Robotica: (2020) volume 38, pp. 1665–1681. © Cambridge University Press 2019
doi:10.1017/S0263574719001681

Unknown External Force Estimation and Collision Detection for a Cooperative Robot

Shirin Yousefizadeh* and Thomas Bak

Department of Electronic Systems, Aalborg University, Fredrik Bajres Vej 7C, Aalborg Øst 9220, Denmark. E-mail: tba@es.aau.dk

(Accepted October 26, 2019. First published online: December 20, 2019)

SUMMARY
In human–robot cooperative industrial manipulators, safety issues are crucial. To control force safely, contact force information is necessary. Since force/torque sensors are expensive and hard to integrate into the robot design, estimation methods are used to estimate external forces. In this paper, the goal is to estimate external forces acting on the end-effector of the robot. The forces at the task space affect the joint space torques. Therefore, by employing an observer to estimate the torques, the task space forces can be obtained. To accomplish this, loadcells are employed to compute the net torques at the joints. The considered observers are extended Kalman filter (EKF) and nonlinear disturbance observer (NDOB). Utilizing the computed torque obtained based on the loadcells measurements and the observer, the estimates of external torques applied on the robot end-effector can be achieved. Moreover, to improve the degree of safety, an algorithm is proposed to distinguish between intentional contact force from an operator and accidental collisions. The proposed algorithms are demonstrated on a robot, namely WallMoBot, which is designed to help the operator to install heavy glass panels. Simulation results and preliminary experimental results are presented to demonstrate the effectiveness of the proposed methods in estimating the joint space torques generated by the external forces applied to the WallMoBot end-effector and to distinguish between the user-input force and accidental collisions.

KEYWORDS: External force estimation; Collision detection; Cooperative robot; Extended Kalman filter (EKF); Nonlinear Disturbance Observer (NDOB).

1. Introduction

Traditional industrial manipulators have caging around them or are deployed in structured environments to ensure safety. However, many new robotic applications involve a close cooperation between humans and robots.[1] Cooperative tasks may entail ergonomic benefits for operators, such as reduced working load, improved body posture, or fewer repetitive movements.[1] Additionally, combining human dexterity with the strength of the robot may improve productivity and efficiency.[2] Furthermore, considering safety issues in human–robot cooperation is of great importance. Safety requirements for collaborative operations are provided in the technical specification ISO/TS 15066.

The conventional way of controlling industrial robots is to program them to follow the desired trajectories.[3] This approach is named pure position control; pure position control is inadequate or even unstable for tasks that involve motions in uncertain environments.[4,5] Since accidental collisions are probable when a robot is intended to work alongside an operator, force control is needed. Force control demands supervision of external forces exerted on the manipulator,[6] which can be accomplished

* Corresponding author. E-mail: shy@es.aau.dk

with external sensors such as capacitive skin[7] or force sensors. Since these sensors can be very expensive,[8] force estimation has attracted attention in the research communities. A rather intuitive force estimation scheme is subtracting the nominal model command, that is, the expected torque in the absence of external forces, from the commanded torque. Even assuming the robot dynamics is completely known, this method uses inverse dynamics computation, which requires acceleration estimation. In such acceleration-based methods, computing the derivatives from the empirical measurements may degrade the estimation process.[9,10] Another approach for force estimation is to use an observer. Two main kind of observers are stochastic and deterministic observers. One kind of deterministic observers is the disturbance observer (DO), in which the contact force is treated as an unknown disturbance force. Then, the equivalent disturbance is estimated by modifying the difference between the estimated output and the output of the nominal response model.[11,12] DOBs guarantee converging state estimations to the vicinity of actual states.[13] However, the performance of the DOs can be degraded in the presence of measurement noises.[14–16] Moreover, since the estimated disturbance in DOs lumps the effects of external disturbances and model uncertainties, a good dynamic model of the system is required to ensure a good external disturbance estimation.[17] Even though stochastic estimators, such as Kalman filtering (KF) and its derivatives, are developed to be tolerant to process and measurement noises, there is no guarantee that state estimates converge to actual ones. KF is an algorithm to reduce noise effects in the state estimation of linear discrete-time systems.[18] EKF is an extension of the Kalman filter to estimate the states of nonlinear dynamic systems,[19] by linearizing the nonlinear process at each time step.[18] In general, robots have highly nonlinear and coupled dynamics. However, the nonlinearities are based on sines and cosines, which points out that linearization and thus ultimately, the EKF should perform well.[20]

Several researchers have attempted to estimate the external forces without using force sensors. In ref. [21], a NDOB is proposed for estimating the contact force between an ear surgical device and the subject body. However, due to sensor noise, some spikes are appeared in the force estimation. Chan et al.[12] have employed a variation of Kalman filter, namely extended active observer, to estimate external forces. The approach is applied to a 2DOF robotic manipulator through computer simulation but it lacks experimental results. In ref. [22], neural networks are used to estimate the contact force in a haptic device. Even though this method does not require knowledge of the dynamical model of the robot, it has a high computational burden. Moreover, its performance depends on the accuracy of the sensed forces during the training phase.

The WallMoBot is a construction robot. Construction robots must be able to move and adjust movements within a changing environment and handle large components.[23] The WallMoBot is supposed to assist the operator to lift, transport and install glass panels of up to 150 kg with an accuracy of 1.5 mm. Generally, the glass panel installation is done by a joystick-controlled robot[2] or manually. Manual installation of glass panels involves heavy lifting and might cause injury and fatigue to the operator.[24] Additionally, installation with a joystick-controlled robot is not intuitive for the operator,[25] and installation takes a too long time. The proposed solution is a robot that is first controlled by a joystick for large movements, and then when the glass is close to the installation place, the robot is controlled by the operator force exerted to the glass panel attached to the end-effector of the robot. By using this solution, the glass panel installation can be done faster and more precise.

In this paper, we focus on the interaction of the human operator with the robot, when the human operator applies forces directly to the glass panel. Instead of using expensive force/torque sensors, cost-effective loadcells are used to estimate the torques acting at the joints, which are introduced by the external forces applied to the end-effector. Also, an EKF and a NDOB are developed to estimate the aforementioned torques, by utilizing the loadcells information, as well as position and velocity measurements. The estimated torques can then be converted into task space forces. Also, in order to take into account the safety, an algorithm has been proposed to differentiate accidental collisions and the intentional contact force from the operator. The main contributions and advantages of this paper can be summarized as follows:

(1) To make the WallMoBot a more economical product, instead of using expensive force/torque sensors, cost-effective loadcells are utilized at the WallMoBot joints for force estimation.
(2) To make the WallMoBot operation intuitive for the human operator, we have developed an EKF and a NDOB to estimate the external forces acting on the robot end-effector by using the information of non-ideal cost-effective loadcells. In the proposed approaches, there is no need

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

Table I. Parameters for kinematics, where $\gamma = 0.3458$ rad.

i	$a_{i\text{-}1}$ (m)	d_i (m)	$\theta_i = q_i +$ offset (rad)	α_{i-1} (rad)
1	0	0	$q_1 + 0$	0
2	$a_1 = 0.06$	0	$q_2 - \pi/2$	$-\pi/2$
3	$a_2 = 0.5$	0	$-q_2 + \gamma$	0
4	$a_3 = 0.195$	0	$q_3 + \pi/2 - \gamma$	0
5	$a_4 = 0.85$	0	$q_4 + \pi/2$	0
6	$a_5 = 0.93$	$d_6 = 0.109$	$q_5 + \pi/2$	$\pi/2$
7	0	0	$q_6 - \pi/2$	$-\pi/2$

to compute the angular acceleration from the angular velocity. Also, depending on the desired performance criteria, both proposed techniques are reliable and economical in practice.

(3) To make the WallMoBot operation safe, a simple collision detection is proposed. It takes the estimated torque from the EKF or the NDOB algorithm as the input and, when a collision occurs, gives a fiducial mark as the output. This fiducial mark can be used to stop the robot movement.

The validity of the proposed force estimation scheme and collision detection algorithm is verified by simulations and experimentation. The outline of this paper is as follows. The modeling of the WallMoBot is provided in Section 2. In Section 3, EKF and NDOB algorithms are adapted, developed, and proposed for force estimation. Section 4 presents the proposed collision detection algorithms. The illustrative simulation results and the preliminary experimental results are presented in Sections 5 and 6, respectively. Discussion of reliability and effectiveness of the estimation methods on the WallMoBot is presented in Section 7. Finally, Section 8 concludes the paper.

