

Journal of Engineering Science and Technology Review 7 (1) (2014) 45 - 51

JOURNAL OF Engineering Science and Technology Review

Research Article

www.jestr.org

Study on Predictive Control for Trajectory Tracking of Robotic Manipulator

Yang Zhao

Dept. of Electronic and Information Technology, Jiangmen Polytechnic, Jiangmen ,529090, China

Received 17 October 2013; Accepted 24 February 2014

Abstract

Model predictive control (MPC) differs from other control methods mainly in implementation of the control actions. In this paper, the tracking error performance index of rolling optimization of predictive control is designed for a class of nonlinear systems by the relative degree, then employed to design controller for robot manipulator. In other words, the robot manipulator's reference trajectory is based on Taylor expansion, and the actual trajectory is substituted into the performance cost function which is derivated for minimum to obtain the controller. Then the performance analysis for the closed system is made. Simulation results demonstrate the effectiveness of the method based on Taylor expansion.

Keywords: Robot Manipulator, Predictive Control, Taylor Expansion, Stability Analysis

1. Introduction

In 1978, Garrett et al. proposed the model prediction-related heuristic control algorithm (MPHC). Since then, predictive control has undergone great development, successively generated the Model Algorithm Control (MAC), Dynamic Matrix Control (DMC), Internal Model Control (IMC), Generalized Predictive Control (GPC) and other dozens of algorithms. Through practical industrial applications, these dozens of algorithms have been proved to have better effects. Initially, these above-mentioned predictive control methods were aimed at linear systems. However, in practice, industrial production processes are often manifested as nonlinear characteristics. Although most of the industrial production processes can be modeled in the vicinity of working points by local linearization methods, some strong nonlinear controlled objects or nonlinear objects with special structures are difficult to obtain satisfactory control effects by using conventional linear control methods. At present, there are not many research results of predictive control methods for nonlinear systems. Being implemented through its rolling optimization, the predictive control itself has a certain degree of robustness. If a system only has weak nonlinear characteristics, it can be regarded as a model mismatch; if a system exhibits strong nonlinear characteristics, using a conventional linear predictive control, due to the model having a comparatively great actual deviation, will not able to reach the effect of optimizing the control, so that the use of a non-linear model will be needed for prediction and optimization, thus producing a nonlinear predictive control method. In view of the present research progress at home and abroad, since the structure identification and parameter estimation of a nonlinear controlled object have great difficulty, the online nonlinear

rolling optimization implementation of an algorithm will experience another difficulty even if the dynamic characteristics and predictive model of a process are obtained in some way. For a nonlinear controlled object, because of its own time-varying, coupling and other characteristics, there is not yet a unified predictive control method now to control it. This requires a more effective nonlinear predictive control method.

In order to further reduce the online calculated amount of predictive control, and make it be appropriate for nonlinear controlled object, in recent years, many scholars have conducted a good deal of research into nonlinear predictive control and its improvement methods. This paper [1] draws on the current-output all-order derivatives to construct the future-output Taylor-order prediction model for the smooth affine nonlinear system. Then it derives the controller solution with the predictive output tracking error norm as the minimum indicator, which could improve accuracy of output prediction using the high control orders.In recent years, robot trajectory tracking control has obtained a lot of achievements However, due to the a robot having strong coupling, high nonlinearity, time-varying and other characteristics, its model parameters vary with the changes in its position, attitude and load, and outside interference, model uncertainty and other factors make the difficulty of controller design increase. In order to achieve the online compensation to uncertainty, various control strategies have been put forward in succession. Among them, self-adaptive control [2,3,4] is mainly used for a case of containing parameter uncertainty and of the unknown parameters of a system being linearized, but has strict real-time requirements, making its realization more complex, and is difficult to ensure the stability of a system and to achieve a certain control performance index especially when there is non-parameter uncertainty; robust control [5] can achieve effective control over a manipulator system, but this method

^{*} E-mail address: zhaoyang19781023@gmail.com

ISSN: 1791-2377 © 2014 Kavala Institute of Technology. All rights reserved.

needs knowing the upper bound of an uncertainty, and in practical application, the upper bound of an uncertainty is difficult to get. Taking into account the defects of the above methods, many scholars combine [6,7,8] an intelligent control method and robust control with adaptive control to make the two coordinate each other, guaranteeing that a control system has good dynamic performance and robust performance. The intelligent control method represented by a neural network [9,10] and fuzzy logic [11] does not need a mathematical model of a system but can accurately approximate a nonlinear system, providing an effective way to deal with an uncertain robot system. However, this algorithm has the large amount of calculation and poor real-timeness.

This paper combines the Taylor-based mechanical hand trajectory and reference trajectory, and integrates them into the predictive control performance indicators with rolling optimization. Then take the derivative of it to get the controller, which could prove the stability of the controller. Meanwhile, the algorithm is highly real-time.

2. Controlled Object Model

A robot can be regarded as an open-chain, rigid and multilink mechanism. Its trajectory tracking control problem is that given a joint angle trajectory vector[12] to be tracked and the initial state of the joint angle, it is required to design a controller and provide a joint control torque to make a joint angle of a robot meet a certain track condition. The kinetic equation of a robot having two degrees of freedom can be represented as:

$$M(\theta)\dot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) = \tau$$
(1)

where

 $M(\theta)$:definite symmetric inertia matrix $C(\theta, \dot{\theta})$:centrifugal force and coriolis force vector $G(\theta)$:gravity vector $\boldsymbol{\tau} = [\tau_1, \tau_2]^{\mathrm{T}}$:joint control torque vector $\theta = [\theta_1, \theta_2]^{\mathrm{T}}$:robot joint angle position $\dot{\theta} = [\dot{\theta}_1, \dot{\theta}_2]^{\mathrm{T}}$:robot joint angle velocity

 $\ddot{\boldsymbol{\theta}} = [\ddot{\theta}_1, \ddot{\theta}_2]^{\mathrm{T}}$: robot joint angle acceleration.

