Implementation of a robust data-driven control approach for an omni-directional mobile manipulator based on koopman operator

Xuehong Zhu\textsuperscript{1}, Chengjun Ding\textsuperscript{1}, Chao Ren\textsuperscript{2}, Tong Zhang\textsuperscript{2}, Lizhen Jia\textsuperscript{3} and Lirong Wu\textsuperscript{1}

Abstract
The dynamic modeling and control of omni-directional mobile manipulators (OMM) are challenging since they are highly nonlinear, strongly coupled, and multi-input multi-output uncertainty systems. Koopman operator theory can provide an explicit linear dynamic model for OMM, only based on input-output data. However, the derived dynamic model usually has modeling errors and cannot capture external unknown disturbances. In this paper, a robust data-driven control scheme is designed and implemented for an OMM, based on Koopman operator and a disturbance observer. Firstly, a data-driven Koopman model of OMM is constructed, solely using the collected input-output data. Then a disturbance observer is designed to online estimate the inherent modeling error of the Koopman model as well as external disturbances. The controller is designed by combining linear MPC with feedback compensation of the estimated disturbances. Finally, both simulations and experimental tests are conducted to verify the effectiveness and robustness of the proposed control scheme.

Keywords
Omni-directional mobile manipulator, Koopman operator, disturbance observer, data-driven control

Introduction
Mobile manipulators are composed of a mobile platform and at least one manipulator fixed on it. Compared with fixed-base manipulators, mobile manipulators have much larger workspace\textsuperscript{1}. Specially, for omnidirectional mobile manipulators (OMMs), the omnidrectional mobile platform can simultaneously and independently carry out translational and rotational motion. Therefore, OMMs are suitable for applications with confined spaces, such as factories. Meanwhile, OMM is a highly nonlinear, strongly coupled, and multi-input multi-output uncertainty system.

In the literature, trajectory tracking control of OMMs is always a challenging problem. The dynamic analysis and control design were given by Watanabe et al.\textsuperscript{2} for an OMM with three lateral orthogonal wheel assemblies, wherein computed torque control and resolved acceleration control were used. Viet et al.\textsuperscript{3} has proposed a tracking control method for a three-wheeled OMM considering disturbance and friction, in which sliding mode controller was employed. A hybrid control strategy for trajectory-tracking control of wheeled mobile robotic manipulators was proposed by Shafei et al.\textsuperscript{4} in the presence of model uncertainties and external disturbances. A robust neural network-based sliding mode controller was designed by Xu et al.\textsuperscript{5} for trajectory tracking problem for a redundantly actuated OMM in the presence of uncertainties and disturbances. Li et al.\textsuperscript{6} presented a non-singular terminal
sliding mode control method for an OMM based on extended state observer, wherein the nominal dynamic model is used. An experimental mobile manipulator assembled on an omnidirectional platform with a dual-arm torso with a human-like structure was presented by Suárez et al.7 A controller based on constrained optimization was developed for tracking problems of mobile manipulation by Avanzini et al.9 However, for the research works mentioned above, the controller design completely or partially depends on the nominal dynamic model. Note that the dynamic modeling for OMMs is complicated and time-consuming, due to high nonlinearity, strong coupling, and complex uncertainty.

Koopman operator theory offers a way that allows one to transform a nonlinear dynamical system into an infinite-dimensional linear system, solely using input-output data of the dynamical system.9,10 In other words, the Koopman methodology is capable of discovering the underlying dynamics and control of a given system from data alone. More specifically, the basic idea is to lift of the state-space to a higher dimensional space, where the evolution of this lifted state is approximately linear. This makes it an attractive data-driven control architecture for OMM control design. Koopman operator theory has been applied in many fields, for example, fluid analysis,11 robotics,12–13 power systems,15 etc.

Since the Koopman operator is an infinite-dimensional operator, it makes practical use prohibitive. In recent works, dynamic mode decomposition (DMD)16 and extended dynamic mode decomposition (EDMD)17 are popular methods to obtain finite-dimensional linear models. This linear representation allows one to control the nonlinear system using tools from linear control methods, for example, linear model predictive control (MPC), Linear Quadratic Regulator (LQR). However, three main aspects influence the modeling accuracy of the Koopman operator based linear representation (Koopman model), that is, dimensionality, basis functions and data quality. Unless a finite-dimensional invariant subspace exists,14 the Koopman operator is infinite-dimensional. Thus, finite-dimensional approximation usually introduces modeling error. Moreover, there has not been a systematic way of choosing the Koopman basis functions for general nonlinear systems. Lastly, the quality of the collected data significantly influences the modeling error. Also, the Koopman model cannot capture unknown external disturbances during system operation.

