Koopman operator based model predictive control for trajectory tracking of an omnidirectional mobile manipulator

Xuehong Zhu\textsuperscript{1}, Chengjun Ding\textsuperscript{1,2}, Lizhen Jia\textsuperscript{2,3} and Yubo Feng\textsuperscript{1,2}

Abstract

Omnidirectional mobile manipulators (OMMs) have been widely used due to their high mobility and operating flexibility. However, since OMMs are complex nonlinear systems with uncertainties, the dynamic modeling and control are always challenging problems. Koopman operator theory provides a data-driven modeling method to construct explicit linear dynamic models for the original nonlinear systems, using only input-output data. It then allows to design control system based on well-established model-based linear control methods. This paper designs a Koopman operator based model predictive control (MPC) scheme for trajectory tracking control of an OMM. Firstly, using Koopman operator and extended dynamic mode decomposition method, an approximate high-dimensional linear dynamic explicit expression for the OMM system is obtained. Then MPC is employed to achieve tracking control based on the derived linear Koopman model. Finally, to show modeling accuracy for the OMM, the Koopman model is evaluated via both simulation and experimental tests. The control performances of the Koopman operator based MPC design are also verified in the simulation and experimental results.

Keywords

Koopman operator, model predictive control, omnidirectional mobile manipulator, linear model, tracking control

Date received: 24 March 2021; accepted: 26 February 2022

Introduction

Omnidirectional mobile manipulators (OMMs) consist of an omnidirectional mobile platform and at least one manipulator fixed on the platform. The omnidirectional mobile platform can simultaneously and independently carry out translational and rotational motion. Therefore, OMMs have broad application prospects in many fields.

In the literature, many control approaches have been designed for the trajectory tracking control of OMMs. A decentralized trajectory tracking control method was proposed by Viet et al.\textsuperscript{1} for an OMM considering disturbances and friction. Two controllers were respectively designed to control the mobile platform and manipulator. Fareh et al.\textsuperscript{2} rearranged the dynamic model to take the form of two interconnected subsystems, and the trajectory tracking control law was designed for the two subsystems, respectively. Watanabe et al.\textsuperscript{3} derived a kinematic and dynamical model for a mobile manipulator using the Newton-Euler method. The controller was designed based on computed torque control and resolved acceleration control. A reactive architecture and impedance control was proposed for an OMM by Djebriani et al.\textsuperscript{4} to ensure reliable task execution. Zhang et al.\textsuperscript{5} presented a feedback control scheme based on repetitive motion planning. An experimental mobile manipulator assembled on an omnidirectional platform by a dual-arm torso with a human-like structure was presented by Suárez et al.\textsuperscript{6} Guo et al.\textsuperscript{7} analyzed the kinematic equation parameters that affects the motion accuracy of the omnidirectional system. For teleoperation of a dual-arm omnidirectional mobile robot, Song et al.\textsuperscript{8} presented a shared-control-based design which can perform user-commanded tasks or autonomous operations.

\textsuperscript{1}School of Mechanical Engineering, Hebei University of Technology, Tianjin, China
\textsuperscript{2}School of Transportation Science and Engineering, Civil Aviation University of China, Tianjin, China

Corresponding author:
Chengjun Ding, School of Mechanical Engineering, Hebei University of Technology, No. 5340, Xiping Road, Beichen District, Tianjin 300401, China
Email: dcj@hebut.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
Avanzini\textsuperscript{9} developed a constrained model predictive control for mobile robotic manipulators, which it is assumed that the robot is velocity controlled. However, for control designs mentioned above, the control performances depend on the nonlinear dynamic models, and most control methods cannot deal with constraint problems. The dynamic modeling for OMMs is complicated and time-consuming since OMMs are multi-input and multi-output systems with high nonlinearity, strong coupling and complex uncertainty. The controller design based on complex nonlinear models may lead to poor performances, if the model accuracy cannot be guaranteed.

For complex nonlinear systems, linearization is often considered for the convenience of system analysis. Typical linearization methods include feedback linearization,\textsuperscript{10} Taylor linearization,\textsuperscript{11} piecewise linearization,\textsuperscript{12} and orthogonal function approximation linearization,\textsuperscript{13} etc. However, the limitations of these methods are obvious. Feedback linearization requires the precise mathematical model of the control system. Taylor linearization is only carried out near the working point, and just obtains the approximate expression of the original system. Piecewise linearization requires more information about nonlinear control system model. The orthogonal function approximation linearization method has a large computational burden and the controller design is complex. Koopman operator proposed by B. O. Koopman in 1931 provides a powerful tool for the study of nonlinear systems.\textsuperscript{14} The basic idea is to re-express the state evolution process of the original nonlinear system as a linear process by elevating the state of the system to an infinite dimensional space. In other words, via Koopman operator, the infinite-dimensional linear representation of the original nonlinear system is obtained. Koopman operator theory can be used for global linearization of nonlinear dynamical systems. This data-driven modeling process only requires experimental or simulation data, which is completely different from traditional modeling methods.

