the proposed collision detection algorithm can effectively achieve collision detection and improve the safety of the robot. At the same time, from the different frequency distributions of force signals, the LPOB and BPOB can work together to detect and distinguish the contact and collision between the human and the robot. Moreover, the proposed collision distinction method can be further combined with the corresponding control algorithm to realize the switching of the robot working mode. For example, it can be used for robot automatic or manual switching control and reduce the robot system hardware costs. Therefore, this method has an application prospect in the field of physical human–robot interaction and coexisting cooperative cognitive robots. The proposed algorithm in this paper is a model-based approach. In practical applications, an accuracy dynamic model is required to improve the detection and distinction performance. As an example, the parameter identification can be used to reduce the dynamic parameter deviation and improve the detection sensitivity. In addition, the proposed collision distinction method cannot be used in tasks that contain high-frequency disturbance such as grinding and polishing.

6. Conclusions

In this paper, we proposed a collision detection algorithm based on the convolution filtering dynamic model. A force-sensing observer was designed in this algorithm to realize robot collision detection. The force-sensing observer does not need to calculate acceleration, which eliminates the disadvantage of introducing large calculation errors owing to the acquisition

of joint acceleration. It only needs to sample the driving torque and position information of conventional robot joints, which is beneficial in reducing the cost of the robot system. Moreover, a collision distinction method was further designed on the basis of the force-sensing observer. By using the different frequency distributions of the contact force and collision force, and combining an LPOB with a BPOB, we can realize the real-time detection and distinction of the contact and collision in the field of human–robot interaction. At the same time, this collision distinction method can detect and distinguish the external forces on both the robot and the body. External force sensing experiments show that the force-sensing observer works well and can be used as a virtual force sensor. The experimental results of collision detection between a robot and an unknown environment show that the collision detection algorithm has stable performance and high reliability. The human–robot interaction experiment verified the effectiveness of the proposed collision distinction method. Therefore, the collision detection algorithm proposed in this paper can be used as a method for robot collision detection in practical applications. The collision distinction method can provide an important reference for distinguishing the contact and collision in physical human–robot interaction. Future work will focus on improving the speed of collision detection and distinction, and applying the algorithm to human–robot cooperation.

Acknowledgments

This work was supported by the National Key Research and Development Plan (No. 2018YFB1308603) and the National Natural Science Foundation of China (Nos. 51575111 and 51605093).

References

1 S. Haddadin, A. De Luca, and A. Albu-Schäffer: IEEE Trans. Robot. **33** (2017) 1292.
2 V. Villani, F. Pini, F. Leali, and C. Secchi: Mechatronics **55** (2018) 248.
3 A. De Luca and F. Flacco: Proc. Int. Conf. Biomedical Rob. Biomechatronics (IEEE, 2012) 288–295.
4 F. Flacco, T. Kroeger, A. D. Luca, and O. Khatib: J. Intell. Rob. Syst. **80** (2015) 7.
5 M. Beetz, G. Bartels, A. Albu-Schaffer, F. Balint-Benczedi, R. Belder, D. Bebler, S. Haddadin, A. Maldonado, N. Mansfeld, and T. Wiedemeyer: Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst. (IEEE, 2015) 6528–6535.
6 H. Wu, J. Chen, Y. Su, Z. Li, and J. Ye: Sens. Actuator A **242** (2016) 146.
7 L. Zeng and G. M. Bone: Mech. Mach. Theory **60** (2013) 1.
8 S. Wolf, G. Grioli, O. Eiberger, W. Friedl, M. Grebenstein, H. Hoppner, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano, D. Lefeber, S. Stramigioli, N. Tsagarakis, M. V. Damme, R. V. Harm, B. Vanderborght, L. C. Visser, A. Bicchi, and A. Albu-Schaffer: IEEE/ASME Trans. Mechatron. **21** (2016) 2418.
9 Z. Xie, L. Cao, and L. Hong: Int. J. Intell. Unman. Syst. **1** (2013) 62.
10 Z. J. Li, H. B. Wu, J. M. Yang, M. H. Wang, and J. H. Ye: Int. J. Autom. Comput. **15** (2018) 1.
11 M. Indri, S. Trapani, and I. Lazzero: Sensors **17** (2017) 1148.
12 J. Huang, Z. Xie, M. Jin, Z. Jiang, and H. Liu: Chin. J. Aeronaut. **22** (2009) 105.
13 H. O. Lim and K. Tanie: IEEE/ASME Trans. Mechatron. **4** (1999) 417.
14 S. Robla-Gómez, V. M. Becerra, J. R. Llata, E. González-Sarabia, C. Torre-Ferrero, and J. Pérez-Oria: IEEE Access PP (2017) 1.
15 M. Geravand, F. Flacco, and A. D. Luca: Proc. IEEE Int. Conf. Rob. Autom. (IEEE, 2013) 4000–4007.
16 S. Golz, C. Osendorfer, and S. Haddadin: Proc. IEEE Int. Conf. Rob. Autom. (IEEE, 2015) 3788–3794.
17 A. Kouris, F. Dimeas, and N. Aspragathos: IEEE Rob. Autom. Lett. PP (2018) 1.
18 W. E. Dixon, I. D. Walker, D. M. Dawson, and J. P. Hartranft: IEEE Trans. Rob. Autom. **16** (2000) 689.

19 M. V. Damme, P. Beyl, B. Vanderborght, and V. Grosu: Proc. IEEE Int. Conf. Rob. Autom. (IEEE, 2011) 1108–1113.
20 S. D. Lee, M. C. Kim, and J. B. Song: Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst. (2015) 1581–1589.
21 A. C. Bittencourt and S. Gunnarsson: J. Dyn. Syst. Meas. Contr. **134** (2012) 1581.
22 S. He, J. Ye, Z. Li, S. Li, G. Wu, and H. Wu: Proc. IEEE Int. Conf. Rob. Biom. (IEEE 2015) 1253–1259.