2. Modeling

WallMoBot is a robot arm with six degrees of freedom on the top of a mobile platform which is shown in Fig. 1. The coordinate frames $o_i x_i y_i z_i$, $\forall \in \{0, \ldots, 6, 7, e\}$, are assigned to the robot by complying with the Denavit–Hartenberg (DH) convention, and they are visualized in Fig. 1. Coordinate frames 0 and e denote the base frame and the end-effector frame, respectively, and coordinate frame 3 is associated with the parallelogram link which keeps the same orientation as the base frame. In this paper, we consider the simple case by assuming the robot is planar. It means that the generalized coordinates q_1, q_5, q_6 are assumed to be zero and only the coordinates q_2, q_3, q_4, as shown in Fig. 1, are used. In other words, three degrees of freedom of the robot are considered. The DH parameters required to obtain the kinematics of the robot are found in Table I.

The dynamic model of the robot is computed using Lagrange formulation. The Lagrange equations are expressed as

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}}\right)^T - \left(\frac{\partial \mathcal{L}}{\partial q}\right)^T = \zeta \tag{1}$$

where q is a vector of generalized coordinates, ζ is a vector of generalized torques/forces, and L is the Lagrangian of the mechanical system defined as

$$\mathcal{L}(q, \dot{q}) := T(q, \dot{q}) - U(q) \tag{2}$$

where $\mathcal{L}(q, \dot{q})$ is the kinetic energy and $U(q)$ is the potential energy of the system.
Deriving the Lagrange equations, the dynamic model of the robot can be obtained as

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(\dot{q}) + g(q) + \tau_{ext} = \tau \tag{3}$$

where $B(q)$ is the symmetric inertia matrix, $C(q, \dot{q})$ is the matrix of centrifugal, and Coriolis terms, $g(q)$ contains the gravity terms, $F(\dot{q})$ represents viscous and coulomb friction torques at the joints, τ_{ext} is the unknown joint space torque vector caused by the external forces acting on the robot end-effector, and τ is the joint space actuation torque vector. Mathematically, the joint space torques generated by the external forces acting on the robot end-effector could be estimated by

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

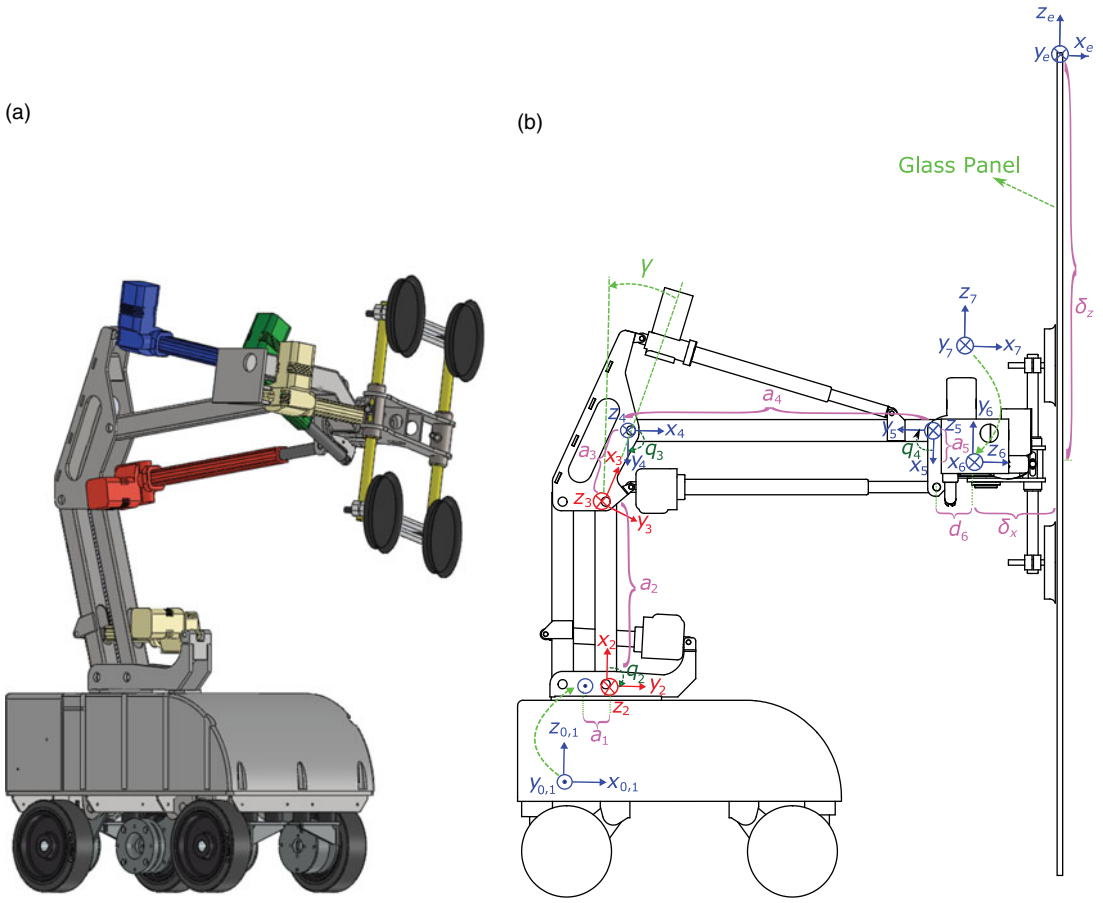

Fig. 1. (a) Illustration of the considered WallMoBot.[2] (b) Illustration of the generalized coordinates on the WallMoBot. Frame 0 coincides with frame 1 when q_1 is zero; where γ is the offset angle between frame 2 and 3, $\delta_z = 1.5\,\text{m}$ is the height of the glass panel from the attachment point to the top and $\delta_x = 0.2025\,\text{m}$ is the distance of the glass panel from the final joint of the robot.

$$\hat{\tau}_{ext} = \tau - B(q)\ddot{q} - C(q, \dot{q})\dot{q} - F(\dot{q}) - g(q) \tag{4}$$

As it can be seen from (4), $\hat{\tau}_{ext}$ calculation needs the value of $\ddot{q}$ which must be measured by an acceleration sensor or empirically computed from the $\dot{q}$. However, installing a new sensor increases the cost, and empirical derivative computation produces error and amplifies the noise. Therefore, in the following, an EKF algorithm is developed to avoid using any extra sensor and amplifying noise and enhance the noise mitigation.

3. Force Estimation

The purpose of this section is to present the development of the EKF and NDOB, which are used to estimate the joint space torques caused by the applied forces to the robot end-effector.

3.1. Extended Kalman filter

The state-space model for the equations of motion can be written as

$$\dot{z} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ B(q)^{-1}(\tau - C(q, \dot{q})\dot{q} - F(\dot{q}) - g(q) - \tau_{ext}) \end{bmatrix} \tag{5}$$

where τ is measured by the loadcells. To estimate joint space torques produced by the external forces acting on the robot end-effector, they should be included in the states of the EKF.[26] To do this, we define the extended state vector x, including τ_{ext} vector, as

$$x = \begin{bmatrix} q \\ \dot{q} \\ \tau_{ext} \end{bmatrix} \tag{6}$$

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

The τ_{ext} dynamic is unknown and it can be produced by intentional human forces or accidental forces. Therefore, it is assumed that $\dot{\tau}_{ext} = 0$. The new augmented state-space model is

$$\dot{x} = \begin{bmatrix} \dot{q} \\ B(q)^{-1}(\tau - C(q, \dot{q})\dot{q} - F(\dot{q}) - g(q) - \tau_{ext}) \\ 0 \end{bmatrix} = f(x, \tau) \tag{7}$$

Considering (6), the measured outputs of the system are described in the equation

$$y = \begin{bmatrix} I\ 0\ 0 \\ 0\ I\ 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \\ \tau_{ext} \end{bmatrix} \tag{8}$$

Putting (7) and (8) together and considering system noise, w, and measurement noise, v, with covariance matrices Q and R, respectively, yields to

$$\begin{cases} \dot{x} = f(x, \tau) + w \\ y = Hx + v \end{cases} \tag{9}$$

Considering T_s as the sampling time, the continuous augmented state-space model equation can be discretized using the forward Euler method as

$$\begin{cases} x_{k+1} = x_k + T_s f(x_k, \tau_k) + w_k \\ y_k = Hx_k + v_k, \end{cases} \tag{10}$$

where

$$f(x_k, \tau_k) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} \dot{q}_k \\ B(q_k)^{-1}\left(\tau_k - C(q_k, \dot{q}_k)\,\dot{q}_k - F(\dot{q}_k) - g(q_k) - \tau_{ext_k}\right) \\ 0 \end{bmatrix} \tag{11}$$

Therefore, the goal is to estimate the state vector x_k by the following iterative EKF procedure:

Due to the nonlinear nature of the process, f cannot be used directly. Rather the Jacobian of f is used. It is defined as

$$F_k = \frac{\partial f}{\partial x}|_{(\hat{x}_k, \tau_k, k)} \tag{12}$$

Then, the EKF algorithm is as follows:

- *Time Update*

$$\begin{aligned} \hat{x}_k^- &= \hat{x}_{k-1} + T_s f(\hat{x}_{k-1}, \tau_k) \\ P_k^- &= F_{k-1} P_{k-1} F_{k-1}^T + Q_{k-1} \end{aligned} \tag{13}$$

- *Measurement Update*

$$\begin{aligned} K_k &= P_k^- H^T (H P_k^- H^T + R_k)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k (y_k - H\hat{x}_k^-) \\ P_k &= (I - K_k H) P_k^-, \end{aligned} \tag{14}$$

where

- $\hat{x}_k^-$, P_k^- are the predicted states and the predicted covariance of the states, respectively, on time step k, before considering the measurement.
- $\hat{x}_k$, P_k are the estimated states and the estimated covariance of the states, respectively, on time step k, after considering the measurement.
- K_k is the filter gain, which tells how much the predictions should be corrected on time step k.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

Finally, the linearized system can be obtained as

$$F_k = I + T_s \frac{\partial f(x_k, \tau_k)}{\partial x}, \tag{15}$$

where

$$\frac{\partial f(x_k, \tau_k)}{\partial x} = \begin{bmatrix} \frac{\partial f}{q} & \frac{\partial f}{\partial \dot{q}} & \frac{\partial f}{\partial \tau_{ext}} \end{bmatrix} \tag{16}$$

By means of the EKF, and only measuring q, $\dot{q}$, and τ, the estimation of τ_{ext} is achieved by being extracted from the state vector x_k.

Remark: Usually, the initial value of the matrix P, that is the covariance of the states error, is diagonal whose diagonal elements are related to the expected variance of the corresponding state. A good guess of the initial values of the states, that is, $\hat{x}_0$, needs a small initial value of the covariance of the states, that is, P_0. Therefore, since the position and velocity of the WallMoBot are measurable, one can choose small corresponding covariance values. On the other hand, the artificial states τ_{ext} are unknown and low information of them is available. Consequently, their corresponding elements of P should have larger values.

The process noise covariance matrix, that is, Q, on one hand, corresponds to system noise covariance and on the other hand corresponds to the uncertainty that is expected in the state-space equations. This could include modeling errors or other uncertainties in the equations themselves. Additionally, the larger (smaller) value of the Q corresponds to faster (slower) convergence by the expense of larger (smaller) steady-state error.[27] Since the last three elements correspond to the external torques, in which their dynamics are completely unknown, larger values of system noise covariance are required.

Also, based on the sensor noise and iterative testing, the covariance matrix of the measurement noise, that is, R, is chosen as below to make the EKF more tolerant to noise.

3.2. Nonlinear disturbance observer

Inspired from ref. [17], the NDOB dynamic equations considered for the system model (3) is in the form

$$\begin{aligned} \dot{z} &= L_d(.)\{-z + C\dot{q} + F + g - \tau - s(.)\} \\ \hat{\tau}_{ext} &= z + s(.) \end{aligned} \tag{17}$$

where $C(q, \dot{q})$, $F(\dot{q})$, $g(q)$ are written as C, F, g for brevity, $z \in R^l$ is the internal state vector of the NDOB, $L_d(.) \in R^{(l \times l)}$ and $s(.) \in R^l$ are the gain matrix and the auxiliary vector of the NDOB, respectively, and $\hat{\tau}_{ext} \in R^l$ is the estimated external torque vector at the joint space.

The following theorem is presented to find $L_d(.)$ and $s(.)$ in order to minimize the estimation error of the unknown external torque and to prove the stability of the estimation error in the sense of Lyapunov. In this theorem, the asymptotic stability of the estimation error for constant joint space external torques and the boundedness of the error in the case of time varying joint space external torques are guaranteed.

Theorem 1. *The unknown external force in* (3) *can be estimated by the NDOB* (17) *with the following parameters:*[28]

$$\begin{aligned} s &= X\dot{q} \\ L_d &= XB^{-1}(q) \end{aligned} \tag{18}$$

where X is a diagonal positive definite matrix.

Proof. Defining the joint space external torque tracking error as $\Delta\tau_e = \hat{\tau}_{ext} - \tau_{ext}$, its dynamics can be obtained using (3) and (17) as

$$\Delta\dot{\tau}_e = L_d\{-z + C\dot{q} + F + g - \tau - s\} + \dot{s} - \dot{\tau}_{ext} \tag{19}$$

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

Since $C\dot{q} + F + g - \tau = -B\ddot{q} + \tau_{ext}$, one has

$$\Delta\dot{\tau}_e = -L_d\Delta\tau_e - L_dB\ddot{q} + \dot{s} - \dot{\tau}_{ext} \tag{20}$$

To make the tracking error of the joint space external torque be independent from $\ddot{q}$, the constraint $\dot{s} = L_dB\ddot{q}$ should be satisfied. Based on the chain-derivative rule, one has $\dot{s} = \frac{\partial s}{\partial q}\dot{q} + \frac{\partial s}{\partial \dot{q}}\ddot{q}$, which yields to

$$\frac{\partial s}{\partial q}\dot{q} + \frac{\partial s}{\partial \dot{q}}\ddot{q} = L_dB\ddot{q} \tag{21}$$

which necessitates that $\frac{\partial s}{\partial q} = 0$ and $\frac{\partial s}{\partial \dot{q}} = L_dB$. Based on the fact that B is invertible, we have

$$L_d = \frac{\partial s}{\partial \dot{q}}B^{-1} \tag{22}$$

A trivial selection of (22) is as (18). Consequently, (20) is continued as

$$\Delta\dot{\tau}_e = -L_d\Delta\tau_e - \dot{\tau}_{ext} \tag{23}$$

To analyze the stability of the joint space external torque error dynamics, the following Lyapunov function is considered:

$$V = \frac{1}{2}\Delta\tau_e{}^TB\Delta\tau_e \tag{24}$$

The time derivative of (24) along the trajectory (23) provides

$$\dot{V} = \Delta\tau_e{}^TB\Delta\dot{\tau}_e = -\Delta\tau_e{}^TBXB^{-1}\Delta\tau_e - \Delta\tau_e{}^TB\dot{\tau}_{ext} \tag{25}$$

which using the following inequalities

$$\begin{gathered}\sigma_{min}(X)\,\|\Delta\tau_e\|^2 \le \Delta\tau_e{}^TB\Delta\dot{\tau}_e \le \sigma_{max}(X)\,\|\Delta\tau_e\|^2\,, \\ \Delta\tau_e{}^TB\dot{\tau}_{ext} \le \|\Delta\tau_e\|\,\|B\|\,\|\dot{\tau}_{ext}\|\end{gathered} \tag{26}$$

results in

$$\dot{V} \le -\sigma_{min}(X)\,\|\Delta\tau_e\|^2 + \|\Delta\tau_e\|\,\|B\|\,\|\dot{\tau}_{ext}\| \tag{27}$$

where $\sigma_{min}(X)$ and $\sigma_{max}(X)$ stand for the minimum eigenvalue and the maximum eigenvalue of the matrix X, respectively. The constraint (27) indicates that the time-derivative of the Lyapunov function (24) is negative in the region

$$D = \left\{\Delta\tau_e |\,\|\Delta\tau_e\| > \frac{\|B\|\,\|\dot{\tau}_{ext}\|}{\sigma_{min}(X)}\right\} \tag{28}$$

Assuming the time derivatives of τ_{ext} and $\dot{q}$ are bounded, respectively, by unknown limits $\bar{q}$ and $\bar{\tau}$, that is, $|\dot{\tau}_{ext}| \le \bar{\tau}$ and $|\dot{q}| \le \bar{q}$, since for the region D, V is positive definite and $\dot{V}$ is negative definite, the tracking error is bounded by

$$\|\Delta\tau_e\| = \frac{\|B\|\,\|\dot{\tau}_{ext}\|}{\sigma_{min}(X)} \tag{29}$$

Thereby, the proof is completed.

Remark. The selection of the matrix X affects the convergence speed and oscillation of the estimations. To illustrate the effect of the X on the convergence speed, (25) should be further analyzed. The solution of (25) with respect to time can be obtained of the form ref. [29]

$$\Delta\tau_e(t) < e^{-\sigma_{min}(X)t}\Delta\tau_e(0) + \Gamma \tag{30}$$

where Γ is a constant value dependent to the X, $\bar{\tau}$, $\|B\|$, and $\|\Delta\tau_e\|$.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

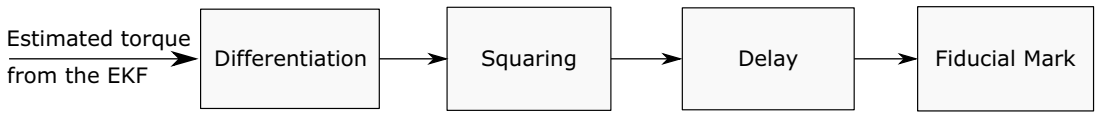

Fig. 2. Block diagram of the collision detection algorithm.