Disturbance observer is one of the most popular robust control tools due to its simplicity, flexibility, and efficacy.18 The basic idea is to estimate both internal (i.e. modeling errors) and external disturbances, and to compensate them in the control signals. Several linear and nonlinear observers have been independently developed in the literature, such as perturbation observer,19 equivalent input disturbance (EID) estimator,20 uncertainty and disturbance estimator (UDE),21 generalized proportional integral observer (GPIO),22 extended state observer (ESO).23 Disturbance observers have been widely applied in various fields, such as robotics,24–26 to name a few.

In this paper, a robust data-driven control scheme is designed and implemented for an OMM, based on Koopman operator and disturbance observer. Firstly, a data-driven Koopman model of OMM was constructed, solely using the collected input-output data. Then a disturbance observer is designed to online estimate the inherent modeling error of the Koopman model as well as external disturbances. The controller is designed by combining linear MPC with feedback compensation of the estimated total disturbances. Finally, both simulations and experimental tests are conducted to verify the effectiveness and robustness of the proposed control scheme.

Finite-dimensional Koopman representations cannot well capture nonlinear effects inherent in many robotic systems. The main contribution of this work is that a robust data-driven MPC scheme is proposed by combining Koopman operator and disturbance observer, without using any model information. Extensive simulations and experimental tests are carried out to verify the effectiveness and robustness of the proposed control scheme, against typical Koopman operator based MPC.

This paper is organized as follows. Firstly, Koopman operator-based dynamic modeling is presented. Then, the proposed control scheme is developed. Next, the simulation and experiment are introduced respectively. Finally, the conclusion is drawn.

**Koopman model**

**Koopman operator theory**

Consider a discrete uncontrolled dynamical system:

\[ x(k + 1) = f(x(k)) \]  \hspace{1cm} (1)

where, \( x(k) \in \mathcal{M} \) is the \( n \)-dimensional system state on a smooth manifold \( \mathcal{M} \), and \( \mathcal{M} \subset \mathbb{R}^n \) is the state space. \( k \) is the discrete time step. \( f \) is the function of state evolution in state space.

The Koopman operator theory provides a method to transform a nonlinear dynamic system into an infinite-dimensional linear system. The Koopman operator \( \mathcal{K} : \mathcal{F} \rightarrow \mathcal{F} \) is an infinite-dimensional linear operator, which advances the evolution of observables \( \Psi \in \mathcal{F} \). \( \mathcal{F} \) is an invariant function space (usually called observable space) under the action of Koopman operator. Thus,

\[ \mathcal{K}\Psi(x(k)) \overset{\Delta}{=} \Psi(f(x(k))) \]  \hspace{1cm} (2)

where \( \Psi(x) = [\psi_1(x), \psi_2(x), \ldots, \psi_N(x)]^T \), \( \psi_i : \mathcal{M} \rightarrow \mathbb{R} \), \( i = 1 \ldots N \) are user-specified (typically nonlinear) lifting functions. It is important that the operator completely captures all the properties of the underlying
A dynamical system, as long as the space of the observation \( \mathcal{F} \) contains the components of the state \( x(k) \).

Combining equations (1) and (2), it has

\[
\mathcal{K}\Psi(x(k)) \overset{\Delta}{=} \Psi(x(k + 1))
\]

In this way, Koopman operator represents a nonlinear system by lifting the state-space to an infinite-dimensional space of observables. Note that (1) is a uncontrolled system, several works have been proposed to extend the Koopman operator to controlled systems.\(^{27-30}\)

### Data-driven finite-dimensional approximation to Koopman operator

Since the Koopman operator is infinite-dimension, obtaining finite-dimensional approximations are necessary in practical applications. Extended dynamic mode decomposition (EDMD) is an effective method to obtain the finite-dimensional approximation of the Koopman operator. Specially, EDMD is extended to the controlled system to construct a matrix \( \mathcal{K} \), as the finite-dimensional approximation of the Koopman operator \( \mathcal{K} \).\(^{30}\) In the following, we directly give the results from Milan Korda.

Consider a discrete-time nonlinear controlled dynamical system:

\[
x(k + 1) = f(x(k), u(k))
\]

where, \( u(k) \in \mathbb{R}^m \) is the control input.

For simplicity, one can choose \( \Psi(x(k), u(k)) = [\Psi_1(x(k)), \Psi_2(u(k))] \). Since there is no need to predict the future control input, choose \( \Psi_2(u(k)) = u(k) \). The lifted states are defined as:

\[
z(k) = \Psi_z(x(k)) = [\psi_{z1}(x(k)), ..., \psi_{zN}(x(k))]^T
\]

Then the constructed Koopman model is in the following discrete linear form as:

\[
z(k + 1) = Az(k) + Bu(k)\]

\[
\hat{x}(k) = Cz(k)
\]

where \( z(k) \in \mathbb{R}^N \) is the lifted state, \( N \) is the lifted dimensionality, \( \hat{x}(k) \) is an estimate of the original nonlinear system state \( x(k) \), and \( A \in \mathbb{R}^{N \times N}, B \in \mathbb{R}^{N \times m}, C \in \mathbb{R}^{m \times N} \) are linear time-invariant matrices.