3. Design of Predictive Controller

For the following MIMO, the nonlinear system is:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) + \boldsymbol{g}(\boldsymbol{x}(t))\boldsymbol{u}(t) \\ y_i(t) = h_i(\boldsymbol{x}(t)) \quad i = 1, 2, \cdots, m \end{cases}$$
(2)

where

 $\mathbf{x}(t) \in \mathbb{R}^{n}$: state variables $\mathbf{u}(t) \in \mathbb{R}^{m}$: system control input $\mathbf{y} = [y_{1}, y_{2}, \dots, y_{m}]^{\mathrm{T}} \in \mathbb{R}^{m}$: system output.

Consider a class of nonlinear systems like those of formula(2) and make the following assumptions.

Assumption 1. The system's reference signal and output for time t are continuous.

Assumption 2. The system's all states can be observed.

Assumption 3. The system's zero dynamic stability $y_i(t)$ has a relative degree ρ , and the system is stable in zero dynamics. For the definition of the relative degree, see Definition 1.

Definition 1. If a nonlinear system like that of formula(2) in area $D_0 \in D$, then

$$L_g L_f^j h_i(\mathbf{x}) = 0, \quad j = 1, 2, L \quad , \rho - 2,$$

$$L_g L_f^{\rho-1} h_i(\mathbf{x}) \neq 0, \quad \forall \mathbf{x} \in D_0$$
(3)

in the formula, $L_f^k h_i(\mathbf{x}) = L_f[L_f^{k-1}h_i(\mathbf{x})], \quad L_f^0 h_i(\mathbf{x}) = 0$

The system has a relative degree ρ . that is, formula(3) is established, equivalent to:

$$\begin{cases} L_g h_i(\mathbf{x}) = 0 \\ L_{ad_fg} h_i(\mathbf{x}) = 0 \\ \mathbf{M} \\ L_{ad_f^{\rho^2}g} h_i(\mathbf{x}) = 0 \\ L_{ad_f^{\rho^1}g} h_i(\mathbf{x}) \neq 0 \end{cases}$$
(4)

which is established or equivalently has:

$$\frac{\partial h_i}{\partial \mathbf{x}}(\mathbf{x})[\mathbf{g}(\mathbf{x}), a \, \mathrm{d}_f \mathbf{g}(\mathbf{x}), a \, \mathrm{d}_f^2 \mathbf{g}(\mathbf{x}), \mathrm{L} ,$$
$$a \, \mathrm{d}_f^{\rho-1} \mathbf{g}(\mathbf{x})] = [0, 0, \mathrm{L} , a(\mathbf{x})] \tag{5}$$

in the formula, $a(x) \neq 0$. $\forall x$ is an appropriate function.

Definition 2. Suppose the consecutive prediction of the future control amount of the control system $\hat{u}(t+\tau)$ in $\tau \in [0,T]$ meets:

$$\begin{cases} \frac{d^{r} \hat{\boldsymbol{u}}(t+\tau)}{d\tau} \neq 0, \tau \in [0,T] \\ \frac{d^{k} \hat{\boldsymbol{u}}(t+\tau)}{d\tau} = 0, \tau \in [0,T], k > r \end{cases}$$
(6)

then, r is the control step of the predictive control system. At first, define the prediction output and reference trajectory at moment $t + \tau$ respectively as $\hat{y}(t + \tau)$ and $y_r(t + \tau)$, and define the prediction error as:

$$\boldsymbol{e}(t+\tau) = \hat{\boldsymbol{y}}(t+\tau) - \boldsymbol{y}_r(t+\tau) \tag{7}$$

Set the performance indicator of the system in rolling prediction time range T as:

$$J = \frac{1}{2} \int_0^T \boldsymbol{e}^{\mathrm{T}}(t+\tau) \boldsymbol{e}(t+\tau) \mathrm{d}\tau$$
(8)

Formula(8) shows that the system takes the tracking error as a performance indicator and it can also be considered that the system's performance indicator is a function of tracking error.

Suppose the control step inputted by the system is r', and make predictive output y(t) of the system at moment t derived to ρ times, then

$$\begin{cases} \dot{\hat{\boldsymbol{y}}}(t) = L_f \, \boldsymbol{h}(\boldsymbol{x}) \\ \vdots \\ \boldsymbol{\hat{y}}^{(\rho-1)}(t) = L_f^{\rho-1} \, \boldsymbol{h}(\boldsymbol{x}) \\ \boldsymbol{\hat{y}}^{(\rho)}(t) = L_f^{\rho-1} \, \boldsymbol{h}(\boldsymbol{x}) + L_g \, L_f^{\rho-1} \, \boldsymbol{h}(\boldsymbol{x}) \boldsymbol{\hat{u}}(t) \end{cases}$$

$$(9)$$

in the formula, $L_f^{i} h(\mathbf{x})$ is the *i* order derivative of $\hat{\mathbf{y}}(t)_{\text{to}} f(x)_{\text{and}} h(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}) \cdots h_m(\mathbf{x})]^{\text{T}}$

$$\begin{cases} L_{f} \mathbf{h}(\mathbf{x}) = \partial \mathbf{h}(\mathbf{x}) / \partial(\mathbf{x})^{\mathrm{T}} \mathbf{f}(\mathbf{x}) \\ L_{f}^{2} \mathbf{h}(\mathbf{x}) = \partial(\partial \mathbf{h}(\mathbf{x}) / \partial(\mathbf{x})^{\mathrm{T}} \mathbf{f}(\mathbf{x}) / \partial(\mathbf{x}))^{\mathrm{T}} \mathbf{f}(\mathbf{x}) \\ \mathrm{M} \end{cases}$$
(10)

The definition of $L_{g}L_{f}^{p-1}h(x)$ is similar to that of $L_{f}^{i}h(x)$,

$$L_{g}L_{f}^{\rho-1}h(x) = (\partial L ((\partial h(x)/\partial (x)^{T} f(x))/\partial x)^{T} g(x)$$
(11)

