Chinese Journal of Aeronautics, (2014),**27**(6): 1562–1570

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Fuzzy adaptive robust control for space robot considering the effect of the gravity

Qin Li, Liu Fucai *, Liang Lihuan, Gao Jingfang

Key Lab of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China

Received 26 November 2013; revised 28 April 2014; accepted 26 August 2014
Available online 18 October 2014

KEYWORDS

Fuzzy adaptive;
Microgravity;
Robustness;
Space robot;
Trajectory tracking control

Abstract Space robot is assembled and tested in gravity environment, and completes on-orbit service (OOS) in microgravity environment. The kinematic and dynamic characteristic of the robot will change with the variations of gravity in different working condition. Fully considering the change of kinematic and dynamic models caused by the change of gravity environment, a fuzzy adaptive robust control (FARC) strategy which is adaptive to these model variations is put forward for trajectory tracking control of space robot. A fuzzy algorithm is employed to approximate the nonlinear uncertainties in the model, adaptive laws of the parameters are constructed, and the approximation error is compensated by using a robust control algorithm. The stability of the control system is guaranteed based on the Lyapunov theory and the trajectory tracking control simulation is performed. The simulation results are compared with the proportional plus derivative (PD) controller, and the effectiveness to achieve better trajectory tracking performance under different gravity environment without changing the control parameters and the advantage of the proposed controller are verified.

1. Introduction

The concept of space robot system was first proposed by the United States in the 1970's. It aimed at performing the extravehicular activity (EVA) by the aid of robotic manipulator in extreme environment of the space, which is hard, heavy or dangerous for people to achieve. At present, space robot is mainly used in spacecraft, satellite and international space station (ISS), especially as a main part of ISS, it is playing a crucial role in the on-orbit assembly, external maintenance and the operations of ISS.[1] As is well-known, the design, assembly and test of space robot is completed under gravity condition, however, it services under microgravity condition finally. The changes in kinematic and dynamic behavior of space robot caused by the change of gravity environment are inevitable tough for the design and verification of the control system. In fact, this problem has been attracted wide attentions in the aerospace field. Various methods for simulated microgravity environment test have been developed to debug the controls parameter and validate the function on ground, including the air floatation, suspension and neutral buoyancy primarily, which are based on the principle of gravity balance. Ranger

* Corresponding author. Tel.: +86 335 8387556.
E-mail address: lfc@ysu.edu.cn (F. Liu).

Peer review under responsibility of Editorial Committee of CJA.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.cja.2014.10.023
1000-9361

test system developed by University of Maryland[2] is representative neutral buoyancy simulated test. It can realize three-dimension simulation, but the problem of water resistance cannot be solved, and it is high cost. Air floatation simulated test system provides a frictionless two-dimensional motion experiment environment. It is applied to formation flight by Stanford.[3] Suspension simulated test counterbalances the force of gravity by using gravity compensation equipment, which still belongs to the two-dimensional simulation method. It only applies to the deployment process of planar antenna or solar panel.[4] In order to ensure the on-orbit service performance of space robot, test about dynamic properties and effectiveness of controller were carried out in microgravity compared with the simulated microgravity environment. Manipulator flight demonstration (MFD) was launched in August 1997 on the space shuttle discovery. In Ref.[5] the flight experiment result was reported. The positioning errors differed from the ground test results was shown. It is presumed that these differences were due to that the ground simulated environment could not provide true microgravity, which may restrain the robot movement slightly and affect the tip position. More microgravity experiments at Tokyo Institute of Technology are introduced in Ref.[6]. The results of microgravity flight experiments show that the error of rotation position is within an acceptable range. However, there is considerable difference about the dynamic characteristic and control current compared with the ground data. In recent years, many combination simulation strategy based on the above simulated methods were proposed to eliminate the defects in simulated microgravity environment test,[7–10] but there are still problems of low fidelity and limited applicability. Given all above, the effect of gravity cannot be ignored, and cannot be solved only through the microgravity environment simulation test.

The realization of the space robot movement function depends on the control system. Therefore, the differences caused by the change of gravity environment should be taken into account when designing the controller. In particular, with the development of the space industry, space operation task is getting more and more diverse and complicate, followed by higher requirements for the accuracy and performance of the space robot system. The traditional method based on margin may bring too large output torque, it not only consume more propellant but also may degrade the positioning accuracy of manipulator terminal as well as the dynamic performance. Accordingly, this problem created by change of gravity environment should be taken on greater consideration because of the higher demand in accuracy and stability for complicated on-orbit servicing missions. For now, many dynamic analysis method and control algorithms have been proposed for trajectory tracking control of robotic system. In comprehensive consideration of flexibility of the gear tooth, mesh damping, clearance between gear teeth and mesh error, dynamics model of the large space manipulator was established in Ref. [11] by using the lumped parameter method. The concept of "system centroid equivalent manipulator" was proposed in Ref.[12]. Based on this concept, the multi-body dynamic model of a dual-arm space robotic system was developed. In Ref.[13] an adaptive control scheme was developed in the case of unknown inertial parameters for the tracking control of space robots with an attitude controlled base by combining the backstepping design approach and adaptive control theory. In Ref.[14] a new model-free control law, called proportional plus derivative (PD) with sliding mode control law was proposed for trajectory tracking control of multi-degree-of-freedom linear translational robotic systems. In Ref.[15] a control scheme with the help of a virtual space vehicle was presented for trajectory control of a two arm rigid-flexible space robot. Li and Chean[16] used a novel regional feedback method for robot task-space control, the feedback information is employed in a local region, and the combination of regional information ensures the global convergence of robot motion. Marco et al.[17] discussed the application of the image based visual serving strategy to space manipulators and experimental results are reported. However, most of these researches so far focus on the unconstrained base,[18] the uncertainty of model,[19] the external disturbances caused by orbital environment and the high flexibility of the links and control problem in capturing task.[20,21] However, there are few researches on the self-adaptation ability of controller for the model variations due to the gravitational differences.

This paper focus on the gravity effect on the kinematic and dynamic behavior of space robot, aims to solve the problem of parameter adjustment and verification on ground through designing of control scheme. That is to design a controller applied to different models of space robot both on ground and in space without changing the structure and parameters. A fuzzy adaptive robust control (FARC) strategy is proposed for the space robot system, which can control the space manipulator to achieve good trajectory tracking effect in different gravity environments without changing the structure and parameters. The uncertain nonlinear terms of the space manipulator dynamic model caused by the change of gravity is approximated by using the fuzzy system. An adaptive control law is designed to estimate parameters online. With the help of the robust controller, the estimation error is compensated. Then the stability of the closed-loop system is proved based on the Lyapunov principle. To verify the effectiveness of the proposed method, different controllers are adopted to control two models respectively. The simulation results show that the proposed fuzzy adaptive controller can achieve better trajectory tracking performance under different gravity environments.

2. System descriptions

In this paper, two conditions of the space manipulator are discussed, so it is necessary to establish the ground alignment model and the space application model. Fig. 1 shows the

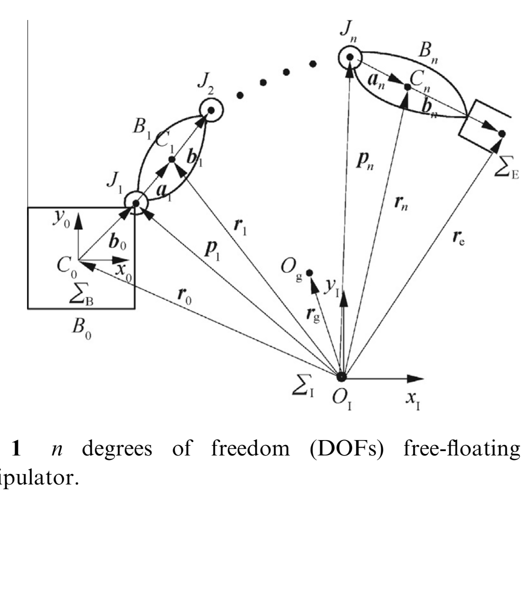


Fig. 1 n degrees of freedom (DOFs) free-floating space manipulator.

schematic diagram of n degrees of freedom (DOFs) free-floating space manipulator.