However, although the nonlinear systems can be converted into infinite dimensional linear systems, Koopman operator is difficult to be applied in practice due to the complexity of its infinite dimension. Dynamic mode decomposition (DMD) method\textsuperscript{15} and extended dynamic mode decomposition (EDMD) method\textsuperscript{16} analyze the spectrum characteristics of the Koopman operator, to achieve the finite dimensional approximation of the infinite dimensional Koopman operator and get the dominant linear expression of the system model. The finite dimensional linear model of the original system is applicable to the model-based linear controller design approaches. Moreover, the DMD and EDMD are mainly based on least-squares regression algorithm, which are computationally flexible and easy to implement. The emergence and development of these data-driven methods make it possible for the Koopman operator method to be applied to many practical problems, and it also provides a new development direction for the control scheme in the era of big data.

In the literature, the EDMD-based Koopman operator approximation methods have been applied to dynamic control systems. Ian et al.\textsuperscript{17} applied Koopman operator theory to the robot control system. A fully data-driven model was obtained using Koopman operator, and then the optimal control was designed based on this model. Simulation and experimental validations were carried out on the Sphero SPRK robot and an inverted pendulum system. Based on the Koopman dynamics model, a linear quadratic regulator (LQR) feedback control scheme was designed by Mamakoukas et al.\textsuperscript{18} Simulations and experiments verified the effectiveness of the modeling and control of the tail-actuated robotic fish, and the Koopman-based LQR control was experimentally verified for the first time. Bruder et al.\textsuperscript{19} obtained the Koopman high-dimensional linear model of a soft robot firstly based on EDMD method. Then the model was identified using a nonlinear autoregressive model with external input and a feed-forward neural network, which was compared with the Koopman high-dimensional linear model to verify the advantage of Koopman operator in the approximation accuracy of the real model. Model predictive control (MPC) scheme was designed and the trajectory tracking control performances were evaluated in the experiments. The active learning algorithm was introduced into the selection of the Koopman operator data sets by Ia.\textsuperscript{20} By designing an active learning controller, the information carrying rate of Koopman operator was improved. Then a LQR control scheme was designed for a Van der Pol oscillator to verify the control performance of the model. In addition, experiments on a free-falling quadrotor have verified the ability of combining active learning active learning with the Koopman operator to perform a single execution model learning.

In this paper, a Koopman operator based MPC scheme is designed for trajectory tracking control of an OMM. The proposed control method is data-driven and does not need any prior knowledge. And Koopman operator can transform nonlinear MPC into linear MPC, thus non-convex optimization is avoided, which can meet the real-time requirements. Moreover, extensive experiments are carried out to verify the effectiveness of the MPC based on Koopman operator for rigid mobile manipulators. The OMM is composed of a mobile platform with three omnidirectional wheels and a parallelogram manipulator. Firstly, the Koopman operator for dynamics systems is briefly introduced. Then a high-dimensional linear model of the OMM, known as the Koopman model, is obtained using EDMD-based Koopman operator approximation method. This process is purely input-output data-driven and does not require any prior knowledge or dynamic model information of the OMM system. Linear MPC scheme is selected to design a trajectory tracking controllers for the robot, based on the
Koopman model. Finally, simulations and experiments are carried out to validate the modeling accuracy of the derived Koopman model. The control performances of the MPC scheme based on the Koopman model, are also verified via both simulations and experimental tests.

The remainder of this paper is as follows. Firstly, the prior knowledge of Koopman operator and EDMD algorithm is introduced. The Koopman representation for the control systems is presented. Secondly, MPC design is presented based on the Koopman model obtained. Thirdly, simulation results are given. Then, experimental tests are shown to evaluate the Koopman model and verify the effectiveness of the Koopman operator based MPC design. Finally, the conclusions are drawn.

**Prior knowledge**

**Koopman operator for dynamics systems**

The Koopman operator is a powerful tool to simplify the analysis of nonlinear dynamic systems, which can transform a finite-dimensional nonlinear dynamic system into an infinite-dimensional linear system. Consider the following discrete-time nonlinear dynamics system:

\[ \xi(k+1) = f(\xi(k)), \]

where \( \xi(k) \in S \) is the \( n \)-dimensional system state on a smooth manifold \( S \), and \( S \subseteq \mathbb{R}^n \) is the state space. The parameter \( k \) is the discrete time step, and \( \xi(k) \) represents the state of the system at the time step \( k \). \( f \) is the function which maps the state \( \xi(k) \) forward time step \( k \) into the future to \( \xi(k+1) \), that is, the evolution of system states.

