

A Brief Survey of Observers for Disturbance Estimation and Compensation

Teng Li¹, Hongjun Xing², Ehsan Hashemi³, Hamid D. Taghirad⁴ and Mahdi Tavakoli^{*1}

¹Department of Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, Edmonton T6G 1H9, Alberta, Canada.

²College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China.

³Department of Mechanical Engineering, Faculty of Engineering, University of Alberta, Edmonton T6G 1H9, Alberta, Canada.

⁴Advanced Robotics and Automated Systems (ARAS), Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran.

*Corresponding author. E-mail: mahdi.tavakoli@ualberta.ca.

Received: xx xxx xxx; **Revised:** xx xxx xxx; **Accepted:** xx xxx xxx

Keywords: disturbance observer, generalized momentum observer, Kalman filter, extended state observer, constant disturbance, time-varying disturbance, impedance control.

Abstract

An accurate dynamic model of a robot is fundamentally important for a control system, while uncertainties residing in the model are inevitable in a physical robot system. The uncertainties can be categorized as internal disturbances and external disturbances in general. The former may include dynamic model errors and joint frictions, while the latter may include external payloads or human-exerted force to the robot. Disturbance observer is an important technique to estimate and compensate for the uncertainties of the dynamic model. Different types of disturbance observers have been developed to estimate the lumped uncertainties so far. In this paper, we conducted a brief survey on five typical types of observers from a perspective of practical implementation in a robot control system, including generalized momentum observer (GMO), joint velocity observer (JVOB), nonlinear disturbance observer (NDOB), disturbance Kalman filter (DKF), and extended state observer (ESO). First, we introduced the basics of each observer including equations and derivations. Two common types of disturbances are considered as two scenarios, *i.e.*, constant external disturbance and time-varying external disturbance. Then, the observers are separately implemented in each of the two simulated scenarios, and the disturbance tracking performance of each observer is presented while their performance in the same scenario has also been compared in the same figure. Finally, the main features and possible behaviors of each type of observer are summarized and discussed. This survey is devoted to helping readers learn the basic expressions of five typical observers, and implement them in a robot control system.

1. Introduction

In a robot control system, an accurate dynamic model of the robot is fundamentally important for accurate and stable control [1]. This is true for all kinds of robots, such as industrial robots, humanoid robots, medical robots, soft robots, and exoskeletons. However, accurate dynamic models only exist in theory but not in practice, since various uncertainties can be residing in the dynamic model inevitably. Examples of such uncertainties are joint friction, inaccurate center of mass location and link weight, extra payload, and robot-environment interaction [1]. Therefore, it is a fundamental topic for estimating and compensating for uncertainties in the field of robot control.

Many methods have been developed for estimating dynamic uncertainties thus eliminating their effects on robot dynamics. Disturbance observer is a main solution that can observe the dynamic uncertainties in an online manner, thus making compensation accordingly when necessary. Besides the observers, many learning techniques have been also applied for disturbance estimation and compensation, such as using feedforward neural network (NN) [2, 3, 4], and nonlinear autoregressive network with exogenous inputs (NARX) [5]. Since the learning-based methods for disturbance estimation are beyond the scope of this paper, they will not be introduced here.

A variety of observers have been developed [6, 7], but not all of them can be implemented into a robotic system. The typical observers selected in this paper are identified according to the following procedures. First, a bunch of observers is collected based on three recent review papers [8, 9, 10] and one research paper [11]. Then, the following two rules are applied, (a) the observer is applicable in practice in a physical robotic system; and (b) the observer is independent of the controllers, *i.e.*, not relying on a specific controller. Finally, five typical observers are identified, including generalized momentum observer (GMO) [8], joint velocity observer (JVOB) [8], extended state observer (ESO) [12], nonlinear disturbance observer (NDOB) [13], and disturbance Kalman filter (DKF) method [11, 4]. In this paper, we will focus on surveying these five typical types of disturbance observers, *i.e.*, GMO, JVOB, ESO, NDOB, DKF. For the ESO, three of its variants are presented, including the original ESO (ESO.orig), a modified ESO (ESO.modi), and an improved ESO (ESO.impr).

Generalized momentum observer (GMO), also known as classic first-order momentum observer, is originally proposed for actuator fault detection and isolation, aiming to avoid joint acceleration measurements and inverse of the robot inertia matrix in the control system [14, 15]. Note that in practice, measuring accelerations is usually avoided due to their high price or installation size restrictions. Furthermore, obtaining accelerations via numerical differentiation of velocity or position is not preferred since it will introduce noise into the system and thus may affect the system's stability [8]. The inverse of the robot inertia matrix can increase the computational load on the system. Besides not requiring the accelerations and inverse of inertia matrix, the GMO also has the advantage of being simple, compact, and easy to implement [16], all of which make it one of the most commonly used observers. It is usually used as a benchmark for comparison when designing new observers [17, 11, 4].

Joint velocity observer (JVOB) is a similar observer to the GMO in terms of the procedures of derivation and the final expression, but the inversion of the robot inertia matrix is needed [8]. The JVOB is derived based on the acceleration expressed by the robot dynamics, then the integration of the acceleration is taken as an estimate of the velocity where the true velocity is assumed to be known. Therefore, the JVOB is a reduced (first-order) observer with a reduced state of dimension n , where n is the number of the generalized coordinates of the robot.

Extended state observer (ESO) was originally proposed by Han in the 1990s, and it is considered as the critical part of active disturbance rejection control (ADRC) which is developed to estimate the lumped uncertainties including both unknown dynamic uncertainties and external disturbances [18, 19]. For a specific survey on ADRC and ESO, please refer to [20]. In another historical survey on observers in 2006 [6], ESO has been viewed as an indicator of an initial shift of design methodology from modern estimators (*e.g.*, Kalman Filter) to disturbance estimators (*e.g.*, ESO). The ESO employs a simple canonical form which is considered as a practical design, thus it receives many applications in different fields, such as power converters, web tension, and bio-mechanics [6]. In the field of robot control, various variations of ESO have been developed in different application scenarios such as collision detection [21] and time-varying interaction force estimation [12].

Nonlinear disturbance observer (NDOB) is originally proposed by Chen, which is considered to overcome the shortcomings of linear disturbance observer (DOB) that is designed or analyzed by linear system techniques [22]. Although this version of NDOB was developed for constant disturbances in theory, it also revealed satisfactory performance on estimating time-varying disturbances like friction. However, it is merely used for planar robots with revolute joints. To solve this problem, Mohammadi *et*

al. [13] proposed a general framework for NDOB by unifying linear and nonlinear disturbance observers which released the restrictions on the number of degree-of-freedom (DOF), the types of joints (revolute or prismatic), or the robot configuration.

Kalman filter (KF) is an early approach to be used for disturbance estimation, which is also one of the first estimators that involve the formulation of disturbances [6]. Based on the original KF, disturbance Kalman filter (DKF) is developed to estimate the dynamic uncertainties in a robot control system [11, 23]. The DKF reveals optimal disturbance tracking performance, but the implementation complexity could be a limitation for its wide use [6].

All the five types of observers introduced above estimate a lumped uncertainty term. While the lumped uncertainties include various components (*e.g.*, model error, joint friction, external payload, human-exerted force, and beyond), it is not possible to separate a specific component out of the lumped uncertainties term. Especially in human-robot interaction scenarios, an operating observer will take the human-exerted force as a part of uncertainties and thus reject it [24].

These five types of observers are well-developed techniques that have been implemented and evaluated in various experimental scenarios. Therefore, in this survey paper, we focus on observer implementation rather than theoretical analysis, and exploration of the observer behaviors in simulations rather than physical experiments. One of the best advantages of simulations is that every component of the lumped uncertainties can be precisely manipulated, thus the disturbance tracking performance of each observer can be clearly revealed.

The main goal of this survey is (1) to present the basic expressions of the five typical observers with which the observers can be quickly implemented into a robotic system, (2) and to help readers to get an intuitive sense on the behaviors of different observers by presenting them in the same specific simulated scenarios. Experiments are also conducted to demonstrate the effectiveness of the observers in a real application scenario.

The rest of this paper is organized as follows: Section 2 describes impedance control, the basic expressions, and derivations of each type of observer. Section 3 presents simulations and corresponding results of each observer on disturbance tracking performance in two simulated scenarios. Section 4 presents experimental results when observers are implemented in the same scenario. Section 5 provides some summary and discussions on the features and behaviors of the observers.

2. Methods

2.1. Robot Dynamics and Impedance Control

A general dynamic model for an n -degree-of-freedom (DOF) rigid robot [25] can be given by

$$\underbrace{M(q)}_{\hat{M}+\Delta M} \ddot{q} + \underbrace{S(q, \dot{q})}_{\hat{S}+\Delta S} \dot{q} + \underbrace{g(q)}_{\hat{g}+\Delta g} + \tau_{fric}(q, \dot{q}) = \tau + \underbrace{\tau_{ext}}_{J^T F_{ext}} \quad (1)$$

where $M \in \mathbb{R}^{n \times n}$ denotes the inherent inertia matrix, $S \in \mathbb{R}^{n \times n}$ denotes a matrix of the Coriolis and centrifugal forces, $g \in \mathbb{R}^n$ represents the gravity vector. $\hat{M}, \hat{S}, \hat{g}$ represent users' model estimates, while $\Delta M, \Delta S, \Delta g$ are the corresponding estimate errors. $\tau_{fric} \in \mathbb{R}^n$ is joint friction, $\tau \in \mathbb{R}^n$ is the commanded joint torque vector, $\tau_{ext} \in \mathbb{R}^n$ is the torque caused by external force, $F_{ext} \in \mathbb{R}^6$ is the external force in Cartesian space, and $J \in \mathbb{R}^{6 \times n}$ is the Jacobian matrix.

A desired impedance model [26, 27, 24] for robot-environment interaction can be expressed as

$$F_{imp} = M_m(\ddot{x} - \ddot{x}_d) + (S_x + D_m)(\dot{x} - \dot{x}_d) + K_m(x - x_d) \quad (2)$$

where M_m, D_m, K_m are user-designed matrices for inertia, damping, and stiffness, respectively. Note that $x_d, \dot{x}_d, \ddot{x}_d$ are the desired position, velocity, and acceleration, respectively in Cartesian space, while

x, \dot{x}, \ddot{x} are the actual ones. S_x is the Coriolis and centrifugal matrix of the robot in Cartesian space and $S_x = J^{-T} S J^{-1} - M_x \dot{J} J^{-1}$, where $M_x = J^{-T} M J^{-1}$ is the inherent inertia of the robot in Cartesian space [28].

To avoid the measurement of external forces, the designed inertia matrix can be set as the inherent inertia matrix of the robot, *i.e.*, $M_m = M_x$. Then, to reach (2) as the closed-loop dynamics governing the robot-environment interaction in an ideal scenario of no model errors and no joint friction, the impedance control law [24] can be given by

$$\tau = M J^{-1} (\ddot{x}_d - \dot{J} J^{-1} \dot{x}_d) + S J^{-1} \dot{x}_d + g + J^T [D_m (\dot{x}_d - \dot{x}) + K_m (x_d - x)] \quad (3)$$

where J^{-1} will be replaced with the pseudo-inverse of the Jacobian $J^\# = J^T (J J^T)^{-1}$ when J is not a square matrix. In this work, singularity problem is not considered, thus J is invertible. For dealing with the singularity problem, please refer to [26]. Note that when implementing the impedance controller (3) in practice, the estimates $\hat{M}, \hat{S}, \hat{g}$ will be used for the calculation since an accurate model of a physical robot is usually not available.

