

# The Design of Unmanned Aerial Vehicle Cruising Formation Predictive Controller Based on the Multi-model Method

Xudong Wang<sup>1,a,\*</sup>, Xiaowei Wang<sup>1,b</sup>, Wenlong Hao<sup>1,c</sup>, Zewei Dong<sup>1,d</sup>

<sup>1</sup>Army Aviation Institute, Taihu Street 9, Tongzhou, Beijing, China

<sup>a</sup>wanghx96@163.com, <sup>b</sup>wangxiaoweis@163.com, <sup>c</sup>574321214@qq.com, <sup>d</sup>dongzewei00@163.com

\*corresponding author

**Keywords:** Multi-model, Predictive control, UAV, Formation flight.

**Abstract:** In the process of unmanned aerial vehicle (uav) cruising formation flight control, traditional step predictive control method limits the maneuverability of the aircraft, prone to excessive regulation, which leads to the unmanned aerial vehicle collision problems. In this paper, a scheme of using multi-model predictive control method to design cruising formation controller will be described, which converts the nonlinear rolling optimization problem into linear quadratic optimization problem, improves the real-time nonlinear prediction, and finally realizes the reliable control of the unmanned aerial vehicle formation. Through the simulation experiment it can be seen that based on multi-model predictive control for UAV cruising formation control can not only meet the control requirements, but also can satisfy the real-time requirements.

## 1. Introduction

More drones in cruising flight to keep a certain formation, which can have full access to environmental information, increase the resistance to the outside world strike capability, improve the system efficiency and robustness, has been widely concerned. In the process of unmanned aerial vehicle (UAV) for cruising formation flight, according to the task when they need to maintain or change the formation, need to control the posture, speed and relative position of the UAV etc, but as a result of UAV control system itself is nonlinear coupling system, combined with the complex battlefield environment constraints, prompted the formation of cruise formation flight design, change and maintain technical requirements are also increasing.

Nonlinear model predictive control method for solving the control problem of nonlinear system provides an effective means, but due to the short distance between cruising fleet UAV produces collide with each other more easily, to demand a higher control precision, the traditional step predictive control method limits the maneuverability of aircraft, prone to excessive regulation, which leads to the UAV collision problems. In this paper, a scheme of using multi-model predictive control method to design controller cruising formation will be described, which converts the nonlinear rolling optimization problem into linear quadratic optimization problem, improves the real-time nonlinear prediction, and finally realizes the reliable control of the UAV formation.

## 2. UAV Cruising Formation Predictive Controller Design

### 2.1. Formation Control Model's Feature Point Setting

According to the basic principle of multi-model predictive control, before you get multi-model is first to determine the feature points of the nonlinear model. As shown in figure 1, the basic idea is as follows: To determine the first feature point first, and then according to the error between the reference trajectory and the tangent which through the feature point, compared with the maximum permissible error, if is greater than the maximum permissible error, criterion to determine the feature point, get the next feature point, repeated calculation until to get the last feature point, so as to determine the feature points of nonlinear system and its applicable areas<sup>[1][4]</sup>.

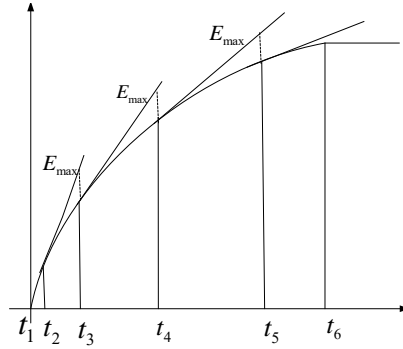


Figure 1 The feature point setting description.

For UAV cruising formation control, the determination of feature points method is as follows:

UAV formation control, assuming initial distance of two machine is  $x_d(0)$ ,  $y_d(0)$ , and target is  $x_c$ ,  $y_c$ , X axis and Y axis to the expectations of the target at the same time, reference trajectory of the UAV cruising formation are as follows:

$$y_{1r}(t) = \xi^t x_d(0) + (1 - \xi^t) \cdot x_c \quad (1)$$

$$y_{2r}(t) = \xi^t y_d(0) + (1 - \xi^t) \cdot y_c \quad (2)$$

Due to there are two output produces two reference trajectory, so the above algorithm cannot be directly used, but due to the expectations of the target at the same time, the reference trajectory in the form is basic consistent, can use an output state of reference trajectories to determine the feature points. Assume that use the X axis track coordinates the relative position of the reference trajectory state to determine the feature points.