Where $B_i(i = 1, 2, \ldots, n)$ is the ist link-rod of manipulator, and B_0 represents the spacecraft platform; J_i $(i = 1, 2, \ldots, n)$ is the joint which connects B_{i-1} with B_i;$C_i(i = 0, 1, \ldots, n)$ is the mass center of B_i; $\boldsymbol{a}_i(i = 1, 2, \ldots, n)$ is the position vector from J_i to C_i; $\boldsymbol{b}_i(i = 0, 1, \ldots, n)$ is the position vector from C_i to J_{i+1}; $\boldsymbol{r}_i(i = 0, 1, \ldots, n)$ is the position vector of the mass center B_i; $\boldsymbol{r}_g$ is the unknown vector of the centroid of the system; $\boldsymbol{r}_e$ is the position vector of the end effector; $\boldsymbol{p}_i(i = 1, 2, \ldots, n)$ is the position vector of J_i; O_g is the centroid of the whole system; O_I is the inertial origin; $\Sigma_B - x_0\ C_0 y_0$ is the coordinate system of the base; $\Sigma_I - x_I\ O_I y_I$ is the inertial coordinate system; Σ_E is the coordinate system of the end effector. Besides, define that I_i is the inertia of B_i relative to its centroid and m_i is the mass of B_i.

The space manipulator in this paper assumes the following properties:

(1) The system is considered as a rigid system.
(2) In space the microgravity is ignored, and the system is in a free-floating condition. It is assumed that no external forces and torques applied on the system.
(3) The system consists of a base and several links. The pose of the base is not controlled actively, and every joint between links can rotate freely within a degree under active control.

2.1. Mathematical model on the ground

When the space manipulator is aligned on the ground, the base is fixed. According to the Lagrange formulation, the dynamic equation of the space robotic manipulator can be formulated as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau} \tag{1}$$

where $\boldsymbol{q} = [q_1, \quad q_2, \quad \ldots \quad , q_n]^{\mathrm{T}} \in \mathbf{R}^n$ is the vector of the joint variables; $\boldsymbol{M}(\boldsymbol{q}) \in \mathbf{R}^{n \times n}$ is the inertia matrix; $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbf{R}^{n \times n}$ is the vector of the coriolis and centrifugal forces; $\boldsymbol{G}(\boldsymbol{q}) \in \mathbf{R}^n$ is the vector of gravitational force and $\boldsymbol{\tau} = [\tau_1, \quad \tau_2, \quad \ldots \quad , \tau_n]^{\mathrm{T}} \in \mathbf{R}^n$ is the vector of joint actuator torques and forces.

Define the vector as $\boldsymbol{x}_1 = \boldsymbol{q},\ \boldsymbol{x}_2 = \dot{\boldsymbol{q}},\ \boldsymbol{x} = \left[\boldsymbol{x}_1^{\mathrm{T}}, \boldsymbol{x}_2^{\mathrm{T}}\right]^{\mathrm{T}}$, then the state equation of the system is

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = \boldsymbol{M}^{-1}(\boldsymbol{x}_1)(\boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{x}_2 - \boldsymbol{G}(\boldsymbol{x}_1)) \end{cases} \tag{2}$$

The output equation is

$$\boldsymbol{y} = \boldsymbol{x}_1 \tag{3}$$

Define

$$\begin{cases} \boldsymbol{U}(\boldsymbol{x}) = -\boldsymbol{M}^{-1}(\boldsymbol{x}_1)(\boldsymbol{C}(\boldsymbol{x}_1, \boldsymbol{x}_2)\boldsymbol{x}_2 + \boldsymbol{G}(\boldsymbol{x}_1)) \\ \boldsymbol{W}(\boldsymbol{x}) = \boldsymbol{M}^{-1}(\boldsymbol{x}_1) \end{cases}$$

then the model of space robot on the ground can be rewritten as

$$\ddot{\boldsymbol{y}} = \boldsymbol{U}(\boldsymbol{x}) + \boldsymbol{W}(\boldsymbol{x})\boldsymbol{\tau} \tag{4}$$

where

$$\begin{cases} \boldsymbol{U}(\boldsymbol{x}) = [u_1(\boldsymbol{x}), u_2(\boldsymbol{x}), \ldots, u_n(\boldsymbol{x})]^{\mathrm{T}} \\ \boldsymbol{W}(\boldsymbol{x}) = \begin{bmatrix} w_{11}(\boldsymbol{x}) & w_{12}(\boldsymbol{x}) & \ldots & w_{1n}(\boldsymbol{x}) \\ w_{21}(\boldsymbol{x}) & w_{22}(\boldsymbol{x}) & \ldots & w_{2n}(\boldsymbol{x}) \\ \vdots & \vdots & & \vdots \\ w_{n1}(\boldsymbol{x}) & w_{n2}(\boldsymbol{x}) & \ldots & w_{nn}(\boldsymbol{x}) \end{bmatrix} \end{cases}$$

and $\boldsymbol{W}(\boldsymbol{x})$ is a positive definite matrix and satisfies that $\boldsymbol{W}(\boldsymbol{x}) \geqslant \sigma \boldsymbol{I}_n$, where $\sigma > 0$ related to the system physical properties, and $\boldsymbol{I}_n$ is a unit matrix of n dimensions.

2.2. Mathematical model in space

When the space manipulator is in the on-orbit service stage, its base cannot be fixed for the microgravity environment. Therefore six additional DOFs will be added to the system. The attitude and position of the base will change since the manipulator will exert a force/torque on it when the manipulator moves or rotates. In this case, the Lagrange function is equal to the kinetic energy of the system. The dynamic equation of the free-floating space manipulator can be written as

$$\boldsymbol{M}_s(\boldsymbol{q}_s)\ddot{\boldsymbol{q}}_s + \boldsymbol{C}_s(\boldsymbol{q}_s, \dot{\boldsymbol{q}}_s)\dot{\boldsymbol{q}}_s = \boldsymbol{\tau}_s \tag{5}$$

where $\boldsymbol{M}_s(\boldsymbol{q}_s) \in \mathbf{R}^{(n+6)\times(n+6)}$ is the inertia matrix; $\boldsymbol{C}_s(\boldsymbol{q}_s, \dot{\boldsymbol{q}}_s) \in \mathbf{R}^{(n+6)\times(n+6)}$ is the vector of the coriolis and centrifugal forces; $\boldsymbol{q}_s = \left[\boldsymbol{q}_b^{\mathrm{T}}, \boldsymbol{q}_m^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbf{R}^{n+6}$ is the general position vector, with $\boldsymbol{q}_b = \left[\boldsymbol{p}_b^{\mathrm{T}}, \quad \boldsymbol{\omega}_b^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbf{R}^6$ the position and attitude vector of the base, $\boldsymbol{p}_b = [x_b, \quad y_b, \quad z_b]^{\mathrm{T}} \in \mathbf{R}^3$ (x_b,y_b and z_b are the displacements in the directions of x, y and z) the position vector and $\boldsymbol{\omega}_b = [\theta, \quad \omega, \quad \phi]^{\mathrm{T}} \in \mathbf{R}^3(\theta, \omega$ and ϕ are the displacements in the directions of roll, yaw and pitch) the attitude vector. $\boldsymbol{q}_m = [q_1, \quad q_2, \quad \ldots \quad , q_n]^{\mathrm{T}} \in \mathbf{R}^n$ the vector of the joint variables; $\boldsymbol{\tau}_s = \left[\mathbf{0}_{6\times1}^{\mathrm{T}}, \boldsymbol{\tau}_{n\times1}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbf{R}^{n+6}$ is the vector of joint actuator torques and forces, with $\mathbf{0}_{6\times1}$ the driving torque exerted on the base, and $\boldsymbol{\tau}_{n\times1}$ the joint torque of the manipulator.

Let $\boldsymbol{x}_{1s} = \boldsymbol{q}_s, \boldsymbol{x}_{2s} = \dot{\boldsymbol{q}}_s, \boldsymbol{x}_s = \left[\boldsymbol{x}_{1s}^{\mathrm{T}}, \boldsymbol{x}_{2s}^{\mathrm{T}}\right]^{\mathrm{T}}$, and the state equation of the system can be written as

$$\begin{cases} \dot{\boldsymbol{x}}_{1s} = \boldsymbol{x}_{2s} \\ \dot{\boldsymbol{x}}_{2s} = \boldsymbol{M}_s^{-1}(\boldsymbol{x}_{1s})(\boldsymbol{\tau}_s - \boldsymbol{C}_s(\boldsymbol{x}_{1s}, \boldsymbol{x}_{2s})\boldsymbol{x}_{2s}) \end{cases} \tag{6}$$

The output equation is

$$\boldsymbol{y}_s = \boldsymbol{x}_{1s} \tag{7}$$

Define that

$$\begin{cases} \boldsymbol{U}_s'(\boldsymbol{x}_s) = -\boldsymbol{M}_s^{-1}(\boldsymbol{x}_{1s})\boldsymbol{C}_s(\boldsymbol{x}_{1s}, \boldsymbol{x}_{2s})\boldsymbol{x}_{2s} \\ \boldsymbol{W}_s(\boldsymbol{x}_s) = \boldsymbol{M}_s^{-1}(\boldsymbol{x}_{1s}) \end{cases}$$

the model in the on-orbit operational process can be rewritten as

$$\ddot{\boldsymbol{y}}_s = \boldsymbol{U}_s'(\boldsymbol{x}_s) + \boldsymbol{W}_s(\boldsymbol{x}_s)\boldsymbol{\tau}_s \tag{8}$$