Define a real-valued observable function on the state space \( g : S \rightarrow \mathbb{R} \), which is an element of an infinite-dimensional Hilbert space. The Koopman operator acts on the real-valued observation function \( g \), which is an infinite-dimensional linear operator:

\[ (Kg)(\xi(k)) = g(f(\xi(k))), \]

where \( g \) indicates function composition. In addition, the Koopman operator \( K \) acts on the function \( g \) of the state space, not on the state \( \xi(k) \), thus even though the state of the original nonlinear system is finite-dimensional, \( K \) retains the property of infinite dimensions.

It can be obtained that:

\[ (Kg)(\xi(k)) = g(f(\xi(k))) = g(\xi(k+1)). \]

In other words, using the Koopman operator \( K \), an infinite-dimensional linear dynamic system is defined, which represents the evolution of the state observation \( g(\xi(k)) \) to the next time step.

For more detailed information about Koopman operator theory, the reader may refer to the work by Proctor.\(^{21}\)

**Finite-dimensional approximation of the data-driven Koopman operator**

In this subsection, the EDMD method is introduced, which approximates the Koopman operator and the Koopman eigenvalue, eigenfunction, and mode tuples.\(^{16}\)

The eigendecomposition of the linear Koopman operator \( K \) is as follows:

\[ K\psi_j(\xi(k)) = \lambda_j \psi_j(\xi(k)), \quad j = 1, 2, \cdots, \infty, \quad (4) \]

where \( \psi_j(\xi(k)) \) represents the eigenfunctions of the Koopman operator, and \( \lambda_j \) represents the corresponding eigenvalues.

Then the observation function \( g \) can be rewritten as:

\[ g(\xi(k)) = \sum_{j=1}^{\infty} \psi_j(\xi(k))v_j, \quad (5) \]

where the vector coefficients \( v_j \) are called Koopman modes. The Koopman eigenvalues, eigenfunctions and modes triplet are the Koopman tuple. Also, the Koopman eigenfunctions \( \psi_j(\xi(k)) \) are considered as basis observable functions.

Due to the infinite dimension of the Koopman operator, it is challenging to calculate and difficult to implement in practical applications. Therefore, the finite-dimension approximation to the Koopman operator is very necessary. Then a finite subspace approximation to the operator \( K \in \mathbb{C}^{N_k \times N_k} \) is required.

A subset of the basis functions \( \psi(\xi) = \{ \psi_1(\xi), \psi_2(\xi), \cdots, \psi_{N_k}(\xi) \} \) is defined, the span of which is the finite subspace denoted as \( F_p \). Then the operator \( K \) acting on \( \psi(\xi) \) is then represented in discrete time as:

\[ (Kg)(\xi(k)) = K\psi_j(\xi(k))v + r(\xi(k)), \quad (6) \]

where \( v \) represents the weight coefficient and \( r(\xi(k)) \) is the residual term.

Provide a dataset \( D = \{ \xi(k) \}_{k=1}^{P} \). The approximate Koopman operator \( K \) can be obtained using least-squares minimization over the parameters of \( K \):

\[ \min_{K} \frac{1}{2} \sum_{k=1}^{P-1} \| \psi_j(\xi(k+1)) - K\psi_j(\xi(k))v \|^2. \quad (7) \]

where \( P \) is the number of the collected data.
The solution is given by
\[ K = EN^+ , \] (8)
where \( ^+ \) denotes the Moore-Penrose pseudoinverse and
\[ E = \frac{1}{P} \sum_{k=1}^{P-1} \psi(\xi(k + 1))\psi(\xi(k))^T , \] (9)
\[ N = \frac{1}{P} \sum_{k=1}^{P-1} \psi(\xi(k))\psi(\xi(k))^T . \]

Convergence of EDMD to the Koopman operator has been analyzed.\(^2\)

In addition, the Koopman operator theory is also applicable to dynamic systems with control. According to the work by Korda and Mezić,\(^2\) consider the following nonlinear dynamic system with control:
\[ \xi(k + 1) = f(\xi(k), u(k)) , \] (10)
where \( \xi(k) \in \mathbb{R}^n \) is the state of the system, and \( u(k) \in \mathbb{R}^m \) is the control input.

The system state \( \xi(k) \) and the control input \( u(k) \) are merged into an augmented state: \( \chi = [\xi^T, u^T]^T \), \( u \) denotes the control sequences, that is \( (u_i)_{i=0}^\infty \). Then the augmented system can be written as:
\[ \chi(k + 1) = F(\chi(k)) = \begin{bmatrix} f(\xi(k), u(k)) \\ u(k + 1) \end{bmatrix} , \] (11)
where \( u(k) \) denotes the \( k \)th element of the sequence \( u \).