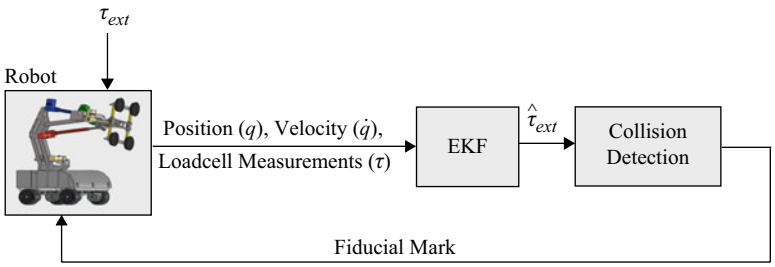

Fig. 3. Overall block diagram of the proposed approaches.

It is evident from (30) that increasing X results to increasing the convergence speed. On the other hand, from (3), (17), and (18), one has:

$$\dot{z} = -X\ddot{q} + XB^{-1}(\tau_{ext} - \hat{\tau}_{ext}), \tag{31}$$

where $\tau_{ext} - \hat{\tau}_{ext}$ denotes the correction term. It can be seen that by increasing X, the effect of the correction term on the $\dot{z}$ is increased which based on (17), results in higher oscillations in $\hat{\tau}_{ext}$. Therefore, there is a trade off between the convergence speed and the oscillation of the estimations. In other words, the larger (smaller) value of the X corresponds to faster (slower) convergence by the expense of larger (smaller) oscillation.

The estimated τ_{ext} using EKF and NDOB can be converted into the task space force, using a Jacobian matrix or geometrical calculations.

4. Collision Detection

The other goal of this paper is to discriminate between human forces and accidental collisions, which is a necessity for safe operation of cooperative robots. Since the WallMoBot is designed to carry glass panels with different weights, using a threshold for collision detection is not suitable in this case. Instead, a method is required to detect collisions regardless of the glass panel's weight. Flash and Hogan[30] found that human arm performs minimum jerk movements. Based on this, a collision detection algorithm has been developed as the block diagram shown in Fig 2. The proposed algorithm is inspired by the Pan–Tompkins Method,[31] which proposes an algorithm to detect peaks in the electrocardiography signal. The steps of the algorithm are described below.[32]

- *Differentiation*: The derivative operator suppresses the low-frequency components and enlarges the high-frequency components from the high slopes.
- *Squaring*: The squaring operation makes the results positive and emphasizes large differences and makes the small differences being suppressed.
- *Delay*: The delay makes to keep the previous value for a time which makes it easier to catch up with fast changes.
- *Fiducial Mark*: Fiducial point is defined as the location of the peak which is detected by defining a threshold.

The output of the fiducial block indicates if the robot collides an obstacle and then is used to stop the robot. The overall block diagram of the proposed approaches is shown in Fig. 3.

5. Simulation

In this section, we demonstrate the proposed EKF, NDOB, and collision detection methods.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

Table II. Robot parameters, where COM_i and Izz_i are the relative position vector of the center of mass to the ith joint, and the moment of inertia about the z-axis of the elements between frame i and $i+1$ relative to the coordinate frame i, respectively, and m_i is the mass of these elements.

i	COM_i (m)	m_i (kg)	Izz_i (kg·m^2)
2	$[0.260; 0; 0.041]^T$	3.31	0.309
3	$[0.683; -0.008; 0.041]^T$	10.42	–
4	$[0.440; -0.008; 0.028]^T$	6.21	1.624
5	$[0.089; -0.317; 0.027]^T$	177	69.79

Table III. Estimators' performance when τ_{ext} is step

Performance / External torque	EKF			NDOB		
	$\tau_{ext,2}$	$\tau_{ext,3}$	$\tau_{ext,4}$	$\tau_{ext,2}$	$\tau_{ext,3}$	$\tau_{ext,4}$
Settling time (2%) (sec)	0.61	0.52	0.35	0.06	0.22	0.19
Rise time (sec)	0.18	0.49	0.44	0.145	0.925	0.619
Simulation time (30,000 samples) (sec)	189.11			187		

5.1. Simulation setup

The simulation is done in Simulink/MATLAB 2017 and performed on the WallMoBot described in (3), and shown in Fig. 1. The robot parameters used in the simulation are obtained from the SolidWorks drawing and listed in Table II. Since the considered robot is planar and all its joints rotate only around y_0, just the moment of inertia of the elements between frame i and $i+1$ about the z-axis, that is, I_{zz_i}, which coincides with y_0, is needed. Moreover, since the elements between frames 3 and 4 only have translational motion and not rotational motion, I_{zz_3} is not needed.

All simulations started with the robot in the initial position as shown in Fig. 1, where q and $\dot{q}$ are zero; thus, the initial value of the states, $\hat{x}_0$, is chosen to be a zero matrix.

5.2. Simulation results

To employ the EKF, P_0, Q, and R are chosen as

$$\begin{aligned} P_0 &= \mathrm{diag}(1, 1, 1, 1, 1, 1, 10, 10, 10)10^{-3} \\ Q &= \mathrm{diag}(10^{-3}, 10^{-3}, 10^{-3}, 10^{-6}, 10^{-6}, 10^{-6}, 4, 2.5, 4) \\ R &= \mathrm{diag}(8, 0.01, 0.01, 0.3, 0.01, 0.01)10^{-10} \end{aligned} \tag{32}$$

Also, the chosen X for NDOB is $X = 2000 \times \mathrm{diag}(0.3, 0.5, 0.3)$. Figure 4 shows the joint positions, joint velocities, and the joint space torques caused by the applied forces on the WallMoBot end-effector, alongside their estimations, calculated by the proposed estimators when the τ_{ext} is a step, which simulates the fast changing of the external forces. The performance of the estimators is evaluated, and the results are shown in Table III. It should be noted that by changing the estimators' parameters, these values can also change.

As can be seen in Table III, the NDOB has a smaller settling time than EKF. The simulation time in both cases is almost the same.

Figure 5 shows the same signals when the τ_{ext} is a ramp which simulates the slow changing of the external forces. Also, to further challenge the proposed estimator, a more complicated signal as $\tau = 10\sin(t) + 5\cos(2t + \pi/2) - 5$ is considered as the τ_{ext}. The results of the EKF in this case are shown in Fig. 6.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

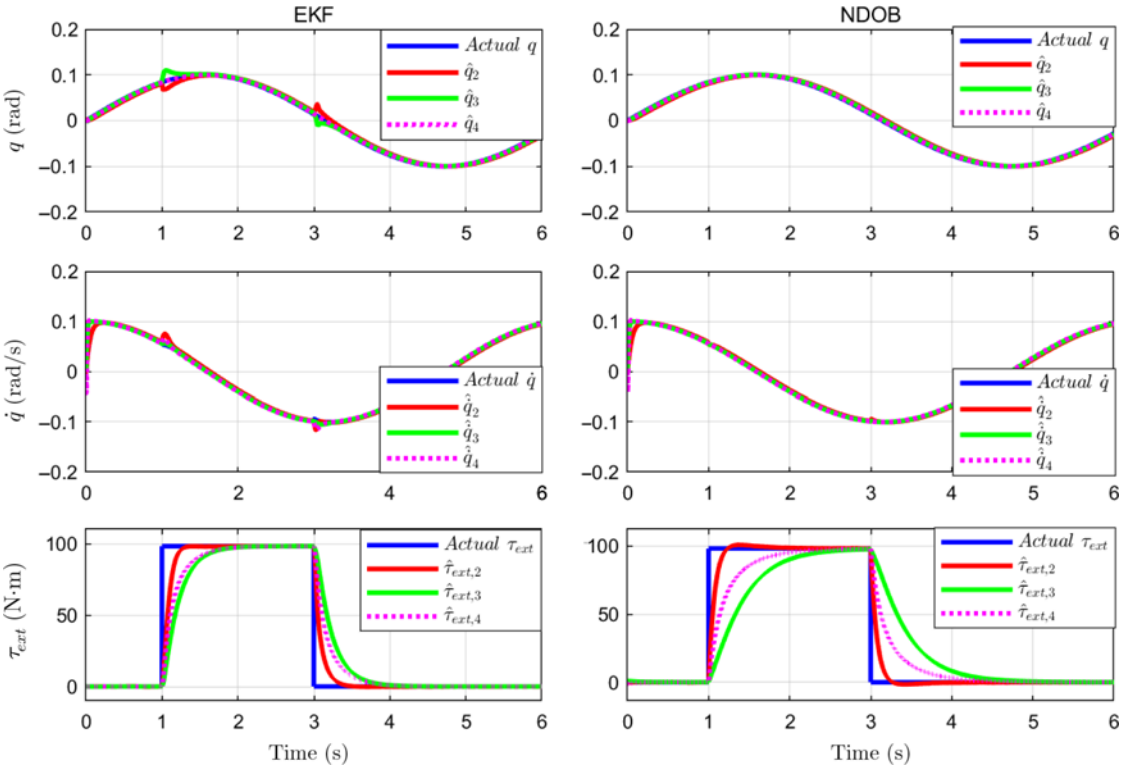

Fig. 4. The augmented states of the WallMoBot when the τ_{ext} is 98.2 N·m and $q_{ref} = 0.1\sin(t)$.