3. Fuzzy algorithm

Assuming that the fuzzy algorithm is a mapping from $\boldsymbol{U} \subseteq \mathbf{R}^n$ to $\mathbf{R}$ and the fuzzy knowledge base comprises a collection of IF–THEN rules in the following form:

$R^{(j)}$: IF x_1 is F_1^j and x_2 is F_2^j ... and x_n is F_n^j,
THEN y_f^j is C^j $(j = 1, 2, \ldots, m)$

where $\boldsymbol{x}_f = [x_1, x_2, \ldots, x_n]^T \in \boldsymbol{U}$ is the input of fuzzy logic system and $y_f \in \mathbf{R}$ is the output of fuzzy logic system . F_n^j is the fuzzy set of input and C^j is the fuzzy set of output. By using the singleton fuzzification, product inference engine and center average defuzzification, the output value of the fuzzy system can be expressed as

$$y_f(\boldsymbol{x}_f) = \frac{\sum_{j=1}^m \bar{y}_f^j \left(\prod_{i=1}^n \mu_{F_i^j}(x_i)\right)}{\sum_{j=1}^m \left(\prod_{i=1}^n \mu_{F_i^j}(x_i)\right)} \tag{9}$$

with

$$\bar{y}_f^j = \max_{y_f^j \in \mathbf{R}} \mu_{C^j}\left(y_f^j\right), \mu_{C^j}\left(\bar{y}_f^j\right) = 1$$

where $\mu_{F_i^j}$ is membership function of input, and μ_{C^j} is membership function of output. Define the fuzzy basis functions as

$$\xi_j(x_f) = \frac{\prod_{i=1}^n \mu_{F_i^j}(x_i)}{\sum_{j=1}^m \left(\prod_{i=1}^n \mu_{F_i^j}(x_i)\right)} \tag{10}$$

Then the fuzzy system can be rewritten as

$$y_f(x_f) = \boldsymbol{\xi}^T(x_f)\boldsymbol{\theta} \tag{11}$$

where $\boldsymbol{\xi}(\boldsymbol{x}_f) = [\xi_1(\boldsymbol{x}_f), \xi_2(\boldsymbol{x}_f), \ldots, \xi_m(\boldsymbol{x}_f)]^T$ is the regression vector; $\boldsymbol{\theta} = \left[\bar{y}_f^1, \bar{y}_f^2, \ldots, \bar{y}_f^m\right]^T$ is the parameter vector of the fuzzy system.

In this paper, fuzzy logic systems are employed to approximate the unknown nonlinear functions $u_i(\boldsymbol{x})$ and $w_{ij}(\boldsymbol{x})$ $(i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, n)$ and to design the adaptive control law.

4. Fuzzy adaptive robust controller design

The dynamic model of ground robot is different from that of free-floating robot because of the different gravity environment. However, all researches so far haven't design a controller performs well during the two stages from the perspective of different gravity environment. In view of the insufficiency of the existing control strategies, a fuzzy adaptive control strategy is proposed in this paper. The adaptive control law is designed by employing the fuzzy system to approximate the unknown nonlinear function online and the approximation errors are compensated by a robust controller.

4.1. Control algorithm design

When the space robot is aligned on the ground, its model is shown in Eq. (4). The control algorithm can be designed as follows. Define the following equations:

$$e(t) = q_d(t) - q(t) \tag{12}$$

$$r(t) = \dot{e}(t) + \boldsymbol{\Lambda} e(t) \tag{13}$$

$$\ddot{q}_r(t) = \ddot{q}_d(t) + \boldsymbol{\Lambda}\dot{e}(t) \tag{14}$$

where $\boldsymbol{e}(t)$ is the joint position tracking error; $\boldsymbol{q}_d(t)$ is the desired manipulator joint trajectory; $\boldsymbol{q}(t)$ is the actual manipulator joint trajectory; $\boldsymbol{r}(t)$ is a filtering error sliding surface; $\boldsymbol{\Lambda}$ is a positive definite diagonal matrix; $\boldsymbol{q}_r(t)$ is the reference trajectory.

Define $(\hat{\cdot})$ is the estimate value of $(\cdot)$, while $(\cdot^*)$is the optimal estimation.

Define that the fuzzy logic systems of the approximation system model nonlinear function $u_i(\boldsymbol{x})$ and $w_{ij}(\boldsymbol{x})$ are

$$\hat{u}_i(x, \hat{\boldsymbol{\theta}}_{u_i}) = \boldsymbol{\xi}_{u_i}^T(x)\hat{\boldsymbol{\theta}}_{u_i} \quad (i = 1, 2, \ldots, n) \tag{15}$$

$$\hat{w}_{ij}(x, \hat{\boldsymbol{\theta}}_{w_{ij}}) = \boldsymbol{\xi}_{w_{ij}}^T(x)\hat{\boldsymbol{\theta}}_{w_{ij}} \quad (i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, n) \tag{16}$$

where $\boldsymbol{\xi}_{u_i}(x)$ and $\boldsymbol{\xi}_{w_{ij}}(x)$ are the fuzzy basis function vectors; $\hat{\boldsymbol{\theta}}_{u_i}$ and $\hat{\boldsymbol{\theta}}_{w_{ij}}$ are the adaptive control parameter vectors.

Assume that $\boldsymbol{x} \in \boldsymbol{D}_x$, let $\boldsymbol{\theta}_{u_i}^*$ and $\boldsymbol{\theta}_{w_{ij}}^*$ be the optimal approximation parameter vectors of $\hat{\boldsymbol{\theta}}_{u_i}$ and $\hat{\boldsymbol{\theta}}_{w_{ij}}$; Let $\varepsilon_{u_i}(x)$ and $\varepsilon_{w_{ij}}(x)$ be the fuzzy logic system minimum approximation errors which satisfy $|\varepsilon_{u_i}(x)| \leqslant \bar{\varepsilon}_{u_i}$ and $\left|\varepsilon_{w_{ij}}(x)\right| \leqslant \bar{\varepsilon}_{w_{ij}}$, where $\bar{\varepsilon}_{u_i}$ and $\bar{\varepsilon}_{w_{ij}}$ are unknown constants. The definitions of parameters are as follows:

$$\boldsymbol{\theta}_{u_i}^* = \arg\min_{\hat{\boldsymbol{\theta}}_{ui}}\{\sup_{x \in \boldsymbol{D}_x} |u_i(x) - \hat{u}_i(x, \hat{\boldsymbol{\theta}}_{u_i})|\} \tag{17}$$

$$\boldsymbol{\theta}_{w_{ij}}^* = \arg\min_{\hat{\boldsymbol{\theta}}_{w_{ij}}}\{\sup_{x \in \boldsymbol{D}_x} |w_{ij}(x) - \hat{w}_{ij}(x, \hat{\boldsymbol{\theta}}_{w_{ij}})|\} \tag{18}$$

$$\tilde{\boldsymbol{\theta}}_{u_i} = \boldsymbol{\theta}_{u_i}^* - \hat{\boldsymbol{\theta}}_{u_i} \tag{19}$$

$$\tilde{\boldsymbol{\theta}}_{w_{ij}} = \boldsymbol{\theta}_{w_{ij}}^* - \hat{\boldsymbol{\theta}}_{w_{ij}} \tag{20}$$

$$\varepsilon_{u_i}(x) = u_i(x) - \hat{u}_i(x, \boldsymbol{\theta}_{u_i}^*) \tag{21}$$

$$\varepsilon_{w_{ij}}(x) = w_{ij}(x) - \hat{w}_{ij}\left(x, \boldsymbol{\theta}_{w_{ij}}^*\right) \tag{22}$$