The feature points are:

$$\dot{x}_d = 0, \dot{y}_d = 0, \dot{\varphi}_w = 0 \quad (3)$$

Assume that the feature point as  $i$  is  $(V_w^i, \omega_w^i, x_d^i, y_d^i, \varphi_d^i)$ , then have the following equation:

$$\begin{bmatrix} -1 & y_d^i \\ 0 & -x_d^i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_w^i \\ \omega_w^i \end{bmatrix} + \begin{bmatrix} V_l \cos(\varphi_l - \varphi_w^i) \\ V_l \sin(\varphi_l - \varphi_w^i) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4)$$

At the same time :

$$x_d^i = y_{1r}(t), y_d^i = y_{2r}(t) \quad (5)$$

Calculating get:

$$\omega_w^i = 0, \varphi_w^i = \varphi_l, v_w^i = v_l, x_d^i = y_{1r}(i), y_d^i = y_{2r}(i) \quad (6)$$

Assume that the initial point is the first feature point, the value of the feature point as follows:

$$\omega_w^1 = 0, \varphi_w^1 = \varphi_l, v_w^1 = v_l, x_d^1 = x_d(0), y_d^1 = y_d(0) \quad (7)$$

The tangent equation is:

$$y_{1k}(t) = \dot{y}_{1r}(t)t + x_d(0) \quad (8)$$

And then to determine the maximum allowable error  $E_{\max}$ , depending on the above method to calculate the second feature point corresponding time value  $t_2$ , can get feature point:

$$\omega_w^2 = 0, \varphi_w^2 = \varphi_l, v_w^2 = v_l, V_w^2 = V_l, x_d^2 = x_d(t_2), y_d^2 = y_d(t_2) \quad (9)$$

The tangent equation is:

$$y_{1k}(t) = \dot{y}_{1r}(t)(t-t_2) + x_d(t_2) \quad (10)$$

Repeat the above steps, can get the corresponding time value  $t_i$  of the feature points in cruising formation model regional, at the same time can get the feature point which corresponds to the time value  $t_i$ :

$$\omega_w^i = 0, \varphi_w^i = \varphi_l, v_w^i = v_l, x_d^i = x_d(t_i), y_d^i = y_d(t_i) \quad (11)$$

Until to get the final feature point, operation shall be terminated.

## 2.2. Cruising Formation Control Discrete Model Set Generating

After getting the feature points, and linearizing the system in each feature point, the discrete model set for cruise formation can be gained, linearizing method is as follows<sup>[5]</sup>:

The nonlinear systems described by the discrete time dynamic equation is:

$$x(k+1) = f(x(k), u(k)) \quad (12)$$

$$y(k) = g(x(k), u(k)) \quad (13)$$

Suppose the system has m different feature points,  $f(x(k), u(k))$  and  $g(x(k), u(k))$  is the first-order continuous partial function, linearize the system in each feature point, the standard discrete state space model which contain m linear models of the original system will be gained:

$$x(k+1) = A_i x(k) + B_i u(k) - \alpha_i \quad (14)$$

$$y(k) = C_i x(k) + D_i u(k) - \beta_i \quad (15)$$

Among them:

$$A_i = \frac{\partial f}{\partial x} \Big|_{(x_i, u_i)}, B_i = \frac{\partial f}{\partial u} \Big|_{(x_i, u_i)}, C_i = \frac{\partial g}{\partial x} \Big|_{(x_i, u_i)}, D_i = \frac{\partial g}{\partial u} \Big|_{(x_i, u_i)} \quad (16)$$

$$\alpha_i = \frac{\partial f}{\partial x} \Big|_{(x_i, u_i)} x_i + \frac{\partial f}{\partial u} \Big|_{(x_i, u_i)} u_i - x_i, \beta_i = \frac{\partial g}{\partial x} \Big|_{(x_i, u_i)} x_i + \frac{\partial g}{\partial u} \Big|_{(x_i, u_i)} u_i - y_i \quad (17)$$

These m linearized models constitute the linearized multi-model of the system.