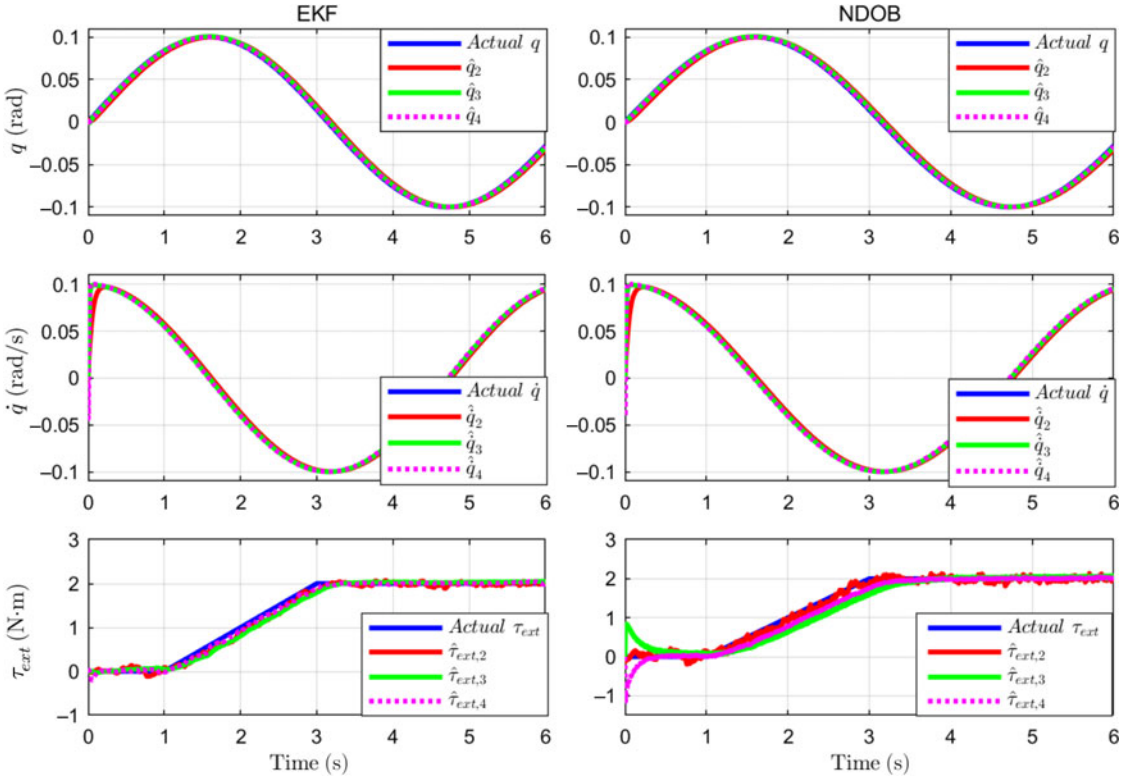

Fig. 5. The augmented states of the WallMoBot when the τ_{ext} is a ramp and $q_{ref} = 0.1\sin(t)$.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

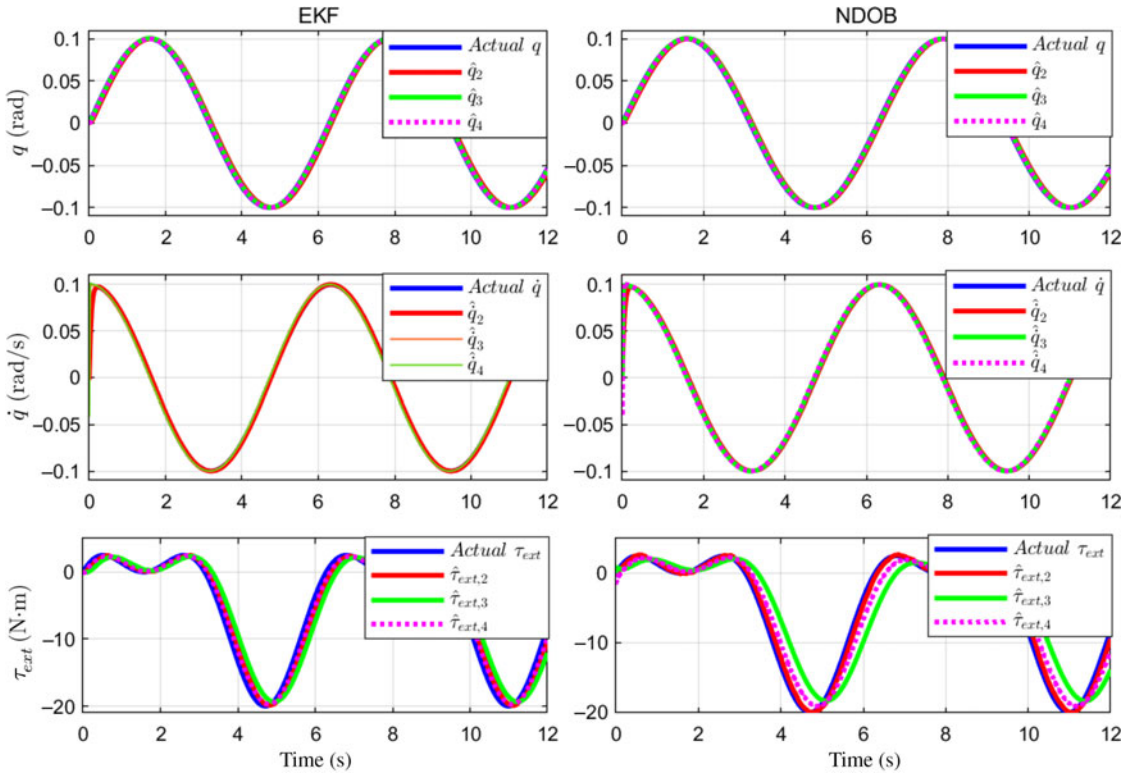

Fig. 6. The augmented states of the WallMoBot when $\tau_{ext} = 10\sin(t) + 5\cos(2t + \pi/2) - 5$ [N·m] and $q_{ref} = 0.1\sin(t)$.

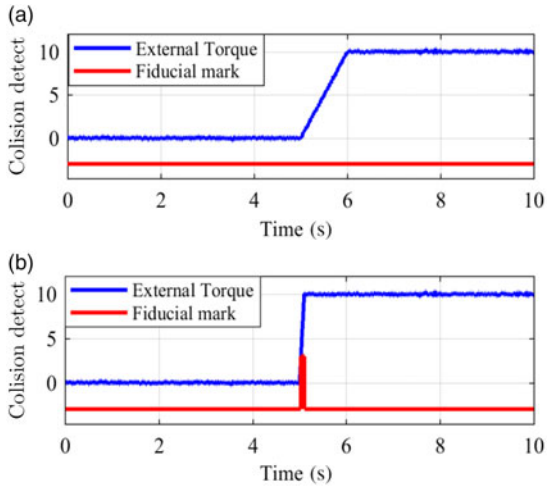

Fig. 7. Simulation result of the collision detection algorithm: (a) without any collision and (b) collision has occurred at $t = 5$ s.

5.3. Collision detection results

The simulation results of the proposed algorithm for collision detection are shown in Fig. 7. The joint space external torque is changing slowly which means no collision has occurred, while Fig. 7 shows a collision occurrence at $t = 5$ s which the proposed algorithm can detect it correctly.

6. Experimental Validation

The purpose of this section is to test the proposed approach on the testbed of the WallMoBot.