Let

$$\begin{cases} \hat{U}(x) = [\hat{u}_1(x), \hat{u}_2(x), \ldots, \hat{u}_n(x)]^T \\ \hat{W}(x) = \begin{bmatrix} \hat{w}_{11}(x) & \hat{w}_{12}(x) & \ldots & \hat{w}_{1n}(x) \\ \hat{w}_{21}(x) & \hat{w}_{22}(x) & \ldots & \hat{w}_{2n}(x) \\ \vdots & \vdots & & \vdots \\ \hat{w}_{n1}(x) & \hat{w}_{n2}(x) & \ldots & \hat{w}_{nn}(x) \end{bmatrix} \\ \boldsymbol{\varepsilon}_u(x) = [\varepsilon_{u_1}(x), \varepsilon_{u_2}(x), \ldots, \varepsilon_{u_n}(x)]^T \\ \boldsymbol{\varepsilon}_w(x) = \begin{bmatrix} \varepsilon_{w_{11}}(x) & \varepsilon_{w_{12}}(x) & \ldots & \varepsilon_{w_{1n}}(x) \\ \varepsilon_{w_{21}}(x) & \varepsilon_{w_{22}}(x) & \ldots & \varepsilon_{w_{2n}}(x) \\ \vdots & \vdots & & \vdots \\ \varepsilon_{w_{n1}}(x) & \varepsilon_{w_{n2}}(x) & \ldots & \varepsilon_{w_{nn}}(x) \end{bmatrix} \\ \hat{\boldsymbol{\theta}}_u(x) = \left[\hat{\boldsymbol{\theta}}_{u_1}(x), \hat{\boldsymbol{\theta}}_{u_2}(x), \ldots, \hat{\boldsymbol{\theta}}_{u_n}(x)\right]^T \\ \hat{\boldsymbol{\theta}}_w(x) = \begin{bmatrix} \hat{\boldsymbol{\theta}}_{w_{11}}(x) & \hat{\boldsymbol{\theta}}_{w_{12}}(x) & \ldots & \hat{\boldsymbol{\theta}}_{w_{1n}}(x) \\ \hat{\boldsymbol{\theta}}_{w_{21}}(x) & \hat{\boldsymbol{\theta}}_{w_{22}}(x) & \ldots & \hat{\boldsymbol{\theta}}_{w_{2n}}(x) \\ \vdots & \vdots & & \vdots \\ \hat{\boldsymbol{\theta}}_{w_{n1}}(x) & \hat{\boldsymbol{\theta}}_{w_{n2}}(x) & \ldots & \hat{\boldsymbol{\theta}}_{w_{nn}}(x) \end{bmatrix} \end{cases}$$

Then

$$U(x) - \hat{U}(x, \hat{\boldsymbol{\theta}}_u) = \hat{U}(x, \boldsymbol{\theta}_u^*) - \hat{U}(x, \hat{\boldsymbol{\theta}}_u) + \boldsymbol{\varepsilon}_u(x) \tag{23}$$

$$W(x) - \hat{W}(x, \hat{\boldsymbol{\theta}}_w) = \hat{W}(x, \boldsymbol{\theta}_w^*) - \hat{W}(x, \hat{\boldsymbol{\theta}}_w) + \boldsymbol{\varepsilon}_w(x) \tag{24}$$

For the convenience of explaining the design and prove process, define $|\boldsymbol{A}|$ is a vector or a matrix composed by the absolute value of each corresponding elements in $\boldsymbol{A} \in \mathbf{R}^{i \times j}$ and $\|\cdot\|$ denotes the 2-norm of the matrix.

When the space manipulator is applied in the space, its model is shown in Eq. (8). Comparing Eq. (8) with Eq. (4), we can see that the specific form of the matrixes are different, it can be regarded as the modeling error. In order to decrease the modeling error and obtain good control performance for both ground alignment and space application stage, the robust control terms are introduced. The specific design about the space manipulator could be written as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_c + \boldsymbol{\tau}_r \tag{25}$$

where

$$\begin{aligned}\boldsymbol{\tau}_c = {} & \hat{W}^T\left(x,\hat{\boldsymbol{\theta}}_w\right)\left[\varepsilon I_n + \hat{W}\left(x,\hat{\boldsymbol{\theta}}_w\right)\hat{W}^T\left(x,\hat{\boldsymbol{\theta}}_w\right)\right]^{-1} \\ & \cdot\left(-\hat{U}\left(x,\hat{\boldsymbol{\theta}}_u\right) + \ddot{q}_r + \boldsymbol{\alpha} r + \beta\hat{W}\left(x,\hat{\boldsymbol{\theta}}_w\right)r\right)\end{aligned} \tag{26}$$

$$\boldsymbol{\tau}_r = \frac{r|r^T|(\hat{\bar{\boldsymbol{\varepsilon}}}'_u + \hat{\bar{\boldsymbol{\varepsilon}}}'_w|\boldsymbol{\tau}_c| + \hat{\varepsilon}_\tau|\boldsymbol{\tau}_0|)}{\|r\|^2 + \delta} \tag{27}$$

where $\hat{\bar{\boldsymbol{\varepsilon}}}'_u$, $\hat{\bar{\boldsymbol{\varepsilon}}}'_w$ and $\hat{\varepsilon}_\tau$ are online estimations of $\bar{\boldsymbol{\varepsilon}}'_u = \sigma^{-1}\bar{\boldsymbol{\varepsilon}}_u$, $\bar{\boldsymbol{\varepsilon}}'_w = \sigma^{-1}\bar{\boldsymbol{\varepsilon}}_w$ and $\varepsilon_\tau = \sigma^{-1}$; $\bar{\boldsymbol{\varepsilon}}_u$ is the upper bound of $\boldsymbol{\varepsilon}_u$; $\bar{\boldsymbol{\varepsilon}}_w$ is the upper bound of $\boldsymbol{\varepsilon}_w$; and

$$\begin{aligned}\boldsymbol{\tau}_0 = {} & \varepsilon\left[\varepsilon I_n + \hat{W}(x,\hat{\boldsymbol{\theta}}_w)\hat{W}^T(x,\hat{\boldsymbol{\theta}}_w)\right]^{-1} \\ & \cdot\left(-\hat{U}\left(x,\hat{\boldsymbol{\theta}}_u\right) + \ddot{q}_r + \boldsymbol{\alpha} r + \beta\hat{W}\left(x,\hat{\boldsymbol{\theta}}_w\right)r\right)\end{aligned} \tag{28}$$

where ε is an arbitrary small positive real number; $\boldsymbol{\alpha}$ is a positive definite diagonal matrix; β is a positive real number; and δ is a time varying parameter. Details of the adaptive design are as follows:

$$\dot{\hat{\boldsymbol{\theta}}}_{u_i} = -\lambda_{u_i}\boldsymbol{\xi}_{u_i}(x)r_i \tag{29}$$

$$\dot{\hat{\boldsymbol{\theta}}}_{w_{ij}} = -\lambda_{w_{ij}}\boldsymbol{\xi}_{w_{ij}}(x)r_i(\tau_{cj} - \beta r_j) \tag{30}$$

where, τ_{cj} is the jth element in $\boldsymbol{\tau}_c$; r_i is the ith element in $\boldsymbol{r}$; r_j is the jth element in $\boldsymbol{r}$.

$$\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_u = \eta_0|r| \tag{31}$$

$$\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_w = \eta_0|r||\boldsymbol{\tau}_c^T| \tag{32}$$

$$\dot{\hat{\varepsilon}}_\tau = \eta_0|r^T||\boldsymbol{\tau}_0| \tag{33}$$

$$\dot{\delta}(t) = -\lambda_0|r^T|(\hat{\bar{\boldsymbol{\varepsilon}}}'_u + \hat{\bar{\boldsymbol{\varepsilon}}}'_w|\boldsymbol{\tau}_c| + \hat{\varepsilon}_\tau|\boldsymbol{\tau}_0|)/(\|r\|^2 + \delta) \tag{34}$$

where $\bar{\boldsymbol{\varepsilon}}'_u = \sigma^{-1}\bar{\boldsymbol{\varepsilon}}_u$, $\bar{\boldsymbol{\varepsilon}}'_w = \sigma^{-1}\bar{\boldsymbol{\varepsilon}}_w$, $\varepsilon_\tau = \sigma^{-1}$, λ_{u_i}, $\lambda_{w_{ij}}$, η_0, λ_0 and $\delta(0)$ are positive constants. Define y_p as the trajectory coordinates in operational space, and y_{pd} as the desired trajectory coordinates in operational space. The control frame of the system can be shown in Fig. 2.