For the control of UAV cruising formation flight, the feature points are:

$$\omega_w^i = 0, \varphi_w^i = \varphi_l, v_w^i = v_l, x_d^i = x_d(t_i), y_d^i = y_d(t_i) \quad (18)$$

Has the following expressions:

$$A_i = \frac{\partial f}{\partial x} \Big|_{(x_i, u_i)} = \frac{\partial \left( \begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(k) \\ 0 & -x_d(k) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} + \begin{bmatrix} v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ v_l(k) \sin(\varphi_l(k) - \varphi_w(k)) \\ 0 \end{bmatrix} \Delta T \right)}{\partial x} \Big|_{(x_i, u_i)}$$

$$= \begin{bmatrix} 0 & \omega_w(k) & v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ -\omega_w(k) & 0 & -v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ 0 & 0 & 0 \end{bmatrix} \Delta T \Big|_{(x_i, u_i)} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -v_l(k) \Delta T \\ 0 & 0 & 1 \end{bmatrix} \quad (19)$$

$$B_i = \frac{\partial f}{\partial u} \Big|_{(x_i, u_i)} = \frac{\partial \left( \begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(k) \\ 0 & -x_d(k) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} + \begin{bmatrix} v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ v_l(k) \sin(\varphi_l(k) - \varphi_w(k)) \\ 0 \end{bmatrix} \Delta T \right)}{\partial u} \Big|_{(x_i, u_i)}$$

$$= \begin{bmatrix} -1 & y_d(k) \\ 0 & -x_d(k) \\ 0 & 1 \end{bmatrix} \Big|_{(x_i, u_i)} = \begin{bmatrix} -1 & y_d(t_i) \\ 0 & -x_d(t_i) \\ 0 & 1 \end{bmatrix} \Delta T \quad (20)$$

$$\alpha_i = \frac{\partial f}{\partial x} \Big|_{(x_i, u_i)} x_i + \frac{\partial f}{\partial u} \Big|_{(x_i, u_i)} u_i - x_i$$

$$= \frac{\begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(k) \\ 0 & -x_d(k) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} + \begin{bmatrix} v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ v_l(k) \sin(\varphi_l(k) - \varphi_w(k)) \\ 0 \end{bmatrix} \Delta T}{\partial x} \Big|_{(x_i, u_i)} x_i \Delta T$$

$$+ \frac{\begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(k) \\ 0 & -x_d(k) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} + \begin{bmatrix} v_l(k) \cos(\varphi_l(k) - \varphi_w(k)) \\ v_l(k) \sin(\varphi_l(k) - \varphi_w(k)) \\ 0 \end{bmatrix} \Delta T}{\partial u} \Big|_{(x_i, u_i)} u_i \Delta T - x_i$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -v_l(k) \Delta T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(t_i) \\ 0 & -x_d(t_i) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} \Delta T - \begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_w(k) \end{bmatrix}$$

$$= \begin{bmatrix} -v_w(k) + \omega_w(k) y_d(k) \\ -v_l(k) \varphi_w(k) - x_d(k) \omega_w(k) \\ \omega_w(k) \end{bmatrix} \Big|_{(x_i, u_i)} \Delta T = \begin{bmatrix} -v_l(k) \Delta T \\ -v_l(k) \varphi_l(k) \Delta T \\ 0 \end{bmatrix} \quad (21)$$

