


RESEARCH ARTICLE

Synchronization control of blanket remote maintenance robot based on MPC-CCC algorithm

Dongyi Li^{1,2,3,4} , Kun Lu¹, Yong Cheng^{1,3}, Huapeng Wu⁴, Heikki Handroos⁴, Xuanchen Zhang^{1,2,3}, Xinpeng Guo^{1,2,3}, Songzhu Yang^{1,3}, Liansheng Du^{1,3} and Yu Zhang^{1,3}

¹Institute of Plasma Physics, Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China, ²University of Science and Technology of China, Hefei 230026, China, ³Anhui Extreme Environment Robot Engineering Laboratory, Hefei 230031, China, and ⁴Lappeenranta University of Technology, Lappeenranta, Finland

Corresponding author: Huapeng Wu; Email: huapeng.wu@lut.fi

Received: 22 May 2023; **Revised:** 28 June 2023; **Accepted:** 13 July 2023; **First published online:** 14 August 2023

Keywords: blanket remote maintenance robot; Mover; hydraulic system; MPC-CCC; synchronization control

Abstract

This paper studies the synchronization control of the blanket remote maintenance robot (BRMR) of the China fusion engineering test reactor (CFETR). First, the general state space mathematical model of BRMR was established by using a physical-based method. Second, based on the receding horizon optimization of model predictive control (MPC) and cross-coupling error reduction in cross-coupling control (CCC), the innovative MPC-CCC controller was proposed to realize the single-system and multisystem error convergence and high accuracy transportation of blanket through the high accuracy synchronization control of BRMR. Third, to verify the control effectiveness of the MPC-CCC controller, two types of simulations and experiments were implied compared with the original proportional-integral (PI) controller in Mover. Results showed that simulation and experiments were highly consistent. It is found that the use of an MPC-CCC controller can result in up to a 70% reduction in displacement error and up to a 59% reduction in synchronization error compared to the PI controller. And the accuracy of the MPC-CCC controller satisfies the real requirement of the maintenance process of the blanket. This work provides the theoretical basis and practical experience for the highly stable, safe, and efficient maintenance of blankets in the future.

1. Introduction

The China fusion engineering test reactor (CFETR) is a tokamak-type fusion reactor, which is designed in China and fills the gap between international thermonuclear experimental reactor (ITER) and the DEMO-class reactor.

As a fusion reactor, CFETR's components will be severely impacted in terms of their lifespan and performance due to the complex operating conditions during its operation, including extreme environments such as high temperatures, strong magnetic fields, and intense radiation. This is especially true for components that are directly exposed to the plasma, such as the first wall, blanket, diverters, and heating devices, which will be subjected to varying degrees of high-energy neutron radiation and will become irradiated and activated due to interaction with the plasma or neutrons.

The CFETR blanket is a large and heavy module, with each module being 7.6 m tall and weighing 60 tons. In the maintenance process, the blanket is needed to transfer from the vacuum vessel (VV) to the maintenance port. The whole process is completed by blanket remote maintenance robot (BRMR). BRMR includes two Movers (as shown in Fig. 1). Each Mover consists of a driving unit, a Jacking Unit, and a locking unit, all of which are hydraulic transmission systems [1]. The Mover driving unit (MDU) connects the locking module (LM) and jacking module (JM) as shown in Fig. 2. The Mover jacking unit (MJU) consists of four jacking cylinders (JC) in JM. The Mover locking unit (MLU) consists of a pair of main locking cylinders (MLC) in LM and a pair of deputy locking cylinders (DLC) in JM. In the

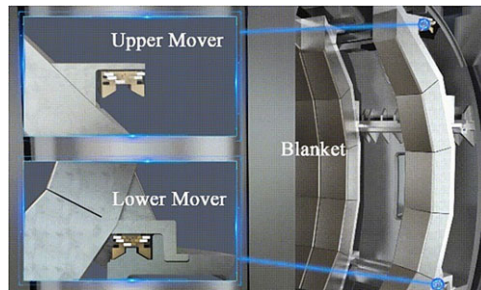


Figure 1. Schematic diagram of blanket transportation.

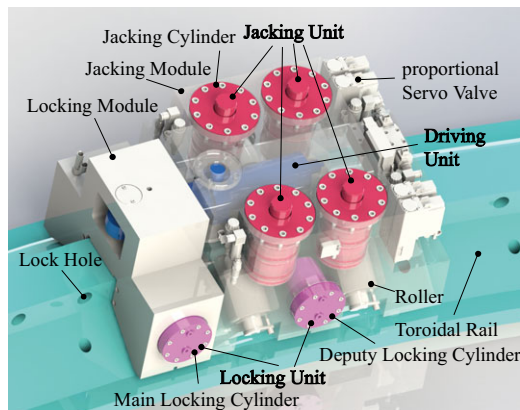


Figure 2. Mover schematic diagram.

maintenance process, firstly, JCs of MJU lift the blanket 30 mm upwards and unlock the blanket from VV. Second, pistons of DLC of MLU stretch out into the lock hole lock the JM, and blanket on the rail. Pistons of MLC of MLU retract from the lock hole to unlock the LM. Then, the driving cylinder (DC) of MDU stretches out and drives the LM move along the toroidal rail. Third, pistons of DLC of MLU retract from the lock hole to unlock the JM and blanket. Pistons of MLC of MLU stretch out into the lock hole and lock the LM. Then, the DC of MDU retracts and drives the JM and blanket move along the toroidal rail. After the repetition of the process above, the blanket can be transferred to the maintenance port. If the blanket moves in the opposite direction, vice versa. The whole synchronous movement of the blanket requires synchronous movement of the BRMR which is the synchronous movement of the upper and lower Mover. The blanket has a narrow, long structure and high gravity center, which is easy to get stuck during the toroidal movement. In the process of blanket toroidal transportation, the dynamic tracking accuracy of the displacement is required to be ± 5 mm, the steady-state accuracy of the displacement is ± 2 mm, and the dynamic tracking accuracy of the synchronous displacement is ± 5 mm. So, synchronous movement requires high precision of pose and synchronization control of the blanket.

There are many motion control algorithms, such as robust control, sliding mode control, proportional–integral–derivative (PID) control, etc. are widely used in research and factorial application [2, 3]. However, robust control design often tends to be conservative and control effect will decrease when the uncertainty goes beyond the specified limits [4]. The mostly used robust control often combines with other methods, such as adaptive method, etc., which will lead to higher controller order and lower steady state accuracy [5]. Sliding mode control has chattering phenomenon. This method is often approved by changing reaching law to decrease chattering, but it may compromise control performance. PID is well known in the industry but PID control is not well-suited for systems with nonlinear dynamics, time-varying parameters, or complex behaviors. And PID controllers may have slow responses for

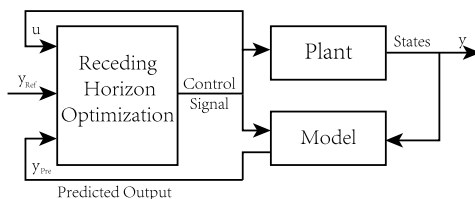


Figure 3. MPC schematic diagram.

processes with fast dynamics, as the controller may not be able to keep up with rapid changes in the process variable. These methods are not suitable for BRMR due to their characteristics.

Model predictive control (MPC) is a control algorithm that uses a mathematical model of the system being controlled to make predictions about its future behavior as shown in Fig. 3. These predictions are then used to determine an optimal control action that minimizes cost function over a finite time horizon. This control algorithm can find the optimal control values repeatedly at each time step. In the actual application, it can predict system output, and do the optimization process based on the newest system prediction output and actual state feedback. MPC can handle systems with complex dynamics and multiple inputs and outputs and overcome disturbances and uncertainties in the system, as it can adjust the control action based on real-time measurements of the system's state. It can optimize the performance of the system over a finite time horizon, leading to improved efficiency and reduced costs. MPC can also ensure safe and reliable operation in the real system based on constraints in the optimizing process [6–8]. Tiancheng Ouyang et al established a model predictive controller for heavy-duty automatic transmission shifting pressure control [9]. The simulation results indicated that the MPC can be applied to heavy-duty vehicles for improving shift quality. Joseph T. Jose et al used MPC for the dynamic performance enhancement of an excavator boom and gained a significant reduction of cost function and vibration [10]. Buyoun Cho et al used MPC to improve the energy efficiency of legged robots and achieved the commanded motion while minimizing energy consumption [11]. Balint Varga et al proposed MPC in mid-sized heavy-duty vehicle manipulator systems with a hydraulic working arm and got smoother motions after the dual-trajectories optimization [12]. Qin Shi et al and Mingming Mei et al used MPC to get better performance in pressure-demand control of electro-hydraulic braking systems [13, 14]. Heybroek Kim et al proposed MPC in hydraulic multichamber actuator force control and got good force tracking performance [15]. Xiaohua Zeng et al designed an MPC feedback controller (FMPC) to calculate the control input sequence of engine torque and hydraulic variable pump displacement [16]. Simulation results showed that the designed FMPC has the best performance, and control performance can be guaranteed in a real-time environment. Frank A. Bender et al used the MPC algorithm to control a hydraulic mini excavator and experimental results showed the developed control effect in hydraulic machinery [17]. O. A. Dahunsi et al presented a controller based on neural network (NN) and MPC [18]. Simulation results demonstrate the superior performance of the NN MPC over the generic PID based on adapting to the deterministic road disturbance. Jouko Kalmari et al presented nonlinear model predictive control (NMPC) to do the anti-sway control of a hydraulic forestry crane and results showed that it can reduce sway significantly [19].

In synchronization control, there are mainly three methods: equal control, master-slave control, and cross-coupling control. In equal control, multiple hydraulic actuators simultaneously track the same ideal input signal to achieve synchronous control as shown in Fig. 4. The synchronization accuracy of equal control depends on the consistency of each hydraulic component in the hydraulic system. If the consistency of the components is good, the synchronization error will be small. However, if components have low consistency, the system has a bad synchronization effect. And this method has a weak anti-interference ability. In master-slave control, multiple hydraulic actuators track the output of one of the actuators as a standard, and the other actuators are controlled to track this standard to achieve synchronous operation as shown in Fig. 5. Without adding feedforward control, the slave hydraulic cylinder

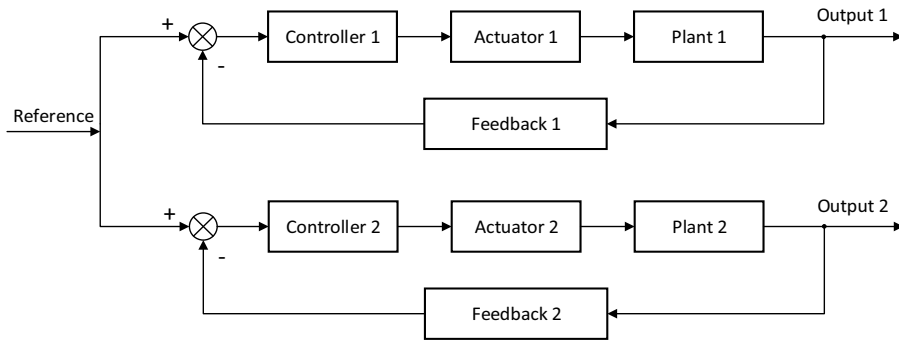


Figure 4. Equal control schematic diagram.