For robot end-effector (EE) moving to a fix point, *i.e.*, set-point regulation, we have $\ddot{x}_d = 0, \dot{x}_d = 0$. Then, the impedance control law (3) can be simplified to (4), which is also known as task-space proportional–derivative (PD) controller with gravity compensation.

$$\tau = J^T [K_m (x_d - x) - D_m \dot{x}] + g \quad (4)$$

2.2. Disturbance and Disturbance Observer

By collecting all the disturbances together, the dynamic model (1) of a robot can be re-written as

$$\hat{M} \ddot{q} + \hat{S} \dot{q} + \hat{g} = \tau + \underbrace{\tau_{ext} - [\tau_{fric} + (\Delta M \ddot{q} + \Delta S \dot{q} + \Delta g)]}_{\tau_{dist}} \quad (5)$$

where τ_{dist} denotes the lumped uncertainties that usually include three main aspects, *i.e.*, the model error $(\Delta M \ddot{q} + \Delta S \dot{q} + \Delta g)$, the joint friction τ_{fric} , and the external disturbances τ_{ext} , where the last aspect may involve constant disturbance and/or time-varying disturbance. The constant disturbance may be a constant payload attached to the robot end-effector (EE) or body, while time-varying disturbance may be robot-environment interaction forces such as human-applied forces during human-robot interaction. A disturbance observer usually estimates the lumped uncertainties τ_{dist} [13], but cannot discriminate any single component when more than one component exists. In this paper, we denote the disturbance observer output as $\hat{\tau}_{dist}$ since it is an estimate of the true disturbance τ_{dist} . Note that measurement noise of position and velocity and any other unknown uncertainties (if there have any) will also be included in the lumped term τ_{dist} .

In this paper, we will focus on simulations thus each component of the lumped uncertainties can be precisely controlled. For the simulations, here we assume that (1) no model errors, *i.e.*, $\hat{M} = M, \hat{S} = S, \hat{g} = g$, thus $\Delta M = 0, \Delta S = 0, \Delta g = 0$; (2) no joint friction, *i.e.*, $\tau_{fric} = 0$; (3) joint velocities are available. By applying these assumptions, the dynamic model (5) will become (6). Therefore, under these assumptions, the lumped uncertainties will purely come from the external disturbances τ_{ext} .

$$M \ddot{q} + S \dot{q} + g = \tau + \underbrace{\tau_{ext}}_{\tau_{dist}} \quad (6)$$

In the remaining of this section, the basic expressions and derivations of the five typical types of disturbance observers, *i.e.*, NDOB, GMO, JVOB, DKF, and ESO with three variants (ESO.orig, ESO.modi,

ESO.impr), will be introduced, respectively. It should be noted that some variable names may be re-used with different meanings by different observers since observers are independent of each other.

2.3. Observer 1: NDOB

A nonlinear disturbance observer (NDOB) can be used to estimate all dynamic uncertainties as a lumped term [13]. It has the advantage of estimating the nonlinearities in the dynamics. An adapted NDOB design based on [13] can be expressed as

$$\begin{cases} L = YM^{-1} \\ p = Y\dot{q} \\ \dot{z} = -Lz + L(S\dot{q} + g - \tau - p) \\ \hat{\tau}_{dist} = z + p \end{cases} \quad (7)$$

where $L \in \mathbb{R}^{n \times n}$ is the observer gain matrix, $Y \in \mathbb{R}^{n \times n}$ is a constant invertible matrix that needs to be designed, z is an auxiliary variable, p is an auxiliary vector determined from Y , $\hat{\tau}_{dist}$ is the estimated lumped uncertainties from the NDOB observer. For the full designing procedures of this NDOB and complete theoretical analysis please refer to [13].

Remarks:

The advantages of the NDOB [13] are the following.

- (1) This is a generalized disturbance observer without restrictions on the number of DOF, the types of joints, and the manipulator configuration;
- (2) The disturbance tracking error can converge exponentially to zero in the case of slow-varying disturbances, while the tracking error is globally uniformly ultimately bounded for fast-varying disturbances.

2.4. Observer 2: GMO

Generalized momentum observer (GMO) is one of the most widely used observers with the advantages of simple, no need to use the inverse of the robot inertia matrix (M^{-1}), and no need of joint accelerations. It is usually used as a reference when developing new observers. Here below are the derivations of the GMO.

In robot dynamics, a basic property is the skew-symmetry of matrix $\dot{M} - 2S$, which can be expressed as

$$\dot{M} = S + S^T \quad (8)$$

The generalized momentum p of the robot is given by

$$p = M\dot{q} \quad (9)$$

where $M \in \mathbb{R}^{n \times n}$ denotes the robot inertia matrix.

From (9), the time evolution of p can be written as

$$\dot{p} = \dot{M}\dot{q} + M\ddot{q} \quad (10)$$

Combining (6), (8), and (10), yields

$$\dot{p} = S^T \dot{q} - g + \tau + \tau_{dist} \quad (11)$$

Correspondingly, the dynamics of the estimated generalized momentum can be given by

$$\dot{\hat{p}} = S^T \dot{q} - g + \tau + r \quad (12)$$

where r is the residual vector.

Then, the disturbance force associated with joint torque can be observed by using

$$r = K_O e_p = K_O(p - \hat{p}) = K_O(p - \int_0^t (S^T \dot{q} - g + \tau + r) d\tau) \quad (13)$$

where K_O is a positive diagonal gain matrix of the observer, e_p is the estimation error of the generalized momentum p . The dynamic evolution of r has a stable structure which can be given by

$$\dot{r} = K_O(\tau_{dist} - r) \quad (14)$$

When the gain matrix K_O is large enough, we will have

$$\tau_{dist} \approx r \quad (15)$$

which means that r can be taken as the estimated lumped uncertainties given by

$$\hat{\tau}_{dist} = r = K_O(M\dot{q} - \int_0^t (S^T \dot{q} - g + \tau + r) d\tau) \quad (16)$$

As a further step, we can project the estimated lumped disturbances into Cartesian space by

$$\hat{F}_{dist} = J^{-T} r \quad (17)$$

where \hat{F}_{dist} is the estimated lumped uncertainties $\hat{\tau}_{dist}$ expressed in Cartesian space. Note that, under ideal conditions, *i.e.*, $\tau_{fric} = 0$, $\hat{M} = M$, $\hat{S} = S$, $\hat{g} = g$, the estimation from the observer is the external force ($\hat{F}_{dist} = \hat{F}_{ext}$).

Remarks:

The advantages of the GMO [8, 29] are the following,

- (1) No need for the inversion of the inertia matrix (M^{-1}).
- (2) No need for the measurement of joint acceleration.

2.5. Observer 3: JVOB

Joint velocity observer (JVOB) is a first-order system. It is a similar observer to the GMO, while the main difference is that in the JVOB observer, acceleration is involved in the derivation instead of the generalized momentum. In what follows, the derivations of the JVOB are given.

From the dynamic model (6), the joints acceleration is

$$\ddot{q} = M^{-1}(\tau - S\dot{q} - g + \tau_{dist}) \quad (18)$$

The observer dynamics is

$$\begin{cases} \hat{\ddot{q}} = M^{-1}(\tau - S\dot{q} - g + r) \\ \dot{r} = K_O(\ddot{q} - \hat{\ddot{q}}) \end{cases} \quad (19)$$

The monitored residual vector r by the observer is

$$r = K_O(\dot{q} - \int_0^t M^{-1}(\tau - S\dot{q} - g + r)d\tau) \quad (20)$$

Then, the observer dynamics will be

$$\dot{r} = K_O M^{-1}(\tau_{dist} - r) \quad (21)$$

Again, when the gain matrix K_O is sufficiently large, it will have

$$\tau_{dist} \approx r \quad (22)$$

which means that r can be taken as the estimated lumped uncertainties, *i.e.*, $\hat{\tau}_{dist} = r$.

Remarks:

The features of the JVOB [8] are the following,

- (1) The inversion of the inertia matrix (M^{-1}) is needed.
- (2) No need for the measurement of joint acceleration.

2.6. Observer 4: DKF

Various versions of Kalman filter have been developed to estimate forces, while some need acceleration sensor, some based on linear system model thus not applicable to nonlinear system like multilink robot, some need system's Jacobian which may bring numerical problem, and some need inverse of inertia matrix [11], [30], [31]. The following presented disturbance Kalman filter (DKF) overcame these disadvantages [11]. Note that the DKF is an unconstrained Kalman filter, while an example of a constrained Kalman filter implemented in a vehicle control system for real-time road bank estimation can refer to [32].

The DKF method [11, 4] uses a disturbance model and a state space representation of the robot system to estimate the disturbances as a lumped term. The detailed methods are introduced below.

First of all, a dynamic model of the disturbance needs to be constructed via an exogenous system which can be given by

$$\begin{cases} {}^i\dot{\omega} = {}^iS \cdot {}^i\omega \\ {}^i\tau_{dist} = {}^iH \cdot {}^i\omega \end{cases} \quad (23)$$

where the left superscript i is the i -th joint, ${}^i\omega \in \mathbb{R}^{l \times 1}$ is the disturbance dynamics variables, ${}^iS \in \mathbb{R}^{l \times l}$ and ${}^iH \in \mathbb{R}^{1 \times l}$ are system matrices of the disturbance.

For a constant or piecewise constant disturbance, it will have

$$\begin{cases} {}^iS = 0 \\ {}^iH = 1 \end{cases} \quad (24)$$

Correspondingly, for a 3DOF robot, simply it will be

$$\begin{cases} \bar{S} = \text{diag}\{{}^1S, {}^2S, {}^3S\} = \text{diag}\{[0, 0, 0]\} \\ \bar{H} = \text{diag}\{{}^1H, {}^2H, {}^3H\} = \text{diag}\{[1, 1, 1]\} \end{cases} \quad (25)$$

For a high-order disturbance with $(r - 1)$ -th order polynomial, there will have

$$\begin{cases} {}^iS = \begin{bmatrix} 0_{(r-1) \times 1}, & I_{r-1} \\ & 0, 0_{1 \times (r-1)} \end{bmatrix} \\ {}^iH = \begin{bmatrix} 1, 0_{1 \times (r-1)} \end{bmatrix} \end{cases} \quad (26)$$

where r is the size of a vector for representing a polynomial.

Note that when $r = 1$, the polynomial disturbance will be reduced as the constant (0-order polynomial) disturbance scenario (24).

For a 1st-order polynomial ($r = 2$) disturbance $([{}^i\omega^1, {}^i\omega^0]^T)$, it will have

$$\begin{cases} {}^iS = \begin{bmatrix} 0, 1 \\ 0, 0 \end{bmatrix} \\ {}^iH = \begin{bmatrix} 1, 0 \end{bmatrix} \end{cases} \quad (27)$$

Correspondingly, for a 3DOF robot with a 1st-order polynomial ($r = 2$) disturbance $([{}^i\omega^1, {}^i\omega^0]^T)$, it will have

$$\begin{cases} \bar{S} = \begin{bmatrix} 0, 1, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 1, 0, 0 \\ 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 1 \\ 0, 0, 0, 0, 0, 0 \end{bmatrix} \\ \bar{H} = \begin{bmatrix} 1, 0, 0, 0, 0, 0 \\ 0, 0, 1, 0, 0, 0 \\ 0, 0, 0, 0, 1, 0 \end{bmatrix} \end{cases} \quad (28)$$

For a harmonic disturbance with a frequency of f (rad/s), it will have

$$\begin{cases} {}^iS = \begin{bmatrix} 0, f \\ -f, 0 \end{bmatrix} \\ {}^iH = \begin{bmatrix} 1, 0 \end{bmatrix} \end{cases} \quad (29)$$

Note that for a relatively low-frequency harmonic disturbance (*e.g.*, in the specific case of this paper, the sinusoidal disturbance $[0.5, 0.2, 0.5]$ Hz, the harmonic disturbance $[0.1, 0.1, 0.1]$ Hz), the disturbance model can be constructed as a polynomial type (26) instead of a harmonic type (29).