For system (11), Koopman operator is defined as follows:
\[ (Kg)(\chi(k)) = g_0 F(\chi(k)) . \] (12)

For the sake of controller design, a high-dimensional linear dynamic model in state space is built using Koopman operator. The high-dimensional linear model constructed is as follows:
\[ z(k + 1) = Az(k) + Bu(k) \]
\[ \hat{\xi}(k) = Cz(k) , \] (13)
where \( z(k) = \psi(\xi) \in \mathbb{R}^N \) is the state of the high-dimensional system, and \( N \) is the dimension of high-dimensional system. \( \hat{\xi}(k) \in \mathbb{R}^n \) is the estimation of the state \( \xi(k) \).

According to the work by Korda and Mezić,\(^2\) an analytic solution to find \( A, B \) and \( C \) in (13) can be easily obtained.

Provide real state and control input data of the original nonlinear system and define data sets as follows:
\[ \xi = [\xi(1) \cdots \xi(P)] \]
\[ Y = [\tilde{\xi}(2) \cdots \tilde{\xi}(P + 1)] , \] (14)
\[ U = [u(1) \cdots u(P)] \]
where \( \xi(k) \) and \( u(k) \) satisfy the relation shown by (10). The matrices \( A, B, \) and \( C \) in (13) can be obtained as the solutions to the following optimization problems:
\[ \min_{A,B} \| Y_{lift} - A\xi_{lift} - BU \| \]
\[ \min_{C} \| \xi - C\xi_{lift} \| , \] (15)
where \( \xi_{lift} = [\psi(\xi(1)), \ldots, \psi(\xi(P))] \), \( Y_{lift} = [\psi(\xi(2)), \ldots, \psi(\xi(P + 1))] \).

The analytical solutions to (15) are:
\[ [A, B] = Y_{lift}\xi_{lift}, U\] (16)
\[ C = XX_{lift}^+ . \]

Now, a Koopman model of the nonlinear system (10) is obtained by (13) and (16) which can be used to predict the future state and design the controller.

**Remark 1.**\(^2,23\) If the basis observable functions \( \{\psi_1(\xi), \psi_2(\xi), \ldots, \psi_N(\xi)\} \) contain the state itself, that is, \( \psi_i(\xi) = \xi_i \), where \( i = 1, 2, \ldots, n \) and \( N \) is one state of the system, the solution to (16) is \( C = [I_n, 0] \).

**Remark 2.**\(^20,24\) The basis functions of state are required to contain nonlinearity. Appropriate basis functions can effectively improve the estimation accuracy of the finite dimensional Koopman operator obtained by EDMD. Often the basis functions are artificially defined, and this process requires specific knowledge and a lot of attempts. In recent years, the development of neural networks provides a new solution for the discovery of basis functions. A deep learning approach was proposed by Yeung et al.\(^24\) to discover the basis functions from data, which shows that deep neural networks can be used to discover the basis functions actively and obtain an accurate approximation of the Koopman operator.

**Remark 3.** For the OMM control system, there is no positive correlation between the dimension of the lifted state-space \( N \) and the Koopman model accuracy. Therefore, in practice, the calculation cost and model accuracy should be balanced to select the appropriate dimension.

**Remark 4.** For more detailed information about equation (13), the readers can refer to the work by Korda and Mezić.\(^2\)
Model predictive control for the OMM

The model-based control method is the most mature and widely used control method among the many control methods. At present, MPC is the most popular model-based control design technique, which uses the model to predict the future output and optimizes control input within a finite time horizon, considering various constraints. The optimized input is applied to a single time step, and then iterate on the optimization. The predictive control principles of linear and nonlinear systems are similar, that is, model-based prediction, online rolling optimization, and feedback correction. However, the optimization problem in nonlinear MPC suffers from heavy calculation burden due to a nonlinear cost function and nonlinear constraints. Koopman high-dimensional model can transform nonlinear MPC into linear MPC, thus non-convex optimization is avoided. Therefore, the control task of nonlinear dynamic systems can be accomplished by designing a linear MPC scheme based on Koopman model.

For the OMM control system, the original state vector is \( \mathbf{x}(k) = [q(k)^T \ \nu(k)^T]^T \in \mathbb{R}^10 \). \( q(k) = [x(k) \ \ y(k) \ \ \theta_1(k) \ \ \theta_2(k)]^T \in \mathbb{R}^3 \) is the position state of OMM, where \( x(k) \), \( y(k) \), and \( \theta(k) \) are three plane degrees of freedom representing the x-coordinate, y-coordinate, and rotation angle of the OMM respectively, while \( \theta_1(k) \) and \( \theta_2(k) \) are the two joint angles of the manipulator. \( \nu(k) = \sum_{i=1}^{N_c} q(k-i)^T \in \mathbb{R}^3 \) is the velocity state, which adopts the backward Euler method. \( \Delta t \) is the sampling time. Control input \( u(k) = [u_1(k) \ \ u_2(k) \ \ u_3(k) \ \ u_4(k) \ \ u_5(k)]^T \in \mathbb{R}^5 \) is the vector consisting of the input voltages of five motors. The goal of trajectory tracking control of an OMM is to give a desired trajectory, and a controller is designed to enable the OMM to track the desired trajectory in real time under the action of the control input. In other words, under effective control, the real outputs (position and velocity) of the OMM at the moment can be infinitely close to the desired values. Based on the method mentioned above, the Koopman model (KM) of the OMM can be obtained as equation (13). The prediction equation for the future state of the high-dimensional system \( z \) can be written as follows:

\[
\begin{align*}
    z(k+1) &= A z(k) + Bu(k) \\
    z(k+2) &= A z(k+1) + Bu(k+1) \\
    & \vdots \\
    z(k+N_c) &= A z(k + N_c - 1) + Bu(k + N_c - 1) \\
    & \vdots \\
    z(k+N_p) &= A z(k + N_p - 1) + Bu(k + N_p - 1) \\
\end{align*}
\]

(17)

where \( N_c \) is the control horizon, \( N_p \) is the prediction horizon, \( z(k+i|k) \) and \( u(k+i|k) \) represent the predictive state and control input at time \( k \), respectively.

Define \( y(k) \) as the output of the OMM system. For this system, the system output is the position, that is, \( y(k) = q(k) \). Then, according to equation (13), the predicted output is represented as:

\[ \hat{y}(k) = \hat{q}(k) = C z(k). \]  

The prediction equation of the system is rewritten in the compact form as follows:

\[ Y(k) = F z(k) + \Phi U(k), \]

(19)

where

\[
    \begin{align*}
        Y(k) &\triangleq \begin{bmatrix} y(k) & y(k+1) & \ldots & y(k+N_p) \end{bmatrix}^T, \\
        F &\triangleq \begin{bmatrix} A & 0 & \ldots & 0 \\
                         C A & \ddots & \ldots & \ldots \\
                         \vdots & \ddots & \ddots & \ddots \\
                         C A^{N_p-1} & \ldots & \ldots & \ldots & \ldots & C A^{N_p-N_p} \end{bmatrix}, \\
        \Phi &\triangleq \begin{bmatrix} \Phi_{0} & \ldots & 0 \\
                         \Phi_{1} & \ldots & 0 \\
                         \vdots & \ddots & \ddots \\
                         \vdots & \ddots & \ddots \end{bmatrix}.
    \end{align*}
\]

The objective of MPC is to minimize the following cost function:

\[ J = (Y(k) - Y_d(k))^T Q (Y(k) - Y_d(k)) + U(k)^T R U(k), \]

(21)

where \( Q \) and \( R \) are positive definite matrices, and

\[
    \begin{align*}
        J &= (F z(k) + \Phi U(k) - Y_d(k))^T Q \\
        &\quad + (F z(k) + \Phi U(k) - Y_d(k))^T \Phi (F z(k) - Y_d(k)) \\
        &\quad + 2 U(k)^T \Phi^T Q U(k) \\
        &\quad + U(k)^T (R + \Phi^T Q \Phi) U(k)
    \end{align*}
\]

(22)

Because the first term on the right hand of (22) is independent of \( U(k) \), it has:

\[
    \min_{U} J \Rightarrow \min_{U} \hat{J} \\
    \text{s.t. } z(k) = \Psi(U(k)), \\
    |u_i| \leq u_{\text{max}}
\]

(23)
where $|u_i| \leq u_{\text{max}}$ represents the voltage constraints of the OMM system, limited by the DC motors, and $H = \Phi^T Q (Fz(k) - Y_i(k)), W = \Phi^T Q \Phi + R$. The above equation can be solved as a standard linear quadratic programing problem, and the global optimal solution can be obtained quickly. By solving equation (24), the optimal voltage sequence $U(k)$ in the predicted horizon $N_p$ can be obtained. The first element in $U(k)$ is the output voltage at time $k$. Block diagram of the proposed control design is show as Figure 1.

In order to improve the trajectory tracking control performances of MPC, set $Q_N \in \mathbb{R}^{5 \times 5}$ in $Q = \text{diag}(Q_1, Q_2, ..., Q_N)$ be larger than $Q_{p} \in \mathbb{R}^{5 \times 5}, R = \text{diag}(R_1, R_2, ..., R_N)$ is the voltage optimization term. In practical application, $Q$ and $R$ matrices need to be adjusted, which can not only satisfy the high-precision trajectory tracking, but also make the actuator outputs meet the constraints.

### Simulations

In simulations, a mathematical dynamic model established by Euler-Lagrange method is employed to approximate the real OMM. The approximate mathematical dynamic model in continuous-time form with voltage input is described as follows:

$$\ddot{x} + C_0 \dot{q} + G = Bu, \quad (25)$$

where $q = [x \ y \ \phi \ \theta_1 \ \theta_2]^T$ is the position state of OMM, $M \in \mathbb{R}^{5 \times 5}$ is the inertia matrix, $C_0 \in \mathbb{R}^{5 \times 5}$ represents the centrifugal and Coriolis matrix, $G \in \mathbb{R}^{5 \times 1}$ is the gravity term, $u = [u_1 \ u_2 \ u_3 \ u_4 \ u_5]^T$ is the supplied voltage of five motors.