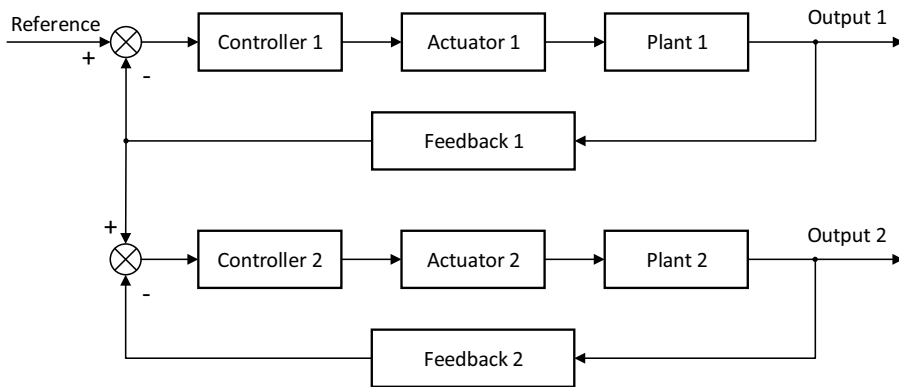


Figure 5. Master-slave control schematic diagram.

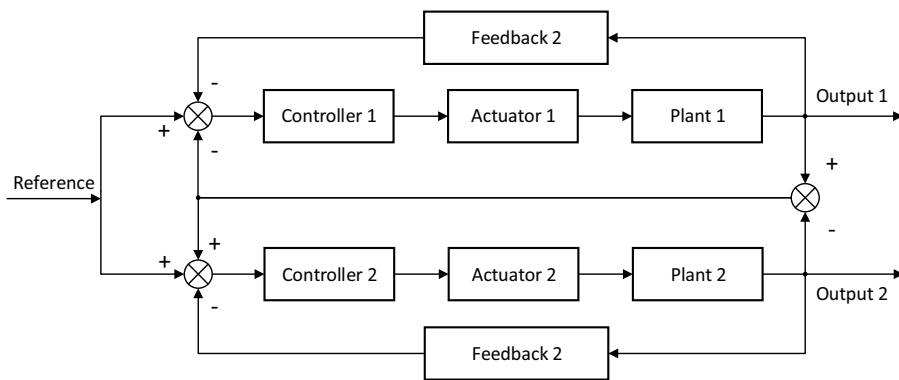


Figure 6. Cross-coupling control schematic diagram.

has a large tracking delay compared to the master hydraulic cylinder, resulting in a large error. In cross-coupling control (CCC), all hydraulic actuators track the same input signal, and the error between the actuators is coupled for synchronous control as shown in Fig. 6. This method can achieve convergence of trajectory and synchronization error simultaneously, ensuring synchronization accuracy without affecting the position accuracy of a single cylinder. Xinyu Yang et al proposed a finite-time tracking and cross-coupling synchronization control method for a dual-motor servo system [20]. Several simulations proved the controller’s stability and effectiveness. Dong Sun et al proposed a model-free cross-coupled controller for position synchronization of multiaxis motions [21]. Dealing with the coupled position

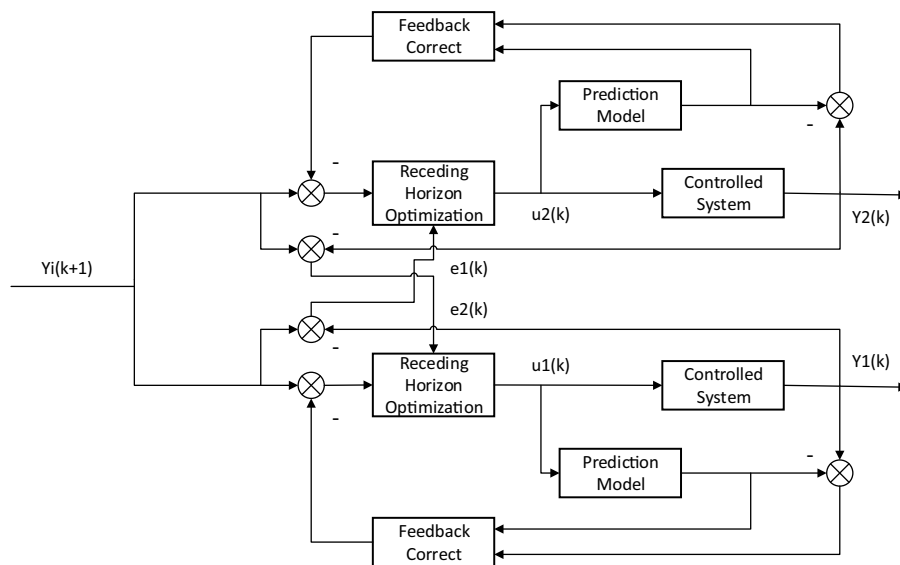


Figure 7. The MPC-CCC control algorithm.

error experiments showed the effectiveness of the controller. Hao Yuan et al used CCC method to realize the synchronous cooperation of two parallel permanent magnet linear synchronous motors [22]. Experimental results demonstrated the efficiency and superior performance of the CCC method. Zhian Kuang et al presented CCC and sliding mode control to reduce the incoordination among driving linear motors [23]. The simulation and experiment results proved their effectiveness and advantages over conventional methods. Guoqiang Han et al presented improved sliding mode control and CCC in the speed synchronization control of dual-switched reluctance motor and got a good performance in simulation and experiment [24]. Songchun Zou et al proposed CCC to enhance the synchronization performance of vehicle dual-motor steer-by-wire system and proved the effectiveness of CCC by dual-motor experiment [25].

From the above studies, it is found that MPC is widely used in hydraulic systems and has good performance in actual control that improves the stability in the real process. And in synchronization control, CCC is often used in the dual motor system and has good performance of accurate synchronous control in a complex system. So, it is an innovative way to use it in the hydraulic system. In this paper, an innovative MPC-CCC control algorithm is proposed as shown in Fig. 7. To realize accurate synchronization control of BRMR and the stable and efficient transfer and maintenance of blanket, inspired by the above studies, the main contributions in this paper can be summarized as the following items:

(1) The mathematical model of BRMR with forward and backward motion models for two asymmetric cylinders is derived based on the physical-based method which can also be used in other same type synchronization hydraulic systems. It is applied to BRMR to provide a model basis for synchronous controller design.

(2) The innovative MPC-CCC control method corresponding controller is proposed based and improved by MPC algorithm and CCC algorithm in this paper. MPC can realize a good control effect in complex system which is one of the control algorithms that can be well used in real system. And it also has good control effects in single Mover. To solve the synchronization control problem of BRMR, we extracted the synchronous errors of BRMR in controller's cost function to decrease synchronous errors and give compensations to each single Mover in the circumstance of ensuring each single Mover system control accuracy. It is a new attempt and an effective way to deal with the high accuracy synchronization control of BRMR with the real-time hydraulic system. And the optimization equation is derived to a standard QP equation to speed up the solving process. The qpSWIFT is a light-weight real-time sparse

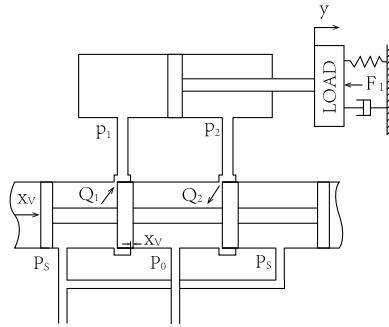


Figure 8. Schematic diagram of valve-controlled asymmetric hydraulic cylinder.

quadratic program solver [26]. It can decrease the solving time, is faster compare to other QP solvers [27], and is then applied in this paper.

(3) The simulation and experiments of heteromorphic (MJU-MDY) and isomorphic (MJU-MJY) hydraulic systems are carried out to verify the control effect of MPC-CCC algorithm. The controller feasibility and stability is also verified in the real operation. In the comparison with Mover’s original PI controller, the experimental results fully demonstrate the effectiveness of this MPC-CCC control strategy on matter heteromorphic and isomorphic systems, providing theoretical and experimental experience for the successful operation of fusion reactors in the future.

The remainder of this paper is organized as follows. The BRMR modeling is shown in Section 2. The MPC-CCC controller design is shown in Section 3. Simulation analysis of BRMR with heteromorphic and isomorphic systems is shown in Section 4. Experimental results of the proposed controller compared with original PI controller accordance to Section 4 is shown in Section 5. The conclusion is found in Section 6.

2. BRMR modeling

2.1. Simplified model of the proportional servo valve

The proportional servo valve in BRMR includes an internal closed-loop control system that regulates the relationship between the input signal u and the output valve spool displacement x_v . As a result, the proportional servo valve module can be represented by the following simplified linear equation:

$$x_v = K_1 u \tag{1}$$

where

x_v —The spool displacement;

K_1 —The gain;

u —The input signal.

2.2. Valve-controlled asymmetric hydraulic cylinder modeling

As the effective areas of the two cavities of the asymmetric hydraulic cylinder are different, it is necessary to establish piecewise linear models for both forward and backward movements of the hydraulic cylinder as shown in Fig. 8. Positive and negative values of x_v represent forward and backward movements of the hydraulic cylinder, respectively. By doing so, a complete and general valve-controlled asymmetric hydraulic cylinder model can be established.

2.2.1. $x_v > 0$ asymmetric hydraulic cylinder model

At the beginning, the following definitions are obtained:

$$n = A_2/A_1 \tag{2}$$

$$P_L = P_1 - nP_2 \tag{3}$$

where

A_1 —Piston effective area with rodless cavity;

A_2 —Piston effective area with rod cavity;

P_L —Load pressure;

P_1 —Pressure with rodless cavity;

P_2 —Pressure with rod cavity;

From Eqs. (2)–(3) and valve flow equation, the flow with load equation is derived:

$$Q_L = A_1 \dot{y} + C_{ief} P_L - C_{eef} P_s + \frac{V_{tf}}{\beta_e} \dot{P}_L \tag{4}$$

where

y —Hydraulic cylinder piston displacement

V_{tf} —Effective volume;

C_{ief} —Equivalent leakage coefficient;

C_{eef} —Additional leakage coefficient.

With linearizing flow with load equation near the balance point of the valve spool [28], the following equation is obtained:

$$Q_L = K_{gf} x_v - K_{cf} P_L \tag{5}$$

where

K_{gf} —The flow gain;

K_{cf} —Flow-pressure coefficient.

Then, the hydraulic cylinder force equilibrium equation is shown below.

$$A_1 P_L = M \ddot{y} + B_p \dot{y} + Ky + F_l \tag{6}$$

where

M —Total piston and load mass;

B_p —Viscous damping coefficient of piston and load;

K —Load elastic stiffness;

F_l —The external force.

By combining Eqs. (1) and (5)–(6), the following equations are obtained:

$$\dot{\mathbf{x}} = \mathbf{A}_f \mathbf{x} + \mathbf{B}_f u + \mathbf{d}_f \tag{7}$$

$$\mathbf{y} = \mathbf{C}_f \mathbf{x} \tag{8}$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \ddot{y} \end{bmatrix};$$

$$\mathbf{A}_f = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\frac{\beta_e C_{ief} K}{V_{tf} M} & -\left(\frac{K}{M} + \frac{\beta_e C_{ief} B_p}{V_{tf} M} + \frac{\beta_e A_1^2}{V_{tf} M}\right) & -\left(\frac{B_p}{M} + \frac{\beta_e C_{ief}}{V_{tf}}\right) \end{bmatrix};$$

$$\mathbf{B}_f = \begin{bmatrix} 0 \\ 0 \\ \frac{\beta_e A_1 K_{gf} K_1}{V_{tf} M} \end{bmatrix};$$

$$\mathbf{C}_f = [1 \ 0 \ 0];$$

$$d_f = \begin{bmatrix} 0 \\ 0 \\ -\left(\frac{\beta_e C_{ief}}{V_{tf} M} F_l + \frac{1}{M} \dot{F}_l\right) \end{bmatrix};$$

β_e —The effective bulk modulus of elasticity;

C_{ief} —Equivalent leakage coefficient;

V_{tf} —Effective volume;

This is the mathematical model when the hydraulic cylinder moves forward.

