of the HPPM is key important. Many scholars have been studied it in depth. In [5], a reduced dynamic model of a five-bar linkage mechanism is proposed, and a PD plus full gravity compensation control strategies is designed. In [6], a computed-torque control method for trajectory tracking of the HPPM is investigated. In [7] and [8], a sliding model control strategy was employed to drive the HPPM to follow a predesigned trajectory. Iterative learning control (ILC) method originated in the study of industrial robot manipulators, which repeats the same task from trial to trial. ILC is a technique for improve trajectory tracking performance of the system that executes a same operation repeatedly. Through decades of development, ILC has been widely applied in the control field, which robot and other difficult to establish accurately model[9]. The natural of repetitive motion and nonlinear feature of the HPPM makes it possible to apply ILC to improve the tracking performance from iteration to iteration.

In this investigation it will be focused on dynamic simulation of the HPPM based on MATLAB/ SimMechanics and Traditional PD control and closed loop PD-type iterative learning control for trajectory tracking of end-effector of the HPPM.

2. System description and Modeling

The design of the 2-DOF HPPM follows a built-up modular system as illustrated in Fig.1. The HPPM is composed of five-bar linkage mechanism.

A simple schematic of the HPPM representing the coordinate systems is shown in Fig. 2. Link *AB* which is driven by CV motor and link *DE* which is driven by servomotor are two driving links. Link *BC* and link *CD* are driven by the two driving links, so that the end-effector can output the flexible trajectory. This paper assumes that the length of links are l_1, l_2, l_3, l_4, l_5, the angel between links and X-axis are $\theta_1, \theta_2, \theta_3, \theta_4, 0^\circ$, the mass of links are m_1,m_2,m_3, m_4 , the rotational inertia of links are I_1, I_2, I_3, I_4.

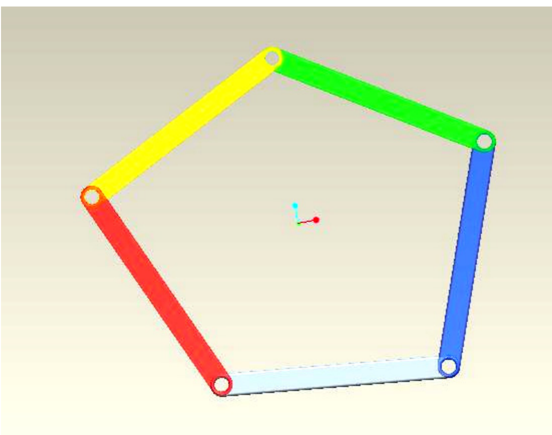

Figure 1. Model of the HPPM

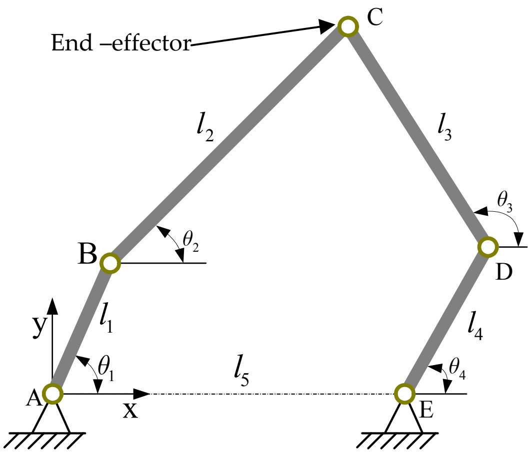

Figure 2. Schematic diagram of the HPPM

MATLAB/SimMechanics extends Simulink with the tools for modeling and simulating mechanical systems. SimMechanics is one of the physical modeling methods, which modeling and design for systems according to basic physical principals. A SimMechanics model includes a set of blocks (body, joint, constraint, coordinate, sensor and so on), which indicate physical components. In the module environment, SimMechanics uses rigid body and kinematic pair to describe the movement relationship between every rigid body. Dynamic SimMechanics model of the HPPM is shown in Fig.3.

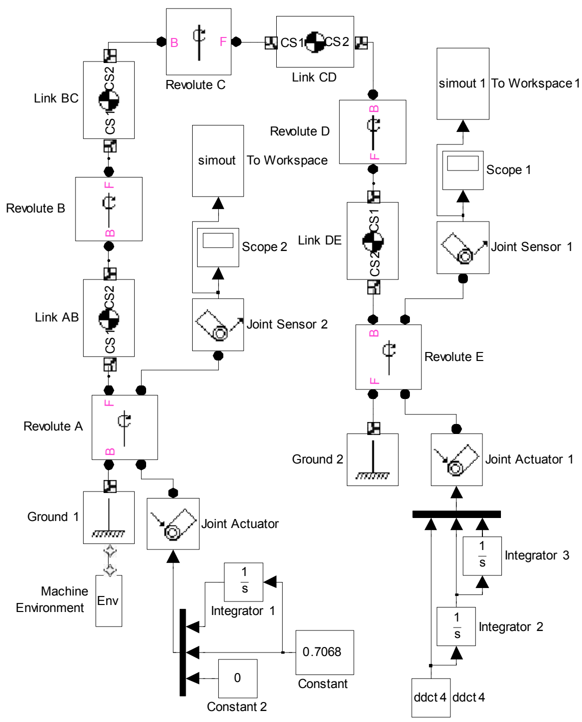

Figure 3. Dynamic simulation model of HPPM in SimMechanics

Initial position of links can be obtained by kinematic analysis of the HPPM. According to geometric relations of mechanism, as depicted in Fig.2, it can be derived the coordinates of point C:

$$x_c = l_1\cos\theta_1 + l_2\cos\theta_2 = l_5 + l_4\cos\theta_4 + l_3\cos\theta_3 \quad (1)$$

$$y_c = l_1\sin\theta_1 + l_2\sin\theta_2 = l_3\sin\theta_3 + l_4\sin\theta_4 \quad (2)$$