Make the control step of control input $\boldsymbol{u}(t)$ take r, and make predictive output $\hat{\boldsymbol{y}}(t)$ of the system at moment t continue to derive time t, and you will obtain:

$$\hat{y}^{[\rho+1]}(t) = L_f^{\rho+1} h(x) + p_{11}(\hat{u}(t), x(t)) + L_g L_f^{\rho-1} h(x) \dot{\hat{u}}(t)$$
(12)

in the formula, $p_{11}(\hat{\boldsymbol{u}}(t),\boldsymbol{x}(t)) = L_g L_f^{\rho^{-1}} \boldsymbol{h}(\boldsymbol{x}) \hat{\boldsymbol{u}}(t) + \frac{\mathrm{d} L_g L_f^{\rho^{-1}} \boldsymbol{h}(\boldsymbol{x})}{\mathrm{d} t} \hat{\boldsymbol{u}}(t)$

Similarly, make the predictive output $\hat{y}(t)$ of the system derived to $\rho + r$ times:

$$\hat{y}^{[\rho+r]}(t) = L_{f}^{\rho+r}\mathbf{h}(\mathbf{x}) + p_{1r}(\hat{\mathbf{u}}(t), \mathbf{x}(t)) + \mathbf{L} + p_{rr}(\hat{\mathbf{u}}(t), \mathbf{L}, \hat{\mathbf{u}}^{[r-1]}(t), \mathbf{x}(t)) + L_{g}L_{f}^{\rho-1}\mathbf{h}(\mathbf{x})\hat{\mathbf{u}}^{[r]}(t)$$
(13)

in the formula, $\hat{\boldsymbol{u}}(t) = [\hat{\boldsymbol{u}}(t)^{\mathrm{T}}, \dot{\hat{\boldsymbol{u}}}(t)^{\mathrm{T}}, \cdots, \hat{\boldsymbol{u}}^{[\rho+r]}(t)^{\mathrm{T}}]$ and $p_{1r}(\hat{\boldsymbol{u}}(t), \boldsymbol{x}(t)), \cdots, p_{rr}(\hat{\boldsymbol{u}}(t), \cdots, \hat{\boldsymbol{u}}^{[r-1]}(t), \boldsymbol{x}(t))$ is a complex nonlinear function on $\hat{\boldsymbol{u}}(t), \boldsymbol{x}(t)$.

It is known from the above theoretical derivation that in rolling predictive time domain T, predictive output $\hat{y}(t+\tau)$ at moment $t+\tau$, using the Taylor series, can be approximately expressed as:

$$\hat{\mathbf{v}}(t+\tau) = \boldsymbol{\Gamma}(\tau)\overline{\mathbf{Y}}(t) \tag{14}$$

 $\boldsymbol{\Gamma}(\tau) = \left[I_m \bar{\tau}, I_m \frac{\bar{\tau}^2}{2!}, \cdots, I_m \frac{\bar{\tau}^{\rho+r}}{(\rho+r)!} \right] \in \mathbb{R}^{m \times m(\rho+r+1)}$ in the formula, and $\bar{\tau} = \text{diag} \{ \tau, \cdots, \tau \} \in \mathbb{R}^{m \times m}$ and I_m is the unit matrix of $m \times m_{1s}$

$$\hat{\bar{Y}}(t) = \begin{bmatrix} \hat{y}^{[0]}(t) \\ \hat{y}^{[1]}(t) \\ \vdots \\ \hat{y}^{[\rho+1]}(t) \\ \hat{y}^{[\rho+2]}(t) \\ \vdots \\ \hat{y}^{[\rho+r]}(t) \end{bmatrix} = \begin{bmatrix} h(x) \\ L_{f}^{l}h(x) \\ \vdots \\ L_{f}^{\rho}h(x) \\ \vdots \\ L_{f}^{\rho+r}h(x) \end{bmatrix} + \begin{bmatrix} 0_{m\times l} \\ \vdots \\ 0_{m\times l} \\ H(\hat{u}(t)) \end{bmatrix}$$
(15)

in the formula,

$$\begin{bmatrix}
L_{g} L_{f}^{p-1} \mathbf{h}(\mathbf{x}) \dot{\hat{\mathbf{u}}}(t) + p_{11}(\hat{\mathbf{u}}(t), \mathbf{x}(t)) + L_{g} L_{f}^{p-1} \mathbf{h}(\mathbf{x}) \hat{\mathbf{u}}(t) \\
\vdots \\
p_{1r}(\hat{\mathbf{u}}(t), \mathbf{x}(t)) + \dots + p_{rr}(\hat{\mathbf{u}}(t) \\
\vdots \\
\hat{\mathbf{u}}^{[r-1]}(t) L_{g} L_{f}^{p-1} \mathbf{h}(\mathbf{x}) \hat{\mathbf{u}}^{[r]}(t)
\end{bmatrix}$$

 $H(\hat{u}(t))$

Similarly, make reference trajectory $y_r(t+\tau)$ at moment $t+\tau$, using the Taylor series, approximately expressed as:

$$\hat{\boldsymbol{y}}_r(t+\tau) = \boldsymbol{\Gamma}(\tau)\boldsymbol{Y}_r(t) \tag{16}$$

in the formula, $\overline{Y}_{r}(t) = [y_{r}(t)^{\mathrm{T}}, \hat{y}_{r}(t)^{\mathrm{T}}, L y_{r}^{\rho+r}(t)^{\mathrm{T}}].$ $\overline{\Gamma}(T) = \int_{0}^{\mathrm{T}} \Gamma^{\mathrm{T}}(\tau) \Gamma(\tau) d\tau \in R^{m(\rho+r+1)\times m(\rho+r+1)}$, it is found from $\hat{y}(t+\tau) = \Gamma(\tau)\hat{\overline{Y}}(t)$ that the elements of this matrix are as follows:

$$\overline{\Gamma}(T)_{(i,j)} = \frac{\overline{T}^{i+j-1}}{(i-1)!(j-1)!(i+j-1)!}$$
(17)

in the formula, $\overline{T} = \text{diag}\{T, \dots, T\} \in \mathbb{R}^{m \times m}, i, j = 1, 2, \dots, \rho + r + 1$.