2.2.2. $x_v < 0$ asymmetric hydraulic cylinder model

The definitions are obtained as shown in the following equations:

$$P_L = P_2 - P_1/n \tag{9}$$

In the same way as in Section 2.2.1, the following equations are obtained:

$$\dot{x} = A_b x + B_b u + d_b \tag{10}$$

$$y = C_b x \tag{11}$$

where

$$A_b = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\frac{\beta_e C_{ieb} K}{V_{tb} M} & -\left(\frac{K}{M} + \frac{\beta_e C_{ieb} B_p}{V_{tb} M} + \frac{\beta_e A_2^2}{V_{tb} M}\right) & -\left(\frac{B_p}{M} + \frac{\beta_e C_{ieb}}{V_{tb}}\right) \end{bmatrix};$$

$$B_b = \begin{bmatrix} 0 \\ 0 \\ \frac{\beta_e A_2 K_{qb} K_1}{V_{tb} M} \end{bmatrix};$$

$$C_b = [1 \ 0 \ 0];$$

$$d_b = \begin{bmatrix} 0 \\ 0 \\ -\left(\frac{\beta_e C_{ieb}}{V_{tb} M} F_l + \frac{1}{M} \dot{F}_l\right) \end{bmatrix};$$

C_{ieb} —Equivalent leakage;

V_{tb} —Effective volume;

K_{qb} —The flow gain.

This is the mathematical model when the hydraulic cylinder moves backward.

2.2.3. General state space model of asymmetric hydraulic cylinder

Combining the piecewise mathematical model above, the general and complete asymmetric hydraulic cylinder state space function type model can be obtained as shown in the following equations.

$$\dot{x} = Ax + Bu + d \tag{12}$$

$$y = Cx \tag{13}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_1 & a_2 & a_3 \end{bmatrix};$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ b_1 \end{bmatrix};$$

$$\mathbf{C} = [1 \ 0 \ 0];$$

$$\mathbf{d} = \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix};$$

where

$$a_1 = -\frac{\beta_e K}{M} \left(g_f(u) \frac{C_{ief}}{V_{if}} + g_b(u) \frac{C_{ieb}}{V_{ib}} \right);$$

$$a_2 = -\left(\frac{K}{M} + \frac{\beta_e B_p}{M} \left(g_f(u) \frac{C_{ief}}{V_{if}} + g_b(u) \frac{C_{ieb}}{V_{ib}} \right) + \frac{\beta_e}{M} \left(g_f(u) \frac{A_1^2}{V_{if}} + g_b(u) \frac{A_2^2}{V_{ib}} \right) \right);$$

$$a_3 = -\left(\frac{B_p}{M} + \beta_e \left(g_f(u) \frac{C_{ief}}{V_{if}} + g_b(u) \frac{C_{ieb}}{V_{ib}} \right) \right);$$

$$b_1 = \frac{\beta_e K_1}{M} \left(g_f(u) \frac{A_1 K_{qf}}{V_{if}} + g_b(u) \frac{A_2 K_{qb}}{V_{ib}} \right);$$

$$d_1 = -\left(\frac{\beta_e}{M} \left(g_f(u) \frac{C_{ief}}{V_{if}} + g_b(u) \frac{C_{ieb}}{V_{ib}} \right) F_l + \frac{1}{M} \dot{F}_l \right);$$

$$g_f(u) = \frac{1 + \text{sign}(u)}{2};$$

$$g_b(u) = \frac{1 - \text{sign}(u)}{2};$$

$$\text{sign}(u) = \begin{cases} -1, & u < 0 \\ 0, & u = 0 \\ 1, & u \geq 0 \end{cases}.$$

This is the general mathematical model of the valve-controlled asymmetric hydraulic cylinder.

2.3. BRMR mathematical state space function establishment

The BRMR has two Movers. Using the general system model representation above, the MDU model of upper Mover can be obtained.

$$\dot{\mathbf{X}}_1 = \mathbf{A}_1 \mathbf{X}_1 + \mathbf{B}_1 u_1 + \mathbf{d}_1 \tag{14}$$

$$\dot{\mathbf{Y}}_1 = \mathbf{C}_1 \mathbf{X}_1 \tag{15}$$

where

$$\mathbf{X}_1 = \begin{pmatrix} y_1 \\ \dot{y}_1 \\ \ddot{y}_1 \end{pmatrix} - \text{Upper Mover MDU states};$$

y_1 -Upper Mover MDU displacement.

As same to Eqs. (14) and (15), the MDU model of lower Mover can be obtained.

$$\dot{\mathbf{X}}_2 = \mathbf{A}_2 \mathbf{X}_2 + \mathbf{B}_2 u_2 + \mathbf{d}_2 \tag{16}$$

$$\dot{Y}_2 = C_2 X_2 \tag{17}$$

where

$$X_2 = \begin{pmatrix} y_2 \\ \dot{y}_2 \\ \ddot{y}_2 \end{pmatrix} \text{—lower Mover MDU states;}$$

y_2 —lower Mover MDU displacement.

For the convenience of establishing the controller, the two Movers are considered as a single entity. The BRMR entire system mathematical state space function is established as follows:

$$\dot{X} = AX + BU + d \tag{18}$$

$$Y = CX \tag{19}$$

where

$$X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix};$$

$$A = \begin{pmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{pmatrix};$$

$$B = \begin{pmatrix} B_1 & \mathbf{0} \\ \mathbf{0} & B_2 \end{pmatrix};$$

$$U = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix};$$

$$d = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix};$$

$$C = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}.$$

3. MPC-CCC controller design

Inspired by the studies in Section 1, taking advantage of MPC and CCC, the following innovative MPC-CCC is improved in this paper (see Fig. 7). The diagram of the MPC-CCC controller in BRMR is shown in Fig. 9.

3.1. System prediction output

The sampling period of the Mover controller is 10 ms. The systems Eqs. (18)–(19) are transformed into the form of discrete state space functions (Eqs. (20)–(21)). Since the external force changes slowly, it is simplified to a known constant. The converted system equations are then obtained as follows:

$$X(k + 1) = A_d X(k) + B_d U(k) + d_d \tag{20}$$

$$Y(k) = C_d X(k) \tag{21}$$

where

$$X(k + 1) = \begin{pmatrix} X_1(k + 1) \\ X_2(k + 1) \end{pmatrix};$$

$$A_d = \begin{pmatrix} A_{d1} & \mathbf{0} \\ \mathbf{0} & A_{d2} \end{pmatrix};$$

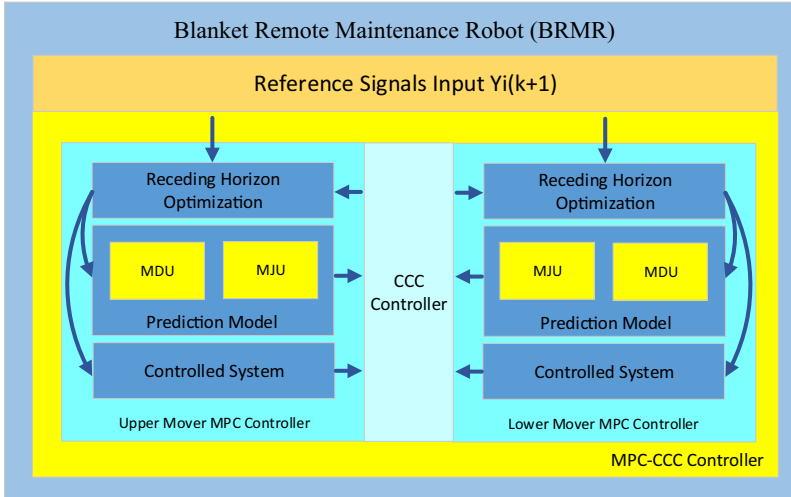


Figure 9. The MPC-CCC controller in BRMR.

$$\mathbb{B}_d = \begin{pmatrix} \mathbf{B}_{d1} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{d2} \end{pmatrix};$$

$$\mathbb{U}(k) = \begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix};$$

$$\mathbf{d}_d = \begin{pmatrix} d_{d1} \\ d_{d2} \end{pmatrix}.$$

$$\mathbf{C}_d = \begin{pmatrix} \mathbf{C}_{d1} \\ \mathbf{C}_{d2} \end{pmatrix}.$$

where

$\mathbf{A}_d, \mathbf{B}_d, \mathbf{d}_d,$ and \mathbf{C}_d are discrete forms of $\mathbf{A}, \mathbf{B}, \mathbf{d}, \mathbf{C}$, respectively.

According to Eqs. (20)–(21), system states for the subsequent N_p sample points are predicted by the following equation:

$$\begin{aligned} \mathbb{X}(k+1|k) &= \mathbb{A}_d \mathbb{X}(k) + \mathbb{B}_d \mathbb{U}(k) + \mathbf{d}_d \\ \mathbb{X}(k+2|k) &= \mathbb{A}_d^2 \mathbb{X}(k) + \mathbb{A}_d \mathbb{B}_d \mathbb{U}(k) + \mathbb{B}_d \mathbb{U}(k+1) + \mathbb{A}_d \mathbf{d}_d + \mathbf{d}_d \\ &\vdots \\ \mathbb{X}(k+N_p|k) &= \mathbb{A}_d^{N_p} \mathbb{X}(k) + \mathbb{A}_d^{N_p-1} \mathbb{B}_d \mathbb{U}(k) + \dots + \mathbb{A}_d^{N_p-N_c} \mathbb{B}_d \mathbb{U}(k+N_c-1) + \mathbb{A}_d^{N_p-1} \mathbf{d}_d + \dots + \mathbb{A}_d^{N_p-N_c} \mathbf{d}_d \end{aligned} \tag{22}$$

Eq. (22) can be transformed to matrix type:

$$\begin{aligned} \begin{pmatrix} \mathbb{X}(k+1|k) \\ \mathbb{X}(k+2|k) \\ \vdots \\ \mathbb{X}(k+N_p|k) \end{pmatrix} &= \begin{pmatrix} \mathbb{A}_d \\ \mathbb{A}_d^2 \\ \vdots \\ \mathbb{A}_d^{N_p} \end{pmatrix} \mathbb{X}(k) + \begin{pmatrix} \mathbb{B}_d & \dots & 0 \\ \vdots & \ddots & \vdots \\ \mathbb{A}_d^{N_p-1} \mathbb{B}_d & \dots & \mathbb{A}_d^{N_p-N_c} \mathbb{B}_d \end{pmatrix} \begin{pmatrix} \mathbb{U}(k) \\ \mathbb{U}(k+1) \\ \vdots \\ \mathbb{U}(k+N_c-1) \end{pmatrix} \\ &+ \begin{pmatrix} \mathbf{d}_d & \dots & 0 \\ \vdots & \ddots & \vdots \\ \mathbb{A}_d^{N_p-1} \mathbf{d}_d & \dots & \mathbb{A}_d^{N_p-N_c} \mathbf{d}_d \end{pmatrix} \end{aligned} \tag{23}$$

where

N_p —Prediction time domain;
 N_c —Control time domain.