For output is linear, so  $C_i, D_i, \beta_i$  don't need to solve. the linearizing equation in the feature points are as follows:

$$\begin{bmatrix} x_d(k+1) \\ y_d(k+1) \\ \varphi_d(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -v_l(k) \Delta T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d(k) \\ y_d(k) \\ \varphi_d(k) \end{bmatrix} + \begin{bmatrix} -1 & y_d(t_i) \\ 0 & -x_d(t_i) \\ 0 & 1 \end{bmatrix} \Delta T \begin{bmatrix} v_w(k) \\ \omega_w(k) \end{bmatrix} - \begin{bmatrix} -v_l(k) \Delta T \\ -v_l(k) \varphi_l(k) \Delta T \\ 0 \end{bmatrix} \quad (22)$$

For different feature point, can get a different time zone linear model of UAV formation control, realizes to get the UAV cruising formation model set, and to know these models for  $M(1), M(2), \dots, M(S)$ .

### 2.3. Cruising Formation Flight Control Model Set Combination Method

For multi-model sets generally has the following two combination methods<sup>[6][8]</sup>:

(1)Weighted combination method based on controller output.

The basic idea is as follows:linearize the nonlinear system in each feature point, establish corresponding linearized model set, and design predictive functional controller independently according to the linearized model and the given performance index and the constraint conditions.In each control period, the controller calculate the output independently, and then linear weighted output according to the design of the weighted beforehand.

(2)Weighted combination method based on the model.

The basic idea is as follows: linearize the nonlinear system in each feature point, establish corresponding linearized model set, determine the model weights at the sample point according to the model of weighted function at each sampling time, and Then calculate the linear control model of the control object in the sample point according to the weighted function. The linear predictive controller is designed by using the model, and get the final output of the controller.

Through the above feature points gained method to get feature points,when the error value is maximum switch to the new model, ensure the maximum error value between forecast track and the reference trajectory is constant, so as to realize the determination of the model applicable area. It can be seen that the model applicable area is based on time division, and so on the different samples have different models, and predictive control belong to the forecast of the future in the time domain, so in this paper, according to the time domain determine the applicable model of the forecasting point ,and the model is used to calculate the predicted value, time domain prediction model judgment rule is as follows:

Assume that the feature points corresponding time is  $t_1 < t_2 < \dots < t_{s-1} < t_s$ , predictive time domain is  $[t, t+N]$ , and then are:

If  $[t, t+N] \in [t_i, t_{i+1}]$ , then the final prediction model of all points is denoted as  $M_T = M_i$ ;

If  $[t, t+N] \in [t_i, t_{i+N}]$ , then it can be determined in turn that all points belong to interval  $[t_h, t_{h+1}]$  from predictive point  $t+1$  to  $t+N$ , and the model is denoted as  $M_T = M_h$ ;

If  $[t, t+N] \in [t_s, +\infty]$ , then the final prediction model of all points is denoted as  $M_T = M_s$ ;

Through the above methods, you can get next time domain prediction function, so as to determine the optimization index. But due to the boundary point in predicting area may be more close to the next linear model, as shown in figure 2, in the sample point  $p_1$  may be more close to the the linearized model of  $t_4$ , but according to the above method, the calculation model for sampling point  $p_1$  is the linearized model of  $t_3$ ; The same as above, in the sample point  $p_2$  may be more close to the the linearized model of  $t_5$ , but according to the above method, the calculation model for sampling point  $p_2$  is the linearized model of  $t_4$ .

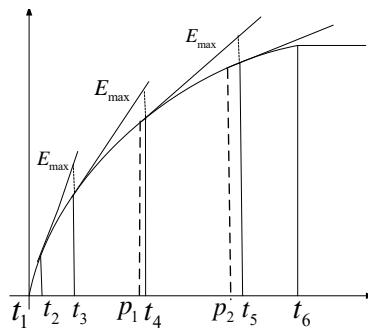


Figure 2 The tangent error description.

So the above method in regional boundary approximation ability declined, model switch not smooth enough. T-S fuzzy model as an intelligent control method, mainly using the approximation

capability of fuzzy reasoning for nonlinear systems, the input space is divided into several fuzzy subspace, in each fuzzy subspace to set up a local linear model, then use the membership function to smooth link each local model, so as to form the global fuzzy model of the nonlinear function, the model has finally been identified as a linear model. So this section referenced by the ideas of T-S fuzzy, and simplify reasoning, draw a UAV formation motion diagram of multiple model control method, as shown in figure 3.