The data set formed by the system states and control input of the original nonlinear system is defined as follows:

\[
X = [x(1) \cdots x(P)]
\]

\[
Y = [x(2) \cdots x(P + 1)] ,
\]

\[
U = [u(1) \cdots u(P)]
\]

which satisfy \( x(k + 1) = f(x(k), u(k)) \). To obtain the matrices \( A, B, C \), the collected data are lifted as:

\[
X_{lift} = [\Psi_z(x(1)), ..., \Psi_z(x(P))]\]

\[
Y_{lift} = [\Psi_z(x(2)), ..., \Psi_z(x(P))]\]

The numerical solutions of the matrices \( A, B, C \) are obtained by substituting the lifted data into (7) and (8):

\[
[A, B] = Y_{lift}[X_{lift}, U]^\dagger
\]

\[
C = XX_{lift}^\dagger
\]

So far, the data-driven Koopman model of the original nonlinear system (4) is obtained as (6) and (12).

### Building a linear model based on Koopman operator

The OMM prototype consists of a three-wheeled omni-directional mobile platform and a parallelogram manipulator. The structure of the OMM is shown in Figure 1. Define the state vector of OMM at time as \( x(k) = [q(k)^T, \dot{q}(k)^T]^T \in \mathbb{R}^{10} \). \( q(k) = [x(k), y(k), \theta_1(k), \theta_2(k)]^T \in \mathbb{R}^5 \) is the position and posture of OMM, where \( x(k), y(k) \) and \( \theta_1(k), \theta_2(k) \) represents the \( x \)-coordinate, \( y \)-coordinate, and orientation angle of the mobile platform in the world coordinate system, respectively, while \( \theta_1(k) \) and \( \theta_2(k) \) are the two joint angles of the manipulator. \( \dot{q}(k) = \frac{q(k) - q(k - 1)}{\Delta t} \in \mathbb{R}^5 \) is the
velocity state of the OMM, Δt is the sampling time. \( \mathbf{u}(k) = [u_1(k), u_2(k), u_3(k), u_4(k), u_5(k)]^T \in \mathbb{R}^5 \) is the control input vector consisting of the voltages of five motors.

In real applications, when the control input sequence \( \mathbf{u}(k) \) is applied to the OMM, the state of OMM \( x(k) \) can then be obtained. By collecting the input/output data, the Koopman model for the OMM can be obtained in the form of (6).

**Control system design**

As mentioned before, inherent modeling errors are between the Koopman model and the robot, including the unknown external disturbances. In this section, a robust data-driven MPC framework is proposed for the OMM, wherein a disturbance observer is employed to estimate the modeling errors of the Koopman model.

**Design of disturbance observer**

The Koopman model for OMM with uncertainty can be written as:

\[
\begin{align*}
\mathbf{z}(k + 1) &= A\mathbf{z}(k) + B\mathbf{u}(k) + \mathbf{w}(k) \\
\mathbf{y}(k) &= C\mathbf{z}(k)
\end{align*}
\]

where \( A \) and \( B \) are the matrix solved by equation (12), \( \mathbf{w} \in \mathbb{R}^{5x1} \) is the modeling errors between the Koopman model and the robot. \( C \) is the first five rows of \( \mathbf{C} \). Actually, many disturbance observers in the literature can be selected to estimate the modeling errors. In this paper, the disturbance observer proposed by Yang et al.\(^{31}\) is employed, and the discrete form is designed as:

\[
\begin{align*}
\mathbf{b}(k) &= \tilde{\mathbf{w}}(k) - \mathbf{p}(k) \\
\mathbf{b}(k + 1) &= (I_{5x5} - T_s\mathbf{L})\mathbf{b}(k) + T_s\mathbf{L}(\mathbf{p}(k) - \mathbf{C}B\mathbf{u}(k))
\end{align*}
\]

where \( T_s \) is the sampling period, \( \tilde{\mathbf{w}}(k) \) is the estimated value of the disturbance \( \mathbf{w}(k) \), \( \mathbf{b}(k) \) is the state variable of the disturbance observer, \( \mathbf{L} \) is the gain of the disturbance observer, \( \mathbf{p}(k) \) is the function to be designed related to the system states and is simply set as \( \mathbf{p}(k) = \rho \mathbf{y}(k) \), \( \rho \) is the coefficient vector. Actually, the selection of disturbance observer is not limited to the proposed by Yang et al.,\(^{31}\) and many disturbance observers can be selected in the proposed control scheme, such as UDE,\(^{21}\) GPIO,\(^{22}\) and ESO.\(^{23}\)