According to formulas(14), (16)and(17), performance index formula (8) can be rewritten as:

$$J = \frac{1}{2} \left[\hat{\overline{Y}}(t) - \overline{\overline{Y}}_{r}(t) \right]^{\mathrm{T}} \overline{\Gamma}(T) \left[\hat{\overline{Y}}(t) - \overline{\overline{Y}}_{r}(t) \right] \mathrm{d}\tau$$
(18)

In order to make the above-mentioned performance index minimum, make

$$\frac{\partial J}{\partial \hat{\boldsymbol{u}}(t)} = 0 \tag{19}$$

The following theorems can be obtained:

Theorem 1. For a class of nonlinear systems composed of formula(2), on satisfying assumptions 1 to 3, use the Taylor formula to make the system output expand to $\rho + r$ steps. Among them, r > 0 is the control step. The

following predictive control law can ensure that predictive control performance formula(8) is optimal:

$$\boldsymbol{u}(t) = -(L_g L_f^{\rho-1} \boldsymbol{h}(\boldsymbol{x}))^{-1} (\boldsymbol{K} \boldsymbol{M}_{\rho} + L_f^{\rho} \boldsymbol{h}(\boldsymbol{x}) - \boldsymbol{y}_r^{[\rho]})$$
(20)

where the non-singular matrix: $\begin{bmatrix} I & I^{n-1}I \\ I & I \end{bmatrix}$

$$L_{g}L_{f}^{\rho-1}h(\mathbf{x}) = \begin{bmatrix} L_{g}L_{f}^{\rho} \cdot h_{1}(\mathbf{x}) & \cdots & L_{gm}L_{f}^{\rho} \cdot h_{1}(\mathbf{x}) \\ L_{g1}L_{f}^{\rho-1}h_{2}(\mathbf{x}) & \cdots & L_{gm}L_{f}^{\rho-1}h_{2}(\mathbf{x}) \\ \vdots & \cdots & \vdots \\ L_{g1}L_{f}^{\rho-1}h_{m}(\mathbf{x}) & \cdots & L_{gm}L_{f}^{\rho-1}h_{m}(\mathbf{x}) \end{bmatrix}$$

$$L_{f}^{\rho}h(\mathbf{x}) = \begin{bmatrix} L_{f}^{\rho}h_{1}(\mathbf{x}) \\ \vdots \\ L_{f}^{\rho}h_{m}(\mathbf{x}) \end{bmatrix} , \qquad \mathbf{y}_{r}^{[\rho]} = \begin{bmatrix} \mathbf{y}_{r1}^{\rho}(\mathbf{x}) \\ \vdots \\ \mathbf{y}_{rm}^{[\rho]}(\mathbf{x}) \end{bmatrix}$$

$$M_{\rho} = \begin{bmatrix} \mathbf{h}(\mathbf{x}) - \mathbf{y}_{r}(t) \\ L_{f}^{1} \mathbf{h}(\mathbf{x}) - \mathbf{y}_{r}^{(1)}(t) \\ \vdots \\ L_{f}^{\rho-1} \mathbf{h}(\mathbf{x}) - \mathbf{y}_{r}^{[\rho-1]}(t) \end{bmatrix},$$

$$\begin{split} \boldsymbol{K} &\in \boldsymbol{R}^{m \times m p} \text{ is a new matrix composed of the first } ^{\boldsymbol{M}} \text{ lines of } \\ \text{matrix } \quad \boldsymbol{\overline{\Gamma}(T)}_{rr}^{-1} \boldsymbol{\overline{\Gamma}(T)}_{\rho r}^{\mathsf{T}} \text{ and } \quad \boldsymbol{\overline{\Gamma}(T)}_{rr} \quad \boldsymbol{\overline{\Gamma}(T)}_{\rho r} \text{ is obtained by } \\ \text{partitioning } & \text{matrix } \quad \boldsymbol{\overline{\Gamma}(T)} \quad , \quad \text{that} \\ \text{is } \quad \boldsymbol{\overline{\Gamma}(T)}_{rr} = \begin{bmatrix} \boldsymbol{\overline{\Gamma}(T)}_{(\rho+1,\rho+1)} & \cdots & \boldsymbol{\overline{\Gamma}(T)}_{(\rho+1,\rho+1)} \\ \vdots & \vdots & \vdots \\ \boldsymbol{\overline{\Gamma}(T)}_{(\rho+r+1,\rho+1)} & \cdots & \boldsymbol{\overline{\Gamma}(T)}_{(\rho+r+1,\rho+r+1)} \end{bmatrix} \quad , \\ \boldsymbol{\overline{\Gamma}(T)}_{\rho r} = \begin{bmatrix} \boldsymbol{\overline{\Gamma}(T)}_{(1,\rho+1)} & \cdots & \boldsymbol{\overline{\Gamma}(T)}_{(1,\rho+r+1)} \\ \vdots & \vdots & \vdots \\ \boldsymbol{\overline{\Gamma}(T)}_{(\rho,\rho+1)} & \cdots & \boldsymbol{\overline{\Gamma}(T)}_{(\rho,\rho+r+1)} \end{bmatrix} \\ \vdots & \vdots & \vdots \\ \boldsymbol{\overline{\Gamma}(T)}_{(\rho,\rho+1)} & \cdots & \boldsymbol{\overline{\Gamma}(T)}_{(\rho,\rho+r+1)} \end{bmatrix} \end{split}$$

The specific derivation process can be found in literature [1].