The input and output data are collected using random initial states and random control input sequences which can stimulate the robot’s dynamic characteristics. The data set is established including the generated output data and control input data. Taking the current state, control input and output state data as a data set, a total of 23,500 sets of data are collected in the simulations used to calculate the high-dimensional Koopman model. The dimension of the lifted state-space is $N = 12$. Thin plate spline radial basis function with center at $\xi_0$ is chosen as basis function, that is, $\phi(\xi) = \|\xi - \xi_0\|^2 \log (\|\xi - \xi_0\|)$. Then using the method proposed, Koopman model (13) is obtained, and MPC scheme is then designed based on the linear model.

To verify the effectiveness of Koopman operator based MPC (K-MPC), two trajectories with different characteristics are used, which are considered as sinusoidal trajectory and linear trajectory according to the desired trajectory of the mobile platform. Define the desired position of the OMM as $q_d = [x_d \ y_d \ \phi_d \ \theta_{1d} \ \theta_{2d}]^T$, then the two desired trajectories are set as follows:

1. The desired sinusoidal trajectory:

$$\begin{align*}
x_d[m] &= 0.8 \cos \left( \frac{\pi t}{40} \right) \\
y_d[m] &= 0.8 \cos \left( \frac{\pi t}{60} \right) \\
\phi_d[\text{rad}] &= \sin \left( \frac{\pi t}{30} \right) \quad 0 \leq t \leq 60s. \\
\theta_{1d}[\text{m}] &= 0.6 \sin \left( \frac{\pi t}{10} \right) \\
\theta_{2d}[\text{m}] &= 0.6 \sin \left( \frac{\pi t}{10} \right)
\end{align*}$$
The desired piecewise linear trajectory:

\[
\begin{align*}
    x_d[m] &= \begin{cases} 
        0.1t, & 0 \leq t < 10s \\
        1, & 10s \leq t < 20s \\
        1 - 0.1(t - 20), & 20s \leq t < 30s \\
        0, & 30s \leq t < 40s \\
        0, & 0 \leq t < 10s 
    \end{cases} \\
    y_d[m] &= \begin{cases} 
        0.1(t - 10), & 10s \leq t < 20s \\
        1, & 20s \leq t < 30s \\
        1 - 0.1(t - 30), & 30s \leq t < 40s \\
        0, & 0 \leq t < 10s 
    \end{cases} \\
    \phi_d[rad] &= \begin{cases} 
        0.1(t - 10), & 10s \leq t < 20s \\
        1, & 20s \leq t < 30s \\
        1 - 0.1(t - 30), & 30s \leq t < 40s 
    \end{cases} \\
    \theta_1[rad] = \theta_2[rad] = 0.6 \sin \left( \frac{\pi t}{20} \right).
\]

In simulations, the parameters used in K-MPC are set as follows:

\[
    Q_a = \text{diag}(0.1, 0.1, 0.1, 0.1, 0.1), \\
    Q_{Np} = \text{diag}(100, 100, 100, 100, 100), \\
    R_c = \text{diag}(1 \times 10^{-6}, 1 \times 10^{-6}, 1 \times 10^{-6}, 1 \times 10^{-6}).
\]

Simulations results are displayed in Figures 2 and 3. It can be seen that the OMM can complete the trajectory tracking task of the two reference trajectories. The proposed control scheme K-MPC achieves satisfactory performances, without any prior knowledge and model information.

Experiments

Experimental setup

Figure 4 shows the prototype platform. The omnidirectional mobile platform consists of three omnidirectional wheels arranged at 120° intervals. Each wheel is actuated by a DC motor. The manipulator is a parallelogram arm with four links. The parallelogram arm has two degrees of freedom (i.e. \( \theta_1 \) and \( \theta_2 \)), which are respectively driven by one DC motor. The five DC motors are identical and are Maxon DC motors with gear reduction ratio of 186 and the nominal voltage of 24 V.

The proposed control approach is implemented on a personal computer (Intel(R) Core(TM) i7-4770 CPU@3.40GHz) via Matlab/Simulink. The control signals generated from a personal computer are then transmitted to the OMM via WIFI module (part NO. ESP8266). The MCU (part NO. STM32F103VET6) in the control box receives the control signals and then sends the corresponding pulse-width modulation (PWM) signal to the five motor drivers (part NO. LMD18200). A motion capture system (OptiTrack) is employed to measure the position and orientation of the OMM. Figure 5 shows the schematic of the experimental setup.