**Controller design**

The designed controller \( \mathbf{u}(k) \) includes two parts:

\[
\mathbf{u}(k) = \mathbf{u}_d(k) + \mathbf{u}_e(k)
\]

where \( \mathbf{u}_d(k) \) represents the control efforts for disturbance compensation, and \( \mathbf{u}_e(k) \) denotes the control efforts for the trajectory tracking control. \( \mathbf{u}_d(k) \) is designed as:

\[
\mathbf{u}_d(k) = \mathbf{u}_d(k) = -(\mathbf{C}B)^{-1}\tilde{\mathbf{w}}(k)
\]

Combining (13) and (16), the system is reduced as:

\[
\begin{align*}
\mathbf{z}(k + 1) &= A\mathbf{z}(k) + B\mathbf{u}(k) \\
\mathbf{y}(k) &= C\mathbf{z}(k)
\end{align*}
\]

where \( \mathbf{u}(k) \) can be designed under MPC framework.

The prediction equation of the future states can be described as:

\[
\begin{align*}
\mathbf{z}(k + 1|k) &= A\mathbf{z}(k) + B\mathbf{u}(k) \\
\mathbf{z}(k + 2|k) &= A\mathbf{z}(k + 1|k) + B\mathbf{u}(k + 1|k) \\
&\vdots \\
\mathbf{z}(k + N_p|k) &= A\mathbf{z}(k + N_p - 1|k) + B\mathbf{u}(k + N_p - 1|k)
\end{align*}
\]

The equation (17) can be written compactly as follows:

\[
\mathbf{Y}(k) = \mathbf{F}\mathbf{z}(k) + \mathbf{U}_t(k)
\]

wherein:

\[
\begin{align*}
\mathbf{Y}(k) &= \begin{bmatrix} \mathbf{y}(k + 1|k)^T, \ldots, \mathbf{y}(k + N_p|k)^T \end{bmatrix}^T \\
\mathbf{U}_t(k) &= \begin{bmatrix} \mathbf{u}_t(k)^T, \ldots, \mathbf{u}_t(k + N_t - 1|k)^T \end{bmatrix}^T
\end{align*}
\]

\[
\mathbf{F} = \begin{bmatrix} \mathbf{C}A, \mathbf{C}A^2, \ldots, \mathbf{C}A^{N_p} \end{bmatrix}^T
\]

\[
\mathbf{\Phi} = \begin{bmatrix} \mathbf{CB} & 0 & \ldots & 0 \\
\mathbf{CAB} & \mathbf{CB} & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{C}A^{N_p - 1}B & \mathbf{C}A^{N_p - 2}B & \ldots & \mathbf{C}A^{N_p - N_t}B \end{bmatrix}
\]

The reference trajectory in the prediction domain is: \( \mathbf{Y}_r(k) = \begin{bmatrix} \mathbf{y}_r(k + 1)^T, \ldots, \mathbf{y}_r(k + N_p)^T \end{bmatrix}^T \). The goal of controller design is to minimize the following performance index functions:

\[
J = (\mathbf{Y}(k) - \mathbf{Y}_r(k))^T\mathbf{Q}(\mathbf{Y}(k) - \mathbf{Y}_r(k)) + \mathbf{U}_t(k)^T\mathbf{R}\mathbf{U}_t(k)
\]

where \( \mathbf{Q} \) and \( \mathbf{R} \) are positive diagonal matrices. The above formula can be transformed as follows:
\[ J = (Fz(k) - Y_r(k))^TQ(Fz(k) - Y_r(k)) \]
\[ + 2U_0(k) \Phi^TQ(Fz(k) - Y_r(k)) \]
\[ + U_0(k)^T(\Phi^TQ\Phi)U_0(k) \]  

(25)

Since the first term on the right side of the above formula has no relation with \(u_i(k)\), thus:

\[
\min_{u_i} J \Rightarrow \min_{u_i} \hat{J} \\
\text{s.t.} z(i + 1) = A z(k) + B u_i(k) \\
\quad z = \Psi(x) \\
\quad |u| \leq u_{\max} \\
\hat{J} = 2U_0(k)^TH + U_0(k)^TEU_0(k) 
\]

where

\[
H = \Phi^TQ(Fz(k) - Y_r(k)), E = \Phi^TQ\Phi + R \\
Q = \text{diag} \quad Q_{a1}, Q_{a2}, \ldots, Q_d, Q_{N_c} \\
R = \text{diag} \quad R_c, R_c, \ldots, R_c 
\]

Finally, the designed controller can be obtained as:

\[
u(k) = -(CB)^{-1}\dot{\hat{w}}(k) + u_i(k) \]  

(27)

where, \(u_i(k)\) is solved by (26).