4. Prediction equation and controller solving

At first, with given desired reference trajectory $\theta_r(t)$, define the predicted value of joint angle $\theta(t+t_s)$ at moment $t+t_s$ as $\hat{\theta}(t+t_s)$ the reference trajectory prediction at time $t+t_s$ as $\hat{\theta}_r(t+t_s)$, and the prediction error as: $\epsilon(t+t_s) = \hat{\theta}(t+t_s) - \hat{\theta}_r(t+t_s)$

The performance index function is taken as:

$$J = \frac{1}{2} \int_0^T [\boldsymbol{\varepsilon}^{\mathrm{T}}(t+t_s)\boldsymbol{\varepsilon}(t+t_s) + \lambda \dot{\boldsymbol{\varepsilon}}^{\mathrm{T}}(t+t_s) \dot{\boldsymbol{\varepsilon}}(t+t_s)] \mathrm{d}t_s$$
(21)

in the formula, λ is a weighting factor and a constant greater than zero; *T* is the predicted time domain. $\boldsymbol{\varepsilon}$ and $\dot{\boldsymbol{\varepsilon}}$ among the performance indicators reflect the optimization requirements of a closed-loop system state.

In order to make the rolling optimization meaningful, you should make T include the dynamic part of a controlled object, that is, you should include all the effects with more control influence. Under a normal circumstance, make T close to the rise time of a system. In actual industrial application, choosing a greater T is more appropriate so that it will exceeds the reverse part caused by the non-minimum phase characteristics or the time delay part of the pulse response of the system and will cover the main dynamic response of a controlled object. The selection of T has a greater impact on the stability and rapidity of a system. Although after selecting a smaller prediction time domain, a

system's rapidity is guaranteed, it has poor robustness and stability; if you select a greater prediction time domain, the robustness of the system will be stronger, but its dynamic response is slower, which increases the computation time and reduces the system's real-time property. In actual selection, you may take a value between the two. If the rapidity is bad, reduce T; if the stability is bad, increase T. Only the two coordinating each other can make a closed loop system have required robustness and desired rapidity.

Use Taylor's formula to make the predictive values of joint angles $\theta(t+t_s)$ and $\dot{\theta}(t+t_s)$ at moment $t+t_s$ approximately expanded as follows:

 $\begin{cases} \hat{\theta}(t+t_{s}) \approx \sum_{i=0}^{4} \frac{t_{s}^{i}}{i!} \theta^{(i)}(t) = H_{1}(t_{s}) \mathbf{y} \\ \hat{\theta}(t+t_{s}) \approx 0_{2\times 2} \times \theta(t) + \sum_{i=1}^{4} \frac{t_{s}^{i-1}}{(i-1)!} \theta^{(i)}(t) = H_{2}(t_{s}) \mathbf{y} \end{cases}$ (22)

in the formula,

$$\boldsymbol{H}_{1}(t_{s}) = \left[I_{2}, t_{s}I_{2}, \frac{t_{s}^{2}}{2}I_{2}, \frac{t_{s}^{3}}{6}I_{2}, \frac{t_{s}^{4}}{24}I_{2}\right], \boldsymbol{H}_{2}(t_{s}) = \left[0_{2\times2}, I_{2}, t_{s}I_{2}, \frac{t_{s}^{2}}{2}I_{2}, \frac{t_{s}^{3}}{6}I_{2}\right],$$

$$\boldsymbol{y} = \left[\boldsymbol{\theta}(t)^{\mathrm{T}}, \dot{\boldsymbol{\theta}}(t)^{\mathrm{T}}, \boldsymbol{\theta}^{(3)}(t)^{\mathrm{T}}, \boldsymbol{\theta}^{(4)}(t)^{\mathrm{T}}\right]^{\mathrm{T}}, \quad 0_{2\times2}, I_{2} \quad \text{respectively}$$

are 2×2 order zero matrix and 2-order unit matrix.

Similarly, we can get the Taylor expansion of $\theta_r(t)$ and its derivative at moment $t + t_s$ approximated as:

$$\begin{cases} \hat{\theta}_{r}(t+t_{s}) \approx \sum_{i=0}^{4} \frac{t_{s}^{i}}{i!} \theta_{r}^{(i)}(t) = H_{1}(t_{s}) y_{r} \\ \hat{\theta}_{r}(t+t_{s}) \approx 0_{2\times 2} \times \theta_{r}(t) + \sum_{i=1}^{4} \frac{t_{s}^{i-1}}{(i-1)!} \theta_{r}^{(i)}(t) = H_{2}(t_{s}) y_{r} \end{cases}$$
(23)

in the formula, $\mathbf{y}_r = \left[\boldsymbol{\theta}_r(t)^{\mathrm{T}}, \dot{\boldsymbol{\theta}}_r(t)^{\mathrm{T}}, \boldsymbol{\theta}_r^{(3)}(t)^{\mathrm{T}}, \boldsymbol{\theta}_r^{(4)}(t)^{\mathrm{T}}\right]^{\mathrm{T}}$ According to formula(22) and formula(23), performance

indicator formula (21) can be approximately expressed as:

$$J = \frac{1}{2} (\boldsymbol{y} - \boldsymbol{y}_r)^{\mathrm{T}} \boldsymbol{R} (\boldsymbol{y} - \boldsymbol{y}_r)$$
(24)

in the formula, $\mathbf{R} = \int_0^T [\mathbf{H}_1^{\mathsf{T}} \mathbf{H}_1 + \lambda \mathbf{H}_2^{\mathsf{T}} \mathbf{H}_2] dt_s$ is a 10×10 constantrelated symmetric matrix and consistent with the format of formula(8).

