An Improved Fuzzy Predictive Control Algorithm

Deng wenl-lian

Computer and Information Engineering Department, Heze University, Heze, 274015, China,
E-mail:dxy5dwl@163.com

Abstract

In this paper, we investigate fuzzy predictive control algorithm based on T-S. The main calculation of the predictive control lies in solving quadratic programming problem, and this method need to solve \( M+1 \) times. If predictive control time domain take great value, calculation quantity will increase exponentially, which reduce the real-time of control algorithm. We propose a simplified calculation method, which make up for the lack of single-step linear model prediction, and has a smaller amount of calculation compared to the multi-step linear prediction method. In a control cycle, proposed algorithm only performs two objective function optimization calculation. We also do experiment to verify the correctness of proposed algorithm and obtain that our proposed algorithm is significantly better than the single-step linearization method.

Keywords: Fuzzy Predictive Control Algorithm, Error Compensation Correction, Control Accuracy, Complexity

1. Introduction

Fuzzy control and predictive control are independently developed two areas of control theory. Fuzzy predictive control has its inherent reasonableness, which is formed by the idea of fuzzy thinking and predictive control thinking[1-6]. Fuzzy control and predictive control will further improve the control effect. The development trend of Fuzzy control is the transformation of rules to model[7-10], and the object model of predictive control can be used as the communication bridge. Predictive control is a kind of precise control method based on the object mathematical model. The complexity of the system and accuracy that analysis system can achieve are restricted each other. Therefore, predictive control under the fuzzy environment is of great significance for expanding the application scope of predictive control.

The remainder of this paper is set out as follows. In the next section, we introduce the basic algorithm of T-S fuzzy model. In Section 3 we puts forward a simplified algorithm that uses error compensation method. In Section 4, we give an illustrative example to verify the developed approach and to demonstrate its feasibility and practicality. In Section 5 we conclude the paper and give some remarks.

2. Preliminaries

Nonlinear system control based on T-S fuzzy model mainly use approximation ability of fuzzy reasoning for nonlinear system. The nonlinear system is divided into multiple linear system, then design the controller by using linear system theory[11,12].

The nonlinear object of a controlled system is

\[
y(k) = g(y(k-1), \ldots, y(k-n), u(k), \ldots, u(k-m-1))
\]

(1)

\( g(\cdot) \) is nonlinear function, \( y(k-1), \ldots, y(k-n) \) is the input sequence of system, \( u(k), \ldots, u(k-m-1) \) is the output sequence. \( n, m \) is the order of system. \( \varphi = [y(k-1), \ldots, y(k-n), u(k), \ldots, u(k-m-1)] \), \( Z \) is data sets for all of the system input and output data, and can be divided into \( c \) cluster center. \( Z = [z^1, \ldots, z^c] \). The number of rule of T-S fuzzy model of the system is \( c \), that can be expressed as
\( R^i : \text{if } \varphi \in z^i, \text{then } y^i(k) = f^i(\varphi) \)
\[
\vdots
\]
\( R^i : \text{if } \varphi \in z^i, \text{then } y^i(k) = f^i(\varphi) \)
\( R^e : \text{if } \varphi \in z^e, \text{then } y^e(k) = f^e(\varphi) \)

\( y^i(k) = f^i(\varphi) \) is local linear model of every cluster center, the output of fuzzy model is

\[
y_m(k) = \sum_{i=1}^{k} u_i y^i(k) / \sum_{i=1}^{k} u_i
\]  

(3)

\( u^i \) is the membership degree of the ith rule. \( u^i = u^i_z(\varphi) \). Formula 3 can be expressed as

\[
y_m(k) = \sum_{i=1}^{k} w^i y^i(k)
\]  

(4)

\[
w^i = u^i / \sum_{j=1}^{k} u^j
\]  

(5)

Formula 4 is predictive controller, Rolling optimization objective function is

\[
J(k) = \sum_{j=1}^{N} q_j [\hat{y}(k+j) - y^*_r(x + j)]^2 + \sum_{j=1}^{M} \lambda_j \Delta u(k + j - 1)^2
\]  

(6)

\( N \) is the length of the prediction horizon; \( M \) is the length of the control horizon, \( q_j, \lambda_j \) is predictive deviation and weighting factor of control increment respectively. \( \hat{y}(k+j) \) is the output of jth step by formula 3, \( y^*_r(k+j) \) is reference trajectory of jth step. \( \Delta u(k + j - 1) \) is control increment of (j-1)th.