**Remark 1:** In the literature, many disturbance observers have been proposed in the past 30 years. Actually, the estimation performance of all disturbance observers depends on the modeling accuracy of the plant. In other words, if the plant model is accurate, a disturbance observer can well estimate the disturbances. In our manuscript, the internal modeling errors of Koopman model (e.g. parameter uncertainties) and external unknown disturbances (such as payload changes) are viewed as lumped disturbances. Therefore, the estimation performance of modeling errors for a disturbance observer depends on the model accuracy of the obtained Koopman model.

**Simulation**

In this section, effectiveness of the proposed control scheme is compared with Koopman model based MPC (K-MPC) via Matlab. Firstly, a Koopman model is derived from input/output data. Then two typical reference trajectories are selected to compare the two control schemes.

**Simulation setup**

For data collection in the Koopman modeling, 47 randomly generated trajectories are selected and each trajectory runs for 5s, that is, each trajectory generates 500 data points (sampling time is 0.01s). Thus, the total number of data pairs is 23,500. The dimension of the lifting space is set as 50, wherein the first 10 dimensions are the position and speed of the robot. The control signals for the mobile platform are randomly generated and uniformly distributed motor voltages between \(-20V\) and \(20V\). The control signals for the manipulator are randomly generated and uniformly distributed motor voltages between \(-8V\) and \(8V\). The basis functions are selected as \(\phi_i(x) = \|x - x_c\|^2 \log(\|x - x_c\|)\), and \(x_c\) is a constant vector with the elements uniformly distributed in a given interval. Here, \(x_c = [-0.6848, 0.9412, 1.3715, -0.0439, 0.09008, -0.3581, -0.0782, 0.8315, 1.7532, 2.7570]^T\).

Since both control schemes are designed based on MPC, the control parameters for MPC are set as the same for both control schemes: \(Q_o = 0.1I_{10 \times 10}, Q_{a} = 100I_{10 \times 10}, R_c = 1 \times 10^{-6}I_{5 \times 5}, N_c = N_o = 5\). The parameters for the disturbance observer are set as: \(p(k) = [0.001x(k), 0.001y(k), 0.001\phi(k), 0.011\theta_1(k), 0.012\theta_2(k)]^T\), \(L = \text{diag}(50, 50, 50, 10, 10)\). The sampling period of the control system is 0.01s.

Two typical reference trajectories are selected in the simulations. For the first trajectory, the mobile platform tracks a circle with a radius of 0.8m, and the joint angles of the manipulator respectively track two sinusoidal trajectories. The initial position and posture of the robot were set as \(q_0 = [0.8 m, 0 m, 0 rad, 0 rad, 0 rad]^T\), and the simulation time is 60s. In order to verify the robustness of the proposed control scheme against external disturbances, an external step disturbance \(d_1 = [2000, 2000, 500, 50, 50]^T\) was added within 5–15s and an external time-varying disturbance \(d_2 = [2000\sin(t), 2000\cos(t), 1000\sin(t), 50\sin(t), 50\cos(t)]^T\) was added within 40–50s. The first reference trajectory is as follows:

\[
\begin{align*}
\chi_d[m] &= 0.8 \cos \left( \frac{\pi}{60} \right) \\
\chi_d[m] &= 0.8 \sin \left( \frac{\pi}{60} \right) \\
\phi_d[rad] &= \sin \left( \frac{\pi}{60} \right) \\
\theta_1_d[rad] &= 0.6 \sin \left( \frac{\pi}{60} \right) \\
\theta_2_d[rad] &= 0.6 \sin \left( \frac{\pi}{60} \right)
\end{align*}
\]  

(28)

In the second trajectory, the mobile platform tracks a square trajectory, and the joint angles of the manipulator respectively track two sinusoidal trajectories. The initial position of the robot is as \(q_0 = [0m, 0m, 0rad, 0rad, 0rad]^T\) in the world coordinate system, and the simulation time is 40s. Similarly, an external disturbance \(d_3 = [1000, 1000, 500, 50, 50]^T\) was added within 10–15s and \(d_4 = [1000\sin(t), 1000\cos(t), 1000\sin(t), 50\sin(t), 50\cos(t)]^T\) was added within...
In this section, two experimental scenarios are carried out. In the first experimental scenario, the control scheme is compared with Koopman model based MPC scheme, to verify the effectiveness in terms of compensation of modeling error. To further compare the robustness of the two control schemes, a payload is added at the end of the manipulator in the second experimental scenario.

**Implementation**

Figure 4 shows the robot prototype. The experimental setup is shown in Figure 5. MATLAB is used on a personal computer (Intel(R) Core(TM) i7-4770 CPU@3.40 GHz) to generate the control input signal (motor voltages). The control signal is sent to the control board on the OMM via a WiFi module. The position and posture feedback in this paper is realized by OptiTrack real time tracking system. As shown in Figure 4, four markers are fixed on the top of the OMM mobile platform, and three markers are fixed on the manipulator.