6.1. Experimental setup

The testbed, as shown in Fig. 8, has one degree of freedom. The relative position and velocity of the rotor are directly obtained from the integrated encoder in the installed actuator. Moreover, Compact

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

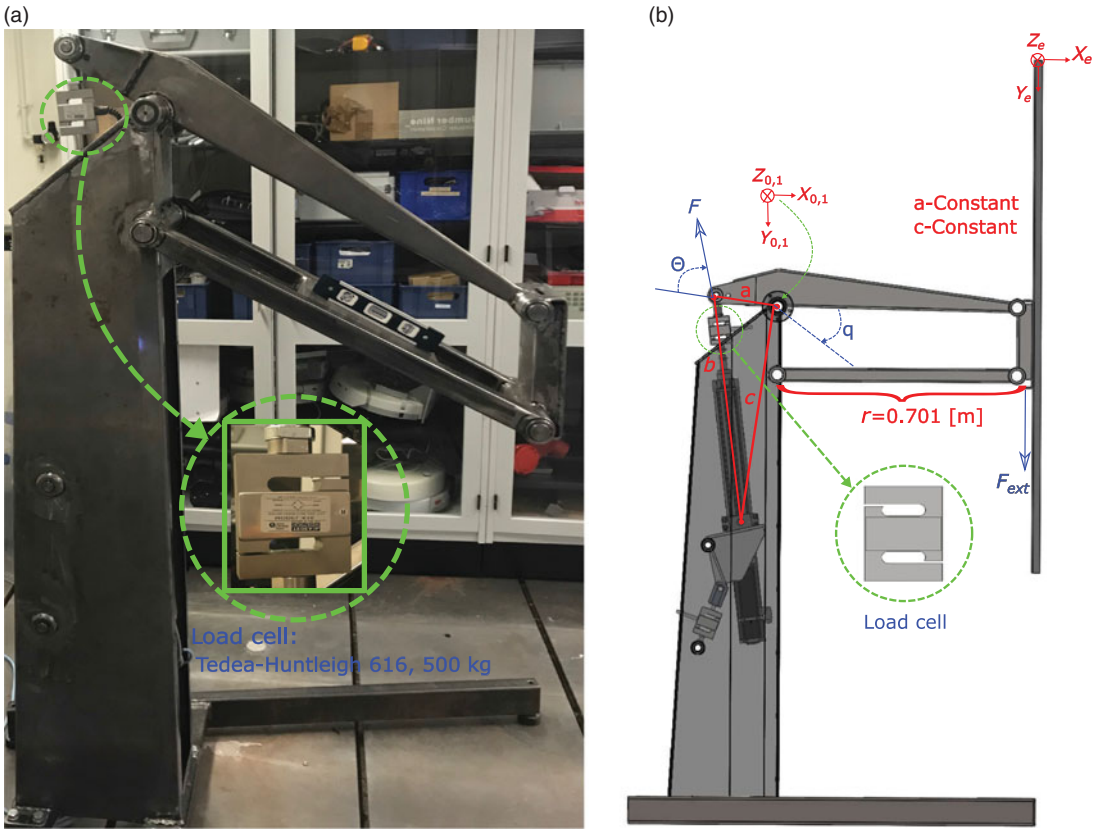

Fig. 8. (a) Illustration of the testbed of WallMoBot and the loadcell. (b) Illustration of the generalized coordinates of the testbed of WallMoBot. Frame 0 coincides with frame 1 when q is zero, and b is the length of the linear screw.

Reconfigurable Input/Output (cRIO) is used as the embedded controller. The cRIO, which is shown in Fig. 9, is made by National Instruments (NI) for industrial control systems. It is a combination of a real-time controller with real-time processor and reconfigurable FPGA, and reconfigurable IO Modules. Programming language of cRIO is LabVIEW, and it communicates with the actuator using EtherCAT. Since loadcells are substantially cheaper than force/torque sensors, a loadcell is connected to the end of the linear screw to measure the net of the exerted linear screw forces and the external forces, that is, F. Considering Fig. 8, the torque acting at the joint, that is, τ, can be obtained using (33) and (34).

$$\tau = Fa \sin \Theta, \tag{33}$$

where

$$\Theta = \arccos\left(\frac{a^2 + b^2 - c^2}{2ab}\right), \tag{34}$$

where τ is used in the EKF and NDOB algorithms to obtain $\hat{\tau}_{ext}$. Also, the actual value of the torque exerted by the external force, F_{ext}, at the joint equals to

$$\tau_{ext} = F_{ext} r \cos q \tag{35}$$

The testbed dynamical model is as

$$\tau = \theta_1 \ddot{q} + \theta_2 g \cos(q) + \theta_3 g \sin(q) + \theta_4 \operatorname{sign}(\dot{q}), \tag{36}$$

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

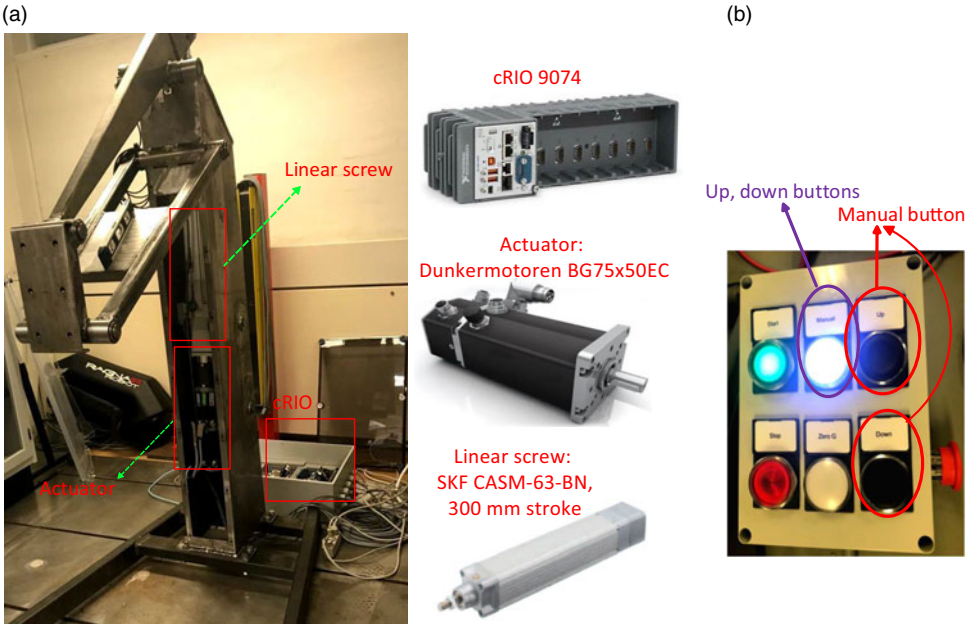

Fig. 9. (a) Illustration of the linear screw, actuator, and cRIO in the testbed. (b) Testbed movements modes.

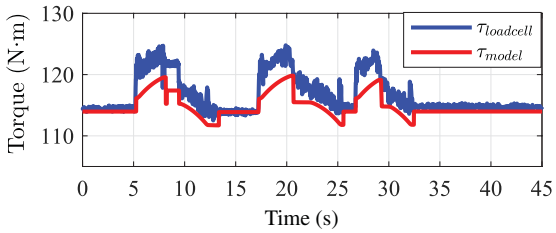

Fig. 10. Examining the testbed dynamical model.

where its parameters are obtained using linear least-square estimation as

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 0.41937 \\ 11.668 \\ -0.74 \\ 2.163 \end{bmatrix} \tag{37}$$

The correctness of the obtained model is examined by moving the robot end-effector down and up. As can be seen in Fig. 10, the model can follow the changes in the loadcell torque effectively. The discrepancy between measured and model torques can be attributed to the mechanical construction of the testbed and loadcell, such as loadcell nonlinearity, backlash effects, and other unmodeled dynamics.

The testbed moves in two different modes:

1. Manual mode: the robot end-effector moves by the human force, which is the aim of the WallMoBot. This mode is activated by the manual button shown in Fig. 9. In this case, the arm moves smoothly between the lower and upper limits of the robot.