Data collection and model identification

In the experiment, irreversible damage may be caused to the motor if the control voltage input changes too fast. Thus it is not feasible to give a random voltage sequence directly, as in simulations. Therefore, in order to balance the randomness and smoothness of the voltage, a voltage generation strategy is designed. The voltage range of the motors for the mobile platform is between −24 and 24 V. Due to the limited movement space of the manipulator, the voltage is limited between −5 and 5 V in order to avoid the excessive movement speed. A matrix \( Y_{5 \times 1500} \) of uniformly distributed random numbers between the voltage range is proposed first. Then set the time period \( T_{u1} \) or \( T_{u2} \), which is much longer than the sampling time, as the interval of time between two adjacent elements of \( Y \). In order to ensure the smoothness of the voltage signal, the random control inputs are obtained by followings:

![Figure 2](image-url) Simulation results of the sinusoidal trajectory: (a) Tracking responses of the OMM. (b) Tracking errors of the OMM. (c) Control input \( u \).
where \( k_1 = \text{floor}(t/T_{u1}) \) and \( k_2 = \text{floor}(t/T_{u2}) \) are the current indexes into the lookup table at time \( t \). The transition periods were set as \( T_{u1} = 0.3 \) s and \( T_{u2} = 0.15 \) s. Data sets of 57 trajectories were collected in the experiments, of which the first 50 sets were training sets used to calculate the Koopman high-dimensional model, and the last seven sets were test sets used to evaluate the approximation accuracy of the high-dimensional model. Each data collection process lasts for 5 s and the sampling time is 0.025 s. Then Koopman model was fit from the collected data, including state (current state and output state) data and control input (voltage) data.

**Experimental results**

**Koopman model evaluation.** The performance of Koopman model was benchmarked against Lagrange model (LM) of the OMM. Two sets of the collected data in the experiments were employed in the model evaluation. The two voltage sequences in the test sets are shown in Figure 6, and the comparative results of the output state trajectory are presented in Figure 7. It is obvious that KM has better predicted performance than LM for the evolution of the system and its predicted state trajectory is closer to the real model. Accurate prediction capability can improve the MPC performances. In order to quantitatively illustrate the advantages of the KM in the experiments, relative root
mean squared error (RRMSE) is defined to evaluate the model accuracy:

\[
RRMSE(\%) = 100 \times \frac{\sqrt{\sum_k \| \xi_{\text{pred}}(kT_s) - \xi_{\text{true}}(kT_s) \|^2}}{\sqrt{\sum_k \| \xi_{\text{true}}(kT_s) \|^2}},
\]

where \(T_s\) is the sampling period.

In the experiments, RRMSE was calculated to measure the approximation accuracy of the high-dimensional linear model to the real OMM experimental system. Seven sets of the collected data were selected randomly to calculate the RRMSE of different trajectories respectively, and the average value was obtained. The calculation results of KM and LM are displayed in Table 1. The RRMSE results show that KM has higher modeling accuracy than LM of the OMM system. Averagely, the accuracy of KM is improved by about 25%, compared with LM.

**Validations of the control effectiveness.** In this part, experimental tests were carried out to verify the trajectory tracking performances of K-MPC. The desired trajectories used in experiments are the same as those in simulations. In the experiments, the parameters used in K-MPC were set as follows:
\( Q_a = \text{diag}(3, 5, 5, 5) \),
\( Q_{Na} = \text{diag}(265, 300, 320, 480, 480) \),
\( R_c = \text{diag}(8 \times 10^{-5}, 8 \times 10^{-5}, 5 \times 10^{-3}) \).

Experiments results are displayed in Figures 8 and 9. It can be seen that the OMM can complete the trajectory tracking task of the two reference trajectories, which validates the effectiveness of the proposed control design in the experimental environment.

In the complex and changeable practical application environment, it is more difficult to model and control the system. The experiments prove that K-MPC method provides an effective solution to achieve the modeling and control of complex nonlinear systems.

### Conclusions

In this paper, a data-driven modeling and control method based on Koopman operator theory have been successfully applied to an OMM. Using EDMD-based Koopman operator approximation method, KM of the OMM system has been obtained firstly. The KM accuracy has been evaluated compared with LM qualitatively and quantitatively, which shows KM has higher accuracy. The K-MPC scheme has been designed and has been shown to be capable of commanding OMM to follow a reference trajectory in simulations and experiments. This method makes the modeling of the complex nonlinear dynamic systems more easier, leading to the rapid development of new control strategies and applications. To the best of our knowledge, this is the first experimental validation of Koopman-based MPC control for rigid manipulators.
Future work will consider the method of choosing the most effective basis functions to achieve high precision approximation. Also, the real-time uncertainties of the systems in the practical application are expected to be included in the Koopman model in the future development.