3. A low-complexity fuzzy predictive control algorithm

Formula 2 takes a discrete state space model then then T-S fuzzy model can be expressed as

\[
R^i : \text{if } \varphi \in z^i, \text{then } \begin{cases} x'(k+1) = A^i x(k) + B^i u(k) \\ y^i(k) = C^i x(k) \end{cases}
\]  

(7)

Global model at the time \( k \) is

\[
\begin{cases}
    x(k+1) = A_k x(k) + B_k u(k) \\
    y(k) = C_k x(k)
\end{cases}
\]  

(8)
\[ A_k = \sum_{i=1}^{c} w^i A^i, \quad B_k = \sum_{i=1}^{c} w^i B^i, \quad C_k = \sum_{i=1}^{c} w^i C^i \]

\[ A^i \text{ is system matrix of local linear model,} \]

\[ B^i \text{ is control matrix of local linear model,} \]

\[ C^i \text{ is output matrix of local linear model.} \]

\[ A^i_k \text{ is system matrix of global linear model,} \]

\[ B^i_k \text{ is control matrix of global linear model,} \]

\[ C^i_k \text{ is output matrix of global linear model,} \]

Three matrix are time invariant. \( y(k+j|k) \) is predictive value of the time \( k \) to the time \( k+j \), and we obtain the equation as formula 9.

\[ N \text{ and } M \text{ are respectively predict time domain and control time domain.} \]

\[ y(k+1|k) = C_{k+1} A_k x(k) + C_{k+1} B_k u(k) \]

\[ y(k+2|k) = C_{k+2} A_{k+1} A_k x(k) + C_{k+2} A_{k+1} B_k u(k) + C_{k+2} B_{k+1} u(k+1) \]

\[ \vdots \]

\[ y(k+N|k) = C_{k+N} A_{k+N-1} \cdots A_k x(k) + C_{k+N} A_{k+N-1} \cdots A_{k+2} B_{k+2} u(k) + \cdots + C_{k+N} A_{k+N-1} \cdots A_{k+M-2} B_{k+M-2} u(k+M-1) \]

Formula 9 is expressed as matrix form

\[ Y = Gx(k) + FU \]

In formula 10,

\[ Y = [y(k+1|k), y(k+2|k), \ldots, y(k+N|k)]^T, \quad G = \begin{bmatrix} C_{k+1} & 0 & \cdots & 0 \\ C_{k+2} A_{k+1} B_k & C_{k+2} B_{k+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_{k+N} A_{k+N-1} \cdots A_{k+2} B_{k+2} & C_{k+N} A_{k+N-1} \cdots A_{k+2} B_{k+2} & \cdots & C_{k+N} A_{k+N-1} \cdots A_{k+M-2} B_{k+M-2} \end{bmatrix} \]

\[ U = [u(k), u(k+1), \ldots, u(k+M-1)]^T \]

And it can be expressed as formula 11.

\[ Y = Gx(k) + F_1 \Delta U + F_2 u(k-1) \]

In formula 11, \( F_1 = \begin{bmatrix} C_{k+1} B_k \\ C_{k+2} (A_{k+1} B_k + B_{k+1}) \\ \vdots \\ C_{k+N} \left( \sum_{i=0}^{M-1} A_{k+N-i} \cdots A_{k+i+1} B_{k+i+1} \right) \end{bmatrix} \]
\[ F_2 = \begin{bmatrix} C_{k+N}B_k & C_{k+2}B_{k+1} \\ \vdots & \vdots \\ C_{k+N}(\sum_{j=0}^{M-1} A_{k+N-j} \cdots A_{k+j+1}B_{k+j}) & C_{k+N}(\sum_{j=0}^{M-1} A_{k+N-j} \cdots A_{k+j+1}B_{k+j}) \\ \vdots & \vdots \\ \vdots & \vdots \\ C_{k+N}A_{k+N-1} \cdots A_{k+M-2}B_{k+M-2} & C_{k+N}A_{k+N-1} \cdots A_{k+M-2}B_{k+M-2} \\ \vdots & \vdots \\ C_{k+N}(\sum_{j=0}^{M-1} A_{k+N-j} \cdots A_{k+j+1}B_{k+j}) & C_{k+N}(\sum_{j=0}^{M-1} A_{k+N-j} \cdots A_{k+j+1}B_{k+j}) \end{bmatrix} \]

\[ \Delta U = [\Delta u(k), \Delta u(k+1), \cdots, \Delta u(k+M-1)]^T. \]