A composite robot system model in state space form can be expressed by

$$\begin{cases} \dot{x} = Ax + Bu + \xi_{pro} \\ y = Cx + \xi_m \end{cases} \quad (30)$$

where $y = p$ is the momentum assumed to be measurable with measurement noise ξ_m , the system state is $x = [p, \omega]^T$, where $p = M\dot{q}$, $\omega = [{}^1\omega, {}^2\omega, \dots, {}^n\omega] = [{}^1\omega^{r-1}, \dots, {}^1\omega^0, {}^2\omega^{r-1}, \dots, {}^2\omega^0, \dots, {}^n\omega^{r-1}, \dots, {}^n\omega^0]$, $\xi_{pro} \sim N(0, \Sigma_{pro})$ is the process noise of the system with covariance matrix $\Sigma_{pro} = \text{diag}\{\Sigma_p, \Sigma_{dist}\}$, and $\xi_m \sim N(0, \Sigma_m)$ is the measurement noise with covariance matrix Σ_m .

$$\begin{cases} A = \begin{bmatrix} 0_{n \times n}, & \bar{H} \\ 0_{l_{sum} \times n}, & \bar{S} \end{bmatrix} \\ B = \begin{bmatrix} I_n \\ 0_{l_{sum} \times n} \end{bmatrix} \\ C = \begin{bmatrix} I_n, & 0_{n \times l_{sum}} \end{bmatrix} \\ \bar{H} = \text{diag}\{[{}^1H, \dots, {}^nH]\} \\ \bar{S} = \text{diag}\{[{}^1S, \dots, {}^nS]\} \end{cases} \quad (31)$$

where $l_{sum} = r \times n = 2 \times 3 = 6$ for case of $r = 2, n = 3$.

The discretized form of model (30) can be expressed as,

$$\begin{cases} x_k = A_k x_{k-1} + B_k u_k + \xi_{pro,k} \\ y_k = C_k x_k + \xi_{m,k} \end{cases} \quad (32)$$

where $u_k = (S^T \dot{q} - g + \tau)$ is the system input, $\xi_{pro,k}$ is the process noise of the discretized system with covariance matrix Q_k , and $\xi_{m,k}$ is the measurement noise of the discretized system with covariance matrix R_k . A_k, B_k are obtained from the matrix exponential of matrix H_1 as

$$\begin{cases} H_1 = \begin{bmatrix} A, & B \\ 0_{n \times (n+l_{sum})}, & 0_{n \times n} \end{bmatrix} \\ e^{H_1 \times T_s} = \begin{bmatrix} A_k, & B_k \\ 0_{n \times (n+l_{sum})}, & I_n \end{bmatrix} \end{cases} \quad (33)$$

where T_s is the sampling time.

Then the state estimation $x_{k|k-1}$ and covariance estimation $P_{k|k-1}$ can be calculated respectively as

$$\begin{cases} x_{k|k-1} = A_k \hat{x}_{k-1} + B_k u_k \\ P_{k|k-1} = A_k \hat{P}_{k-1} A_k^T + Q_k \end{cases} \quad (34)$$

where it can set $\hat{x}_0 = 0_{(n+l_{sum}) \times 1}$, and $\hat{P}_0 = I_{(n+l_{sum})}$ as the initialization.

Then, the Kalman filter gain K_k is calculated by

$$K_k = P_{k|k-1} C_k^T (C_k P_{k|k-1} C_k^T + R_k)^{-1} \quad (35)$$

Then, the state estimation \hat{x}_k and covariance estimation \hat{P}_k can be updated respectively by using

$$\begin{cases} \hat{x}_k = x_{k|k-1} + K_k (y_k - C_k x_{k|k-1}) \\ \hat{P}_k = (I - K_k C_k) P_{k|k-1} \end{cases} \quad (36)$$

Finally, the disturbance estimation $\hat{\tau}_{dist,k}$ can be obtained by

$$\hat{\tau}_{dist,k} = \bar{H} \cdot [0_{l_{sum} \times n}, \quad I_{l_{sum}}] \cdot \hat{x}_k \quad (37)$$

Remarks:

The features of the DKF method are the following,

(1) No need for the inversion of the inertia matrix (M^{-1}).

- (2) No need for the measurement of joint acceleration.
- (3) No need for the system's Jacobian in the DKF method.
- (4) The system model (30) is a linear time-invariant (LTI), and it is observable for any choices of the disturbances types (24), (26), (29) since its observability matrix can satisfy the full-rank condition [11].
- (5) A disturbance model (23) needs to be constructed.
- (6) The implementation of the DKF method is complicated compared to other observers.
- (7) When designing the process noise related covariance matrix Q_k and the measurement noise related covariance matrix R_k , the larger values on the diagonal position mean larger noise, while smaller values mean smaller noise. Therefore, for the momentum p related diagonal values in Q_k , it can be set relatively small since it is assumed to be measurable, while the disturbance-related diagonal values can be set relatively large to indicate that the disturbance is with large uncertainties.

2.7. Observer 5.1: ESO.orig

The original extended state observer (ESO.orig) is initially proposed by Han [19] for active disturbance rejection control (ADRC). The ESO.orig [21] is presented below.

From the dynamic model (6), The joints acceleration can be re-written as

$$\ddot{q} = M^{-1}(\tau - S\dot{q} - g + \tau_{dist}) = M^{-1}(\tau - S\dot{q} - g) + M^{-1}\tau_{dist} \quad (38)$$

The original 3rd-order linear ESO can be given by

$$\begin{cases} \dot{x}_1 = -\beta_1 e_1 + x_2 \\ \dot{x}_2 = -\beta_2 e_1 + x_3 + M^{-1}(\tau - S\dot{q} - g) \\ \dot{x}_3 = -\beta_3 e_1 \\ e_1 = x_1 - q \end{cases} \quad (39)$$

where $x_1 = \hat{q}$, $x_2 = \dot{\hat{q}}$, $x_3 = M^{-1}\tau_{dist}$ in the state space. $\beta_1, \beta_2, \beta_3$ are diagonal matrices need to be designed.

Then, the estimated lumped disturbances can be given by

$$\hat{\tau}_{dist} = M \cdot x_3 \quad (40)$$

Remarks:

The features of ESO.orig observer are the following,

- (1) The inversion of the inertia matrix (M^{-1}) is needed.
- (2) There are three observer gains that need to be tuned.
- (3) This observer works for constant payload, but not for time-varying payload.

2.8. Observer 5.2: ESO.modi

To avoid the inverse of the inertia matrix, a modified version of extended state observer (ESO.modi) [21] is presented below. The ESO.modi, a modified 2nd-order linear ESO is given by

$$\begin{cases} \dot{x}_1 = -\beta_1 e_1 + x_2 + (\tau + S^T \dot{q} - g) \\ \dot{x}_2 = -\beta_2 e_1 \\ e_1 = x_1 - p \end{cases} \quad (41)$$

where $x_1 = \hat{p}$, $x_2 = \hat{\tau}_{ext}$ in the state space, $p = M\dot{q}$ is the generalized momentum of the robot, β_1, β_2 are diagonal gain matrices which are usually selected such that the corresponding state space matrix A is Hurwitz and has real negative eigenvalues.

Then, the estimated lumped disturbances can be given by

$$\hat{\tau}_{ext} = x_2 \quad (42)$$

Remarks:

The feature of ESO.modi are the following [21],

- (1) The phase lag is decreased by reducing the observer order compared to the ESO.orig observer.
- (2) No need for the inversion of the inertia matrix (M^{-1}).
- (3) There are two observer gains that need to be tuned.
- (4) For scenarios of constant disturbance and time-varying disturbance, the observer gains need to be tuned separately in order for optimal results in each scenario. In other words, one set of observer gains is not suitable for both scenarios.
- (5) The disturbance tracking error of ESO converges asymptotically when the dynamic model is available, while it is bounded when the dynamic model is not available [33, 21].

2.9. Observer 5.3: ESO.impr

The improved version of extended state observer (ESO.impr) [12] that is extended especially for time-varying external disturbances is presented below. The ESO.impr, an improved 3rd-order linear ESO can be given by

$$\begin{cases} \dot{x}_1 = \frac{\beta_1}{\epsilon} e_1 + x_2 \\ \dot{x}_2 = \frac{\beta_2}{\epsilon^2} e_1 + M^{-1}(x_3 + \tau) - M^{-1}(S\dot{q} + g) \\ \dot{x}_3 = \frac{1}{\epsilon^3} e_1 \\ e_1 = q - x_1 \end{cases} \quad (43)$$

where $x_1 = \hat{q}$, $x_2 = \hat{\dot{q}}$, $x_3 = \hat{\tau}_{ext}$ in the state space, $(\beta_1, \beta_2, \epsilon)$ are positive scalar parameters need to be designed. For the full designing procedures and theoretical analysis of this ESO.impr please refer to [12].

Then, the estimated lumped disturbances can be given by

$$\hat{\tau}_{ext} = x_3 \quad (44)$$

Remarks:

The features of ESO.impr are the following [12],

- (1) The inversion of the inertia matrix (M^{-1}) is needed.
- (2) There are three scalar observer gains that need to be tuned which make the tuning relatively simpler than ESO.orig and ESO.modi.
- (3) The ESO.impr is a linear ESO but it works for nonlinear dynamic systems and a large class of external forces.

3. Simulations and Results

3.1. Robotic System

A 3-DOF PHANToM Premium 1.5A robot (3D Systems, Inc., Cary, NC, USA) is used for simulations in this paper in order to test the disturbance tracking performance of different observers. To build a virtual model of this robot, the kinematic model and dynamic model of the PHANToM robot are reconstructed

Table 1: Denavit–Hartenberg (DH) parameters for the 3-DOF Phantom Premium 1.5A robot’s kinematic chain (for the homogeneous transform in the modified convention).

	no.	Joint	$a(m)$	$\alpha(rad)$	$d(m)$	$\theta(rad)$
	1	Joint 01	0	$-\frac{\pi}{2}$	L_2	$-\frac{\pi}{2}$
\oplus	2	Joint 02	$-L_1$	0	0	0
\oplus	3	Joint 1	0	0	0	q_1
	4	Joint 2	0	$\frac{\pi}{2}$	0	q_2
	5	Joint 3	L_1	0	0	$q_3 - q_2 - \frac{\pi}{2}$
\oplus	6	Joint 4 (EE)	L_2	0	0	0
\oplus	7	Joint 5 (EE)	0	$-\frac{\pi}{2}$	0	$\frac{\pi}{2}$

Note:

- L_1 and L_2 are link length.
- Symbol \oplus means that the DH parameters of these two adjacent joints can be directly summed together, respectively, to be as one joint.
- Joint 01, 02, and 5 are virtual joints that are only used for transforming one frame to another desired one via translation and/or rotation.

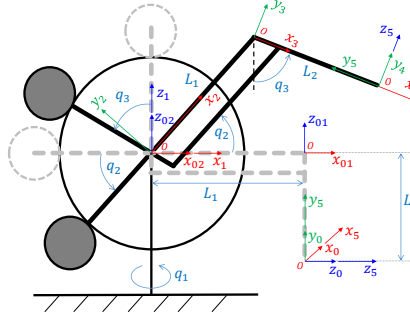


Figure 1: Schematic of the 3-DOF Phantom Premium 1.5A robot and frame attachment to each joint. Frame {0} is the base frame while frame {5} is the end-effector (EE) frame. L_1, L_2 are link lengths. q_1, q_2, q_3 are joint angle variables.

based on [34]. All the simulations are conducted by using MATLAB/Simulink (version R2020a, MathWorks Inc., Natick, MA, USA), which is running on a computer with a 3.70 GHz Intel(R) Core(TM) i5-9600K CPU and a Windows 10 Education 64-bit operating system. The control rate of the virtual robot is set as 1,000 Hz, while the sampling rate for acquiring data is 500 Hz.

3.2. Robot Kinematics

A thorough exploration of the kinematics and dynamics of the 3-DOF Phantom Premium 1.5A can refer [34]. Additionally, a Denavit–Hartenberg (DH) table for this robot is provided in Table 1. The corresponding frames’ definitions are illustrated in Fig. 1. Please note that the base frame of this robot is defined to be coincident with the initial robot end-effector (EE) position (see the gray pose in Fig. 1).