For joint angle vector $\theta(t)$, respectively obtain its 2,3,4order derivatives, and we will get:

$$\ddot{\boldsymbol{\theta}}(t) = \boldsymbol{m}_{1}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + [\boldsymbol{M}(\boldsymbol{\theta})]^{-1}\boldsymbol{\tau}$$
⁽²⁵⁾

$$\boldsymbol{\theta}^{(3)}(t) = \boldsymbol{m}_2(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{p}_1(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}}) + [\boldsymbol{M}(\boldsymbol{\theta})]^{-1} \dot{\boldsymbol{\tau}}$$
(26)

$$\boldsymbol{\theta}^{(4)}(t) = \boldsymbol{m}_{3}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{p}_{2}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \boldsymbol{\theta}^{(3)}\boldsymbol{\tau}, \dot{\boldsymbol{\tau}}) + [\boldsymbol{M}(\boldsymbol{\theta})]^{-1} \boldsymbol{\dot{\tau}}$$
(27)

Where

$$\boldsymbol{m}_{1}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = -[M(\boldsymbol{\theta})]^{-1}[C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + G(\boldsymbol{\theta})]$$

$$\boldsymbol{m}_{2}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \dot{\boldsymbol{\theta}} \frac{\partial \boldsymbol{m}_{1}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}} , \qquad \boldsymbol{m}_{3}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \dot{\boldsymbol{\theta}} \frac{\partial \boldsymbol{m}_{2}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}} ,$$

$$p_{1}(\theta, \dot{\theta}, \ddot{\theta}, \tau) = \ddot{\theta} \frac{\partial m_{1}(\theta, \dot{\theta})}{\partial \theta} + \dot{\theta}\tau \times \frac{\partial [M(\theta)]^{-1}}{\partial \theta}$$

$$p_{2}(\theta, \dot{\theta}, \ddot{\theta}, \theta^{(3)}\tau, \dot{\tau}) = \ddot{\theta} \frac{\partial m_{2}(\theta, \dot{\theta})}{\partial \theta} + \dot{p}_{1} \times \dot{\theta}\dot{\tau} \frac{\partial [M(\theta)]^{-1}}{\partial \theta},$$
define $\hat{\boldsymbol{y}} = [\boldsymbol{\Phi}^{\mathrm{T}} - \boldsymbol{\Phi}_{r}^{\mathrm{T}}, \dot{\boldsymbol{\Phi}}_{r}^{\mathrm{T}} - \dot{\boldsymbol{\Phi}}_{r}^{\mathrm{T}}]^{\mathrm{T}}, \hat{\boldsymbol{\Phi}}_{r} = [\boldsymbol{\tilde{\Phi}}_{r}^{\mathrm{T}}, \boldsymbol{\Phi}_{r}^{(3)\mathrm{T}}, \boldsymbol{\Phi}_{r}^{(4)\mathrm{T}}]^{\mathrm{T}},$
from formula(27), we can see:

$$\boldsymbol{y} - \boldsymbol{y}_r = \begin{bmatrix} \hat{\boldsymbol{y}} \\ \boldsymbol{m} - \hat{\boldsymbol{\theta}}_r \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}}) \end{bmatrix}$$
(28)

in the formula,

$$\hat{\boldsymbol{\tau}} = [\boldsymbol{\tau}^{\mathrm{T}}, \dot{\boldsymbol{\tau}}^{\mathrm{T}}, \ddot{\boldsymbol{\tau}}^{\mathrm{T}}], \quad \boldsymbol{m} = \begin{bmatrix} \boldsymbol{m}_{1}^{\mathrm{T}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}), \boldsymbol{m}_{1}^{\mathrm{T}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}), \boldsymbol{m}_{1}^{\mathrm{T}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \end{bmatrix}^{\mathrm{T}}, \\ \hat{\boldsymbol{m}}(\hat{\boldsymbol{\Phi}}) = \begin{bmatrix} [\boldsymbol{M}(\boldsymbol{\Phi})]^{-1} \boldsymbol{\Phi} \\ \boldsymbol{p}_{1}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}, \ddot{\boldsymbol{\Phi}}, \boldsymbol{\Phi}) + [\boldsymbol{M}(\boldsymbol{\Phi})]^{-1} \dot{\boldsymbol{\Phi}} \\ \boldsymbol{p}_{2}(\boldsymbol{\Phi}, \dot{\boldsymbol{\Phi}}, \ddot{\boldsymbol{\Phi}}, \boldsymbol{\Phi}) + [\boldsymbol{M}(\boldsymbol{\Phi})]^{-1} \ddot{\boldsymbol{\Phi}} \end{bmatrix}, \\ \mathbf{make} \\ R = \begin{bmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ \boldsymbol{R}_{12}^{\mathrm{T}} & \boldsymbol{R}_{22} \end{bmatrix}, \\ \text{where } \boldsymbol{R}_{11} \in \boldsymbol{R}^{4\times4}, \quad \boldsymbol{R}_{12} \in \boldsymbol{R}^{4\times6} \quad \boldsymbol{R}_{22} \in \boldsymbol{R}^{6\times6}. \\ \text{Derive from formula(28), we will see:} \end{cases}$$

$$\frac{\partial J}{\partial \hat{\boldsymbol{\tau}}} = \frac{\partial (\boldsymbol{y} - \boldsymbol{y}_{r})^{\mathrm{T}}}{\partial \hat{\boldsymbol{\tau}}} \boldsymbol{R}(\boldsymbol{y} - \boldsymbol{y}_{r}) = \begin{bmatrix} \boldsymbol{0} \\ \frac{\partial \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}})}{\partial \hat{\boldsymbol{\tau}}} \end{bmatrix}^{\mathrm{T}} \times \begin{bmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ \boldsymbol{R}_{12}^{\mathrm{T}} & \boldsymbol{R}_{22} \end{bmatrix} \times \\ \begin{pmatrix} \begin{bmatrix} \hat{\boldsymbol{y}} \\ \boldsymbol{m} - \hat{\boldsymbol{y}}_{r} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}}) \end{bmatrix} \end{pmatrix} \\ = \begin{bmatrix} \frac{\partial \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}})}{\partial \hat{\boldsymbol{\tau}}} \end{bmatrix}^{\mathrm{T}} \times \begin{pmatrix} \begin{bmatrix} \boldsymbol{R}_{12}^{\mathrm{T}} & \boldsymbol{R}_{22} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{y}} \\ \boldsymbol{m} - \hat{\boldsymbol{\theta}}_{r} \end{bmatrix} + \boldsymbol{R}_{22} \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}}) \end{pmatrix}$$
(29)