That is,

$$\mathfrak{X} = \Psi \mathbb{X}(k) + \Phi \mathfrak{U} + \mathfrak{D} \tag{24}$$

From Eq. (20), states of upper Mover and lower Mover can be obtained.

$$X_1(k + 1) = K_1 \mathbb{X}(k + 1) \tag{25}$$

$$X_2(k + 1) = K_2 \mathbb{X}(k + 1) \tag{26}$$

where

$$K_1 = (I_{(3 \times 3)} \quad \mathbf{0}_{(3 \times 3)});$$

$$K_2 = (\mathbf{0}_{(3 \times 3)} \quad I_{(3 \times 3)}).$$

Then, the prediction states of upper Mover and lower Mover can be obtained as follows:

$$\mathfrak{X}_1 = \mathfrak{K}_1 \mathfrak{X} \tag{27}$$

$$\mathfrak{X}_2 = \mathfrak{K}_2 \mathfrak{X} \tag{28}$$

where

$$\mathfrak{X}_1 = \begin{pmatrix} X_1(k + 1) \\ X_1(k + 2) \\ \vdots \\ X_1(k + N_p) \end{pmatrix};$$

$$\mathfrak{K}_1 = \begin{pmatrix} K_1 & \dots & \dots & \mathbf{0} \\ \vdots & K_1 & & \vdots \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \dots & \dots & K_1 \end{pmatrix};$$

$$\mathfrak{X}_2 = \begin{pmatrix} X_2(k + 1) \\ X_2(k + 2) \\ \vdots \\ X_2(k + N_p) \end{pmatrix};$$

$$\mathfrak{K}_2 = \begin{pmatrix} K_2 & \dots & \dots & \mathbf{0} \\ \vdots & K_2 & & \vdots \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \dots & \dots & K_2 \end{pmatrix}.$$

3.2. Cost function of ordinary MPC

First, in actual system, the control value \mathfrak{U} have limitation. So, it is important to limit the calculation value of \mathfrak{U} .

$$\mathfrak{U}_{min} \leq \mathfrak{U} \leq \mathfrak{U}_{max} \tag{29}$$

Then, Eq. (29) can be turned to standard constraint in MPC.

$$S_1 \mathbf{u} \leq S_2 \tag{30}$$

where

$$S_1 = \begin{pmatrix} \mathbf{I}_{(2N_c \times 2N_c)} \\ -\mathbf{I}_{(2N_c \times 2N_c)} \end{pmatrix};$$

$$S_2 = \begin{pmatrix} \mathbf{u}_{max(2N_c \times 2N_c)} \\ -\mathbf{u}_{max(2N_c \times 2N_c)} \end{pmatrix}.$$

For ordinary MPC cost function in BRMR, it only contains the displacement error of every single MDU and the amplitude of control value of every MDU. This can be seen in the following cost function:

$$J_1 = (\mathfrak{R} - \mathfrak{X})^T Q_1 (\mathfrak{R} - \mathfrak{X}) + \mathbf{u}^T Q_2 \mathbf{u} \tag{31}$$

where

$$\mathfrak{R} = \mathbf{R} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}_{(N_p \times 1)} \quad \text{--Reference signal of BRMR;}$$

$$\mathbf{R} = \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{pmatrix};$$

$\mathbf{R}_{1(3 \times 1)}$ —Reference signal of upper MDU;

$\mathbf{R}_{2(3 \times 1)}$ —Reference signal of lower MDU;

$$Q_1 = \begin{pmatrix} q_{11} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & q_{1N_p} \end{pmatrix}_{(N_p \times N_p)} \quad \text{--Weight matrix on the whole prediction time in displacement, velocity, and acceleration;}$$

$$q_{11} = \begin{pmatrix} q_{x1} & \cdots & \cdots & 0 \\ & q_{v1} & & \vdots \\ & \vdots & q_{a1} & \vdots \\ & \vdots & & q_{x2} \\ & \vdots & & \vdots \\ 0 & \cdots & \cdots & q_{v2} \\ & & & q_{a2} \end{pmatrix} \quad \text{--Weight matrix on the first sample time, } q_{1N_p} \text{ is same as } q_{11};$$

q_{x1} —Displacement weight of upper MDU;

q_{v1} —Velocity weight of upper MDU;

q_{a1} —Acceleration weight of upper MDU;

q_{x2} —Displacement weight of lower MDU;

q_{v2} —Velocity weight of lower MDU;

q_{a2} —Acceleration weight of lower MDU;

$$Q_2 = \begin{pmatrix} q_{21} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & q_{2N_c} \end{pmatrix}_{(N_c \times N_c)} \quad \text{--Weight matrix on the whole prediction time in control value;}$$

$$q_{21} = \begin{pmatrix} q_{u1} & 0 \\ 0 & q_{u2} \end{pmatrix} \text{—Weight matrix on the first sample time, } q_{2N_c} \text{ is same as } q_{21};$$

q_{u1} —Control value weight of upper MDU;

q_{u2} —Control value weight of lower MDU.

3.3. Cost function of CCC algorithm

Then, considering the cross-coupled error between upper MDU and lower MDU of BRMR, the cross-coupling control algorithm is used in this paper. The cost function of cross coupled error term is shown in following equation.

$$J_2 = (\mathfrak{X}_1 - \mathfrak{X}_2)^T Q_3 (\mathfrak{X}_1 - \mathfrak{X}_2) \tag{32}$$

where

$$Q_3 = \begin{pmatrix} q_{31} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & q_{3N_p} \end{pmatrix}_{(N_p \times N_p)} \text{—Weight matrix on the whole prediction time in cross-coupled}$$

displacement, velocity and acceleration;

$$q_{31} = \begin{pmatrix} q_{ex} & 0 & 0 \\ 0 & q_{ev} & 0 \\ 0 & 0 & q_{ea} \end{pmatrix} \text{—Weight matrix on the first sample time, } q_{3N_p} \text{ is same as } q_{31};$$

q_{ex} —Cross-coupled displacement weight;

q_{ev} —Cross-coupled velocity weight;

q_{ea} —Cross coupled acceleration weight.

3.4. Cost function of MPC-CCC algorithm and its optimization type

Then, combining Eqs. (31)–(32), the overall cost function of MPC-CCC controller of BRMR is obtained in Eq. (33).

$$J = J_1 + J_2 \tag{33}$$

By the substitution of Eqs. (23)–(24), (27)–(28), and (30)–(32) into Eq. (33), the above equation is simplified as follows optimization type:

$$\min J = \frac{1}{2} \mathfrak{U}^T E \mathfrak{U} + \mathfrak{U}^T F \tag{34}$$

$$s.t. \mathbf{S}_1 \mathfrak{U} \leq \mathbf{S}_2$$

where

$$E = \Phi^T Q_4 \Phi + Q_2;$$

$$F = \Phi^T Q_4 \Psi \mathfrak{X}(k) + \Phi^T Q_4 \mathfrak{D} - \Phi^T Q_1 \mathfrak{X}.$$

$$Q_4 = Q_1 + \mathfrak{K}_3^T Q_3 \mathfrak{K}_3;$$

$$\mathfrak{K}_3 = \mathfrak{K}_1 - \mathfrak{K}_2.$$

Table I. Key simulation parameters.

Parameters	Values	Units
MDU diameter with rodless cavity	80	mm
MDU diameter with rod cavity	40	mm
MDU cylinder stroke	0.2	m
MJU diameter with rodless cavity	100	mm
MJU diameter with rod cavity	55	mm
MJU cylinder stroke	0.03	m
MDU M	100	K_g
MJU M	5.9581	K_g
B_p	400	
K	4000	N/M
β_e	6.85×10^8	Pa
MDU F_l	250	N
MJU F_l	0	N
K_1	1.25×10^{-3}	
N_p	50	
N_c	4	

After solving the Eq. (34), the input control signal of upper MDU is the first value of \mathbf{u} , and the input control signal of lower MDU is the second value of \mathbf{u} .

4. Simulation analysis

In order to verify the control effectiveness of MPC-CCC controller, because the BRMR is under construction, simulations in heteromorphic (MJU-MDY) and isomorphic (MJU-MJY) circumstances of Mover mockup were implemented.

4.1. Lift cylinder of MJU and driving cylinder of MDU synchronization control simulation

The simulation was performed using MATLAB/Simulink software, with a BRMR dynamic system model that represents the process under control. The system is subject to reference signal changes, and the controllers were designed to regulate the system outputs to follow the desired set points then the effectiveness of MPC-CCC controller and PI controller can be obtained.

The whole simulation time is set to 20 s, the sample time is 0.01 s, which is identical to experiments. The key simulation parameters are shown in Table I.

On the basement of Eq. (34), MPC-CCC and PI controller simulation platforms were built in Simulink, as shown in Fig. 10. Then, the reference signal is set up as shown in Table II.

After the simulation, the displacements of the JC of MJU and the DC of MDU in different controllers are shown in Fig. 11. The displacements errors controlled by different controllers are shown in Figs. 12 and 13. The displacements synchronization errors can be seen in Fig. 14.

From Fig. 12, the response of the MPC-CCC controller approaches the target value faster than the PI controller. In the upward part of the reference signal, the time for the MPC-CCC controller to reach the target is close to 5.84 s, which is faster than the PI controller whose the fastest time is 7.53 s and slowest time is 8.96 s. The biggest time delay is up to about 3 s. In terms of steady-state error, the response of PI overshoots after reaching the target, and the overshoot increases gradually with time until it reaches the maximum value (0.03 mm). In the falling part of the reference signal, the MPC-CCC controller also has a faster response and less overshoot than the PI controller. The response of the MPC-CCC controller reaches the target value of about 16.89 s, which is faster than the PI controller (19.45 s). The biggest

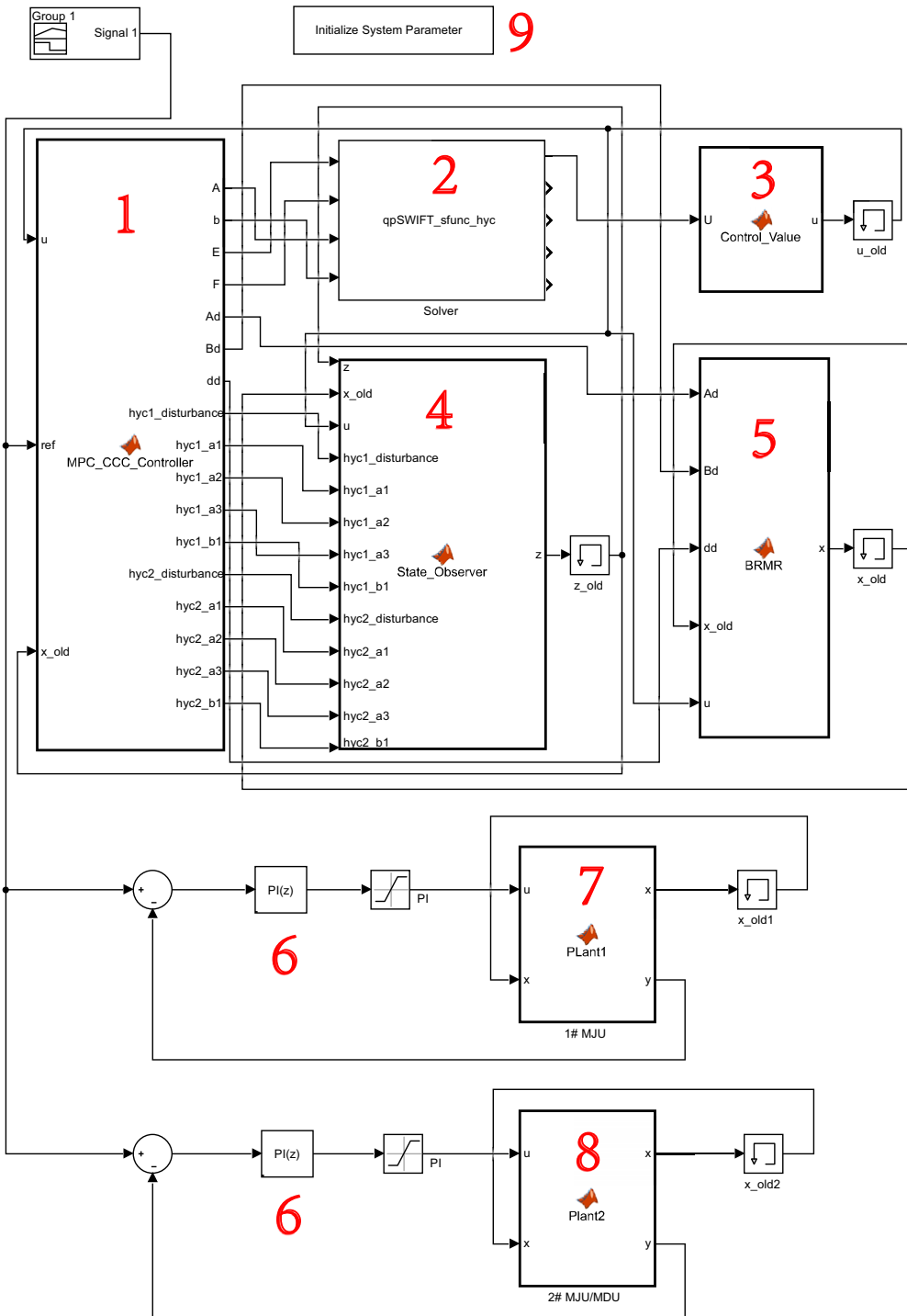


Figure 10. Simulation platform for MPC-CCC and PI controllers 1-MPC-CCC controller; 2-qpSWIFT; 3-Control value extractor; 4-State observer; 5-BRMR; 6-PI controller; 7-MJU; and 8-MJU/MDU.