2. Button mode: the robot end-effector moves up and down by the up and down buttons shown in Fig. 9. This mode is used for more testing of the robot and algorithms. In this case, when we press the up button, the robot end-effector moves upward until it reaches the upper limit position and stops. Similarly, when we press the down button, the robot end-effector moves downward until the robot reaches the lowest limit position and stops.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

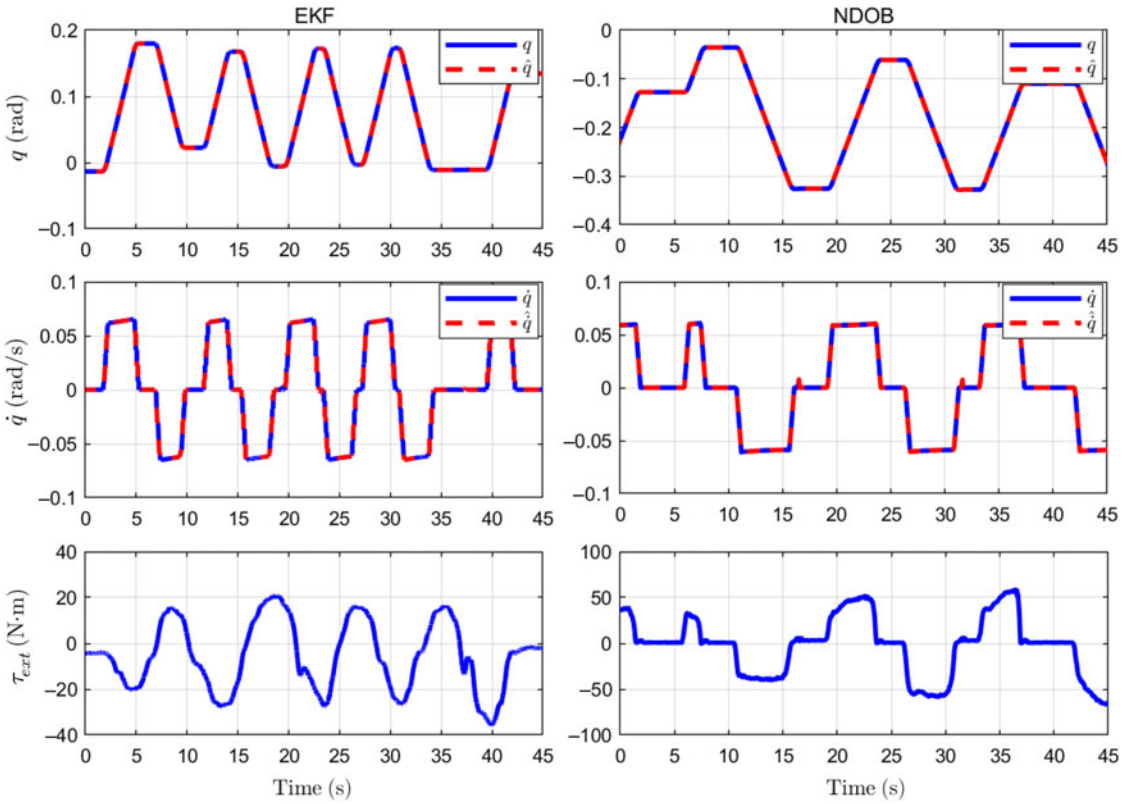

Fig. 11. The augmented states of the testbed when τ_{ext} is caused by the human force exerted on the testbed end-effector.

6.2. *Experimental results*

In order to implement the EKF, the initial value of the states, $\hat{x}_0$, is chosen to be a zero matrix. Also, P_0, Q, R are as

$$\begin{aligned} P_0 &= \text{diag}(10^{-20}, 10^{-20}, 10^{-2}) \\ Q &= \text{diag}(10^{-14}, 10^{-14}, 10^{-2}) \\ R &= \text{diag}(10^{-20}, 10^{-17}) \end{aligned} \tag{38}$$

To show the effectiveness of the proposed EKF and NDOB, different experimental results are provided. First, the efficiency when the human operator moves the end-effector of the testbed up and down is investigated. Figure 11 shows that the estimated position and velocity are very accurate; however, the measurement of the actual human force is not available. Therefore, in the next experiment, a constant weight is hanged to the robot end-effector as the external force.

In order to examine the accuracy of the τ_{ext} estimation, a 3.3 kg load at $t = 25$ s is attached to the robot end-effector, while the position of the end-effector is kept constant at $q = 0.5$ rad. Based on this position and using (35), the actual value of the torque produced by the external load, which acts as an external force applied to the WallMoBot end-effector, is 19.9 N·m. Figure 12 shows that the estimated torque, $\hat{\tau}_{ext}$, converges fast and accurately to its actual value 19.9 N·m.

Additionally, in order to test the sensitivity and precision of the proposed estimators in estimating small changes of τ_{ext}, the added load on the end-effector in the previous experiment is randomly swung within small range of angles with the starting time $t = 13$ s, and then at $t = 32$ s, the load is dropped. Figure 13 shows the $\hat{\tau}_{ext}$ illustrates that both EKF and NDOB are able to estimate small changes in the τ_{ext}.

In order to examine the efficiency of the estimators while the robot end-effector is moving, a 1.2 kg load at $t = 8$ s is attached to the robot end-effector, while the end-effector is moving up and down. Figure 14 shows the $\hat{\tau}_{ext}$ in this case. As can be seen, at the start point of moving the robot down and up, a peak in τ_{ext} estimation occurs, which is caused by the mechanical constraints of the testbed while switching between up, down, and stop functions in the button mode. Meanwhile, the estimators

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

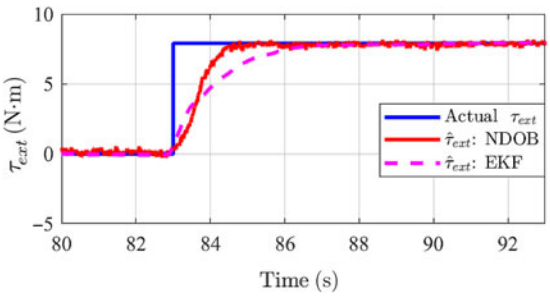

Fig. 12. Experimental result when the τ_{ext} is 19.9 N·m and the robot position is constant.

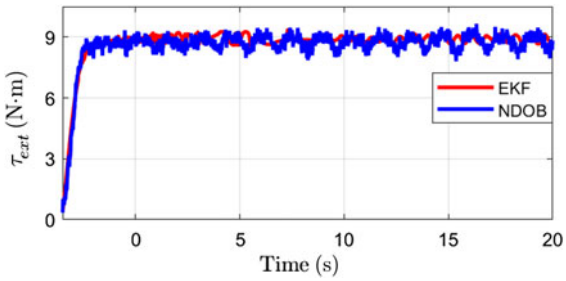

Fig. 13. Experimental results when the load on the end-effector is randomly swung.

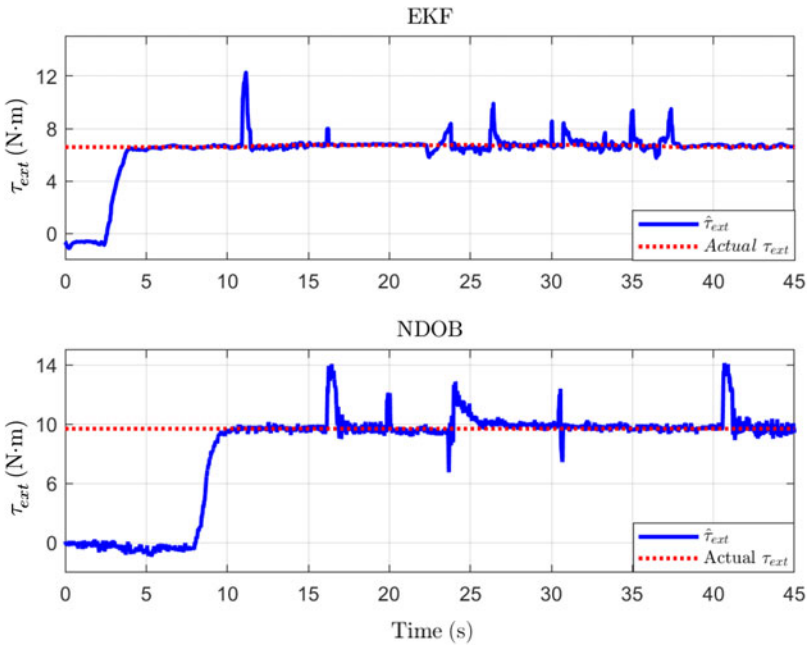

Fig. 14. Experimental results when the robot end-effector and the added load to it move vertically up and down.

are able to estimate the τ_{ext} effectively with a small error. However, since the human operator moves the robot end-effector smoothly, the proposed estimators are able to estimate the human operator force accurately.

6.3. Collision detection results

The final experiment is testing the collision detection algorithm. To this aim, a sudden collision is exerted to the robot end-effector which causes a sudden change in the $\hat{\tau}_{ext}$ as can be seen in Fig. 15. This signal, as explained in Fig. 2, is used to make a fiducial mark. It is shown in Fig. 15, which is used to stop the motor when a collision happens.

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

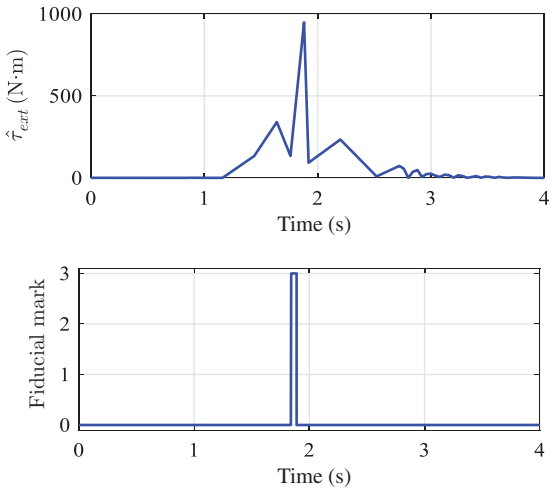

Fig. 15. Experimental result of the collision detection algorithm.