4.2. Stability analysis

Define the following Lyapunov function to prove the stability of the closed-loop system

$$\begin{aligned}V = {} & \frac{1}{2}r^Tr + \frac{1}{2}\sum_{i=1}^{n}\frac{1}{\lambda_{u_i}}\tilde{\boldsymbol{\theta}}_{u_i}^T\tilde{\boldsymbol{\theta}}_{u_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{\lambda_{w_{ij}}}\tilde{\boldsymbol{\theta}}_{w_{ij}}^T\tilde{\boldsymbol{\theta}}_{w_{ij}} \\ & + \frac{\sigma}{2\eta_0}\boldsymbol{\varepsilon}_{\varepsilon u}^T\boldsymbol{\varepsilon}_{\varepsilon u} + \frac{\sigma}{2\eta_0}\mathrm{tr}(\boldsymbol{\varepsilon}_{\varepsilon w}^T\boldsymbol{\varepsilon}_{\varepsilon w}) + \frac{\sigma}{2\eta_0}\varepsilon_{\varepsilon\tau}^2 + \frac{\sigma}{2\lambda_0}\delta^2\end{aligned} \tag{35}$$

where $\boldsymbol{\varepsilon}_{\varepsilon u} = \bar{\boldsymbol{\varepsilon}}'_u - \hat{\bar{\boldsymbol{\varepsilon}}}'_u$, $\boldsymbol{\varepsilon}_{\varepsilon w} = \bar{\boldsymbol{\varepsilon}}'_w - \hat{\bar{\boldsymbol{\varepsilon}}}'_w$, $\varepsilon_{\varepsilon\tau} = \varepsilon_\tau - \hat{\varepsilon}_\tau$.

The time derivative of the Lyapunov function V is

$$\begin{aligned}\dot{V} = {} & r^T\dot{r} - \sum_{i=1}^{n}\frac{1}{\lambda_{u_i}}\tilde{\boldsymbol{\theta}}_{u_i}^T\dot{\hat{\boldsymbol{\theta}}}_{u_i} - \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{\lambda_{w_{ij}}}\tilde{\boldsymbol{\theta}}_{w_{ij}}^T\dot{\hat{\boldsymbol{\theta}}}_{w_{ij}} - \frac{\sigma}{\eta_0}\boldsymbol{\varepsilon}_{\varepsilon u}^T\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_u \\ & - \frac{\sigma}{\eta_0}\mathrm{tr}(\boldsymbol{\varepsilon}_{\varepsilon w}^T\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_w) - \frac{\sigma}{\eta_0}\varepsilon_{\varepsilon\tau}\dot{\hat{\varepsilon}}_\tau + \frac{\sigma}{\lambda_0}\delta\dot{\delta}\end{aligned} \tag{36}$$

Using Eqs. (4),(12),(13),(14) and (25), then Eq. (37) can be obtained as

$$\begin{aligned}r^T\dot{r} & = r^T(\ddot{q}_r - U(x) - W(x)\boldsymbol{\tau}) \\ & = r^T(\ddot{q}_r - U(x) - \hat{W}(x,\hat{\boldsymbol{\theta}}_w)\boldsymbol{\tau}_c - (W(x) - \hat{W}(x,\hat{\boldsymbol{\theta}}_w))\boldsymbol{\tau}_c \\ & \quad - W(x)\boldsymbol{\tau}_r)\end{aligned} \tag{37}$$

From Eq. (26), one yields

$$\begin{aligned}\ddot{q}_r - U(x) - \hat{W}(x,\hat{\boldsymbol{\theta}}_w)\boldsymbol{\tau}_c = {} & -(U(x) - \hat{U}(x,\hat{\boldsymbol{\theta}}_u)) - \alpha r \\ & - \beta\hat{W}(x,\hat{\boldsymbol{\theta}}_w)r + \boldsymbol{\tau}_0\end{aligned} \tag{38}$$

Combine Eqs. (23) and (24), one obtains

$$\begin{aligned}r^T\dot{r} = {} & -r^T\boldsymbol{\alpha} r - r^T\beta\hat{W}(x,\boldsymbol{\theta}_w^*)r - \sum_{i=1}^{n}\boldsymbol{\xi}_{u_i}^T(x)\tilde{\boldsymbol{\theta}}_{u_i}r_i \\ & - \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{\xi}_{w_{ij}}^T(x)\tilde{\boldsymbol{\theta}}_{w_{ij}}r_i(\tau_{cj} - \beta r_j) - r^TW(x)\boldsymbol{\tau}_r + r^T\boldsymbol{\tau}_0 \\ & - r^T\boldsymbol{\varepsilon}_u(x) - r^T\boldsymbol{\varepsilon}_w(x)\boldsymbol{\tau}_c\end{aligned} \tag{39}$$

Substituting Eq. (39) into Eq. (36), then the following equation can be obtained as

Table 1 Simulation parameters of the model.

Link-rod	a_i (m)	b_i (m)	m_i (kg)	I_i (kg · m^2)
B_0	–	0.5	40	6.667
B_1	0.5	0.5	4	0.333
B_2	0.5	0.5	3	0.250

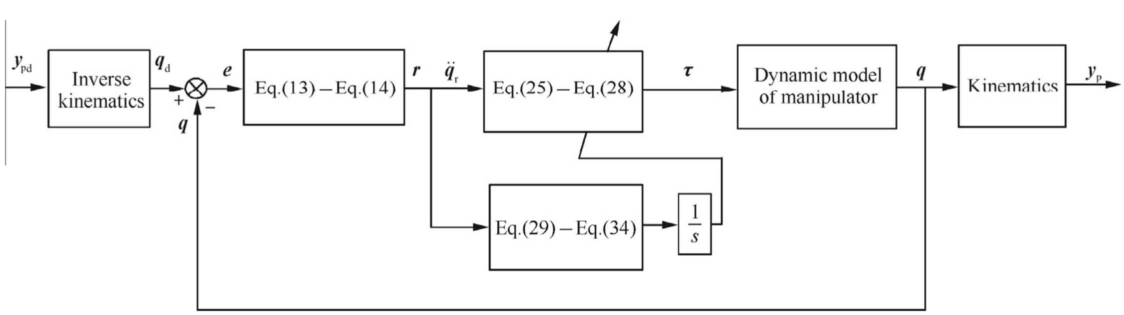


Fig. 2 Fuzzy adaptive robust control frame of the manipulator.

Table 2 Simulation parameters of the controller.

Desired trajectory	Fuzzy membership functions	Gain and adaptive parameters
$y_{\mathrm{pd}} = [x_{\mathrm{d}},\quad y_{\mathrm{d}}]^{\mathrm{T}}$ $\begin{cases} x_{\mathrm{d}} = 0.3\cos(0.5\pi t) + 0.85 \\ y_{\mathrm{d}} = 0.3\sin(0.5\pi t) \end{cases}$	$\mu_{F_i^1}(x_i) = \exp\left(-0.5\left(\frac{x_i+4}{2}\right)^2\right)$ $\mu_{F_i^2}(x_i) = \exp\left(-0.5\left(\frac{x_i}{2}\right)^2\right)$ $\mu_{F_i^3}(x_i) = \exp\left(-0.5\left(\frac{x_i-4}{2}\right)^2\right)$	$\boldsymbol{\alpha} = \mathrm{diag}(500, 500),\quad \lambda_{u_i} = 5,\quad \lambda_{w_{ij}} = 5,$ $\varepsilon = 0.001,\quad \beta = 500,$ $\boldsymbol{\Lambda} = \mathrm{diag}(50, 50),\quad \eta_0 = 0.001,$ $\delta(0) = 1\quad \lambda_0 = 0.001$ $\hat{\varepsilon}_\tau(0) = 0,\quad \hat{\boldsymbol{\varepsilon}}'_{u_i}(0) = 0,\quad \hat{\boldsymbol{\varepsilon}}'_{w_{ij}}(0) = 0.$

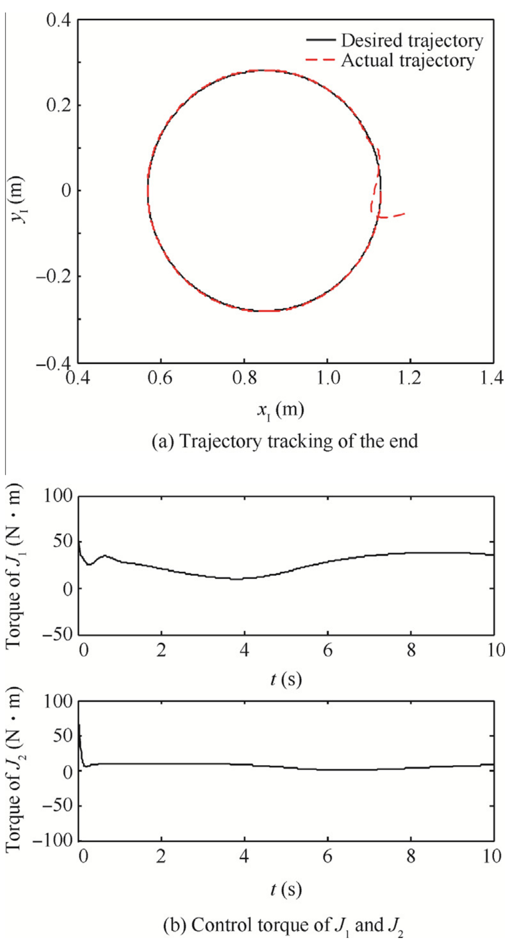


Fig. 3 Simulation results of PD control on the ground.