According to the DH parameters in Table 1 and the frames determined in Fig. 1, the homogeneous transformation matrix T from EE frame {5} to base frame {0} can be obtained as

$$T = \begin{bmatrix} c_1, & -s_1 s_3, & c_3 s_1, & s_1(L_1 c_2 + L_2 s_3) \\ 0, & c_3, & s_3, & L_2 - L_2 c_3 + L_1 s_2 \\ -s_1, & -c_1 s_3, & c_1 c_3, & L_1 c_1 c_2 - L_1 + L_2 c_1 s_3 \\ 0, & 0, & 0, & 1 \end{bmatrix} \quad (45)$$

where s_i, c_i represent $\sin(q_i), \cos(q_i), i = 1, 2, 3$, respectively. L_1, L_2 are link lengths.

The Jacobian matrix J can be expressed by

$$J = \begin{bmatrix} c_1(L_1 c_2 + L_2 s_3), & -L_1 s_1 s_2, & L_2 c_3 s_1 \\ 0, & L_1 c_2, & L_2 s_3 \\ -s_1(L_1 c_2 + L_2 s_3), & -L_1 c_1 s_2, & L_2 c_1 c_3 \\ 0, & 0, & -c_1 \\ 1, & 0, & 0 \\ 0, & 0, & s_1 \end{bmatrix} \quad (46)$$

Note that the Jacobian matrix here is the space Jacobian which can be calculated from the body Jacobian used in [34]. In this paper, we only use the upper 3-by-3 linear part of the Jacobian in (46), which means that the rotational angles in Cartesian space are ignored.

3.3. Parameterization

For all simulations in the remaining part of this paper, the values assigned to the parameters are summarized in Table 2.

It has been proved in our earlier work that by integrating an impedance controller and an observer, an accurate impedance control can be achieved in a trajectory tracking task when the actual velocity and acceleration converge to the desired ones [24]. It is noteworthy that, the observer estimation accuracy is not affected by the trajectory tracking accuracy. In other words, even if the trajectory tracking performance is poor (*e.g.*, due to a small stiffness gain setting in the impedance model, or uncompensated disturbances), the observer will still have an accurate estimation of the lumped uncertainties.

In order to explore the disturbance tracking performance of different observers integrated with an impedance controller, a figure-eight trajectory expressed by (47) is employed in the simulated trajectory tracking task.

$$\begin{cases} x_d = R \sin(\frac{2\pi}{t_1} t) \cos(\frac{2\pi}{t_1} t) \\ y_d = R \sin(\frac{2\pi}{t_1} t) + R \\ z_d = 0 \end{cases} \quad (47)$$

where $R = 0.02$ m is the amplitude of the figure-eight trajectory, $t_1 = 5$ s is the period for generating a full cycle.

Then, two typical types of external disturbance, *i.e.*, constant payload and time-varying payload, are applied to the robot EE separately. The constant payload is 22 grams, and the time-varying payload is represented by a set of sinusoidal curves given by (48).

$$\begin{cases} Fx_d = a_1 \sin(\frac{2\pi}{t_1} t) \\ Fy_d = a_2 \sin(\frac{2\pi}{t_2} t) \\ Fz_d = a_3 \sin(\frac{2\pi}{t_3} t) \end{cases} \quad (48)$$

Table 2: Parameterization for simulations and experiments.

Observer	Parameter	Simulation	Experiment	Location
	Spring	$K_m = 7.5I$	$K_m = 7.5I$	(3)
	Damping	$D_m = 2\sqrt{7.5}I$	$D_m = 2\sqrt{7.5}I$	(3)
NDOB	Observer gain	$Y = 0.0958I$	$Y = 0.00958I$	(7)
GMO	Observer gain	$K_o = 100I$	$K_o = 50I$	(13)
JVOB	Observer gain	$K_o = 0.1I$	$K_o = 0.05I$	(20)
DKF	disturbance model	${}^iS = [0, 1; 0, 0], i = 1, 2, 3$	${}^iS = [0, 1; 0, 0], i = 1, 2, 3$	(26)
DKF	disturbance model	${}^iH = [1, 0], i = 1, 2, 3$	${}^iH = [1, 0], i = 1, 2, 3$	(26)
DKF	initialization	$\hat{x}_0 = 0_{9 \times 1}$	$\hat{x}_0 = 0_{9 \times 1}$	(34)
DKF	initialization	$\hat{P}_0 = I_{9 \times 9}$	$\hat{P}_0 = I_{9 \times 9}$	(34)
DKF	process noise	$Q_k = \text{diag}\{[a_i]\}, i = 1, \dots, 9$ $a_i = 0.001, i = 1, 2, 3$ $a_i = 20, i = 6, \dots, 9$	$Q_k = \text{diag}\{[a_i]\}, i = 1, \dots, 9$ $a_i = 0.001, i = 1, 2, 3$ $a_i = 10, i = 6, \dots, 9$	(34)
DKF	measurement noise	$R_k = \text{diag}\{[0.1, 0.1, 0.1]\}$	$R_k = \text{diag}\{[0.1, 0.1, 0.1]\}$	(35)
ESO.orig	Observer gain	$\beta_1 = 2I$	$\beta_1 = 10I$	(39)
ESO.orig	Observer gain	$\beta_2 = 22I$	$\beta_2 = 110I$	(39)
ESO.orig	Observer gain	$\beta_3 = 20I$	$\beta_3 = 100I$	(39)
ESO.modi	Observer gain	$\beta_1 = 4I$	$\beta_1 = 20I$	(41)
ESO.modi	Observer gain	$\beta_2 = 80I$	$\beta_2 = 400I$	(41)
ESO.impr	Observer gain	$\beta_1 = 100$	$\beta_1 = 100$	(43)
ESO.impr	Observer gain	$\beta_2 = 120$	$\beta_2 = 120$	(43)
ESO.impr	Observer gain	$\epsilon = 0.1$	$\epsilon = 0.1$	(43)

Note: $I \in \mathbb{R}^{3 \times 3}$ denote identity matrix.

where $t_1 = 2, t_2 = 5, t_3 = 2$ are cycles of the desired time-varying EE payload for each axis in units of second, $a_1 = 0.01, a_2 = 0.2, a_3 = 0.01$ are the corresponding amplitudes in units of Newton.

3.4. Individual Observer Simulations

In this section, the trajectory tracking performance of each observer under three types of disturbances is presented, *i.e.*, ideal dynamic model without any disturbances, dynamic model under a constant disturbance, and dynamic model under a sinusoidal disturbance. For the scenario of no disturbances, no observers need to be implemented, and the trajectory tracking task performance under an impedance controller is taken as a reference. For the scenarios of constant disturbance and sinusoidal disturbances, each of the introduced observers, *i.e.*, NDOB, GMO, JVOB, DKF, ESO.orig, ESO.modi, and ESO.impr, will be implemented to track the disturbance and thus do a compensation in the control law. The disturbance tracking performance of each observer will be presented, respectively in the following part. Note that the observer gains in the simulations are tuned by using a trial and error method based on a binary searching strategy.

With an ideal dynamic model, the figure-eight trajectory tracking performance under an impedance controller (3) is shown in Figs. 2a and 2b. The figure shows that the impedance controller with properly tuned impedance gains (M_m, D_m, K_m) can accurately track the desired figure-eight trajectory when none of any disturbances or uncertainties are involved. However, when a constant payload of 22 grams is added to the robot EE, the trajectory tracking performance is affected as shown in Fig. 2c. And when a time-varying sinusoidal payload (48) is added to the robot EE, the affected tracking performance is shown in Fig. 2d. From Figs. 2c, 2d we can see that, if disturbance exists in the control system but not be compensated, the trajectory tracking performance can be significantly affected.

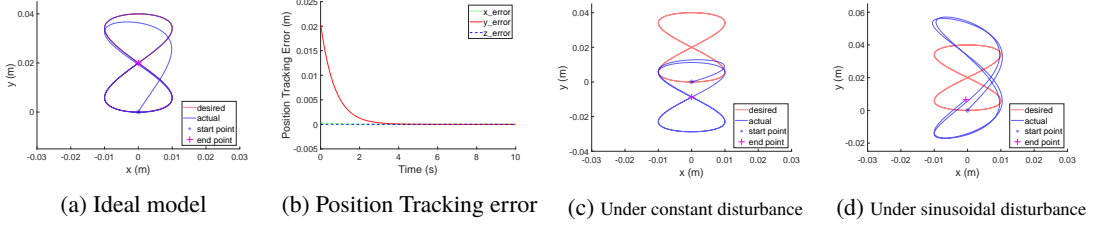


Figure 2: Simulation of figure-eight trajectory tracking under different types of disturbance. (a) ideal dynamic model without any disturbances; (b) position tracking error with the ideal dynamic model; (c) trajectory tracking performance under a constant disturbance of a 22 grams payload; (d) trajectory tracking performance under a sinusoidal disturbance.

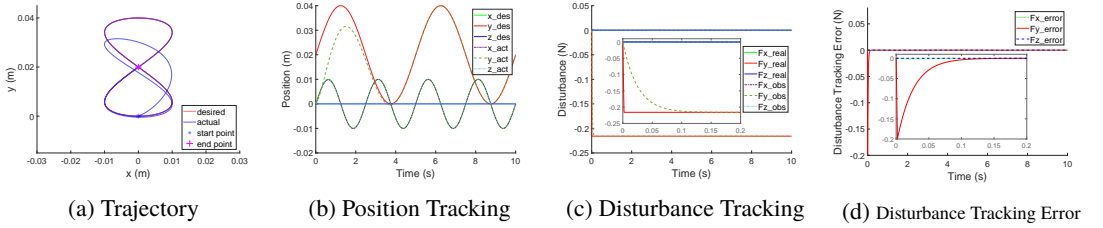


Figure 3: Trajectory tracking performance and disturbance tracking performance with NDOB observer under a constant disturbance of a constant 22 grams payload.

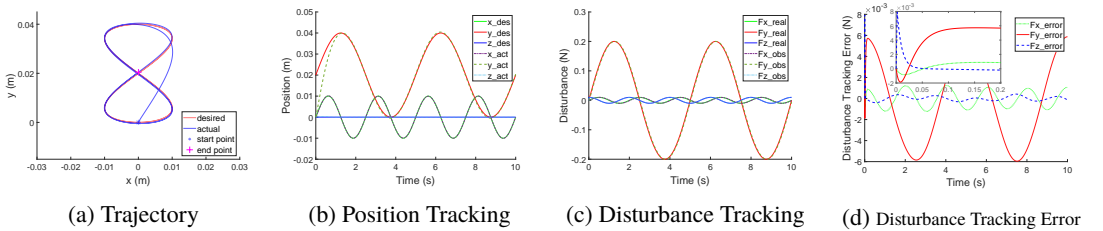


Figure 4: Trajectory tracking performance and disturbance tracking performance with NDOB observer under a time-varying disturbance of a sinusoidal payload.

With implementing the observer of NDOB, the trajectory tracking performance in the constant-payload scenario and time-varying-payload scenario are shown in Figs. 3a, 3b and Figs. 4a, 4b, respectively, while the disturbance tracking performance in two scenarios are shown in Figs. 3c, 3d and Figs. 4c, 4d, respectively. Fig. 3 and Fig. 4 show that with NDOB observer, both the constant and sinusoidal disturbance can be accurately estimated and compensated, thus an accurate trajectory tracking performance can be recovered compared to the reference in Fig. 2. Note that the disturbance tracking can reach a steady state within 0.1 seconds in both scenarios. For the term steady state, we define it as when the actual values are within 5% of the target value (for constant disturbances) or when the disturbance tracking error falls into its upper/lower bound for the first time (for time-varying disturbances).