For $\hat{m}(\hat{\tau})$ in formula(29), obtain the partial derivative of $\hat{\tau}$, and we will see:

$$\frac{\partial \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}})}{\partial \hat{\boldsymbol{\tau}}} = \begin{bmatrix} [M(\boldsymbol{\theta})]^{-1} & \boldsymbol{0}_{2\times 2} & \boldsymbol{0}_{2\times 2} \\ * & [M(\boldsymbol{\theta})]^{-1} & \boldsymbol{0}_{2\times 2} \\ * & * & [M(\boldsymbol{\theta})]^{-1} \end{bmatrix}$$
(30)

Therefore, $\frac{\partial \hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}})}{\partial \hat{\boldsymbol{\tau}}}$ is a reversible matrix. \boldsymbol{R}_{22} is also a $\frac{\partial J}{\partial \boldsymbol{\tau}}$

reversible matrix. From the minimum condition $\frac{\partial J}{\partial \hat{r}} = 0$, for performance indicators, make formula (29) equal to 0, and we will have:

$$\hat{\boldsymbol{m}}(\hat{\boldsymbol{\tau}}) = -\boldsymbol{R}_{22}^{-1}\boldsymbol{R}_{12}^{\mathrm{T}}\hat{\boldsymbol{y}} - (\boldsymbol{m} - \hat{\boldsymbol{\theta}}_{r})$$
(31)

Define $e(t) = \theta_r(t) - \theta(t) K_1$ as a square matrix composed of the first two lines and the first two columns of $R_{22}^{-1}R_{12}^{T}$ and K_2 as a square matrix composed of the first two lines and the last two columns of $R_{22}^{-1}R_{12}^{T}$. From the above formulas, the control law making the performance indicators minimum, the following derivations all omit time t, can be obtained as follows:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{\theta})[\boldsymbol{\ddot{\theta}}_r + \boldsymbol{K}_2 \boldsymbol{\dot{e}} + \boldsymbol{K}_1 \boldsymbol{e}] + \boldsymbol{C}(\boldsymbol{\theta}, \boldsymbol{\dot{\theta}})\boldsymbol{\dot{\theta}} + \boldsymbol{G}(\boldsymbol{\theta})$$
(32)

According to the definitions of K_1, K_2 , they can be calculated as:

$$K_{1} = \begin{bmatrix} k_{1} & 0 \\ 0 & k_{1} \end{bmatrix}, K_{2} = \begin{bmatrix} k_{2} & 0 \\ 0 & k_{2} \end{bmatrix},$$

$$K_{1} = \begin{bmatrix} (K_{11} / K_{12}) / K_{13} \end{bmatrix} / K_{14},$$

$$K_{2} = \begin{bmatrix} (K_{21} / K_{22}) / K_{23} \end{bmatrix} / K_{24},$$

$$k_{11} = 1050T^{6} \times (34T^{4} + 3744\lambda T^{2} + 16128\lambda^{2}),$$

$$k_{12} = (338688\lambda^{3} + 136080\lambda^{2}T^{2} + 5460\lambda T^{4} + 25T^{6}) \times T^{3} + 14700 \times (5T^{4} + 468\lambda T^{2} + 1728\lambda^{2})$$

$$k_{13} = (338688\lambda^2 T^2 + 136080\lambda^2 T^2 + 5460\lambda T^4 + 25T^6) \times T^4 - 1512 \times (25T^4 + 2065\lambda T^2 + 7056\lambda^2)$$

$$\begin{split} k_{14} &= (338688\lambda^3 + 136080\lambda^2 T^2 + 5460\lambda T^4 + 25T^6)T^5 \\ k_{21} &= 6300T^6 \times (34T^4 + 3744\lambda T^2 + 16128\lambda^2) \end{split}$$

$$\begin{split} k_{22} &= (338688\lambda^3 + 136080\lambda^2 T^2 + 5460\lambda T^4 + 25T^6) \times \\ k_{23} &= (\frac{1}{3}38688\lambda^3 + \frac{1}{2}\lambda_1^2) + \frac{1}{3}0080\lambda^2 T^0 + \frac{5}{3}460\lambda T^4 + \frac{1}{2}5T^6} \right) \times \\ (\frac{1}{7}28\lambda^2) + \frac{1}{6}\lambda T^3) - 181440 \times (25T^4 + 2056\lambda T^2 + 7056\lambda^2) \\ k_{24} &= (338688\lambda^3 + 136080\lambda^2 T^2 + 5460\lambda T^4 + 25T^6) \times \\ (1/144T^6 + \frac{1}{2}4\lambda T^2) \end{split}$$

5. Closed-loop system performance analysis

Under a circumstance that the modeling of a robot system is accurate and its interference is zero, a conventional PD controller can also be used to implement the trajectory tracking of its joint angle. Its control law is formula(33):

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{\theta})[\hat{\boldsymbol{\theta}}_r - \boldsymbol{K}_{\mathrm{v}}\dot{\boldsymbol{e}} - \boldsymbol{K}_{\mathrm{p}}\boldsymbol{e}] + \boldsymbol{C}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \boldsymbol{G}(\boldsymbol{\theta})$$
(33)

in the formula, $K_v = aI_2, K_p = bI_2, a > 0$, b > 0.