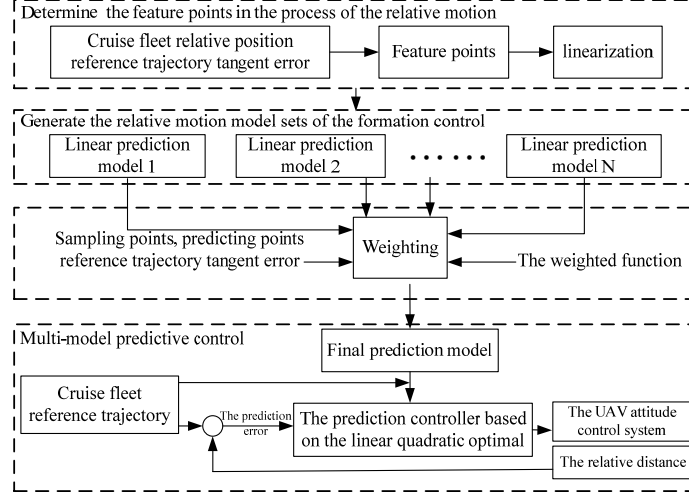


Figure 3 Multi-model control method sketch of UAV formation movement.

For each sampling point, using the tangent of feature points and the sampling points of the reference trajectory error to calculate the sampling points for the feature points after linearized the membership degree of the linear model. Assume that the tangent of  $j$  feature point to  $t$  sampling point reference trajectory error is  $E_j(t)$ , for the point  $t+i$  in predicting area, can use the following weight function:

$$\begin{cases} w_j(t+i) = e^{-\frac{E_j(t+i)^2}{E_{\max}}} & -E_{\max} \leq E_j(t+i) \leq E_{\max} \\ w(t+i) = 0 & E_j(t+i) \leq -E_{\max} \text{ 或 } E_j(t+i) \geq E_{\max} \end{cases} \quad (23)$$

Through this function ensures that the sampling point farther from the feature point its weight is lower. Through the weights of the weighted, in sampling point  $t$  predictive model for:

$$M_t = \frac{\sum_{i=1}^m (w_i M_i)}{\sum_{i=1}^m (w_i)} \quad (24)$$

In the prediction of time domain sampling points are:

$$y_p(t+i) = \frac{\sum_{j=1}^m (w_j(t+i) M_j)}{\sum_{j=1}^m (w_j(t+i))} \quad (25)$$

Among them:

$$\begin{cases} w_j(t+i) = e^{-\frac{E_j(t+i)^2}{E_{\max}}} & -E_{\max} \leq E_j(t+i) \leq E_{\max} \\ w(t+i) = 0 & E_j(t+i) \leq -E_{\max} \text{ 或 } E_j(t+i) \geq E_{\max} \end{cases} \quad (26)$$

By the above method, the linear predictive function of the UAV cruising formation can be obtained:

$$x_{dp}(k+1), x_{dp}(k+2), \dots, x_{dp}(k+N-1), x_{dp}(k+N) \quad (27)$$

$$y_{dp}(k+1), y_{dp}(k+2), \dots, y_{dp}(k+N-1), y_{dp}(k+N) \quad (28)$$

The predictive output value by nonlinear function which contains parameter  $v_w(k), v_w(k+1), \dots, v_w(k+N-1)$  and  $\omega_w(k), \omega_w(k+1), \dots, \omega_w(k+N-1)$  becomes a linear function which contains the above parameters, the control problem becomes into a multi-input multi-output linear predictive control problem.

### 3. Conclusions

By two important scenarios to test the ability of the controller to control the formation, one is the leading UAV do straight level flight, another is the leading UAV do turning flight, as long as be able to complete the cruise fleet under the two scenarios to form and maintain, so generally can complete basic fell cruising formation, and transformation.

Assume that the initial conditions of the formation is as follows:

Table 1 The initial conditions of the UAV formation.