In the experiments, a total of 57 groups of data were collected, in which 50 groups were used to calculate the Koopman model, and the last 7 groups were used to test the approximation accuracy of the Koopman model. Each group contains 200 sampling points and the sampling time is 0.025 s. That is, the final data set contains 11,400 data points. The basis function is the same as that in the simulations. The second reference trajectory used in the simulations was employed in the experimental tests. The control parameters for both control schemes were set as:

\[
\begin{align*}
N_p &= N_c = 15, Q_p = \text{diag}(3, 5, 5, 5) \\
Q_{Ny} &= \text{diag}(265, 300, 320, 480, 480) \\
R_e &= 10^{-5} \text{diag}(8, 8, 8, 5, 5) \\
L &= \text{diag}(1, 1, 1, 0.05, 0.05) \\
p(k) &= 0.001[x(k), y(k), \varphi(k), \theta_1(k), \theta_2(k)]^T
\end{align*}
\]

**Experimental scenarios**

In the first experimental scenario, the proposed control scheme is compared with Koopman model based MPC scheme without any external payload disturbances, to demonstrate the effectiveness in compensating modeling errors. Experimental results are shown in Figure 6.

It is shown that the proposed control scheme can achieve better control performance than Koopman model based MPC scheme, especially for the mobile platform of OMM. Actually, it is found in our simulations and experiments that the Koopman model can well predict the robot states, if the OMM moves without rotational motion. However, when the OMM moves with rotational motion, the Koopman model results in large modeling errors due to couplings in the dynamics. Moreover, the model accuracy of Koopman model for any systems depends on the selection of basis functions, dimension and data distribution. Therefore, the modeling error of the obtained Koopman model

\[
\begin{align*}
\{x_d[m] = 0.1 t \\
y_d[m] = 0 \\
\varphi_d[rad] = 0 \\
(0s \leq t \leq 10s) \\
\theta_1d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
\theta_2d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
x_d[m] = 1 \\
y_d[m] = 0.1(t - 10) \\
\varphi_d[rad] = 0.1(t - 10) \\
(10s \leq t \leq 20s) \\
\theta_1d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
\theta_2d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
\{x_d[m] = 1 - 0.1(t - 20) \\
y_d[m] = 1 \\
\varphi_d[rad] = 1 \\
(20s \leq t \leq 30s) \\
\theta_1d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
\theta_2d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
x_d[m] = 0 \\
y_d[m] = 1 - 0.1(t - 30) \\
\varphi_d[rad] = 1 + 0.1(t - 30) \\
(30s \leq t \leq 40s) \\
\theta_1d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right) \\
\theta_2d[rad] = 0.6 \sin \left(\frac{\pi}{2} t \right)
\end{align*}
\]

In order to quantitatively compare the control performance of the two control algorithms, this paper uses the absolute error integral (IAE) as the evaluation index, which is defined as follows:

\[
\begin{align*}
\text{IAE}_{x_d}[m] &= \int_0^T |x| + |\dot{x}| dt \\
\text{IAE}_{\varphi_d}[rad] &= \int_0^T |\varphi| dt \\
\text{IAE}_{\theta_1d, \theta_2d}[rad] &= \int_0^T |\theta_1| + |\dot{\theta}_2| dt
\end{align*}
\]

**Simulation results**

The simulation results are shown in Figures 2 and 3, respectively. The IAE of simulation results is shown in Tables 1 and 2, respectively. It can be seen that the proposed control scheme can achieve much better performance than Koopman-based MPC, especially when the external disturbances were added. More specifically, the control performance of the proposed control scheme keeps almost unchanged and shows strong robustness against external disturbances. This is because the external disturbances can be online estimated and compensated in the proposed control scheme. Figures 2(d) and 3(d) show the disturbance estimation performance.

**Experiment**

In this section, two experimental scenarios are carried out. In the first experimental scenario, the proposed control scheme is compared with Koopman model based MPC scheme, to verify the effectiveness in terms of compensation of modeling error. To further compare the robustness of the two control schemes, a payload is added at the end of the manipulator in the second experimental scenario.
results from both internal uncertainties and external disturbances.

On the other hand, it is known that the performance of MPC scheme depends on the modeling accuracy. However, the disturbance observer can well estimate the modeling errors and compensate them in the control signal. Therefore, the control performance of the proposed control scheme can be guaranteed. Figure 6(d) show the estimated disturbances. Table 3 shows the IAE of experimental results.