With implementing the observer of GMO, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 5a, 5b and Figs. 6a, 6b, respectively. The disturbance tracking performance in two scenarios are shown in Figs. 5c, 5d and Figs. 6c, 6d,

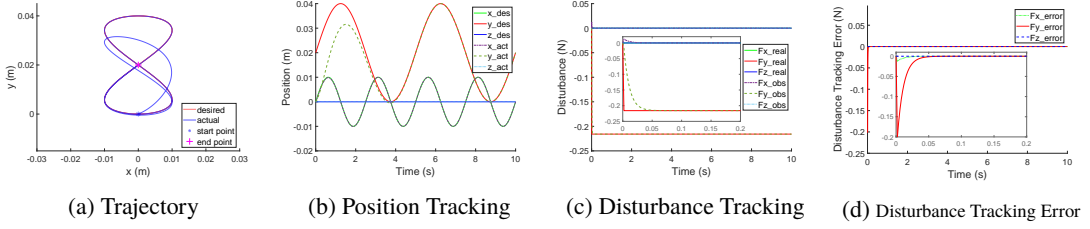


Figure 5: Trajectory tracking performance and disturbance tracking performance with GMO observer under a constant disturbance of a constant 22 grams payload.

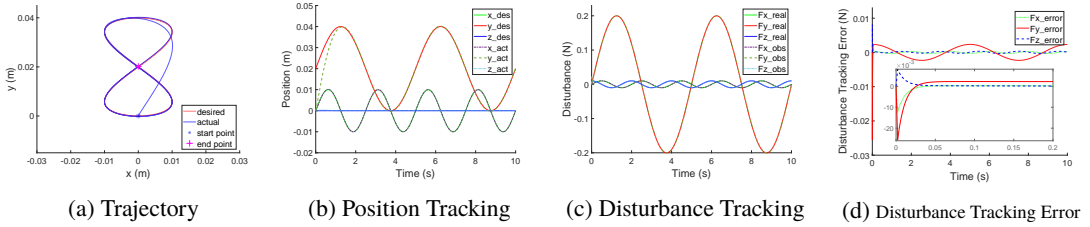


Figure 6: Trajectory tracking performance and disturbance tracking performance with GMO observer under a time-varying disturbance of a sinusoidal payload.

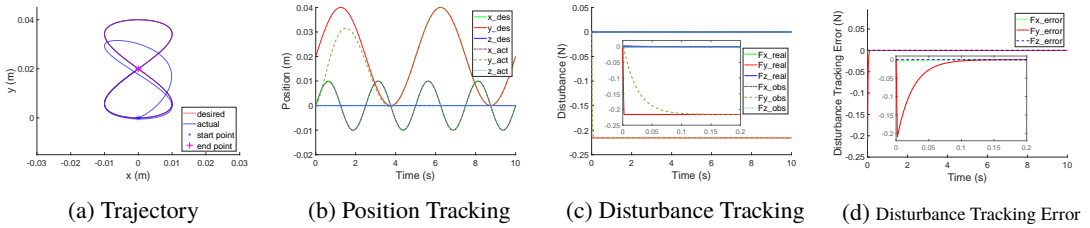


Figure 7: Trajectory tracking performance and disturbance tracking performance with JVOB observer under a constant disturbance of a constant 22 grams payload.

respectively. Fig. 5 and Fig. 6 show that with GMO observer, both the constant and sinusoidal disturbance can be accurately estimated and compensated, thus an accurate trajectory tracking performance can be recovered compared to the reference in Fig. 2. The disturbance tracking can reach a steady state within 0.1 seconds in both scenarios. The disturbance tracking performance of GMO is very similar to that of NDOB.

With implementing the observer of JVOB, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 7a, 7b and Figs. 8a, 8b, respectively. The disturbance tracking performance in two scenarios are shown in Figs. 7c, 7d and Figs. 8c, 8d, respectively. Fig. 7 and Fig. 8 show that, by implementing JVOB observer to compensate for disturbances, an accurate trajectory tracking performance is able to be recovered in both the constant-disturbance scenario and the time-varying-disturbance scenario when comparing to the reference in Fig. 2. The disturbance tracking can reach a steady state within 0.1 seconds in both scenarios. Also, we can find that the disturbance tracking performance of JVOB is very similar to that of NDOB and GMO.

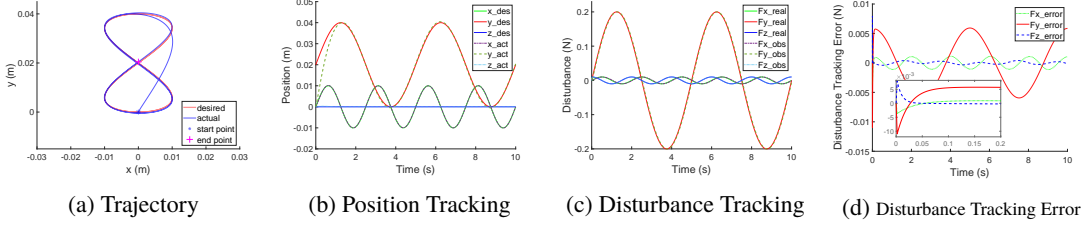


Figure 8: Trajectory tracking performance and disturbance tracking performance with JVOB observer under a time-varying disturbance of a sinusoidal payload.

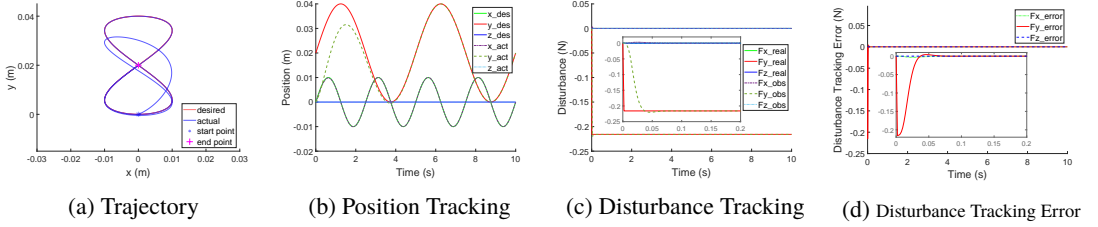


Figure 9: Trajectory tracking performance and disturbance tracking performance with DKF observer under a constant disturbance of a constant 22 grams payload.

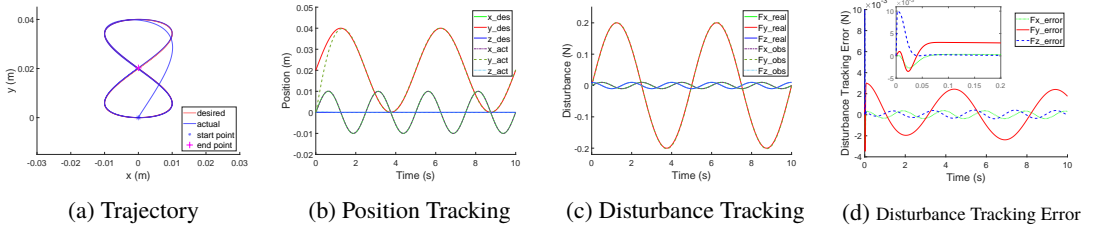


Figure 10: Trajectory tracking performance and disturbance tracking performance with DKF observer under a time-varying disturbance of a sinusoidal payload.

With implementing the observer of DKF, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 9a, 9b and Figs. 10a, 10b, respectively. The disturbance tracking performance in two scenarios are shown in Figs. 9c, 9d and Figs. 10c, 10d, respectively. Fig. 9 and Fig. 10 show that with DKF observer, both the constant and time-varying disturbance can be accurately estimated and compensated, thus an accurate trajectory tracking performance can be recovered to be normal compared to the reference in Fig. 2. The disturbance tracking can reach a steady state within 0.1 seconds in both scenarios. It is interesting to find that the disturbance tracking performance of DKF in this specific case is very similar to that of NDOB, GMO, and JVOB.

With implementing the observer of ESO.orig, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 11a, 11b and Figs. 12a, 12b, respectively. The disturbance tracking performance in two scenarios are shown in Figs. 11c, 11d and Figs. 12c, 12d, respectively. Comparing the disturbance tracking performance of ESO.orig in Fig. 11 and Fig. 12, we can see that ESO.orig observer can track constant disturbance but not time-varying sinusoidal disturbance. Also, for tracking constant disturbance, it takes a significantly long time (about

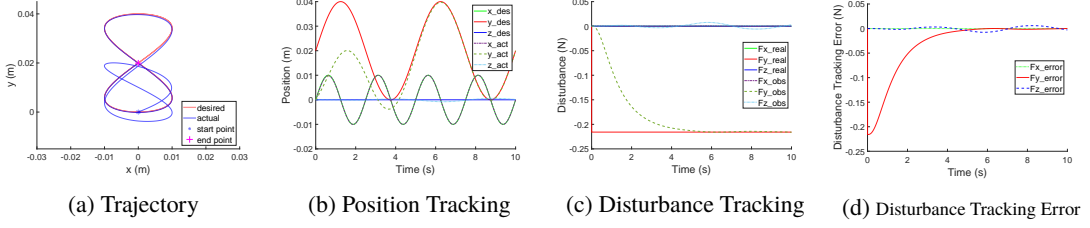


Figure 11: Trajectory tracking performance and disturbance tracking performance with ESO.orig observer under a constant disturbance of a constant 22 grams payload.

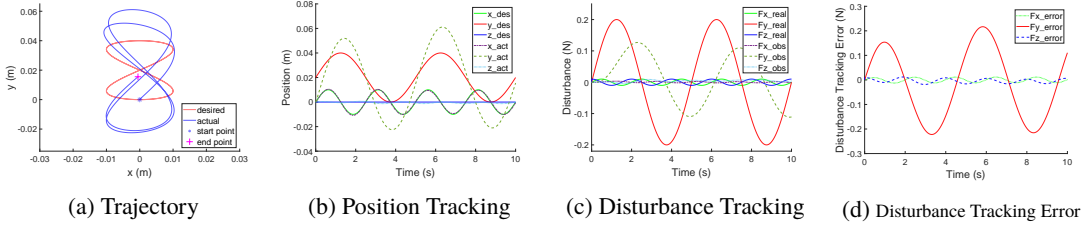


Figure 12: Trajectory tracking performance and disturbance tracking performance with ESO.orig observer under a time-varying disturbance of a sinusoidal payload.

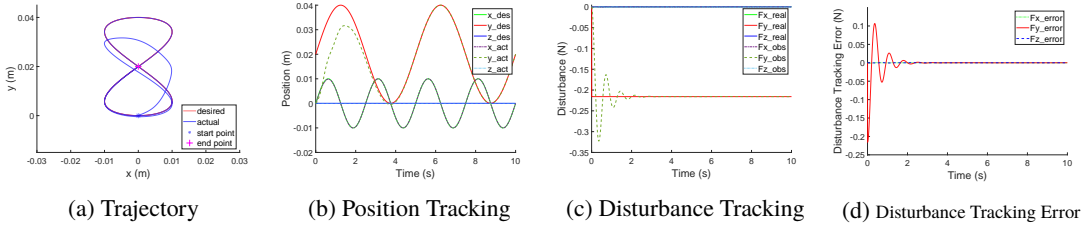


Figure 13: Trajectory tracking performance and disturbance tracking performance with ESO.modi observer under a constant disturbance of a constant 22 grams payload.

6 seconds) to reach a steady state. Since the ESO.orig observer cannot track time-varying disturbance, the figure-eight trajectory tracking is still distorted as shown in Fig. 12a when comparing it with the corresponding reference in Fig. 2d.

With implementing the observer of ESO.modi, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 13a, 13b and Figs. 14a, 14b, respectively. The disturbance tracking performance in two scenarios are shown in Figs. 13c, 13d and Figs. 14c, 14d, respectively. For tracking constant disturbance in Fig. 13, the disturbance can be accurately estimated and compensated, thus trajectory tracking performance is recovered to be normal. For tracking time-varying disturbance in Fig. 14, although the disturbance tracking error is relatively large compared to the results from NDOB, GMO, JVOB, and DKF, the trajectory tracking performance is recovered to a large extent when comparing to the reference in Fig. 2. Note that it also takes a relatively long time for the disturbance tracking to reach a steady state (about 2 seconds) in both scenarios.