Table II. Ramp reference signal.

Parameters	Values	Units	
y_1	$0.004t,$	$t \leq 5$	m
	$0.02,$	$5 < t \leq 11$	
	$-0.004(t - 11)t + 0.02,$	$11 < t \leq 16$	
	$0,$	$16 < t$	

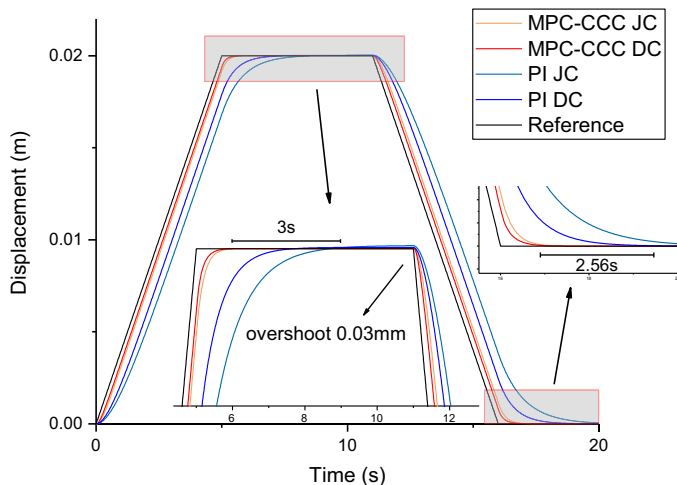


Figure 11. Displacements synchronization control in different controllers.

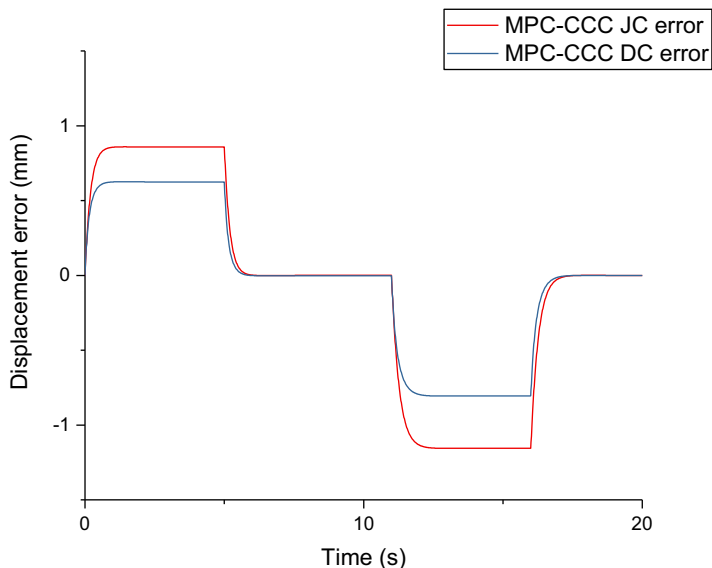


Figure 12. Displacements error in MPC-CCC controllers.

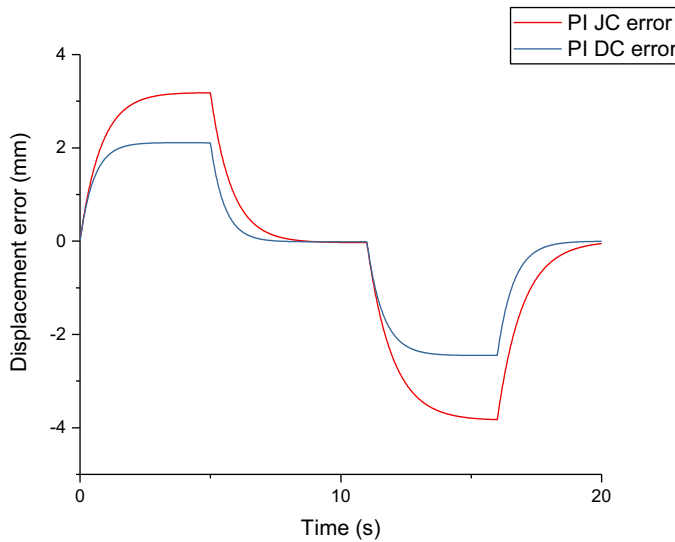


Figure 13. Displacements error in PI controllers.

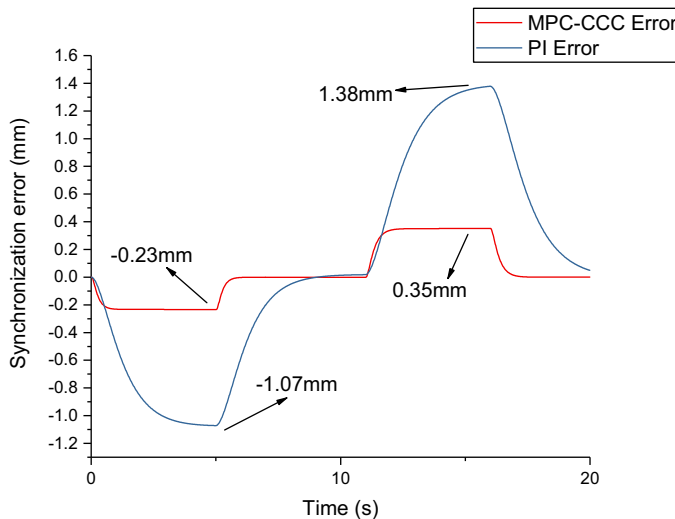


Figure 14. Displacements synchronization errors in different controllers.

time delay between two controllers is up to about 2.56 s. After the response of the PI controller reaches the target, there is still an overshoot, while the overshoot of the MPC controller is close to 0. In Figs. 12 and 13, the maximum displacement error of the MPC-CCC controller is about 3.5 times the PI controller. From Fig. 14, in the upward part of the reference signal, the maximum synchronization error of MPC is -0.23 mm, and the PI's is -1.07 mm which is about 4.7 times MPC's. In the falling part of the reference signal, the maximum synchronization error of MPC is 0.35 mm, and the PI's is 1.38 mm which is about 3.9 times MPC's.

From the simulation results, we can see that the MPC-CCC can not only guarantee the quick convergence of displacement error of a single system such as MDU and MJU but also have a good convergence of synchronization error in multielement systems (BRMR). MPC-CCC controller outperforms the PI controller in terms of response speed and steady-state error even in the condition of the reference signal changes drastically. The MPC-CCC controller can achieve the optimal control effect in both the

Table III. Key simulation parameters.

Parameters	Values	Units
JC diameter with rodless cavity	100	mm
JC diameter with rod cavity	55	mm
JC stroke	0.03	m
M of 1# JC	5.9581	K_g
M of 4# JC	505.9581	K_g
B_p	400	
K	4000	N/M
β_e	6.85×10^8	Pa
F_l of 1# JC	59.581	N
F_l of 4# JC	5059.581	N

Table IV. Sin reference signal.

Parameters	Values	Units
y_2	$0.01, \quad 0 \leq t < 5$	m
	$0.001(t - 5) + 0.01, \quad 5 \leq t < 10$	
	$0.005 \sin(\frac{\pi}{20}(t - 10)) + 0.015, \quad 10 \leq t \leq 50$	
	$0.015, \quad t \leq 55$	

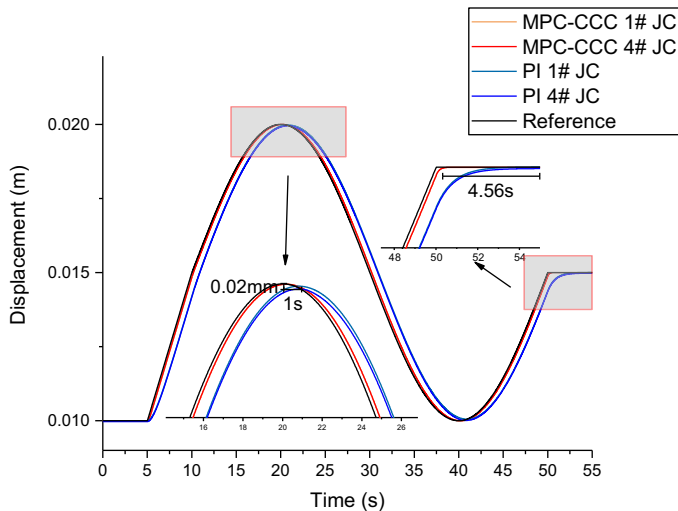


Figure 15. Displacements synchronization control in different controllers.

dynamic tracking error, steady-state error, and synchronization error because of the characteristic of the MPC model-based predictive control algorithm and the continuous optimization of the cross-coupling term in the thought of CCC algorithm under constraints. From Fig. 12, the simulation results have shown that the maximum dynamic tracking error and the maximum steady-state error of this studied MPC controller are $-1.16 \sim 0.86$ and 1.8×10^{-3} mm, respectively, which meet the actual technical accuracy requirements of ± 5 and ± 2 mm, respectively. From Fig. 14 we can see the maximum displacement synchronization error is 0.35 mm, which meets the actual technical accuracy requirements of ± 5 mm.

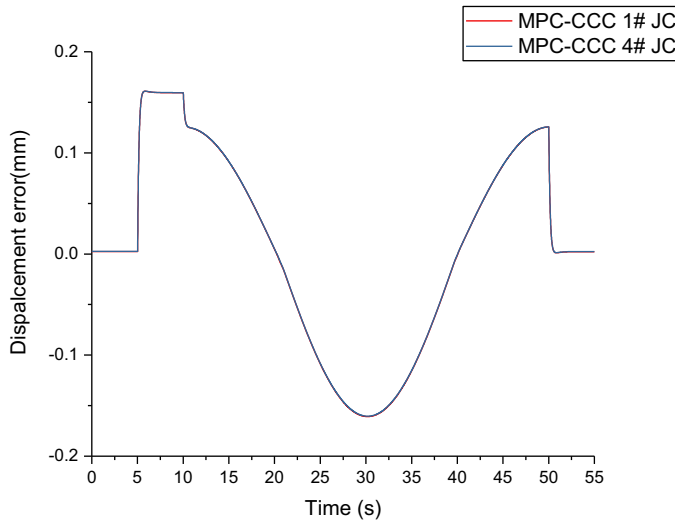


Figure 16. Displacements error in MPC-CCC controllers.