7. Discussion

As been explained, the NDOB in general is sensitive to sensor noise and system uncertainty. As can be seen in the estimation and experimental results, the NDOB estimations are more noisy. However, since the used sensors in our experiments are not very noisy and the system uncertainty is not much, both estimators provide acceptable estimations of the external torques. However, in practice, since the glass panel weight is heavy and there could be some uncertainty in its weight, the EKF can be more reliable.

Moreover, the estimation time of the NDOB in general is less than the EKF. However, since only 3 degrees of freedom of the WallMoBot are considered in our simulations, the simulation time for both NDOB and EKF is almost the same, as can be seen in Table III.

All in all, in our case, the estimation results of the EKF are more reliable.

8. Conclusion

In this paper, instead of employing torque/force sensors, two alternative approaches to estimate external forces acting on the end-effector of a robot were proposed. The presented methods were based on the extended Kalman filter (EKF) and nonlinear disturbance observer (NDOB). In the suggested techniques, first, the net forces arise from the environment and the linear screws are directly measured by loadcell units. Then, the measured forces were converted into torques and utilized in the EKF and NDOB algorithms to achieve the estimations of slow-varying external torques applied on the robot end-effector. Additionally, to enhance the safety of the robot operation, an algorithm was proposed to differentiate the user input force from accidental collisions. The proposed algorithms were applied to the WallMoBot and experimental validations were carried out. It can be concluded that even though both the proposed approaches can effectively estimate the exerted external forces on the end-effector of the robot, the results of the EKF can be more reliable in practice.

Acknowledgments

This research was supported by Innovation Fund Denmark in project number 5150-00007A called WallMoBot. We are thankful to our colleagues at the University of Southern Denmark (SDU) who provided the computer-aided design (CAD) models for the WallMoBot in Fig. 1. We are also grateful to WallMo A/S for assistance with the project.

References

1. M. Hersh, "Overcoming barriers and increasing independence–service robots for elderly and disabled people," *Int. J. Adv. Rob. Syst.* **12**(8), 114 (1972).
2. C. Sloth and R. Pedersen, "Control of wall mounting robot," *IFAC-PapersOnLine* **50**(1), 5648–5653 (2017).
3. M. R. Soltanpour and M. H. Khooban, "A particle swarm optimization approach for fuzzy sliding mode control for tracking the robot manipulator," *Nonlinear Dyn.* **74**(1–2), 467–478 (2013).

https://doi.org/10.1017/S0263574719001681 Published online by Cambridge University Press

4. J. Luo and K. Hauser, "Robust trajectory optimization under frictional contact with iterative learning," *Auton. Rob.* **41**(6), 1447–1461 (2017).
5. K. Asai and G. Ulusoy, *Computer Integrated Manufacturing: Current Status and Challenges*, vol. 49 (Springer Science & Business Media, Berlin, Heidelberg, 2012).
6. A. Colomé, D. Pardo, G. Alenya and C. Torras, "External Force Estimation During Compliant Robot Manipulation," *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe (2013) pp. 3535–3540.
7. J. L. Novak and I. Feddema, "A Capacitance-Based Proximity Sensor for Whole Arm Obstacle Avoidance," *IEEE International Conference On Robotics and Automation (ICRA)*, Nice, France (1992) pp. 1307–1314.
8. K. Nuelle, M. J. Schulz, S. Aden, A. Dick, B. Munske, J. Gaa, J. Kotlarski and T. Ortmaier, "Force Sensing, Low-Cost Manipulator in Mobile Robotics," *IEEE International Conference on Robotics and Automation (ICRA)*, Marina Bay Sands, Singapore (2017) pp. 196–201.
9. I. F. J. Ghalyan, *Force-Controlled Robotic Assembly Processes of Rigid and Flexible Objects* (Springer, Switzerland, 2016).
10. A. De Luca, A. Albu-Schaffer, S. Haddadin and G. Hirzinger, "Collision Detection and Safe Reaction with the DLR-III Lightweight Manipulator Arm," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China (2006) pp. 1623–1630.
11. K. S. Eom, I. H. Suh, W. K. Chung and S. R. Oh, "Disturbance Observer Based Force Control of Robot Manipulator Without Force Sensor," *IEEE International Conference On Robotics and Automation (ICRA)*, Leuven, Belgium, vol. 4 (1998) pp. 3012–3017.
12. L. Chan, F. Naghdy and D. Stirling, "Extended active observer for force estimation and disturbance rejection of robotic manipulators," *Rob. Auton. Syst.* **61**(12), 1277–1287 (2013).
13. G. Liao, Y. Sheng and X. Zeng, "Spacecraft Hovering Around Asteroid via Disturbance Observer Based Exponential Time-Varying Sliding Mode Controller," *Proceedings of the 13th IEEE International Conference on Control & Automation (ICCA)*, Ohrid, Macedonia (2017) pp. 313–318.
14. C. Mitsantisuk, K. Ohishi, S. Urushihara and S. Katsura, "Kalman filter-based disturbance observer and its applications to sensorless force control," *Adv. Rob.* **25**(3–4), 335–353 (2011).
15. Y. Nagatsu and S. Katsura, "High-Order Disturbance Estimation Using Kalman Filter for Precise Reaction-Torque Control," *IEEE International Workshop on Advanced Motion Control (AMC)*, Auckland (2016) pp. 79–84.
16. J. Hu and R. Xiong, "Contact force estimation for robot manipulator using semi-parametric model and disturbance Kalman filter," *IEEE Trans. Ind. Electron.* **65**(4), 3365–3375 (2018).
17. A. Mohammadi, H. J. Marquez and M. Tavakoli, "Nonlinear disturbance observers: Design and applications to Euler-Lagrange systems," *IEEE Control Syst.* **37**(4), 50–72 (2017).
18. M. S. Grewal, "Kalman Filtering," **In**: *International Encyclopedia of Statistical Science* (Springer, Berlin, Heidelberg, 2011) pp. 705–708.
19. J. Hartikainen, A. Solin and S. Särkkä, "Optimal Filtering with Kalman Filters and Smoothers", **16** (Department of Biomedica Engineering and Computational Sciences, Aalto University School of Science, Greater Helsinki, Finland, 2011).
20. A. Zelenak, M. Pryor and K. Schroeder, "An Extended Kalman Filter for Collision Detection During Manipulator Contact Tasks," *American Society of Mechanical Engineers (ASME) Dynamic Systems and Control Conference* (2014) pp. V001T11A005–V001T11A005.
21. W. Liang, S. Huang, S. Chen and K. K. Tan, "Force estimation and failure detection based on disturbance observer for an ear surgical device," *ISA Trans* **66**(7), 476–484 (2017).
22. A. C. Smith, F. Mobasser, and K. Hashtrudi-Zaad, "Neural-network-based contact force observers for haptic applications," *IEEE Trans. Rob.* **22**(6), 1163–1175 (2006).
23. L. Cousineau and N. Miura, *Construction Robots: The Search for New Building Technology in Japan* (ASCE Publications, Virginia, 1998).
24. Y. Chen, S. Turner, R. McNamee, C. N. Ramsay and R. M. Agius, "The reported incidence of work-related ill-health in Scotland (2002–2003)," *Occup. Med.* **55**(4), 252–261 (2005).
25. B. Hein and M. Hensel and H. Worn, "Intuitive and Model-Based On-line Programming of Industrial Robots: A Modular On-line Programming Environment," *IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, California (2008) pp. 3952–3957.
26. Y. G. Park and W. K. Chung, "Unified External Torque-Sensing Algorithm for Flexible-Joint Robot Based on Kalman Filter," *IEEE International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Jeju (2013) pp. 78–79.
27. G. Bishop and G. Welch, "An Introduction to the Kalman Filter," *Proceedings of SIGGRAPH*, course 8, no. 27599–23175, vol. 8 (2001) p. 35.
28. S. Yousefizadeh and T. Bak, "Nonlinear Disturbance Observers for External Force Estimation in a Cooperative Robot," *19th International Conference on Advanced Robotics (ICAR)*, Brazil (2019).
29. C. S. Tseng and C. K. Tompkins, "Fuzzy observer-based fuzzy control design for nonlinear systems with persistent bounded disturbances," *Fuzzy Sets Syst.* **158**(2), 164–179 (2007).
30. T. Flash and N. Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *J. Neurosci.* **5**(7), 1688–1703 (1985).
31. J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.* **32**(3), 230–236 (1985).
32. R. M. Rangayyan, *Biomedical Signal Analysis*, vol. 33 (John Wiley & Sons, Hoboken, NJ, 2015)