In the second experimental scenario, a payload (2.53 kg) is added at the end of the manipulator, as shown in Figure 7. To reduce the shaking of the manipulator in experiments, the reference angle of $\theta_1$ was set as 0 rad. The experimental results are shown in Figure 8. Since the payload was added at the end of the manipulator, the payload disturbance is mainly introduced to the dynamics of manipulator (i.e. $\theta_1$ and $\theta_2$). As shown in Figure 8(b), distinct performance improvement is achieved by the proposed control scheme, due to the effective compensation of the external payload. Note that the data collection in the Koopman modeling does not include the case with external payload. Therefore, the derived Koopman model cannot capture the payload information and the control performance cannot be guaranteed against external disturbances. Figure 8(d) show the estimated disturbances. Table 4 shows the IAE of experimental results.

The dimension selection about Koopman model in terms of modeling accuracy is unclear, mainly depending on trial-and-error. Actually, it is found in our simulations that the higher dimension does not mean higher modeling accuracy. Moreover, there has not been a systematic way of choosing the Koopman basis functions for a given nonlinear systems. But the basis functions of Koopman operator are quite important in improving the modeling accuracy of Koopman model.

The horizon $P$ in (7) means the number of the collected data pairs. In our experiments, it is found that $P$ should be relative larger. However, the model accuracy
of Koopman model does not improve as the increase of the horizon \( P \). The collected data should be rich enough to include the complex nonlinear dynamics. Therefore, the control input and initial states for the data collection are random signals with uniform distributions in our simulations and experiments.

### Conclusion

In this paper, a robust data-driven control scheme has been designed and implemented for an OMM, based on Koopman operator and a disturbance observer.
Firstly, a high dimensional Koopman model has been constructed for the OMM, solely using the collected input-output data from experiments. Then a disturbance observer has been selected to online estimate the modeling error as well as external disturbances. The controller has been designed by combining linear MPC with feedback compensation of the estimated disturbances. Finally, both simulations and experimental tests demonstrate that the proposed control scheme

| Control algorithm | IAE_{\text{x}} [m] | IAE_{\text{\theta}} [rad] | IAE_{\text{\varphi}} [rad] |
|-------------------|-------------------|-------------------|-------------------|
| K-MPC             | 2.8474            | 0.9888            | 1.5679            |
| Proposed          | 1.5470            | 0.4355            | 1.5664            |

| Figure 5. Experimental setup. |

| Figure 6. Experimental tests without external payload: the proposed control scheme and Koopman model based MPC (K-MPC): (a) tracking performance, (b) tracking errors, (c) control input, and (d) estimated disturbances. |

| Table 3. IAE of experimental tests without payload. |
Figure 7. Experimental setup with payload.

Figure 8. Experimental tests with external payload: the proposed control scheme and Koopman model based MPC (K-MPC): (a) tracking performance, (b) tracking errors, (c) control input, and (d) estimated disturbances.
achieves much better control performance, compared with Koopman model based MPC scheme.

**Declaration of conflicting interests**

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

**Funding**

The author(s) received no financial support for the research, authorship, and/or publication of this article.

**ORCID iDs**

Xuehong Zhu https://orcid.org/0000-0003-4824-1869
Chengjun Ding https://orcid.org/0000-0003-0455-7538
Tong Zhang https://orcid.org/0000-0002-9449-1394
Lizhen Jia https://orcid.org/0000-0002-7184-859X
Lirong Wu https://orcid.org/0000-0002-2194-4559