From Eqs. (1) and (2), it can be found that θ_1 and θ_4 are independent in the system, and θ_2 and θ_3 can be determined by θ_1 and θ_4 as follows:

$$\theta_3 = 2\arctan\left[\frac{A \pm \sqrt{A^2 + B^2 - C^2}}{B - C}\right] \quad (3)$$

where

$$\begin{aligned} A &= 2l_3l_4\sin\theta_4 - 2l_1l_3\sin\theta_1 \\ B &= 2l_3l_5 - 2l_1l_3\cos\theta_1 + 2l_3l_4\cos\theta_4 \\ C &= l_1^2 - l_2^2 + l_3^2 + l_4^2 + l_5^2 - 2l_1l_4\sin\theta_1\sin\theta_4 \\ &\quad -2l_1l_5\cos\theta_1 + 2l_4l_5\cos\theta_4 - 2l_1l_4\cos\theta_1\cos\theta_4 \end{aligned}$$

From Eqs. (2) and (3) above, we get

$$\theta_2 = \arcsin\left[\frac{l_3\sin\theta_3 + l_4\sin\theta_4 - l_1\sin\theta_1}{l_2}\right] \quad (4)$$

Dynamic simulation model parameters include control parameters of kinematic pair and physical parameters of rigid body, which could be set by variables input. The initial variable parameters of rigid body and that of kinematic pair can be written in MATLAB m-file and imported in the workspace of MATLAB before dynamic simulation of the HPPM. If the structure parameters of the HPPM need to be changed, only need to change in the M file, there is no need to set parameters for every module in the model.

3. Trajectory tracking control

Based on the results in the [7] and consider the various errors and disturbances, dynamic equation of the HPPM can be described as follows:

$$M(\theta')\ddot{\theta} + C(\theta',\dot{\theta}')\dot{\theta} + G(\theta') = \tau - \tau_d \quad (5)$$

where

$$\theta = [\theta_1 \quad \theta_4]^{\mathrm{T}}\ ;\ \theta' = [\theta_1 \quad \theta_2 \quad \theta_3 \quad \theta_4]^{\mathrm{T}}$$
$$\dot{\theta} = [\dot{\theta}_1 \quad \dot{\theta}_4]^{\mathrm{T}}\ ;\ \dot{\theta}' = [\dot{\theta}_1 \quad \dot{\theta}_2 \quad \dot{\theta}_3 \quad \dot{\theta}_4]^{\mathrm{T}}$$

$M(\theta')$ is the symmetric positive definite inertia matrix; $C(\theta',\dot{\theta}')$ is the centrifugal force and Coriolis force matrix; $G(\theta')$ is the gravity matrix; $\tau = [\tau_1, \tau_4]^{\mathrm{T}}$ is the control torque, τ_1 is the torque input by the CV motor, τ_4 is the torque input by the servomotor; τ_d is the error and disturbance. The elements in the matrix can be found in [7].

3.1 Traditional PD feedback control

The traditional PD control is widely used in control systems because of its simple algorithm and easy implementation. The PD control law for the HPPM is described by the following formulation:

$$\tau = K_d\dot{e} + K_P e \quad (6)$$

In the equation, the angular tracking error and angular velocity tracking error are defined as:

$$e = \theta_d - \theta\ ; \quad \dot{e} = \dot{\theta}_d - \dot{\theta} \quad (7)$$

Where θ_d is the desired angular position vector $[\theta_{1d} \quad \theta_{4d}]^{\mathrm{T}}$; $\dot{\theta}_d$ is the desired angular velocity vector $[\dot{\theta}_{1d} \quad \dot{\theta}_{4d}]^{\mathrm{T}}$; K_d is the derivative gain; K_P is the proportional gain.

The block diagram of traditional PD control for the HPPM is shown in Fig.4.

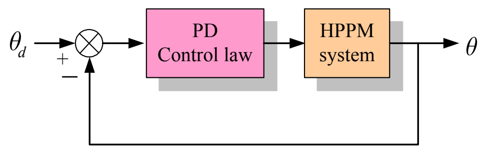

Figure 4. Control diagram of traditional PD for the HPPM

3.2 Iterative learning control

$y_d(t)$ is desired trajectory of the HPPM; $y_k(t)$ is output trajectory of the kth iterative operation; $e_k(t) = y_d(t) - y_k(t)$ is the trajectory tracking error. Iterative learning control objective for the HPPM is to make output trajectory track desired trajectory as closely as possible.

Consider the (k+1)th iterative operation for the system (5), the closed loop PD-type iterative learning control law is