With implementing the observer of ESO.impr, the trajectory tracking performance in constant-payload scenario and time-varying-payload scenario are shown in Figs. 15a, 15b and Figs. 16a, 16b,

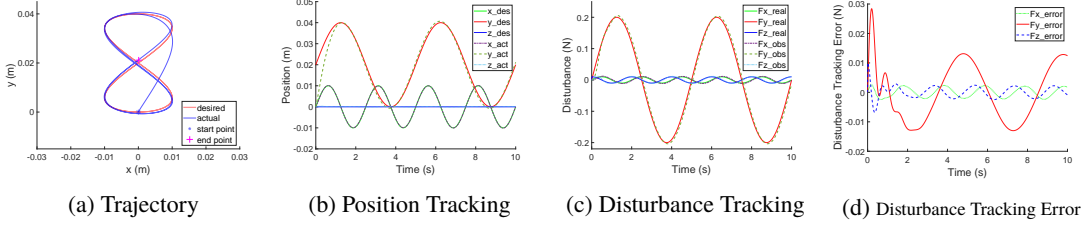


Figure 14: Trajectory tracking performance and disturbance tracking performance with ESO.modi observer under a time-varying disturbance of a sinusoidal payload.

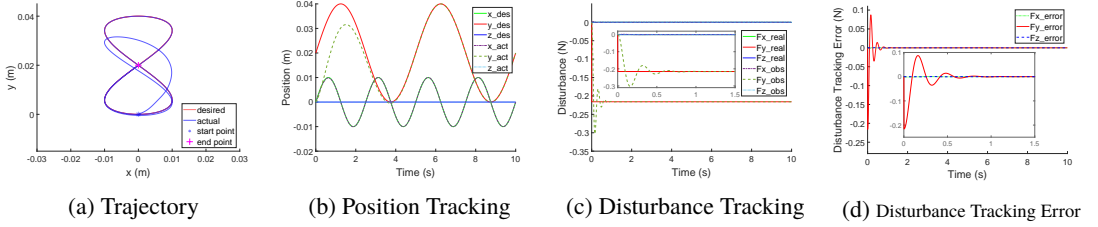


Figure 15: Trajectory tracking performance and disturbance tracking performance with ESO.impr observer under a constant disturbance of a constant 22 grams payload.

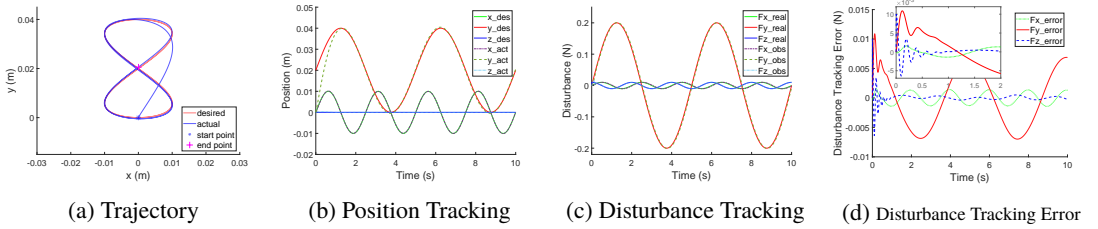


Figure 16: Trajectory tracking performance and disturbance tracking performance with ESO.impr observer under a time-varying disturbance of a sinusoidal payload.

respectively. The disturbance tracking performance in two scenarios are shown in Figs. 15c, 15d and Figs. 16c, 16d, respectively. For tracking constant disturbance in Fig. 15, the disturbance can be accurately estimated and compensated, thus trajectory tracking performance is recovered to be normal. For tracking time-varying disturbance in Fig. 16, ESO.impr is better than ESO.orig and ESO.modi in terms of disturbance tracking error, but not as good as NDOB, GMO, JVOB, and DKF in the same simulated scenario. It takes a relatively long time for the disturbance tracking to reach a steady state (about 1 second) in both disturbance scenarios.

3.5. Comparing Observers under Constant Disturbance

Here in this section, we put the disturbance tracking errors of all observers in the same figure for an intuitive and qualitative comparison. Note that these comparisons are expected to provide an intuitive sense of the possible behaviors of different observers in the same simulated scenario rather than quantitative analyses. It should be noted that the behavior of an observer may be varying due to many factors, such

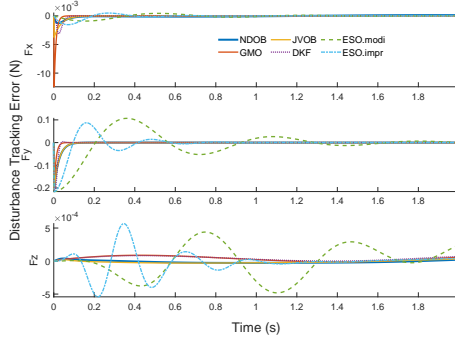


Figure 17: Disturbance tracking errors of different observers in the same constant payload scenario.

Table 3: MSE of constant disturbance tracking in steady state ([5, 10] second)

	observer	$MSE_{F_x} (N^2)$	$MSE_{F_y} (N^2)$	$MSE_{F_z} (N^2)$
1	NDOB	8.5512 e-09	4.0159 e-09	7.1058 e-10
2	GMO	2.5659 e-12	3.1420 e-10	2.6744 e-09
3	JVOB	7.3894 e-13	1.8366 e-09	6.5021 e-10
4	DKF	8.0715 e-11	3.0547 e-10	2.5286 e-09
5.1	ESO.orig	2.5473 e-08	1.1430 e-06	2.3432 e-05
5.2	ESO.modi	2.5466 e-11	7.5043 e-09	2.0335 e-08
5.3	ESO.impr	1.9040 e-11	2.6477 e-09	8.0222 e-10

Note: The tracked external disturbance is a constant payload of 22 grams (-0.2158 N along the y-axis).

as gain tuning or a specific application scenario setting. One observer may have better performance than others in one specific scenario, but may also have worse performance in another scenario.

Fig. 17 shows the disturbance tracking error in Cartesian space when different observers are implemented to track a constant payload of 22 grams (-0.2158 N along the y-axis). Since the ESO.orig observer has poor disturbance tracking performance especially on time-varying disturbance, it does not appear in the figure. As clearly can be seen in the figure, the ESO.modi and ESO.impr observer have remarkably poor disturbance tracking performance compared to other observers.

The mean squared error (MSE) of constant disturbance tracking in steady state ([5, 10] second) is summarized in Table 3, and the corresponding bar chart is shown in Fig. 18 (note that ESO.orig is excluded due to its significantly higher MSE than other observers). Generally, all observers have relatively good disturbance tracking performance considering that the MSE is at the order of magnitude of 10^{-8} . This is reasonable since the disturbance is a simple constant payload.

In summary, for the disturbance-tracking performance in a simple constant-payload scenario, all observers can reach a relatively high tracking accuracy in the steady state except the ESO.orig. However, the observers of NDOB, GMO, JVOB, and DKF have significantly faster convergence speeds than the observers of ESO.modi and ESO.impr.

3.6. Comparing Observers under Sinusoidal Disturbance

Fig. 19 shows the disturbance tracking errors when different observers are implemented to track a time-varying sinusoidal payload with amplitude $[0.01, 0.2, 0.01]$ N. As mentioned earlier, the ESO.orig observer only works for constant disturbances but not time-varying disturbances, so it does not appear in the figure. As clearly can be seen in the figure, the ESO.modi observer has the worst disturbance tracking performance among all observers presented in the figure.

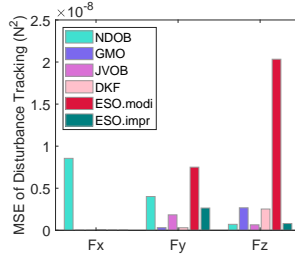


Figure 18: MSE of different observers in the same constant payload scenario in steady state ([5, 10] second).

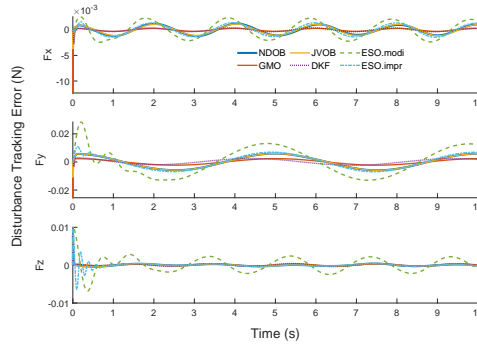


Figure 19: Disturbance tracking errors of different observers in the same time-varying sinusoidal payload scenario.

The MSE of sinusoidal disturbance tracking in steady state ([5, 10] second) is summarized in Table 4, and the corresponding bar chart is shown in Fig. 20 (note that ESO.orig is excluded in the bar chart due to its significantly higher MSE than other observers). As can be seen in the figure, the ESO.modi observer has the worst disturbance-tracking performance among all observers presented in the figure. Note that the sinusoidal time-varying payload is with amplitude [0.01, 0.2, 0.01] N, which means that the disturbance along y-axis has the highest amplitude (0.2 N).

In summary, for the disturbance-tracking performance in a sinusoidal-payload scenario, all observers are capable of tracking the sinusoidal disturbance except ESO.orig. However, unlike their performance in the constant-payload scenario, their tracking accuracy cannot converge to zero errors in the steady state anymore. Instead, their disturbance-tracking errors in the steady state are bounded. In this specific case, the observers of GMO and DKF have the highest disturbance-tracking accuracy, and the observers of NDOB, JVOB, and ESO.impr have relatively lower tracking accuracy, while the observer of ESO.modi has the lowest tracking accuracy. Similar to the constant-payload scenario, the observers of NDOB, GMO, JVOB, and DKF still have significantly faster convergence speeds than the observers of ESO.modi and ESO.impr.

Table 4: MSE of sinusoidal disturbance tracking in steady state ([5, 10] second)

	observer	MSE _{F_x} (N ²)	MSE _{F_y} (N ²)	MSE _{F_z} (N ²)
1	NDOB	5.3739 e-07	1.7244 e-05	3.5122 e-08
2	GMO	4.2431 e-08	2.6948 e-06	5.5458 e-08
3	JVOB	5.9144 e-07	1.7181 e-05	3.6675 e-08
4	DKF	6.5737 e-08	2.8714 e-06	8.3401 e-08
5.1	ESO.orig	6.7428 e-05	0.023265	8.2052 e-05
5.2	ESO.modi	2.4179 e-06	8.3662 e-05	2.6411 e-06
5.3	ESO.impr	9.3663 e-07	2.3691 e-05	5.5001 e-08

Note: The tracked external disturbance is a sinusoidal time-varying payload with amplitude [0.01, 0.2, 0.01] N.

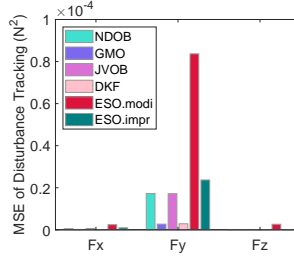


Figure 20: MSE of different observers in the same sinusoidal payload scenario in steady state ([5, 10] second).