Vector form of formula 6 is

\[ J(k) = (Y - Y_r)^T Q(Y - Y_r) + \Delta U^T R \Delta U \quad (12) \]

We solve the increment \( \Delta U \) by a secondary gauge stroke, control amount at the moment is

\[ u(k) = u(k-1) + \Delta u. \]

The key issues of T-S fuzzy model predictive control algorithm of is how to take advantage of the linear time-varying model to predict the behavior of charged objects in the next moment. If using a single-step linear model to predict when the control object is strongly nonlinear, multi-step vector form of formula 6 is ahead forecast deteriorate control performance due to large model errors[13,14]. However, when the control object has a relatively weak nonlinear characteristics, the control performance impact of the processing methods is not big. Due to the predictive control is the whole control process rolling optimization. If the multi-step linearizing model is used to predict future output behavior, we need repeat solving predictive control law and multi-step linearizing model in the prediction time domain. The main calculation of the predictive control lies in solving quadratic programming problem, this method need to solve M+1 times. If predictive control time domain take great value, calculation quantity will increase exponentially[15-18], which reduce the real-time of control algorithm. This section propose a simple approximation algorithm starting from the basic principles of predictive control. Predictive control has three basic elements: predictive model, roll optimization and error correction[19,20]. It is due to introduction error feedback correction, that make the rolling optimization based on prediction model output feedback, which overcome the existing uncertainty, model mismatch and various interference effect in the system, and increase the control precision and robustness of the system. Therefore, the model accuracy requirement of predictive control is relatively low relative to other model based control algorithm. In order to improve the performance of the predictive control, in addition to improving the precision of the prediction model and choosing appropriate rolling optimization method, we still can select proper error compensation method. The proposed algorithm is as follows.

Step1. Calculate control signal \( u(k), u(k+1), \cdots, u(k+M-1) \) and \( \hat{y}_m(k+1), \hat{y}_m(k+2), \cdots, \hat{y}_m(k+N) \), according to the linear model \( \{ A_k, B_k, C_k \} \) at the samling instant \( k \) in the whole predictive time domain.

Step2. Calculate \( \hat{y}(k+1) \), use formula 7 and formula 8 to calculate linear model \( \{ A_{k+1}, B_{k+1}, C_{k+1} \} \) according to \( \hat{y}(k+1) \) and \( u(k) \) at the time \( k+1 \).

Step3. Calculate \( \hat{y}(k+2) \), use formula 7 and formula 8 to calculate linear model \( \{ A_{k+2}, B_{k+2}, C_{k+2} \} \) according to \( \hat{y}(k+2) \) and \( u(k+1) \) at the time \( k+1 \).
Step 4. Repeat these steps throughout the whole prediction time domain until 
\( \hat{y}(k+1), \hat{y}(k+2), \cdots, \hat{y}(k+N) \) is all calculated.

Step 5. Come to sampling instant \( k \), rectify output prediction of \( \{A_k, B_k, C_k\} \) in the future \( N \) steps according to the difference between 
\( \hat{y}(k+1), \hat{y}(k+2), \cdots, \hat{y}(k+N) \) and \( \hat{y}_m(k+1), \hat{y}_m(k+2), \cdots, \hat{y}_m(k+N) \). Re-calculating control signal \( u(k) \) and acting on the control object.

This simplified calculation method make up for the lack of single-step linear model prediction, to a certain extent. Meanwhile, compared to the multi-step linear prediction method, it has a small amount of calculation, and a control cycle only performs two objective function optimization calculation.