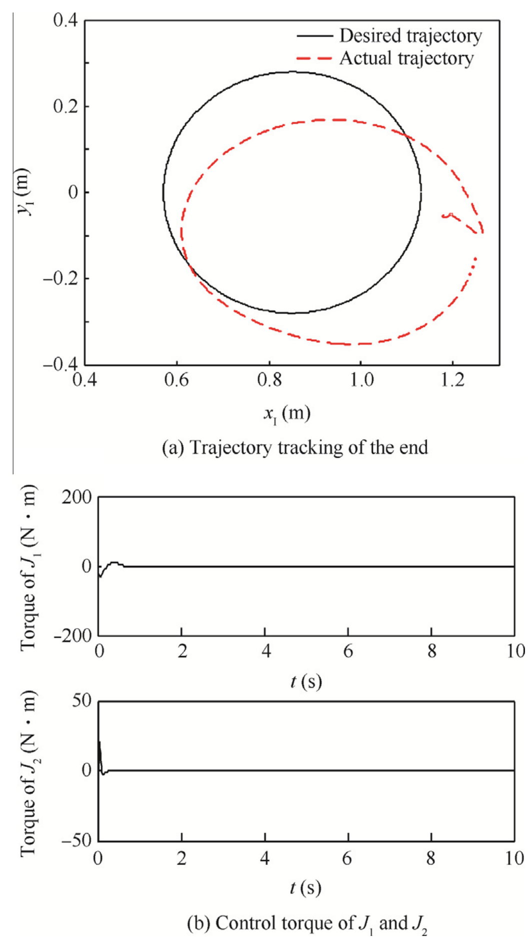


Fig. 4 Simulation results of PD control in the space.

$$\dot{V} = -r^{\mathrm{T}}\boldsymbol{\alpha}r - r^{\mathrm{T}}\beta\hat{W}(x,\boldsymbol{\theta}_w^*)r - \sum_{i=1}^{n}\tilde{\boldsymbol{\theta}}_{u_i}^{\mathrm{T}}\left(\boldsymbol{\xi}_{u_i}(x)r_i + \frac{1}{\lambda_{u_i}}\dot{\hat{\boldsymbol{\theta}}}_{u_i}\right) - \sum_{i=1}^{n}\sum_{j=1}^{n}\tilde{\boldsymbol{\theta}}_{w_{ij}}^{\mathrm{T}}\left(\boldsymbol{\xi}_{w_{ij}}(x)r_i(\tau_{cj} - \beta r_j) + \frac{1}{\lambda_{w_{ij}}}\dot{\hat{\boldsymbol{\theta}}}_{w_{ij}}\right) - r^{\mathrm{T}}W(x)\boldsymbol{\tau}_{\mathrm{r}} + r^{\mathrm{T}}\boldsymbol{\tau}_0 - r^{\mathrm{T}}\boldsymbol{\varepsilon}_u(x) - r^{\mathrm{T}}\boldsymbol{\varepsilon}_w(x)\boldsymbol{\tau}_{\mathrm{c}} - \frac{\sigma}{\eta_0}\boldsymbol{\varepsilon}_{\varepsilon u}^{\mathrm{T}}\dot{\hat{\boldsymbol{\varepsilon}}}'_u - \frac{\sigma}{\eta_0}\mathrm{tr}(\boldsymbol{\varepsilon}_{\varepsilon w}^{\mathrm{T}}\dot{\hat{\boldsymbol{\varepsilon}}}'_w) - \frac{\sigma}{\eta_0}\varepsilon_{\varepsilon\tau}\dot{\hat{\varepsilon}}_\tau + \frac{\sigma}{\lambda_0}\delta\dot{\delta} \tag{40}$$

Substituting Eqs. (29) and (30) into Eq. (40), then Eq. (40) can be rewritten as

$$\dot{V} = -r^{\mathrm{T}}\boldsymbol{\alpha}r - r^{\mathrm{T}}\beta\hat{W}(x,\boldsymbol{\theta}_w^*)r - r^{\mathrm{T}}W(x)\boldsymbol{\tau}_{\mathrm{r}} + r^{\mathrm{T}}\boldsymbol{\tau}_0 - r^{\mathrm{T}}\boldsymbol{\varepsilon}_u(x) - r^{\mathrm{T}}\boldsymbol{\varepsilon}_w(x)\boldsymbol{\tau}_{\mathrm{c}} - \frac{\sigma}{\eta_0}\boldsymbol{\varepsilon}_{\varepsilon u}^{\mathrm{T}}\dot{\hat{\boldsymbol{\varepsilon}}}'_u\frac{\sigma}{\eta_0}\mathrm{tr}\left(\boldsymbol{\varepsilon}_{\varepsilon w}^{\mathrm{T}}\dot{\hat{\boldsymbol{\varepsilon}}}'_w\right) - \frac{\sigma}{\eta_0}\varepsilon_{\varepsilon\tau}\dot{\hat{\varepsilon}}_\tau + \frac{\sigma}{\lambda_0}\delta\dot{\delta} \tag{41}$$

Since $\boldsymbol{W}(\boldsymbol{x}) \geqslant \sigma\boldsymbol{I}_n$ and by combining Eqs. (27),(31),(32),(33),(34), a simplification form can be obtained as

$$-r^{\mathrm{T}}W(x)\boldsymbol{\tau}_{\mathrm{r}}+\frac{\sigma}{\lambda_0}\delta\dot{\delta}\leqslant-\sigma|r^{\mathrm{T}}|(\hat{\bar{\boldsymbol{\varepsilon}}}'_u+\hat{\bar{\boldsymbol{\varepsilon}}}'_w|\boldsymbol{\tau}_{\mathrm{c}}|+\hat{\varepsilon}_\tau|\boldsymbol{\tau}_0|) \quad (42)$$

$$r^{\mathrm{T}}\boldsymbol{\tau}_0-r^{\mathrm{T}}\boldsymbol{\varepsilon}_u(x)-r^{\mathrm{T}}\boldsymbol{\varepsilon}_w(x)\boldsymbol{\tau}_{\mathrm{c}}\leqslant\sigma|r^{\mathrm{T}}|(\bar{\boldsymbol{\varepsilon}}'_u+\bar{\boldsymbol{\varepsilon}}'_w|\boldsymbol{\tau}_{\mathrm{c}}|+\varepsilon_\tau|\boldsymbol{\tau}_0|) \quad (43)$$

$$\frac{\sigma}{\eta_0}\boldsymbol{\varepsilon}^{\mathrm{T}}_{\varepsilon u}\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_u\frac{\sigma}{\eta_0}\mathrm{tr}\left(\boldsymbol{\varepsilon}^{\mathrm{T}}_{\varepsilon w}\dot{\hat{\bar{\boldsymbol{\varepsilon}}}}'_w\right)-\frac{\sigma}{\eta_0}\varepsilon_{\varepsilon\tau}\dot{\hat{\varepsilon}}_\tau=-\sigma|r^{\mathrm{T}}|(\boldsymbol{\varepsilon}_{\varepsilon u}+\boldsymbol{\varepsilon}_{\varepsilon w}|\boldsymbol{\tau}_{\mathrm{c}}|+\varepsilon_{\varepsilon\tau}|\boldsymbol{\tau}_0|) \quad (44)$$

Because

$$-\sigma|r^{\mathrm{T}}|(\hat{\bar{\boldsymbol{\varepsilon}}}'_u+\hat{\bar{\boldsymbol{\varepsilon}}}'_w|\boldsymbol{\tau}_{\mathrm{c}}|+\hat{\varepsilon}_\tau|\boldsymbol{\tau}_0|)+\sigma|r^{\mathrm{T}}|(\bar{\boldsymbol{\varepsilon}}'_u+\bar{\boldsymbol{\varepsilon}}'_w|\boldsymbol{\tau}_{\mathrm{c}}|+\varepsilon_\tau|\boldsymbol{\tau}_0|)$$
$$-\sigma|r^{\mathrm{T}}|(\boldsymbol{\varepsilon}_{\varepsilon u}+\boldsymbol{\varepsilon}_{\varepsilon w}|\boldsymbol{\tau}_{\mathrm{c}}|+\varepsilon_{\varepsilon\tau}|\boldsymbol{\tau}_0|)=0$$

Eq. (45) can be obtained

$$\dot{V}\leqslant-r^{\mathrm{T}}\boldsymbol{\alpha}r-r^{\mathrm{T}}\beta\hat{W}(x,\boldsymbol{\theta}^*_w)r \quad (45)$$

Since $\hat{W}(x,\boldsymbol{\theta}^*_w)$ is a positive definite matrix and β is a positive real number, one yields

$$\dot{V}\leqslant-r^{\mathrm{T}}\boldsymbol{\alpha}r\leqslant 0 \quad (46)$$

5. Simulation results

To verify the effectiveness of the proposed control scheme, simulations were carried out on a two-link planar space manipulator system (n = 2) and the trajectory tracking task under different gravity environments was completed by using the control algorithm in this paper. The simulation was conducted under the environment of matlab7.0 and the simulation time was set 10 s. Table 1 shows the simulation parameters of the model, and Table 2 shows the controller parameters used for numerical analysis.