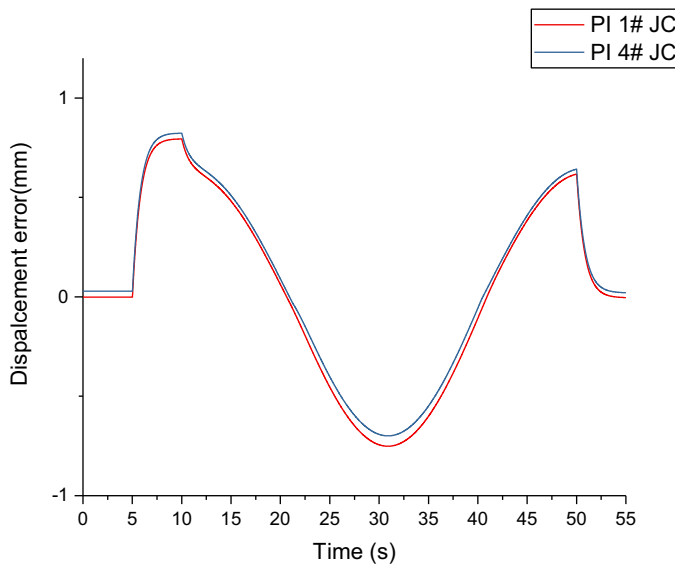


Figure 17. Displacements error in PI controllers.

4.2. JCs of MJU synchronization control simulation

The simulation data is shown in Table III. Noted that the 1# JC with no load and 4# JC with 500 Kg load as shown in Table III. The reference is sin signal as shown in Table IV. The displacements of JCs of MJU in different controllers are shown in Fig. 15. the displacements errors controlled by different controllers are shown in Figs. 16 and 17. The displacements synchronization errors can be seen in Fig. 18.

From the simulation results, taking the reference signal in Table IV, the response of the MPC-CCC controller is much faster than the PI controller. The PI response has a response delay of about 1 s and 0.02 mm displacement error at about time 21 s. In the same way, at the peak of the sin function, the response delay of the MPC-CCC controller is about 0 and the displacement error is about -0.0049 mm. In the displacement error, PI is about 4 times MPC-CCC. At end of the sin signal, PI response has about a 4.56 s delay compared with MPC-CCC. In Figs. 16 and 17, the MPC-CCC error range is

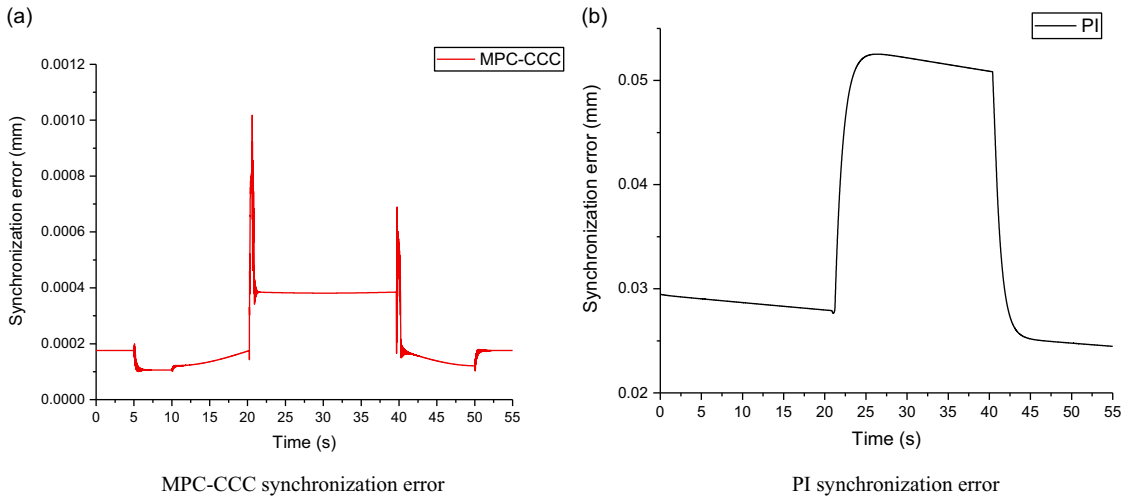


Figure 18. Displacements synchronization errors in different controllers.

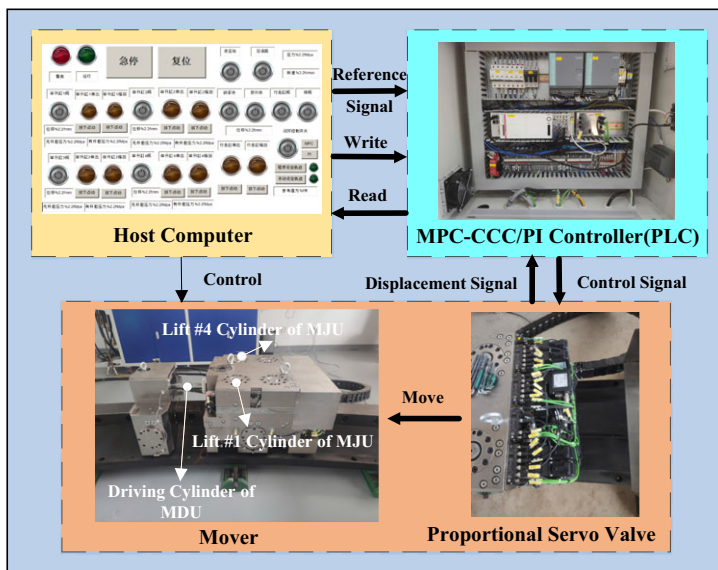


Figure 19. BRMR experiment platform.

about -0.16 to 0.16 mm. And PI error range is about -0.82 to 0.75 mm. The displacement error of PI is about 5 times MPC-CCC. In Fig. 18 the maximum synchronization error of PI is much bigger than MPC-CCC. In general, because of the control value optimization and cross-coupling error decrease made by CCC, the synchronization error of MPC-CCC is in a lower lever compared with PI. In this isomorphic condition, the synchronization error, dynamic, and steady-state error of MPC-CCC also satisfy actual displacement accuracy.

5. Experimental verification

5.1. JC of MJU and DC of MDU synchronization control experiment

The structure of the BRMR experiment platform is shown in Fig. 19. The synchronous experiments were realized by PLC. The experiment parameters are identical to the corresponding simulation in

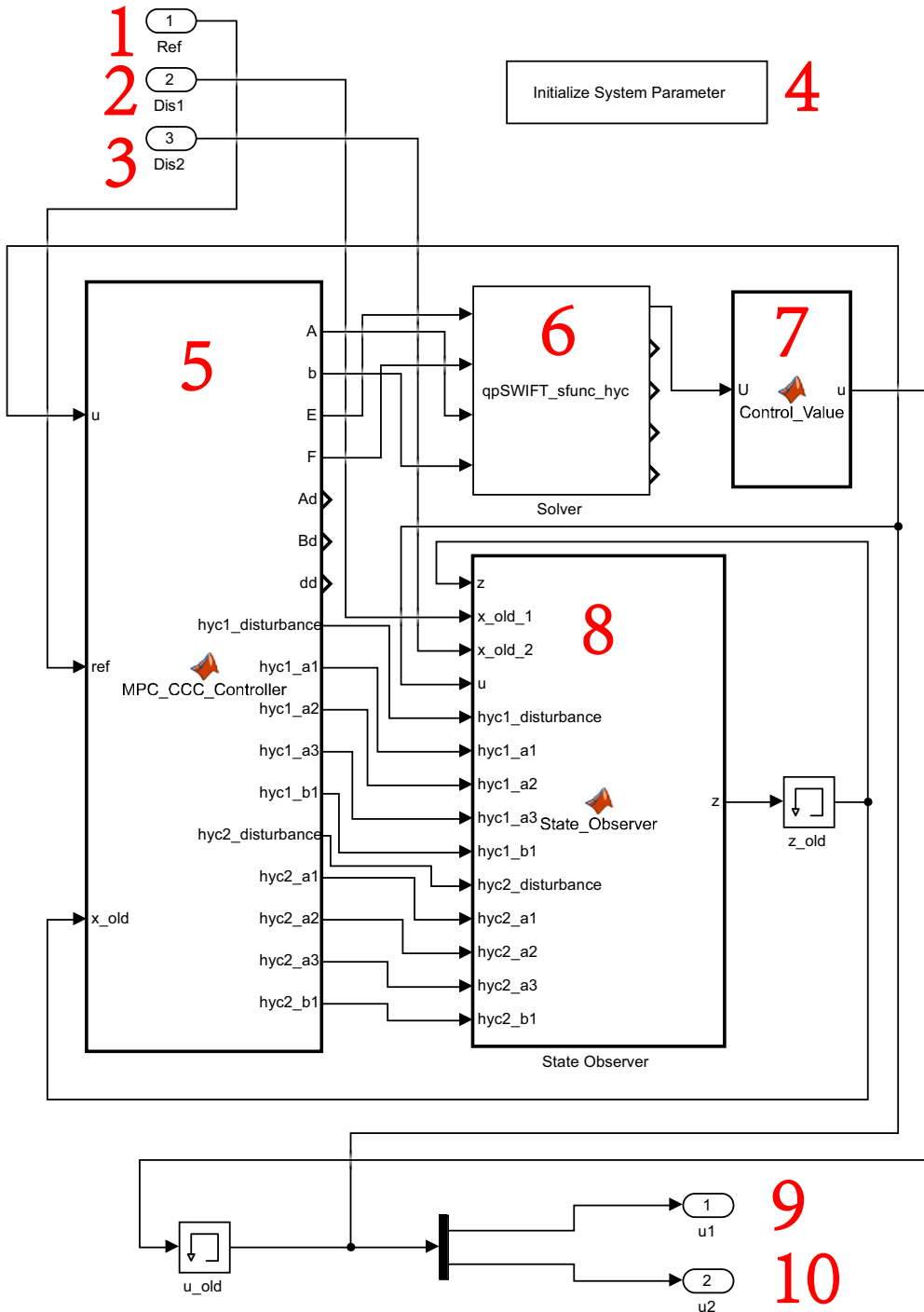


Figure 20. MPC-CCC controller graph in PLC.
 1-Reference signal input; 2-Displacement feedback signal input of first cylinder; 3-Displacement feedback signal input of second cylinder; 4-Initialize system parameter; 5-MPC-CCC controller; 6-qp-SWIFT; 7-Control value extractor; 8-State observer; 9-Control signal output of first cylinder; 10-Control signal output of second cylinder.

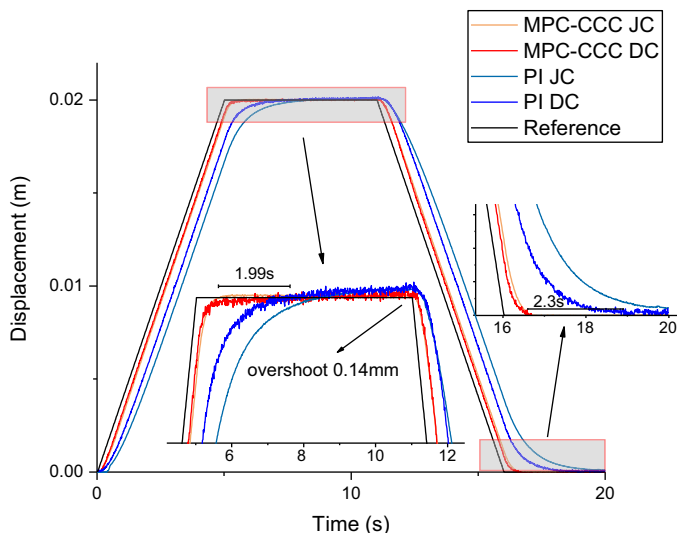


Figure 21. Displacements synchronization control in different controllers.

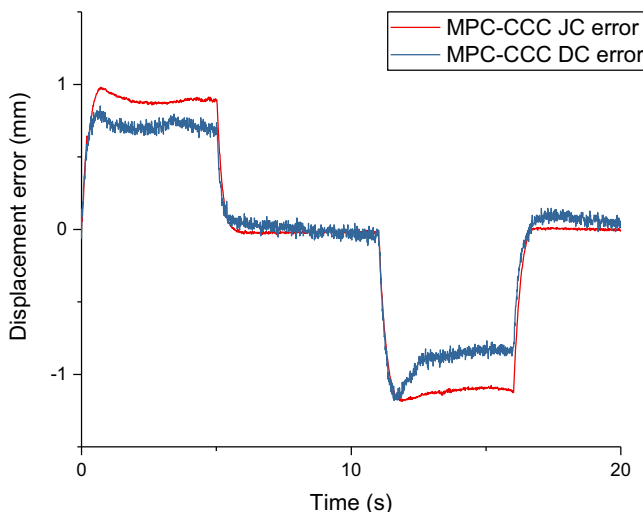


Figure 22. Displacements error in MPC-CCC controllers.