4. Illustrative example

Non-linear model of system is

\[
y(k) = 0.72y(k-1) + 0.1y(k-2)u(k-2) + 0.5u^2(k-3) + 0.2u(k-4)
\] (13)

Establish the system T-S fuzzy model using three rules.

\[R^1: \text{If } y(k-1) \text{ is } A^1_i \text{ and } y(k-2) \text{ is } A^2_i \text{ and } u(k-2) \text{ is } A^3_i \text{ and } u(k-3) \text{ is } A^4_i \text{ and } u(k-4) \text{ is } A^5_i \text{ then } y(k) = 1.257y(k-1) - 0.395y(k-2) - 0.045u(k-2) - 0.6u(k-3) + 0.452u(k-4) + 0.177\]

\[R^2: \text{If } y(k-1) \text{ is } A^2_i \text{ and } y(k-2) \text{ is } A^2_i \text{ and } u(k-2) \text{ is } A^3_i \text{ and } u(k-3) \text{ is } A^4_i \text{ and } u(k-4) \text{ is } A^5_i \text{ then } y(k) = 1.310y(k-1) - 0.372y(k-2) - 0.406u(k-2) + 2.442u(k-3) - 1.154u(k-4) - 0.24\]

\[R^3: \text{If } y(k-1) \text{ is } A^3_i \text{ and } y(k-2) \text{ is } A^2_i \text{ and } u(k-2) \text{ is } A^3_i \text{ and } u(k-3) \text{ is } A^4_i \text{ and } u(k-4) \text{ is } A^5_i \text{ then } y(k) = 1.513y(k-1) - 0.596y(k-2) + 2.954u(k-2) - 1.060u(k-3) - 0.306u(k-4) - 1.022\]

Membership function is Gaussian function \( f(x) = \exp\left(-\left(x - c\right)^2 / 2\sigma^2\right) \), the parameters are shown in Table 1. In order to verify the accuracy of the model, input signal is \( u = 0.5\sin(0.02\pi) + 0.5 \), the output of model and object are shown as figure 1, in which black curve represents the output of model, and blue curve represents the output of object. We can see that both cure are substantially coincident.

<table>
<thead>
<tr>
<th>Rule NO</th>
<th>parameter</th>
<th>(A^1_i)</th>
<th>(A^2_i)</th>
<th>(A^3_i)</th>
<th>(A^4_i)</th>
<th>(A^5_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i = 1)</td>
<td>(\sigma)</td>
<td>7.949</td>
<td>7.949</td>
<td>1.061</td>
<td>1.061</td>
<td>1.068</td>
</tr>
<tr>
<td></td>
<td>(\sigma)</td>
<td>0.983</td>
<td>1.181</td>
<td>0.191</td>
<td>0.229</td>
<td>0.257</td>
</tr>
<tr>
<td>(i = 2)</td>
<td>(\sigma)</td>
<td>7.949</td>
<td>7.949</td>
<td>1.061</td>
<td>1.065</td>
<td>1.067</td>
</tr>
<tr>
<td></td>
<td>(\sigma)</td>
<td>6.078</td>
<td>5.474</td>
<td>1.588</td>
<td>1.533</td>
<td>1.478</td>
</tr>
</tbody>
</table>
Table 1. Simulation results of the controller

<table>
<thead>
<tr>
<th>i = 3</th>
<th>σ</th>
<th></th>
<th>C</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>7.953</td>
<td>7.949</td>
<td>1.061</td>
<td>1.061</td>
<td>1.061</td>
<td></td>
</tr>
<tr>
<td></td>
<td>18.055</td>
<td>17.697</td>
<td>1.809</td>
<td>1.844</td>
<td>1.876</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. Comparison between model output and object output

Figure 2. Simulation result of the controller

Figure 2 are the simulation results of controller, controll horizon is 4, the prediction horizon is 10, at the start of the simulation, established value steps from 0 to 1, it rises to 2.5 in the 40-step, and lasts for 40 steps, then steps down to 1. We can conclude that simplified multi-step linear error correction method is significantly better than the single-step linearization method.

5. Conclusions

We investigate fuzzy predictive control algorithm based on T-S fuzzy method in this paper, which exists a large amount of calculation. This paper puts forward a simplified algorithm that uses error compensation method. It uses correction prediction error method to improve control accuracy and reduce the complexity of computation.
6. References