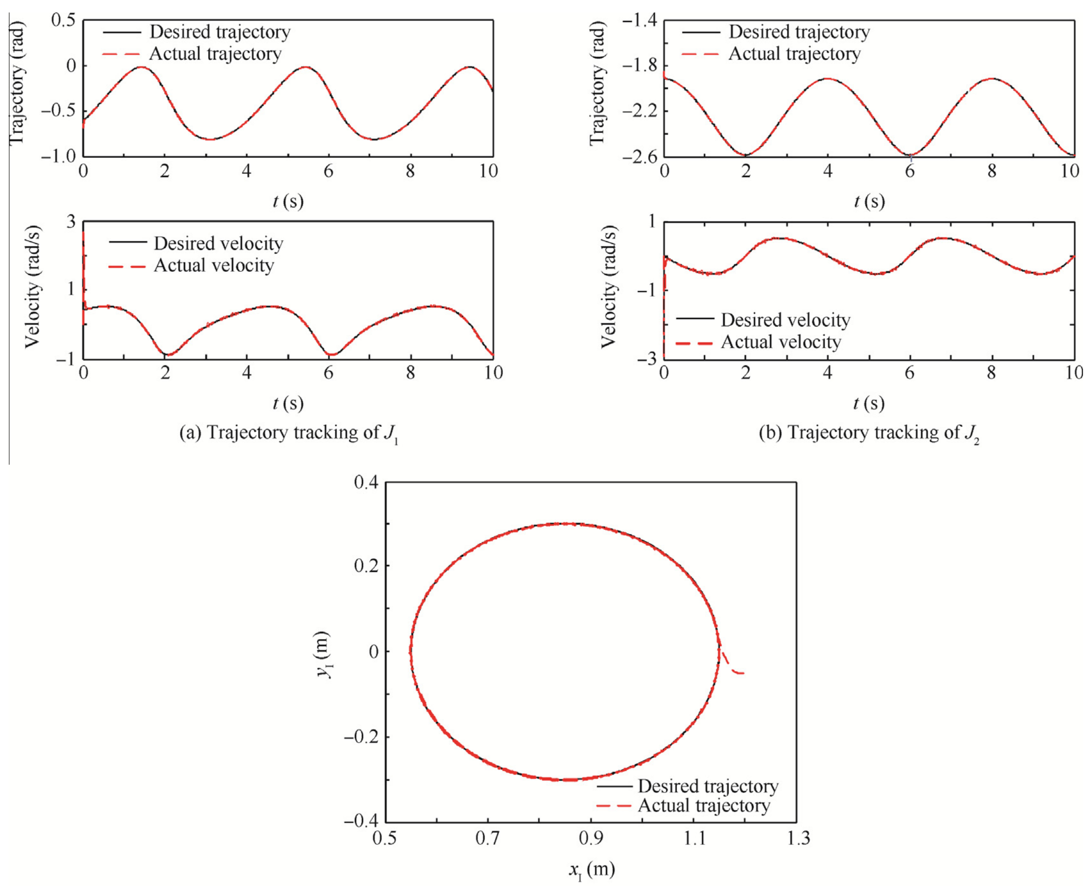
In order to verify the adaptability and robustness of the controller, simulations are compared with the conventional PD control under the same conditions. The PD controller contains a gravity compensation term which is equal to the gravity vector that can be computed as

$$\boldsymbol{\tau}=k_{\mathrm{p}}+k_{\mathrm{d}}+G(q)$$

where k_{p} is proportional gains; k_{d} is differential gains of $\boldsymbol{G}$.

The controller parameters of PD are chosen as k_{p} = 250, k_{d} = 25 when the space manipulator was controlled by the same PD controller for both ground alignment and space application. The simulation results are shown in Figs. 3 and 4. x is the trajectory in x_{I} direction in coordinate system $\Sigma_{\mathrm{I}}-x_{\mathrm{I}}O_{\mathrm{I}}y_{\mathrm{I}}$. y is the trajectory in y_{I} direction in coordinate system $\Sigma_{\mathrm{I}}-x_{\mathrm{I}}$-$O_{\mathrm{I}}y_{\mathrm{I}}$. From Figs. 3 and 4 we can see that when the space manipulator was aligned on the ground it can track the desired trajectory. However, when it was applied in the space, the gravity will disappear, and it cannot track the desired trajectory due to the gravity term that existed in the controller.

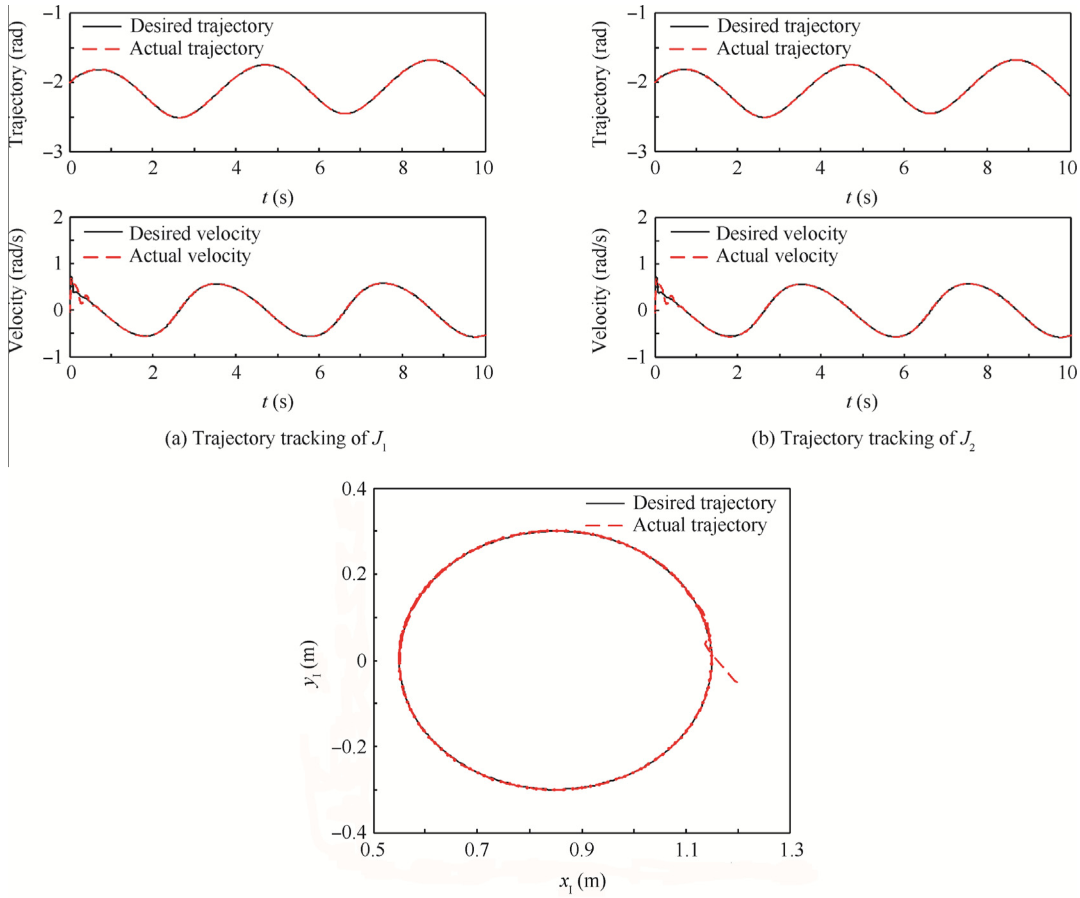
Fig. 5 shows the trajectory tracking results of the space manipulator in joint space and task space during ground align-



(a) Trajectory tracking of J_1 (b) Trajectory tracking of J_2

(c) Trajectory tracking of the end

Fig. 5 Simulation results of the fuzzy adaptive robust control on the ground.