Table I. The conference signals are identical to Table II. In this experiment, PLC is different from MATLAB/Simulink environment, so the MPC-CCC controller graph is arranged as shown in Fig. 20. After the experiment, the displacements synchronization control in different controllers is shown in Fig. 21. The displacements error in different controllers is shown in Figs. 22 and 23. Displacement synchronization errors in different controllers are shown in Fig. 24.

In Fig. 21, PI still has a time delay of up to 1.99 s to reach the reference signal compared with MPC-CCC in the upward part of the signal. And PI still has a maximum overshoot of about 0.14 mm at time 11 s. On the contrary, MPC-CCC’s overshoot is nearly 0. In the downward direction, PI has about a 2.3 s time delay to reach 0 compared with MPC-CCC. The experiment displacement response of MPC-CCC is identical to the response in the simulation shown in Section 4.1. In Fig. 22, the maximum displacement error of MPC-CCC is $-1.18 \text{ mm} \sim 0.98 \text{ mm}$ which is much smaller than $-3.92 \text{ mm} \sim 3.29 \text{ mm}$ of PI as

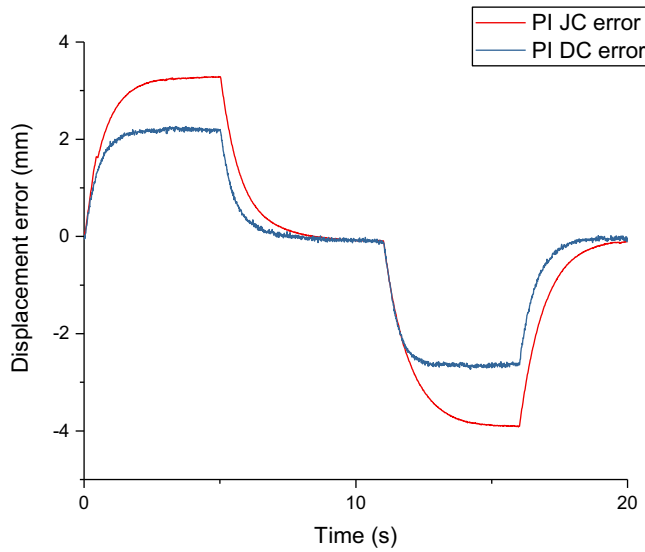


Figure 23. Displacements error in PI controllers.

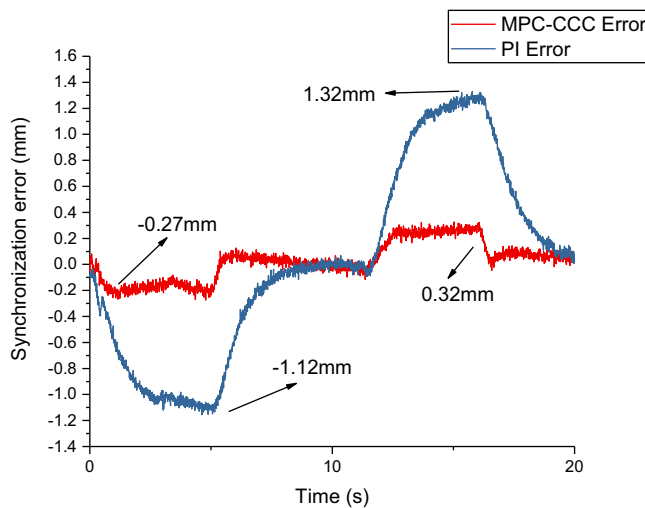


Figure 24. Displacements synchronization errors in different controllers.

shown in Fig. 23. From Fig. 24, the error range of PI is $-1.12\text{mm} \sim 1.32\text{mm}$, the error range of MPC-CCC is $-0.27\text{mm} \sim 0.32\text{mm}$, both of which is identical with the simulation. The experiment results verify the simulation and prove that the MPC-CCC control algorithm has better control effectiveness than PI.

5.2. JCs of MJU synchronization control experiment

The experiment parameters are identical to the corresponding simulation in Table III. The conference signals are identical to Table IV. The structure of the controller is the same as in Fig. 20. In the JCs of MJU synchronization control, the #1 JC of MJU has no load, but the #4 JC of MJU has a 500 kg load. The

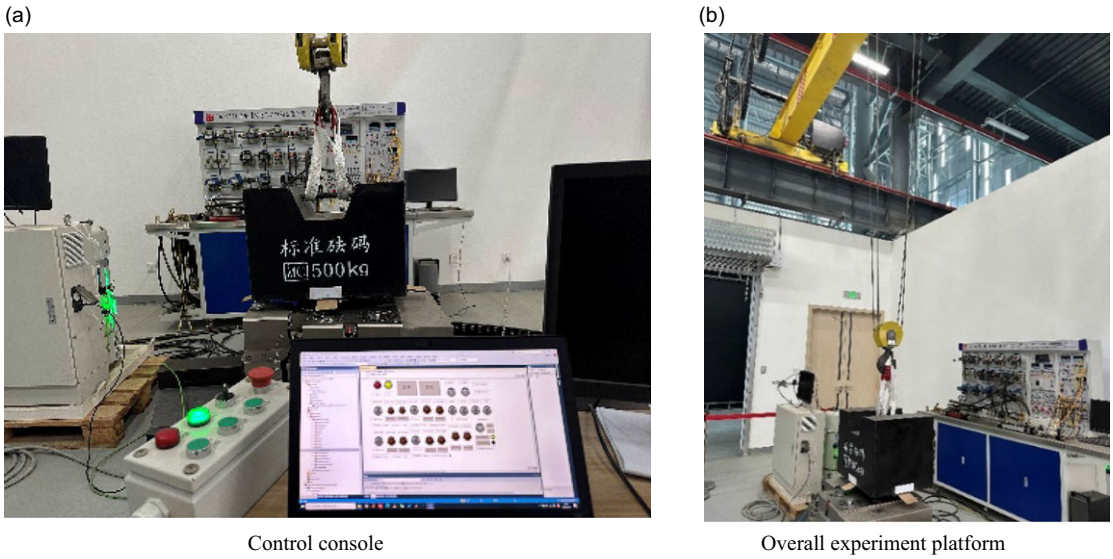


Figure 25. Experiment process diagram.

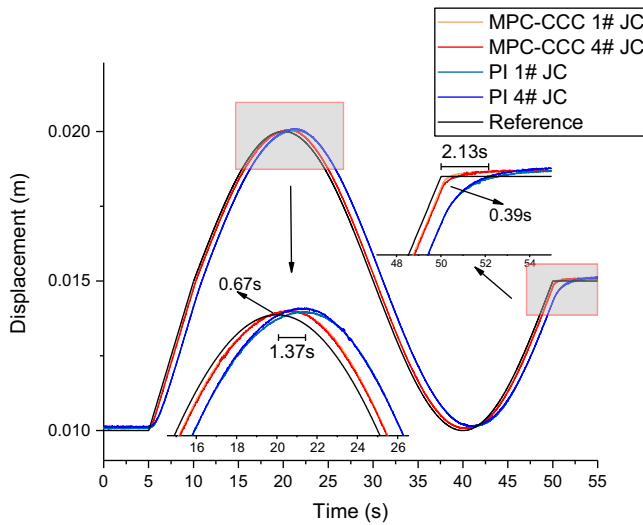


Figure 26. Displacements synchronization control in different controllers.

load was preheated for safety reasons (the sling of the crane is loose), and the experiment process pictures can be seen in Fig. 25. The displacement synchronization control in different controllers is shown in Fig. 26. The displacements error in different controllers is shown in Figs. 27 and 28. Displacement synchronization errors in different controllers are shown in Fig. 29.

In Fig. 26, at top of the reference signal, MPC-CCC delayed 0.67 s to the reference signal, whereas PI delayed 1.37 s, and MPC-CCC's response was 0.7 s faster than PI's. At the end of the signal, MPC-CCC's time delay is about 0.39 s, and PI's time delay is about 2.13 s. The trend is identical to the simulation. From comparing with Figs. 27 and 28, the MPC-CCC's error range is $-0.26 \text{ mm} \sim 0.27 \text{ mm}$ is smaller than $-0.88 \text{ mm} \sim 0.87 \text{ mm}$, and the error of PI is about 3 times MPC-CCC. The trend is also identical

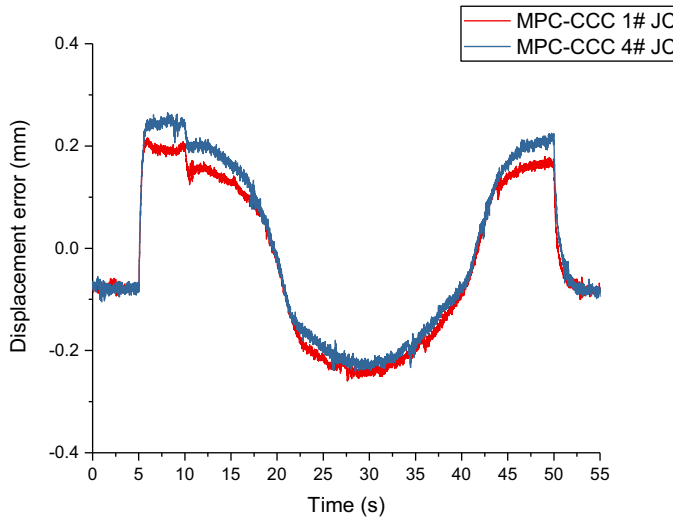


Figure 27. Displacements error in MPC-CCC controllers.

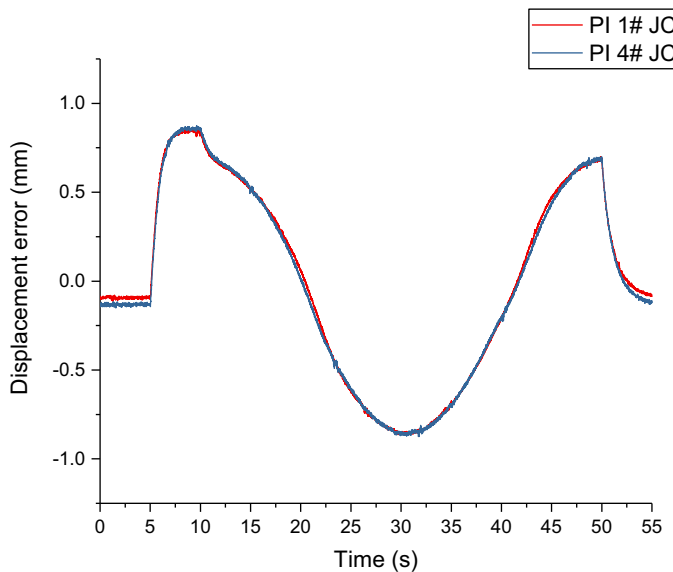


Figure 28. Displacements error in PI controllers.

to the simulation result. In Fig. 29, we can see that there are fluctuations from about time 5 and 42.5 s, respectively, in accordance with Figs. 26, 27 and 28, that is because the #4 cylinder is not completely identical with 1# cylinder of MJU, the #4 cylinder is not usually used leading to a different friction model and the mess of payload also has an error. That is a systemic error. But the displacement error trend is identical between MPC-CCC and PI, and it can also prove the effectiveness of the MPC-CCC controller. In the same synchronization error range of 0.01 mm, the dynamic tracking error of PI is about 3 times MPC-CCC. The control effectiveness of MPC-CCC is verified in actual operation. In the end, the steady-state error, dynamic tracking error, and synchronization error under the control of the MPC-CCC algorithm meet the actual technical accuracy requirements of ± 2 , ± 5 , and ± 5 mm, respectively.