3.7. Comparing Observers under Harmonic Disturbance with Gaussian Noise

In this section, a more realistic scenario is simulated where the external disturbance is a set of Fourier series functions (*i.e.*, harmonics) given by

$$\begin{cases} F_x = a_1 \sin(\frac{2\pi n_1}{t_1} t) + a_2 \sin(\frac{2\pi n_2}{t_1} t - \phi_2) + a_3 \sin(\frac{2\pi n_3}{t_1} t - \phi_3) \\ F_y = a_1 \cos(\frac{2\pi n_1}{t_1} t) + a_2 \cos(\frac{2\pi n_2}{t_1} t - \phi_2) + a_3 \cos(\frac{2\pi n_3}{t_1} t - \phi_3) \\ F_z = a_1 \cos(\frac{2\pi n_1}{t_1} t) + a_2 \sin(\frac{2\pi n_2}{t_1} t - \phi_2) + a_3 \cos(\frac{2\pi n_3}{t_1} t - \phi_3) \end{cases} \quad (49)$$

where F_x, F_y, F_z are the force components of the time-varying payload expressed in Cartesian space, and $t_1 = 10$, $a_1 = 0.1$, $a_2 = 0.15$, $a_3 = 0.05$, $n_1 = 1$, $n_2 = 2$, $n_3 = 3$, $\phi_2 = \frac{1}{2}\pi$, $\phi_3 = \pi$. Based on (49), a pseudo-randomly-generated zero-mean Gaussian noise is added to each axis of the time-varying payload while the seeds for the random number generator remain the same over the simulations of different observers.

Fig. 21 shows the disturbance tracking performance of each observer when a complex harmonic disturbance with Gaussian noise is employed. Again, as mentioned earlier, the ESO.orig observer does not work for time-varying disturbances, so it does not appear in the figure. As can be seen in the figure, all observers have relatively good disturbance tracking performance in the steady state, except that different behaviors may exist for different observers in the initial transition period.

The MSE of noisy harmonic disturbance tracking in steady state ([5, 10] second) is summarized in Table 5, and the corresponding bar chart is shown in Fig. 22 (note that ESO.orig is excluded in the bar chart due to its significantly higher MSE than other observers). As can be seen in the figure, the ESO.modi observer has the worst disturbance-tracking performance among all observers presented in terms of MSE. Note that the harmonic disturbance with Gaussian noise is with amplitude [0.3, 0.3, 0.3] N, which means that the disturbance along each axis has the same amplitude (0.3 N).

In summary, for the disturbance-tracking performance in a complex harmonic-disturbance scenario, all observers can track the harmonic disturbance except ESO.orig. Similar to the constant-payload

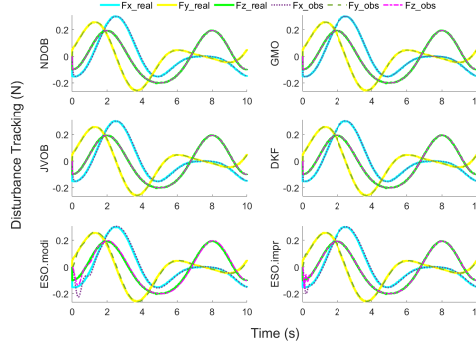


Figure 21: Disturbance tracking performance of different observers when the time-varying payload is complex harmonics with Gaussian noise.

Table 5: MSE of harmonic disturbance with Gaussian noise tracking in steady state ([5, 10] second)

	observer	$MSE_{F_x} (N^2)$	$MSE_{F_y} (N^2)$	$MSE_{F_z} (N^2)$
1	NDOB	1.4764 e-05	1.2384 e-05	9.8996 e-06
2	GMO	8.2024 e-06	8.4555 e-06	1.0378 e-05
3	JVOB	1.4786 e-05	1.2327 e-05	1.0140 e-05
4	DKF	1.0088 e-05	9.6452 e-06	1.0920 e-05
5.1	ESO.orig	0.0017811	0.0037793	0.010515
5.2	ESO.modi	2.3188 e-05	2.6482 e-05	7.3929 e-05
5.3	ESO.impr	1.7919 e-05	1.4301 e-05	1.1400 e-05

Note: The tracked external disturbance is a Fourier Series time-varying payload with amplitude [0.3, 0.3, 0.3] N).

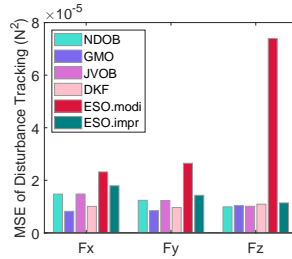


Figure 22: MSE of different observers in steady state ([5, 10] second) when the time-varying disturbance is complex harmonic with Gaussian noise.

scenario and the sinusoidal-payload scenario, the observers of NDOB, GMO, JVOB, and DKF have significantly faster convergence speeds than the observers of ESO.modi and ESO.impr.

4. Experiments and Results

Although the simulations are the main focus of this work, physical experiments are also conducted to demonstrate the effect of different observers on dealing with real-scenario disturbances. The same robotic system used in simulations, *i.e.*, the 3-DOF PHANToM Premium 1.5A robot, is also used for physical experiments. Quarc real-time control software (Quanser Inc., Markham, ON, Canada) is used for real-time control on the 3-DOF physical robot while the control rate is 1,000 Hz. Same as the

simulations, the impedance controller (3) and the figure-eight trajectory (47) are employed in the experimental trajectory tracking task. The parameterization for experiments is summarized in Table 2. A demonstration [video](#) of the experiments with implementing different observers is attached ¹.

For tuning the observer gains in the experiments, all of them are started with the same gains used in the simulations. For NDOB, GMO, JVOB, and DKF, using the exact simulation gains in experiments will cause robot chattering, thus the observer gains are tuned down correspondingly by trial and error method. For ESO.orig and ESO.modi, the gains are tuned up correspondingly in order for better trajectory tracking performance, while ESO.impr uses the exact same gains as that used in the simulations.

For the disturbances in the experiments, dynamic model errors and unmodelled frictions always exist. There are two scenarios taken as references, *i.e.*, Ref.1, and Ref.2, as shown in Fig. 23a and Fig. 23b respectively. Both Ref.1, and Ref.2 are implemented with impedance controller but without any observer, and the only difference is that Ref.1 involves model errors and unmodelled frictions, while Ref.2 involves model errors, unmodelled frictions, and an external constant payload (22 grams). As can be seen in Fig. 23a, the actual trajectory is shrunk compared to the desired one due to the model errors and unmodelled frictions, while in Fig. 23b, the actual trajectory is not only shrunk but also shifted downward due to the additional external payload.

Fig. 23c to Fig. 23i show the trajectory tracking performance when different observers are implemented while the disturbance sources are the same as Ref.2 (Fig. 23b), *i.e.*, dynamic model errors, unmodelled frictions, and an external constant payload (22 grams). As shown in figures of Fig. 23c to Fig. 23i, all observers can effectively estimate and compensate for the disturbances except ESO.orig. Note that ESO.orig results in a shrunk but not downward-shifted actual trajectory (Fig. 23g) that is similar to the result in Ref.1 (Fig. 23a). This indicates that ESO.orig is not capable of dealing with dynamic model errors and unmodelled frictions which can be considered as time-varying disturbances. This is consistent with the results in simulations, *i.e.*, ESO.orig is capable of tracking constant disturbances but not time-varying disturbances.

In summary, for a physical robot system, only implementing a controller may not be able to achieve good trajectory-tracking performance due to the inevitable dynamic model errors and unmodelled/inaccurate joint frictions. The dynamic model errors and unmodelled/inaccurate joint frictions may result in a shrunk trajectory, while an external constant payload attached to the robot EE may cause the actual trajectory to shift downward. When a suitable disturbance observer is implemented together with the controller, a satisfying trajectory-tracking performance can be achieved due to the fact that the lumped uncertainties can be effectively estimated and compensated by the observer.

5. Discussions and Conclusions

In this paper, we reviewed five typical types of disturbance observers, including GMO, JVOB, NDOB, DKF, ESO (ESO.orig, ESO.modi, ESO.impr). We presented the simulation results on disturbance tracking performances of each observer in three scenarios based on the types of disturbance, *i.e.*, a constant payload, a time-varying sinusoidal payload, and harmonic disturbance with Gaussian noise.

For the DKF method implemented in the simulations, the disturbance model is constructed as a polynomial disturbance (27) for both the sinusoidal disturbance scenario and the harmonic disturbance scenario. This is acceptable when the frequency of the disturbance is relatively low, for example, in this paper, the sinusoidal disturbance is [0.5, 0.2, 0.5] Hz, and the harmonic disturbance is [0.1, 0.1, 0.1] Hz. For a high-frequency harmonic disturbance, the disturbance model of harmonic type (29) should be employed for better disturbance tracking performance. According to the bandwidth simulation results for the 1st-order polynomial disturbance from [11], the frequency response revealed low-pass property,

¹online video: <https://drive.google.com/file/d/1qjUwrXcCnk31SYs5ZVuN8pclig7OBXj1/view?usp=sharing>

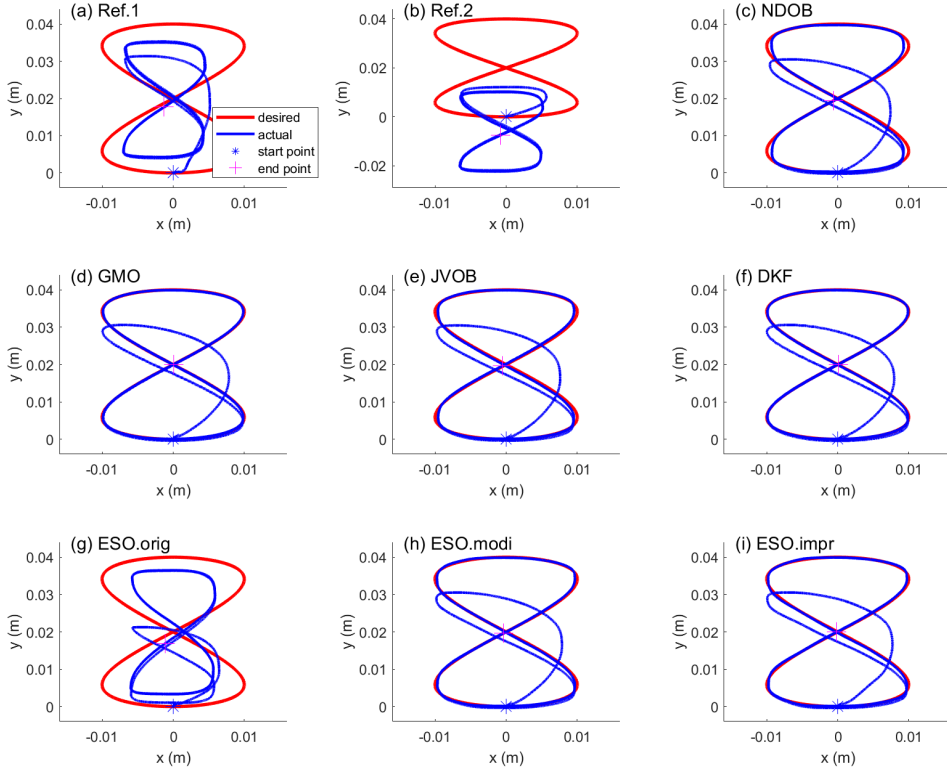


Figure 23: Physical experiments and results when implementing different observers.

and the attenuation rate was small when the frequency was in the range of $[1, 100]$ rad/s, while the signal quickly decayed when the frequency was greater than 100 rad/s.

Although there is no significance revealed in Table 5, the DKF has a relatively strong ability to deal with measurement noise than other presented observers since the DKF has a special feature for explicitly dealing with it. It should be also noted that the sampling rate of the robot control system in our simulations is designed as ideally 1000 Hz since we do not want the observer performance to be affected by this factor. Indeed, a lower sampling rate will make the disturbance tracking performance worse due to the fact that the disturbance cannot be compensated in time. Further study is needed to explore the relationship between sampling rate and disturbance tracking performance. Considering that this paper is focusing on the implementations of the observers, detailed theoretical analyses of the observers' design are ignored, and readers who are of interest can refer to the corresponding reference of each observer which can be found in Section 2.

In Table 6, some main features of each observer are summarized including whether the observer works for constant/sinusoidal disturbance, a subjective score on the complexity of implementation, and whether the acceleration (\ddot{q}) and inverse of inertia matrix (M^{-1}) are needed.