(a) Trajectory tracking of J_1

(b) Trajectory tracking of J_2

(c) Trajectory tracking of the end

Fig. 6 Trajectory tracking results in the on-orbit operational process.

ment stage. From Fig. 5, we can see that when the manipulator system is aligned on the ground, the proposed controller can guarantee that the desired trajectory of joint space can be quickly tracked to, and the end trajectory would rapidly approach the desired value. Fig. 6 shows the trajectory tracking results in the on-orbit operational process. It illustrates that in the on-orbit operational process the desired trajectory would be quickly approached in joint space and task space and the tracking error converges to zero quickly.

The trajectory tracking error curves of fuzzy adaptive robust control method in both joint and task space are shown in Fig. 7 compared with PD control. $e_{x_{\mathrm{I}}}$ and $e_{y_{\mathrm{I}}}$ are the tracking errors in x_{I} and y_{I} directions in the coordinate system $\Sigma_{\mathrm{I}} - x_{\mathrm{I}}O_{\mathrm{I}}y_{\mathrm{I}}$;$e_{q_1}$ and e_{q_2} are the tracking errors of q_1 and q_2, respectively. From Fig. 7, we can see that the proposed fuzzy adaptive robust control strategy can guarantee better tracking performance of the space manipulator from the ground alignment stage to the on-orbit operation stage, and the robust controller can compensate the modeling error during both stages.

6. Conclusions

The changes in kinematic and dynamic model of space robot caused by the change of gravity environment may cause the failure of the trajectory tracking controller in space. For this problem, the main achievements of this work are summarized as follows:

(1) A fuzzy adaptive robust control strategy is proposed to cope with this problem in a new approach. That is to say to develop a controller which can adapt to the variations of mathematical model caused by the change of gravity environment.
(2) The fuzzy algorithm was adopted to estimate the changes in the unknown nonlinear functions of the manipulator model online. In addition, the estimate error is compensated by a robust strategy.
(3) The proposed control strategy is independent of the mathematical model of controlled objects, and in the design of the control law, the dynamic model does not need to be linearized, which avoid complex calculations of regression and unknown parameters estimation, and by this method the computation is reduced.
(4) The closed-loop system stability is proved based on the Lyapunov principle. Simulation shows that the controller can achieve high control accuracy in different gravity environment which demonstrate that the control strategy is effective for both models on the ground and in space.

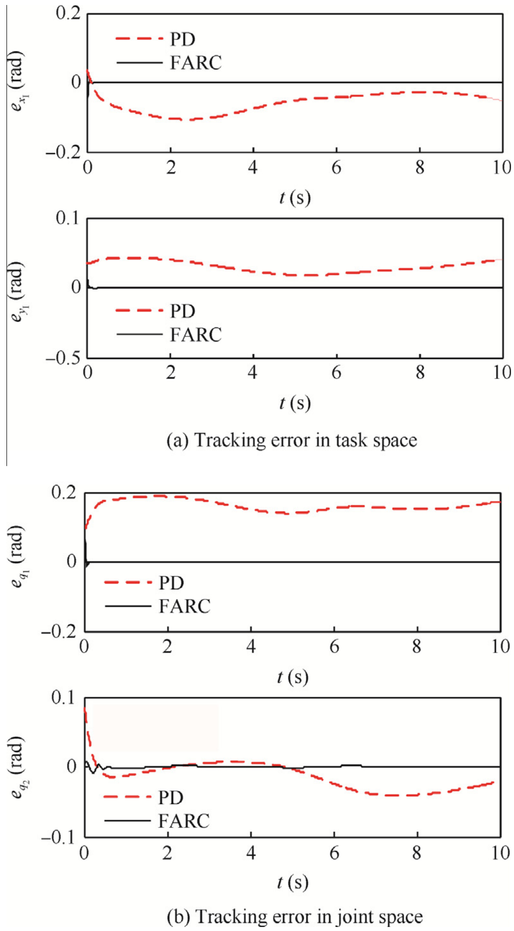


Fig. 7 Curves of trajectory tracking errors.

Acknowledgments

The authors would like to thank the research group for their help in the entire research process. The research of this paper was supported by the National High-tech Research and Development Program of China.

References

1. Zhai G, Zhang JR, Zhou ZC. Coordinated target localization base on pseudo measurement for clustered space robot. *Chin J Aeronaut* 2013;**26**(6):1524–33.
2. Parrish JC, Sullivan BR, Roberts BJ. Planning for the ranger telerobotic shuttle experiment on-orbit operations. 2000. Report No.: AIAA-2000-5291.
3. Robertson A, Inalhan G, How JP. Spacecraft formation flying control design for the orion mission. 1999. Report No.: AIAA-99-4266.
4. Tsunoda H, Hariu K, Kawami Y, Miyosh K. Structural design and deployment test methods for deploy able mesh reflector. 1997. Report No.: AIAA-97-1148.
5. Nagatomo M, Mitome T, Kawasaki K, Kazuhito K. MFD robot arm and its flight experiment. *Proceedings of Sixth ASCE specialty conference and exposition on engineering, construction, and operations in space*; 1998 Apr 26–30; United States. Albuquerque, New Mexico: ASCE; 1998. p. 319–24.
6. Matunaga S. Micro-gravity experiments of space robotics and space-used mechanisms at Tokyo Institute of Technology. *J Jpn Soc Microgravity Appl* 2002;**19**:101–5.
7. Han O, Kienholz D, Janzen P, Kidney S. Gravity-offloading system for large-displacement ground testing of spacecraft mechanisms. *Proceedings of 40th aerospace mechanisms symposium*; 2010 May 12–14; Florida, CA: NASA Kennedy Space Center; 2010. p. 119–32.
8. Chen SF, Mei T, Zhang T, Wang XH. Design of the controller for a crowd simulation system of spatial microgravity environment. *Robot* 2008;**30**(3):201–4 [Chinese].
9. Qi NM, Zhang WH, Gao JZ, Ma J. Design of ground simulation test system for three-dimensional spatial microgravity environment. *Chin J Mech Eng* 2011;**47**(9):16–20.
10. Jiang Z, Liu S, Li H, Dong Q, Chen XP, Huang Q. Mechanism design and system control for humanoid space robot movement using a simple gravity-compensation system. *Int J Adv Robot Syst* 2013;**10**:389–401.
11. Pan B, Yu DY, Sun J. Research on dynamic modeling and analysis of joint in large space manipulator. *J Astronaut* 2010;**31**(11):2448–55 [Chinese].
12. Xu WF, Wang XQ, Xue Q, Liang B. Study on trajectory planning of dual-arm space robot keeping the base stabilized. *Acta Automat Sin* 2013;**39**(1):69–80.
13. Ren YQ, Ma BL. Adaptive control of space robot system based on back-stepping design. *Acta Aeronaut Astronaut Sin* 2007;**28**(2):490–4 [Chinese].
14. Ouyang PR, Acob J, Pano V. PD with sliding mode control for trajectory tracking of robotic system. *Robot Comput Integr Manuf* 2014;**30**(2):189–200.
15. Kumar A, Pushparaj MP, Sukavanam N. Trajectory control of a two DOF rigid-flexible space robot by a virtual space vehicle. *Robot Auton Syst* 2013;**61**(5):58–69.
16. Li X, Cheah CC. Global task-space adaptive control of robot. *Automatica* 2013;**49**(1):58–69.
17. Marco S, Riccardo M, Paolo G, Giovanni B. Adaptive and robust algorithms and tests for visual-based navigation of a space robotic manipulator. *Acta Astronaut* 2013;**83**:65–84.
18. Dubowsky S, Papadopoulos E. The kinematics, dynamics and control of free-flying and free-floating space robotics system. *IEEE Trans Robot Autom* 1993;**9**(5):531–43.
19. Zhang FH, Fu YL, Wang SG. Adaptive control of free-floating space robot with inertia parameters uncertainty. *Acta Aeronaut Astronaut Sin* 2012;**33**(12):2347–54 [Chinese].
20. Liang J, Chen L. Dynamics modeling for free-floating space-based robot during satellite capture and RBF network control for compound body stable movement. *Acta Aeronaut Astronaut Sin* 2013;**34**(4):970–8 [Chinese].
21. Wei C, Liu TX, Zhao Y. Grasping strategy in space robot capturing floating target. *Chin J Aeronaut* 2010;**23**(5):591–8.

Qin Li is a Ph.D. student at school of automation, Yanshan University. Her research interests includes intelligent control, robot technology, simulated microgravity environment.

Liu Fucai is a professor and Ph.D. supervisor at school of automation, Yanshan University. He received the Ph.D. degree from Harbin Institute of Technology in 2003. His current research interests includes fuzzy identification, predictive control, robot technology.

Liang Lihuan is a postgraduate at school of automation, YanShan University. Her research interests is mainly about space manipulator control.

Gao Jingfang is a postgraduate at school of automation, YanShan University. Her research interests is mainly about space manipulator control.