The leading UAV initial position	The following UAV initial position	The leading UAV initial velocity and angular velocity	The following UAV initial velocity and angular velocity	Position measurement error of the leading UAV
(0, 0)	(-100, -100)	(40, 0)	(40, 0)	$\pm 0.5$

(1)The formation simulation for the leading UAV do straight level flight

The predictive time domain  $N=5$ , the sampling interval 0.2s, the UAV angular velocity change rate (-0.1,0.1), the minimum and maximum cruising speed (35,45), the initial Angle for the leading UAV is  $\pi/4$ , in straight level flight cases, assume that the ideal position for the following UAV and the leading UAV in track coordinates is  $X=-60, Y=30$ , the initial relative position in track coordinates is as follows:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} \cos \frac{\pi}{4} & \sin \frac{\pi}{4} \\ -\sin \frac{\pi}{4} & \cos \frac{\pi}{4} \end{bmatrix} \begin{bmatrix} -100 \\ -100 \end{bmatrix} = \begin{bmatrix} -100\sqrt{2} \\ 0 \end{bmatrix} \quad (29)$$

Using Matlab Simulink toolbox to simulate, the simulation results as shown in the figure below:

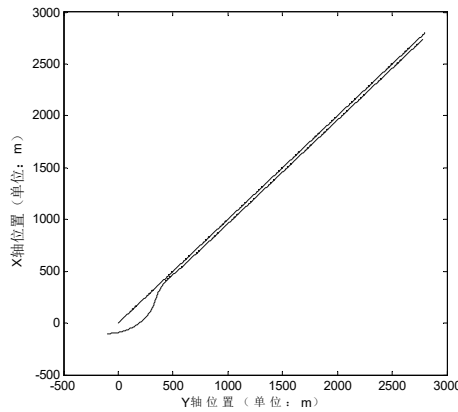


Figure 4 Two aircraft flight path map.

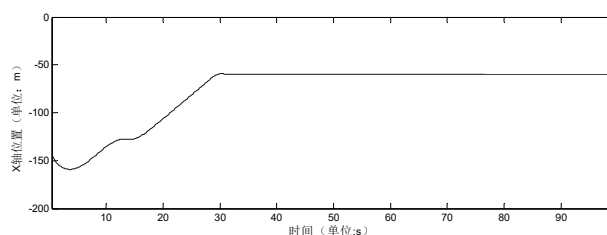


Figure 5 Two machine X axis relative position in track coordinates.

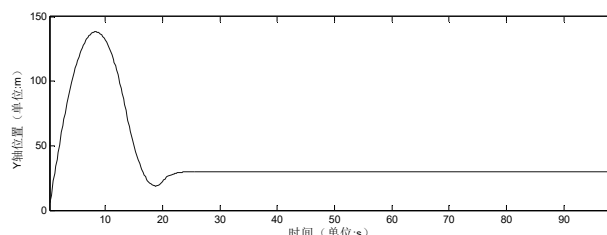


Figure 6 Two machine Y axis relative position in track coordinates.

(2)The formation simulation for the leading UAV do turning flight flight

The leading UAV's initial Angle with the X axis is 0 degree, linear fly for 20 seconds, and then fly with angular velocity  $\pi/200$  for 100 seconds, and then to the direction of the Y axis to linear fly. The predictive time domain  $N=5$ , the sampling interval 0.2s, the UAV angular velocity change rate (-0.1,0.1), the minimum and maximum cruising speed (35,45), assume that the ideal position for the following UAV and the leading UAV in track coordinates is as follows:

$$\begin{bmatrix} x_{dref} \\ y_{dref} \end{bmatrix} = \begin{bmatrix} -50 \\ -50 \end{bmatrix} \quad (30)$$

The initial relative position in track coordinates is as follows:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} \cos 0 & \sin 0 \\ -\sin 0 & \cos 0 \end{bmatrix} \begin{bmatrix} -100 \\ -100 \end{bmatrix} = \begin{bmatrix} -100 \\ -100 \end{bmatrix} \quad (31)$$

Using Matlab Simulink toolbox to simulate, the simulation results as shown in the figure below:

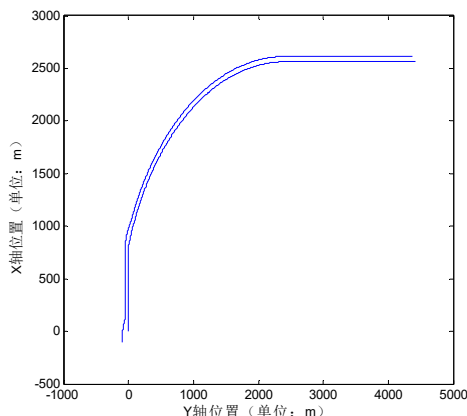


Figure 7 Two aircraft flight path map.



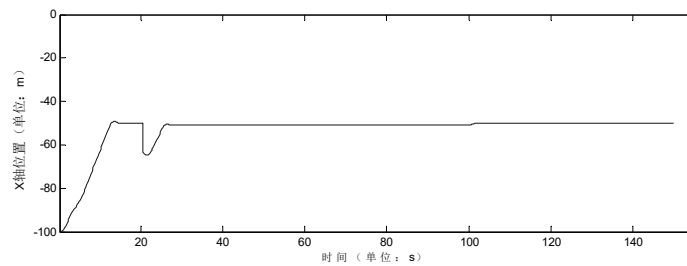


Figure 8 Two machine X axis relative position in track coordinates.

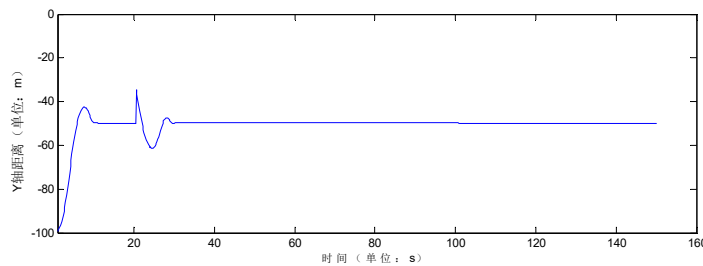


Figure 9 Two machine Y axis relative position in track coordinates.

According to the above two scenarios of UAV formation cruise Matlab simulation , when the sampling interval is 0.2 seconds, in a PC running a rolling optimization procedure of simulation time is lower than 0.2 seconds, when the special chip in high speed running will faster, and can meet the real-time requirement of the control obviously. By track coordinates two machine the relative position of the X axis and Y axis curve can be seen, UAV formation in 40 seconds. Can be seen by simulation based on multi-model predictive control for UAV cruising formation control can not only meet the control requirements, but also can satisfy the real-time requirements.

## References

- [1] Jinnan Zhang.(2008)Guided Bomb Multi-model and Predictive Control Technology Research[D].Changsha, National University of Defense Technology.
- [2] Guolong Hu,Youxian Sun.(2012)The Research Progress of Multi-model Control Method and Its Application Status[J].Information and Control,33(1),72-76.
- [3] Yugeng Xi,Wangfan.(2010)Multi-model Predictive Control of Nonlinear System Method[J]. Acta Automatica, 22(4),456-460.
- [4] Zhikui Zhao,Zongzhao Han,Baiwu Wan.(2013)Multi-model Adaptive Control and Its Application in Three-axis Turntable[J].Control and Decision Making,13(3),212-217.
- [5] Jianfu Cao,Zongzhao Han,Yangwang Wan.(2009)Nonlinear System Theory and Application[M].Xian,Xi 'an Jiaotong University Press.
- [6] Daoying Pi,Youxian Sun.(2014)Multi-model MIMO System of Weighted Fuzzy Control Strategy[J].Acta Automatica,24(3),387-390.
- [7] Narendra KS, Cheng Xiang. (2012)Adaptive Control of Discrete-time Systems Using Multi-models [J]. IEEETrans on Automatic Control,45(9),1669-1686.
- [8] Danielle,D, Doug,C. (2014)A Practical Multi-model Adaptive Strategy for Single-loop MPC [J].Control Engineering Practice,11(2),141-159.