Table 6 shows that all the observers work for tracking constant disturbance. Only NDOB, GMO, JVOB, and ESO.impr work for tracking time-varying sinusoidal disturbances, while ESO.orig does not. For constant disturbances, the disturbance tracking error of observers can asymptotically converge to zero with different convergence speeds (see Fig. 17 for an example). For time-varying disturbances, the disturbance tracking error may not be able to converge to zero thus a bounded error may remain (see Fig. 19 for an example). Intuitively, this is because of that the observers need some time to converge to the target disturbance while some observers may need less time and some may need longer. If the

Table 6: A summary of the main features of observers.

	Observer	Constant	Sinusoidal	Complexity	Acceleration (\ddot{q})	Inertia	Reference
1	NDOB	✓	✓	**	×	M^{-1}	[13]
2	GMO	✓	✓	*	×	M	[29]
3	JVOB	✓	✓	*	×	M^{-1}	[8]
4	DKF	✓	depend.1	*****	×	M	[11, 4]
5.1	ESO.orig	✓	NA	****	×	M^{-1}	[19, 21]
5.2	ESO.modi	✓	depend.2	***	×	M	[21]
5.3	ESO.impr	✓	✓	***	×	M^{-1}	[12]

Note: **Constant**, means constant disturbances; **Sinusoidal**, means sinusoidal disturbances; **Complexity**, means the subjective complexity of implementing the observers, and the more asterisk (*) means the more complexity of the implementation. **Inertia**, means the form of the inertia matrix required.

symbol (✓), means "yes, applicable"; NA, not applicable; symbol (×), means "no need";

depend.1, means that it depends on what dynamics model of the disturbances is used. If a constant disturbance model is used, then it is only valid for constant disturbance, *e.g.*, a constant payload at the end-effector. If a polynomial disturbance model is used, then it is valid for both constant and time-varying disturbance.

depend.2, means that it depends on parameter tuning. The same set of parameters cannot be valid for both constant and time-varying disturbances at the same time. The parameters need to be tuned separately in different disturbance scenarios in order for optimal effect in each.

target disturbance varies with respect to time, it may cause the observer not able to converge to zero in time and thus a bounded error. Theoretically, this is described in [13] when the NDOB was developed, *i.e.*, for slow-varying disturbances, the disturbance tracking error of NDOB can converge asymptotically or exponentially to zero, while for fast-varying disturbances, the disturbance tracking error is globally uniformly ultimately bounded.

For using DKF to track the sinusoidal disturbances, it will depend on what disturbance model is used by the user. As introduced earlier, a disturbance model needs to be constructed by the user in the DKF method. If a constant disturbance model is used, then the DKF will only be valid for constant disturbances. If a polynomial disturbance model is used, then the DKF will be valid for both constant and sinusoidal disturbances. This is also reflected in the corresponding simulation results. For ESO.modi, it will depend on the parameter tuning result. For constant and sinusoidal disturbances, the observer gains may need to be tuned separately in each scenario in order to achieve optimal performance in each. Note that this is not necessary for NDOB, GMO, JVOB, and ESO.impr, for whom one set of observer gains work well for both constant and sinusoidal disturbances.

The complexity of implementing the observers are also subjectively ranked and presented in Table 6. Subjectively, GMO and JVOB are the easiest observers to be implemented, while DKF is the most complicated one. This can also be seen from the complex equations of DKF introduced earlier. It should note that the disturbance tracking performance of DKF is excellent although the complexity of implementation. Also, for ESO.orig and ESO.impr, there are three observer gains, which make them difficult to be tuned in order to find the optimal gain for best disturbance tracking performance.

There is no need for accelerations for all the presented observers, but some of them may need the inverse of the robot inertia matrix, including NDOB, JVOB, ESO.orig, and ESO.impr. For the specific NDOB design (7), the robot inertia matrix can be designed to be a positive definite diagonal matrix, for sure, which will introduce dynamic model error.

This paper is devoted to introducing all the required basics for implementing the five typical types of disturbance observers and presenting the possible behavior of each observer in the same simulated scenario for estimating the uncertainties in a robot control system. It is possible for the readers to quickly implement the introduced observers into a robot system relying on the information provided in this paper. Also, the presented simulation results can help to get an intuitive sense on the possible disturbance-tracking behavior of each observer in the same simulated scenario. At last, this study aims to help the users to better select the most suitable one for their own application scenario. Each observer has its

own unique features, thus has its own advantages in different application scenarios. It would be more meaningful to present the disturbance-tracking behaviors of different observers in the same simulated scenario, rather than to decide which one is superior to another, even though some may have better performance than others in a specific scenario.

Author Contributions. TL and HX conceived and designed the study. TL conducted data gathering and analyses. TL, HX, EH, HT, and MT wrote the article.

Financial Support. This research is supported in part by the Canada Foundation for Innovation (CFI) under grants LOF 28241 and JELF 35916, in part by the Government of Alberta under grants IAE RCP-12-021 and EDT RCP-17-019-SEG, in part by the Government of Alberta's grant to Centre for Autonomous Systems in Strengthening Future Communities (RCP-19-001-MIF), in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada under grants RTI-2018-00681, RGPIN-2019-04662, and RGPAS-2019-00106.

Conflicts of Interest. The authors declare no conflicts of interest exist.

Ethical Approval. Not applicable.

References

- [1] C. Gaz, M. Cagnetti, A. Oliva, P. R. Giordano, and A. De Luca, "Dynamic identification of the Franka Emika Panda robot with retrieval of feasible parameters using penalty-based optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, 2019.
- [2] V. Panwar and N. Sukavanam, "Design of optimal hybrid position/force controller for a robot manipulator using neural networks," *Mathematical Problems in Engineering*, vol. 2007, 2007.
- [3] H.-R. Wang, L. Yang, and L.-X. Wei, "Fuzzy-neuro position/force control for robotic manipulators with uncertainties," *Soft Computing*, vol. 11, pp. 311–315, 2007.
- [4] S. Liu, L. Wang, and X. V. Wang, "Sensorless force estimation for industrial robots using disturbance observer and neural learning of friction approximation," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102168, 2021.
- [5] M. Sharifi, J. K. Mehr, V. K. Mushahwar, and M. Tavakoli, "Adaptive CPG-based gait planning with learning-based torque estimation and control for exoskeletons," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8261–8268, 2021.
- [6] A. Radke and Z. Gao, "A survey of state and disturbance observers for practitioners," in *2006 American Control Conference*. IEEE, 2006, pp. 6–pp.
- [7] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-observer-based control and related methods—an overview," *IEEE Transactions on industrial electronics*, vol. 63, no. 2, pp. 1083–1095, 2016.
- [8] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.
- [9] A. Mohammadi, H. J. Marquez, and M. Tavakoli, "Nonlinear disturbance observers: Design and applications to Euler-Lagrange systems," *IEEE Control Systems Magazine*, vol. 37, no. 4, pp. 50–72, 2017.
- [10] S.-H. Chien, J.-H. Wang, and M.-Y. Cheng, "Performance comparisons of different observer-based force-sensorless approaches for impedance control of collaborative robot manipulators," in *2020 IEEE conference on industrial cyberphysical systems (ICPS)*, vol. 1. IEEE, 2020, pp. 326–331.
- [11] J. Hu and R. Xiong, "Contact force estimation for robot manipulator using semiparametric model and disturbance Kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3365–3375, 2018.

- [12] G. Sebastian, Z. Li, V. Crocher, D. Kremers, Y. Tan, and D. Oetomo, "Interaction force estimation using extended state observers: An application to impedance-based assistive and rehabilitation robotics," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1156–1161, 2019.
- [13] A. Mohammadi, M. Tavakoli, H. J. Marquez, and F. Hashemzadeh, "Nonlinear disturbance observer design for robotic manipulators," *Control Engineering Practice*, vol. 21, no. 3, pp. 253–267, 2013.
- [14] A. De Luca and R. Mattone, "Actuator failure detection and isolation using generalized momenta," in *2003 IEEE international conference on robotics and automation (cat. No. 03CH37422)*, vol. 1. IEEE, 2003, pp. 634–639.
- [15] —, "Sensorless robot collision detection and hybrid force/motion control," in *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE, 2005, pp. 999–1004.
- [16] T. Zhang and X. Liang, "Disturbance observer-based robot end constant contact force-tracking control," *Complexity*, vol. 2019, 2019.
- [17] G. Garofalo, N. Mansfeld, J. Jankowski, and C. Ott, "Sliding mode momentum observers for estimation of external torques and joint acceleration," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6117–6123.
- [18] J. Han, "A class of extended state observers for uncertain systems," *Control and decision*, vol. 10, no. 1, pp. 85–88, 1995, [In Chinese].
- [19] —, "From PID to active disturbance rejection control," *IEEE transactions on Industrial Electronics*, vol. 56, no. 3, pp. 900–906, 2009.
- [20] Y. Huang and W. Xue, "Active disturbance rejection control: Methodology and theoretical analysis," *ISA transactions*, vol. 53, no. 4, pp. 963–976, 2014.
- [21] T. Ren, Y. Dong, D. Wu, and K. Chen, "Collision detection and identification for robot manipulators based on extended state observer," *Control Engineering Practice*, vol. 79, pp. 144–153, 2018.
- [22] W.-H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O'Reilly, "A nonlinear disturbance observer for robotic manipulators," *IEEE Transactions on industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.
- [23] S. Liu, L. Wang, and X. V. Wang, "Sensorless force estimation for industrial robots using disturbance observer and neural learning of friction approximation," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102168, 2021.
- [24] T. Li, A. Badre, H. D. Taghirad, and M. Tavakoli, "Integrating impedance control and nonlinear disturbance observer for robot-assisted arthroscope control in elbow arthroscopic surgery," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 172–11 179.
- [25] J. Fong, H. Rouhani, and M. Tavakoli, "A therapist-taught robotic system for assistance during gait therapy targeting foot drop," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 407–413, 2019.
- [26] S. Bruno, S. Lorenzo, V. Luigi, and O. Giuseppe, "Robotics: modelling, planning and control," 2010.
- [27] P. Song, Y. Yu, and X. Zhang, "A tutorial survey and comparison of impedance control on robotic manipulation," *Robotica*, vol. 37, no. 5, pp. 801–836, 2019.
- [28] A. Torabi, M. Khadem, K. Zareinia, G. R. Sutherland, and M. Tavakoli, "Application of a redundant haptic interface in enhancing soft-tissue stiffness discrimination," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1037–1044, 2019.
- [29] L. Ding, H. Xing, H. Gao, A. Torabi, W. Li, and M. Tavakoli, "VDC-based admittance control of multi-DOF manipulators considering joint flexibility via hierarchical control framework," *Control Engineering Practice*, vol. 124, p. 105186, 2022.

- [30] L. Roveda and D. Piga, "Sensorless environment stiffness and interaction force estimation for impedance control tuning in robotized interaction tasks," *Autonomous Robots*, vol. 45, pp. 371–388, 2021.
- [31] L. Roveda, D. Riva, G. Bucca, and D. Piga, "External joint torques estimation for a position-controlled manipulator employing an extended kalman filter," in *2021 18th International Conference on Ubiquitous Robots (UR)*. IEEE, 2021, pp. 101–107.
- [32] E. Hashemi, A. Khajepour, N. Moshchuk, and S.-K. Chen, "Real-time road bank estimation with disturbance observers for vehicle control systems," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 1, pp. 443–450, 2022.
- [33] Q. Zheng, L. Q. Gao, and Z. Gao, "On estimation of plant dynamics and disturbance from input-output data in real time," in *2007 IEEE International Conference on Control Applications*. IEEE, 2007, pp. 1167–1172.
- [34] M. C. Çavuşoğlu, D. Feygin, and F. Tendick, "A critical study of the mechanical and electrical properties of the phantom haptic interface and improvements for highperformance control," *Presence: Teleoperators & Virtual Environments*, vol. 11, no. 6, pp. 555–568, 2002.