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
Xuchong Zhu https://orcid.org/0000-0003-4824-1869
Chengjun Ding https://orcid.org/0000-0003-0455-7538
Lizhen Jia https://orcid.org/0000-0002-7184-859X
Yubo Feng https://orcid.org/0000-0002-3006-6727

References
1. 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: 2197–2211.
2. Fareh R, Saad MR, Saad M, et al. Trajectory tracking and stability analysis for mobile manipulators based on decentralized control. Robotics 2019; 37: 1732–1749.
3. Watanabe K, Sato K, Izumi K, et al. Analysis and control for an omnidirectional mobile manipulator. J Intell Robot Syst 2000; 27: 3–20.
4. Djebrani S, Benali A and Abdessemed F. Modelling and control of an omnidirectional mobile manipulator. Int J Appl Math Comput Sci 2012; 22: 601–616.
5. Zhang Y, Li W, Liao B, et al. Analysis and verification of repetitive motion planning and feedback control for omnidirectional mobile manipulator robotic systems. J Intell Robot Syst 2014; 75: 393–411.
6. Sáurez R, Palomo-Avellaneda L, Martínez J, et al. Development of a dexterous dualarm omnidirectional mobile manipulator. IFAC-PapersOnLine 2018; 51: 126–131.
7. Guo S, Jin Y, Bao S, et al. Accuracy analysis of omnidirectional mobile manipulator with mecanum wheels. Adv Manuf 2016; 4: 363–370.
8. Song KT, Jiang SY and Lin MH. Interactive teleoperation of a mobile manipulator using a shared-control approach. IEEE Trans Hum Mach Syst 2016; 46: 834–845.
9. Buzi Avanzini G, Zanchettin AM and Rocco P. Constrained model predictive control for mobile robotic manipulators. Robotics 2018; 36: 19–38.
10. Maaiz MK, Boucher P and Dumur D. A new control strategy for induction motor based on non-linear predictive control and feedback linearization. Int J Adapt Control Signal Process 2000; 14: 313–329.
11. Wen C and Hill DJ. Adaptive linear control of nonlinear systems. IEEE Trans Automat Contr 1990; 35: 1253–1257.
12. Xi YG, Wang F and Wu GH. Nonlinear multi-model predictive control. IFAC Proc Vol 1996; 29: 2359–2364.
13. Dumont GA and Fu Y. Nonlinear adaptive control via Laguerre expansion of Volterra kernels. Int J Adapt Control Signal Process 1993; 7: 367–382.
14. Koopman BO. Hamiltonian systems and transformation in Hilbert space. Proc Natl Acad Sci USA 1931; 17: 315–318.
15. Hemati MS, Williams MO and Rowley CW. Dynamic mode decomposition for large and streaming datasets. Phys Fluids 2014; 26: 111701–111728.
16. Williams MO, Kevrekidis IG and Rowley CW. A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. J Nonlinear Sci 2015; 25: 1307–1346.
17. Ian A, Gerardo T and Todd M. Model-based control using Koopman operators. In: N. Amato, S. Srinivasa, N. Ayavanj, & S. Kuindersma (Eds.), Robotics: Science and Systems, Cambridge, Massachusetts, USA: MIT Press Journals, 12–16 July 2017, pp. 52–59.
18. Mamakoukas G, Castano ML, Tan X, et al. Local Koopman operators for data-driven control of robotic systems. In: A. Bicchi, H. Kress-Gazit, & S. Hutchinson (Eds.), Robotics: Science and Systems, Freiburg im Breisgau, Germany: MIT Press Journals, 22–26 June 2019, pp. 54–62.
19. Bruder D, Gillespie B, Remy CD, et al. Modeling and control of soft robots using the Koopman operator and model predictive control. In: A. Bicchi, H. Kress-Gazit, & S. Hutchinson (Eds.), Robotics: Science and Systems, Freiburg im Breisgau, Germany: MIT Press Journals, 22–26 June 2019, pp. 60–69.
20. Abraham I and Murphey TD. Active learning of dynamics for data-driven control using Koopman operators. IEEE Trans Robot 2019; 35: 1071–1083.
21. Proctor JL, Bruntonz SL and Kutzx JN. Generalizing Koopman theory to allow for inputs and control. SIAM J Appl Dyn Syst 2018; 17: 909–930.
22. Korda M and Mezić I. On convergence of extended dynamic mode decomposition to the Koopman operator. J Nonlinear Sci 2018; 28: 687–710.
23. Korda M and Mezić I. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. Automatica 2018; 93: 149–160.
24. Yeung E, Kundu S and Hodas N. Learning deep neural network representations for Koopman operators of nonlinear dynamical systems. In: 2019 American Control Conference (ACC), Philadelphia, PA, USA, 10-12 July 2019, pp.4832–4839. IEEE, Piscataway, NJ, USA.
25. Holkar KS and Whagmure LM. An overview of model predictive control. Int J Control Autom 2010; 3: 47–63.
26. Paulson JA, Mesbah A, Streif S, et al. Fast stochastic model predictive control of high-dimensional system. In: 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, 15-17 December 2014, Piscataway, NJ, USA: IEEE. pp.2802–2809.