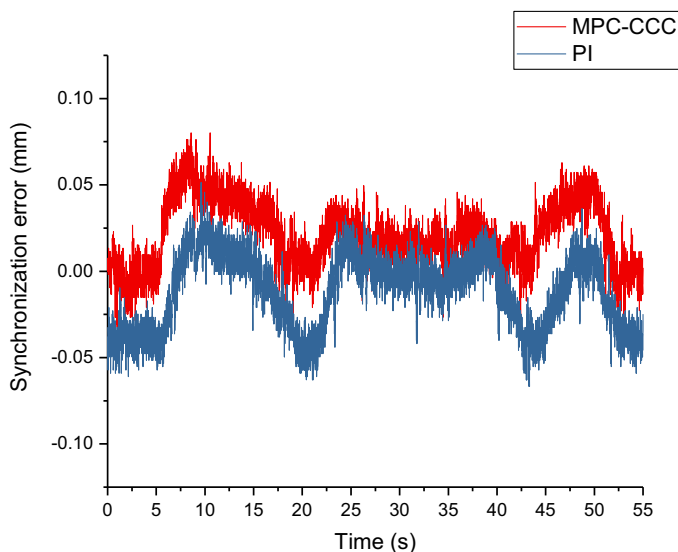


Figure 29. Displacements synchronization errors in different controllers.

6. Conclusion

This study aims to realize the high accuracy, stability, and high-efficiency transport of blankets through BRMR by using an innovative MPC-CCC method. First, the general state space mathematical model of BRMR is derived according to the physical-based method which includes two Movers' hydraulic systems. Second, with the advantage of control value receding horizon optimization based on the model prediction output in the MPC algorithm and the advantage of fast convergence of the cross-coupling displacement error in the CCC algorithm the innovative MPC-CCC controller was proposed in this paper. The main idea of the controller is the cost function establishment which includes the model predictive optimization in MPC and the cross-coupling elimination term in CCC. Through the optimization of both terms, the controller can realize the convergence of single-system displacement error and multisystem synchronization error. This attempt is rarely used in the hydraulic system that needs high synchronization accuracy in the actual project. Third, to verify the BRMR synchronization control effectiveness of the MPC-CCC controller, because of experiment platform limits, the JC of MJU – DC of MDU and JC–JC of MJU were used for the simulation and experiments. And the comparison between MPC-CCC and the original PI used in Mover is obtained in simulation and experiments. From the simulation and experiment results in different reference signals, we can see that the results of the simulation and experiment are highly consistent. The displacement error and synchronization error of PI are 2.22 ~ 3.33 times and 1.06 ~ 2.44 times to MPC-CCC in experiments. MPC-CCC controller can reduce the displacement error and synchronization error by up to 70% and 59% compared to PI, respectively. And the accuracy of MPC-CCC satisfies the real requirement that the dynamic tracking accuracy of the displacement is ± 5 mm, the steady-state accuracy of the displacement is ± 2 mm, and the dynamic tracking accuracy of the synchronous displacement is ± 5 mm.

Despite the strengths of this study, there are some limitations to consider. For example, due to the small difference in friction between #1 and #4 hydraulic cylinders of MJU, these two cylinders are not entirely identical leading to fluctuations in the reference signal turning points. Nevertheless, the results of this study provide valuable insights into the synchronization control in BRMR. This work has also provided a theoretical basis and practical experience for the future applications of the MPC-CCC control algorithm in the synchronous control of the BRMR and the synchronous transportation of the blanket.

Looking forward, future research could enhance these findings and explore better performance in MPC-CCC controller by considering nonlinear parts to optimize the mathematical model.

Author contribution. Dongyi Li: conceptualization, data curation, formal analysis, investigation, methodology, software, validation, visualization, and writing – original article. Kun Lu and Yong Cheng: conceptualization, funding acquisition, project administration, resources, supervision, visualization and writing – review and editing. Huapeng Wu and Heikki Handroos: conceptualization, methodology, supervision, visualization, and writing – review and editing. Xuanchen Zhang: methodology and visualization. Xinpeng Guo, Songzhu Yang, Liansheng Du, and Yu Zhang: data curation, visualization.

Financial support. This work was supported by the National Key R&D Program of China with Grant No. 2017YFE0300503, Comprehensive Research Facility for Fusion Technology Program of China under Contract No. 2018-000052-73-01-001228, and Anhui Extreme Environment Robot Engineering Laboratory and the China Scholarship Council with No. 202206340050.

Competing interests. The authors declare no Competing interests exist.

Ethical approval. Not applicable.

References

- [1] D. Li, K. Lu, Y. Cheng, W. Zhao, S. Yang, Y. Zhang, J. Li and S. Shi, “Dynamic analysis of multi-functional maintenance platform based on Newton-Euler method and improved virtual work principle,” *Nucl. Eng. Technol.* **52**(11), 2630–2637 (2020).
- [2] X. Yu, W. He, Q. Li, Y. Li and B. Li, “Human-robot co-carrying using visual and force sensing,” *IEEE Trans. Ind. Electron.* **68**(9), 8657–8666 (2021).
- [3] X. Yu, B. Li, W. He, Y. Feng, L. Cheng and C. Silvestre, “Adaptive-constrained impedance control for human-robot co-transportation,” *IEEE Trans. Cybern.* **52**(12), 13237–13249 (2021).
- [4] A. Alleyne, F. Allgöwer, A. Ames, S. Amin, J. Anderson, A. Annaswamy, P. Antsaklis, N. Bagheri, H. Balakrishnan and B. Bamieh, “Control for Societal-Scale Challenges: Road Map 2030,” *2022 IEEE CSS Workshop on Control for Societal-Scale Challenges* (2023) pp. 114–118.
- [5] Q. Zhu, J. Wang and G. Xie, “Review of composite adaptive control for servo system,” *Aeronaut. Manuf. Technol.* **64**(22), 14–27 (2021).
- [6] G. B. Avanzini, A. M. Zanchettin and P. Rocco, “Constrained model predictive control for mobile robotic manipulators,” *Robotica* **36**(1), 19–38 (2018).
- [7] M. Yao and M. Zhao, “Unmanned aerial vehicle dynamic path planning in an uncertain environment,” *Robotica* **33**(3), 611–621 (2015).
- [8] T. P. Nascimento, C. E. Dórea and L. M. G. Gonçalves, “Nonholonomic mobile robots’ trajectory tracking model predictive control: A survey,” *Robotica* **36**(5), 676–696 (2018).
- [9] T. Ouyang, Y. Lu, L. Cheng and J. Wang, “Controller design for electro-hydraulic actuator of heavy-duty automatic transmission using model predictive control algorithm,” *IEEE Trans. Transp. Electrification* **1**–1 (2023) <https://doi.org/10.1109/TTE.2023.3249164>.
- [10] J. T. Jose, J. Das and S. K. Mishra, “Dynamic improvement of hydraulic excavator using pressure feedback and gain scheduled model predictive control,” *IEEE Sens. J.* **21**(17), 18526–18534 (2021).
- [11] B. Cho, S.-W. Kim, S. Shin, J.-H. Oh, H.-S. Park and H.-W. Park, “Energy-efficient hydraulic pump control for legged robots using model predictive control,” *IEEE/ASME Trans. Mech.* **28**(1), (2022).
- [12] B. Varga, S. Meier, S. Schwab and S. Hohmann, “Model Predictive Control and Trajectory Optimization of Large Vehicle-Manipulators,” *IEEE International Conference on Mechatronics (ICM)*, vol. 12019 (2019) pp. 60–66.
- [13] Q. Shi and L. He, “A model predictive control approach for electro-hydraulic braking by wire,” *IEEE Trans. Ind. Inform.* **19**(2), 1380–1388 (2022).
- [14] M. Mei, S. Cheng, H. Mu, Y. Pei and B. Li, “Switchable MPC-based multi-objective regenerative brake control via flow regulation for electric vehicles,” *Front. Rob. AI* **10** (2023). <https://doi.org/10.3389/frobt.2023.1078253>.
- [15] K. Heybroek and J. Sjöberg, “Model predictive control of a hydraulic multichamber actuator: A feasibility study,” *IEEE/ASME Trans. Mech.* **23**(3), 1393–1403 (2018).
- [16] X. Zeng, G. Li, G. Yin, D. Song, S. Li and N. Yang, “Model predictive control-based dynamic coordinate strategy for hydraulic hub-motor auxiliary system of a heavy commercial vehicle,” *Mech. Syst. Signal Process.* **101**, 97–120 (2018).
- [17] F. A. Bender, S. Göltz, T. Bräunl and O. Sawodny, “Modeling and offset-free model predictive control of a hydraulic mini excavator,” *IEEE Trans. Autom. Sci. Eng.* **14**(4), 1682–1694 (2017).
- [18] O. Dahunsi, J. Pedro and O. Nyandoro, “Neural network-based model predictive control of a servo-hydraulic vehicle suspension system,” *In: Proceedings of IEEE AFRICON 2009*, Nairobi Kenya (2009). <https://doi.org/10.1109/AFRCON.2009.5308111>.
- [19] J. Kalmari, J. Backman and A. Visala, “Nonlinear model predictive control of hydraulic forestry crane with automatic sway damping,” *Comput. Electron. Agric.* **109**, 36–45 (2014).
- [20] X. Yang, X. Wang, S. Wang, K. Wang and M. B. Sial, “Finite-time adaptive dynamic surface synchronization control for dual-motor servo systems with backlash and time-varying uncertainties,” *ISA Trans.* **137**, 248–262 (2022) <https://doi.org/10.1016/j.isatra.2022.12.013>.

- [21] D. Sun, X. Shao and G. Feng, “A model-free cross-coupled control for position synchronization of multi-axis motions: Theory and experiments,” *IEEE Trans. Control Syst. Technol.* **15**(2), 306–314 (2007).
- [22] H. Yuan and X. Zhao, “A novel precision synchronization control via adaptive jerk control with parameter estimation for gantry servo system,” *Int. J. Control Autom. Syst.* **21**(1), 188–200 (2023).
- [23] Z. Kuang, H. Gao and M. Tomizuka, “Precise linear-motor synchronization control via cross-coupled second-order discrete-time fractional-order sliding mode,” *IEEE/ASME Trans. Mech.* **26**(1), 358–368 (2020).
- [24] G. Han, Z. Lu, J. Hong, M. Wu, S. Xu and B. Zhu, “Speed synchronization control of dual-SRM drive with ISMC-based cross-coupling control strategy,” *IEEE Trans. Transp. Electrification* **9**(2), 2524–2534 (2022).
- [25] S. Zou, W. Zhao, C. Wang, W. Liang and F. Chen, “Tracking and synchronization control strategy of vehicle dual-motor steer-by-wire system via active disturbance rejection control,” *IEEE/ASME Trans. Mech.* **28**(1), 92–103 (2022).
- [26] <https://github.com/qpSWIFT/qpSWIFT>.
- [27] A. G. Pandala, Y. Ding and H.-W. Park, “qpSWIFT: A real-time sparse quadratic program solver for robotic applications,” *IEEE Rob. Autom. Lett.* **4**(4), 3355–3362 (2019).
- [28] D. Li, K. Lu, Y. Cheng, W. Zhao, S. Yang, Y. Zhang, J. Li and H. Wu, “Fuzzy-PID controller for motion control of CFETR multi-functional maintenance platform,” *Nucl. Eng. Technol.* **53**(7), 2251–2260 (2021).