Although this formula(33) and the control law formula (32) obtained from performance index formula(21) rolling optimization in this paper are consistent in form, the two are essentially different. In the PD controller, the selection of coefficients K_{v} and K_{P} has a great deal of randomness, and the tentative selection of parameters does not ensure that a designed controller is globally optimum. Predictive control using rolling optimization to substitute global one-time optimization, that is, the optimization process is not performed offline at a time but undergoes online repeated optimization calculation and rolling implementation, thus making the uncertainty caused by model mismatch, time varying, interference and the like made up in time, always making new optimization built on an actual foundation, and making control keep actually optimal. For an actual complex industrial process, there inevitably is uncertainty, therefore, a rolling optimization strategy built on a finite time domain will be more effective. Formula (32) K_1, K_2 is exactly the embodiment of this characteristic.

 $M(\theta)$ is a reversible matrix, so after making robust predictive control law (32) substituted into equation (20) for a controlled object, we can obtain a closed-loop system equation as follows:

$$\mathbf{e} + \mathbf{K}_2 \mathbf{e} + \mathbf{K}_1 \mathbf{e} = 0$$

(34)

From the form of K_1 and K_2 , we can know it is a positive definite matrix. When a controlled object takes (1) and the control law takes formula (33), equation (34) of a closed-loop system will be globally asymptotically stable.

6. Simulation

Consider that the kinetic model of a two-joint robot system is (1),

where

$$M(\theta) = \begin{bmatrix} v + q_1 + 2q_2 \cos(\theta_2) & q_1 + q_2 \cos(\theta_2) \\ q_1 + q_2 \cos(\theta_2) & q_1 \end{bmatrix}$$

$$C(\theta, \dot{\theta}) = \begin{bmatrix} -q_2 \dot{\theta}_2 \sin(\theta_2) & -q_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_2) \\ q_2 \dot{\theta}_1 \sin(\theta_2) & 0 \end{bmatrix}$$

$$G(\theta) = \begin{bmatrix} 15g \cos(\theta_1) + 8.75g \cos(\theta_1 + \theta_2) \\ 8.75g \cos(\theta_1 + \theta_2) \end{bmatrix}$$
simulation parameters are $v = 13.33, q_1 = 8.98, q_2 = 8.75, g = 9.8$

and the desired tracking signal is $\begin{bmatrix} \theta_{r1} \\ \theta_{r2} \end{bmatrix} = \begin{bmatrix} 1+0.2\sin(0.5\pi t) \\ 1-0.2\cos(0.5\pi t) \end{bmatrix}$.

In performance indicators (21), select $\lambda = 2$, predictive time domain, T = 1s, and the solution will be $K_1 = 2.8041I_2, K_2 = 7.6734I_2$.

After calculation, the relative step and control step of thee robot model used in this section are 2. After using Simulink and S function to carry out control system simulation, the simulation results are shown in Figures 1 (a) to (d).

Figures 1 (a) to 1 (b) show the joint angle output and its reference trajectory, Figure 1 (c) shows the error curve and Figure 1 (d) shows the control torque. From Fig .(a) to Fig .(b), it can be seen that the output curve can quickly track the reference curve. From Figure (c), it can be seen that the tracking error is 0, and the tracking error convergence is relatively rapid. From Figure (d), it can be seen that the control torque and its desired signal are basically the same, with the controller having a good effect.



Fig. 1(a) Joint 1 and its reference trajectory



Fig. 1(b) Joint 2 and its reference trajectory



Fig. 1(c)Error between joint 1 and joint 2



Fig. 1(d) Control torque for joint 1 and joint 2

7. Conclusions

At first, this paper gives a predictive controller design method for a multi-input and multi-output nonlinear system based on Taylor expansion under a nominal condition.

Taking a robot as a controlled object, this paper improves the performance index function in the method we propose, and then uses it for robot control.

For the above methods, the design process is simple, without online calculation, thus greatly reducing the heavy computation burden brought by predictive control rolling optimization.

References

- W. H. CHEN, D. J. BALANCE, P. J. Gawthrop, "Optimal Control of Nonlinear System: a Predictive Control Approach", *Automatica*, 39(4), 2003, pp.633-641.
- TAYEBI A, "Adaptive iterative learning control for robot Manipulators", Automatica, 40(7), 2004, pp.1195–1203.
- 3. LIUZZO S, TOMEI P, "A global adaptive learning control for robotic manipulators", *Automatica*, 44(5).2008, pp.1379–1384.
- LIUZZO S, MARINO R, TOMEI P, "Adaptive learning controlof nonlinear systems by output error feedback", *IEEE Transactions on Automatic Control*, 52(4), 2007, pp. 1232–1248.
- SHI J, LIU H, BAJCINCA N, "Robust control of robotic manipulators based on integral sliding mode", *International Journal of Control*, 81(10).2008, pp.1537–1548.
- LI T H S, HUANG Y C, "MIMO adaptive fuzzy terminal Slidingmode controller for robotic manipulators", *Information Sciences*, 23(180).2010, pp.4641-4660.
- ZUO Y, WANG Y N, LIU X Z, SIMON X, "Neural network robust H∞ tracking control strategy for robot manipulators", *Applied Mathematical Modelling*, 34 (7).2010, pp.1823–1838.

- LI Z J,GE S Z,WANG Z P, "Robust adaptive control of coorinate multiple mobile manipulators",Mechatronics,18(5),2008,pp.239–250.
 YILDIRIM S, ESKI I, "Noise analysis of robot manipulatorusing
- KUMAR N, PANWAR V, SUKAVANAM, S P, "Neural networkbased nonlinear tracking control of kinematically redundant robot manipulators", Mathematical and Computer Modelling, 21(1), 2011, pp.1–13.
- ALAM M S,TOKHI M O, "Hybird fuzzy logic control with genetic optimisation for a single-link flexible manipulator", Engineering Applications of Artificial Intelligence, 6(21), 2008, pp.858–873.
- A. Kovantsov, R. Krumbergs, "Creation of graphs of functions with use of theorems of elementary geometry", *Computer Modelling and New Technologies*, 16(2), 2012, pp. 50-59.