**References**

1. Fareh R, Saad MR, Saad M, et al. Trajectory tracking and stability analysis for mobile manipulators based on decentralized control. *Robotica* 2019; 37(10): 1732–1749.
2. Watanabe K, Sato K, Izumi K, et al. Analysis and control for an omnidirectional mobile manipulator. *J Intell Robot Syst* 2000; 27(1/2): 3–20.
3. Viet TD, Doan PT, Hung N, et al. Tracking control of a three-wheeled omnidirectional mobile manipulator system with disturbance and friction. *J Mech Sci Technol* 2012; 26(7): 2197–2211.
4. Shafei HR, Bahrami M and Talebi HA. Trajectory tracking of an uncertain wheeled mobile robotic manipulator with a hybrid control approach. *J Braz Soc Mech Sci Eng* 2020; 42(1): 1–26.
5. Xu D, Zhao D, Yi J, et al. Trajectory tracking control of omnidirectional wheeled mobile manipulators: robust neural network-based sliding mode approach. *IEEE Trans Syst Man Cybern Part B Cybern* 2009; 39(3): 788–799.
6. Li C, Ren C, Ding Y, et al. Non-singular terminal sliding mode control of an omnidirectional mobile manipulator based on extended state observer. *Int J Intell Robot Appl* 2021; 5(2): 1–16.
7. Suárez R, Palomo-Avellaned L, Martínez J, et al. Development of a dexterous dual-arm omnidirectional mobile manipulator. *IFAC-PapersOnLine* 2018; 51(22): 126–131.
8. Avanzini GB, Zanchetti AM and Rocco P. Constrained model predictive control for mobile robotic manipulators. *Robotica* 2018; 36(1): 19–38.
9. Koopman BO. Hamiltonian systems and transformation in hilbert space. *Proc Natl Acad Sci USA* 1931; 17(5): 315–318.
10. Mezić I. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dyn* 2005; 41(1-3): 309–325.
11. Mezić I. Analysis of fluid flows via spectral properties of the koopman operator. *Annu Rev Fluid Mech* 2013; 45: 357–378.
12. Abraham I and Murphey TD. Active learning of dynamics for data-driven control using koopman operators. *IEEE Trans Robot* 2019; 35(5): 1071–1083.
13. Mamakoukas G, Castano M, Tan X, et al. Local Koopman operators for data-driven control of robotic systems. In: Bicchi A, Kress-Gazit H and Hutchinsonand S (eds) *Robotics: science and systems*, Freiburg im Breisgau, Germany: MIT Press journals, 2019, pp.54–62.
14. Mamakoukas G, Castano ML, Tan X, et al. Derivative-based Koopman operators for real-time control of robotic systems. *IEEE Trans Robot* 2021; 37: 2173–2192.
15. Korda M, Susuki Y and Mezić I. Power grid transient stabilization using Koopman model predictive control. *IFAC-PapersOnLine* 2018; 51(28): 297–302.
16. H. Tu J, W. Rowley C, M. Luchtenburg D, et al. On dynamic mode decomposition: Theory and applications. *J Comput Dyn* 2014; 1(2): 391–421.
17. Williams MO, Kevcikidis IG and Rowley CW. A data-driven approximation of the koopman operator: Extending dynamic mode decomposition. *J Nonlinear Sci* 2015; 25(6): 1307–1346.
18. Sarifyldiz E, Oboe R and Ohnishi K. Disturbance observer-based robust control and its applications: 35th anniversary overview. *IEEE Trans Ind Electron* 2020; 67(3): 2042–2053.
19. Kwon S and Chung WK. A discrete-time design and analysis of perturbation observer for motion control applications. *IEEE Trans Control Syst Technol* 2003; 11(3): 399–407.
20. She JH, Fang M, Ohyama Y, et al. Improving disturbance-rejection performance based on an equivalent-input-disturbance approach. *IEEE Trans Ind Electron* 2008; 55(1): 380–389.
21. Zhong QC, Kuperman A and Stobart RK. Design of ude-based controllers from their two-degree-of-freedom nature. *Int J Robust Nonlinear Control* 2011; 21(17): 1994–2008.
22. Sira-Ramirez H and Oliver-Salazar MA. On the robust control of buck-converter dc-motor combinations. *IEEE Trans Power Electron* 2013; 28(8): 3912–3922.
23. Han J. From pid to active disturbance rejection control. *IEEE Trans Ind Electron* 2009; 56(3): 900–906.
24. Huang J, Ri S, Fukuda T, et al. A disturbance observer based sliding mode control for a class of underactuated robotic system with mismatched uncertainties. *IEEE Trans Automat Contr* 2019; 64(6): 2480–2487.
25. Sariyildiz E, Sekiguchi H, Nozaki T, et al. A stability analysis for the acceleration-based robust position control of robot manipulators via disturbance observer. *IEEE/ASME Trans Mechatron* 2018; 23(5): 2369–2378.
26. Huang H, He W, Li J, et al. Disturbance observer-based fault-tolerant control for robotic systems with guaranteed prescribed performance. *IEEE Trans Cybern* 2022; 52: 772–783.
27. Brunton SL, Brunton BW, Proctor JL, et al. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLoS One* 2016; 11(2): e0150171.

**Table 4. IAE of experimental tests with payload.**

| Control algorithm | $\text{IAE}_{m}$ [m] | $\text{IAE}_{\theta}$ [rad] | $\text{IAE}_{\theta m}$ [rad] |
|-------------------|----------------------|-----------------------------|-----------------------------|
| K-MPC             | 3.0646               | 1.3011                      | 4.6519                      |
| Proposed          | 1.3576               | 0.5345                      | 1.5388                      |
28. Williams MO, Hemati MS, Dawson STM, et al. Extending data-driven Koopman analysis to actuated systems. *IFAC-PapersOnLine* 2016; 49(18): 704–709.

29. Proctor JL, Bruntonz SL and Kutx JN. Generalizing Koopman theory to allow for inputs and control. *SIAM J Appl Dyn Syst* 2018; 17(1): 909–930.

30. Korda M and Mezić I. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica* 2018; 93: 149–160.

31. Yang J, Li S and Yu X. Sliding-mode control for systems with mismatched uncertainties via a disturbance observer. *IEEE Trans Ind Electron* 2013; 60(1): 160–169.