Real Time Control of Twin Rotor MIMO System using Intelligent Controller

Santosh Sadashiv Shinde¹ Abhishek J. Bedekar² Prof. Nitin V. Patel³

Walchand College Of Engg., Sangli

Abstract— This paper discusses the scope of designing and implementing a intelligent hybrid PID control technique to the Twin Rotor Multi-input multi-output System (TRMS). Control of nonlinear system has always been a challenging problem due severe nonlinearities, inaccessibility of some states and output for measurement with significant cross coupling. The control objective is to make this systems move quickly so that it track the given reference inputs accurately. Real time performance of Twin Rotor Multi-input multi-output System (TRMS) with the intelligent controllers has been shown.

Key words: Hybrid; Neural, fuzzy; Twin Rotor Multi Input Multi Output System (TRMS); Nonlinear

I. INTRODUCTION

Controlling nonlinear systems in real physical systems is a difficult problem due to flexible dynamics. The TRMS is a laboratory set up designed for control experiment. With some significant simplification it serves as a model of helicopter. TRMS is benchmark problem for many important application like control of helicopter. The overall mathematical model is supposed to be needless in this paper with respect to the adjustable parameters of PID controller therefore designed process is brief but comprehensive and therefore controller is assumed to be more robust.

This paper is categorized as follows, Section II consist of introduction to Twin Rotor Multi-input multi-output System (TRMS). Section III deals with the method of designing the intelligent controller. In section IV, the simulation results are presented and last section consists of conclusion.

II. TWIN ROTOR MIMO SYSTEM

The TRMS consist of beam, such that it can rotate freely both in vertical and horizontal planes as it pivoted on its base. Two propellers are present at both ends of the beam perpendicular to each other which are driven by permanent magnet DC motors.

Fig. 1: Twin Rotor MIMO system

The mathematical model of the main rotor,

\[ \dot{x} = \begin{bmatrix} x \end{bmatrix}, \quad \dot{y} = \begin{bmatrix} y \end{bmatrix}, \quad \dot{\alpha} = \begin{bmatrix} \alpha \end{bmatrix}, \quad \dot{\beta} = \begin{bmatrix} \beta \end{bmatrix}, \quad \dot{\gamma} = \begin{bmatrix} \gamma \end{bmatrix}, \quad \dot{\omega}_1 = \begin{bmatrix} \omega_1 \end{bmatrix}, \quad \dot{\omega}_2 = \begin{bmatrix} \omega_2 \end{bmatrix} \]

Where,

\[ A = \begin{bmatrix} \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \end{bmatrix} \]

\[ B = \begin{bmatrix} \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \end{bmatrix} \]

\[ C = \begin{bmatrix} \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \\ \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} & \frac{m_a}{2} \end{bmatrix} \]

\[ \alpha, \alpha_p : \text{Horizontal and vertical position of TRMS beam}, \]

\[ \alpha_h, \beta : \text{Horizontal and vertical angular velocity of TRMS beam}, \]

\[ U_h, U_z : \text{Horizontal and vertical DC motor voltage control input}, \]

\[ P_r, P_v : \text{Nonlinear part of DC motor with tail rotor and main rotor}, \]

\[ w_c, w_m : \text{Rotational speed of tail rotor and main rotor}, \]

\[ l_f, l_m : \text{Effective arm of aerodynamic force from tail and main rotor}, \]

\[ J_f, J_m : \text{Nonlinear function of moment of inertia with respect to horizontal axis and vertical axis}, \]
\( I_{tr}, I_{mr} \) Moment of inertia in DC motor tail and main propeller subsystem,

\( S_h, S_v \) Angular momentum in horizontal plane and vertical plane for the beam,

\( s_e \) Constant scaling factor,

\( k_h, k_v \) Positive constant,

\( F_h, F_v \) Moment of friction force in horizontal plane and vertical plane,

\( m_{mr} \) Mass of the main DC motor with the main rotor,

\( m_m \) Mass of the main part of the beam,

\( m_r \) Mass of the tail motor with the tail rotor,

\( m_b \) Mass of the tail part of the beam,

\( m_{cw} \) Mass of the counter-weight,

\( m_{mb} \) Mass of the counter-weight beam,

\( m_s \) Mass of the main shield,

\( m_t \) Mass of the tail shield,

\( l_p \) Length of the counter-weight beam,

\( l_{eh} \) Distance between the counter-weight and the joint,

\( u_{mg} \) Output of the vertical DC motor \( g \) Gravitational acceleration.

Fig. 2: Gravity and propulsive forces in the vertical plane.

The mathematical model of the tail rotor,

\[
\frac{dS_h}{dt} = l_s S_f F_h (w_r) \cos \omega_r - \Omega h_k \sin \omega_r k_h
\]

(7)

\[
\frac{dS_v}{dt} = \Omega h_k S_f \sin \omega_r + w_{mc} \cos \omega_r \frac{\sin \omega_r^2 + \cos \omega_r^2 + F}{u_{hh}}
\]

(8)

\[
u_{hh} = \frac{T_{tr} s + 1}{u_{hh}}
\]

(9)

\[
\frac{du_{hh}}{dt} = \frac{1}{T_{tr}} \left( -u_{hh} + u_h \right)
\]

(10)

\[
w_{mc} = P_h (u_{hh})
\]

(11)

Where,

\( r_{ms} \) Radius of the main shield,

\( r_{ts} \) Radius of the tail shield,

\( u_{hh} \) Output of the horizontal DC motor.

III. HYBRID PID CONTROLLER USING NEURAL NETWORK

A. Neural Network Design Steps

The neural networks are used to solve problems in four application areas: pattern recognition, clustering, function fitting, and time series analysis. The procedure for any of these problems has six primary steps. (Data collection generally occurs outside the MATLAB environment, so it is step 0.)

1) Collection of data
2) Creation of the neural network
3) Configuration of the neural network
4) Initialization of the weights and biases
5) Training of the neural network
6) Validation of the network
7) Use the neural network.

For designing neural network, input and target data is required in certain format.

For controlling Twin rotor MIMO system we required a network having three inputs and single output. As the neural network has to be incorporated after PID and has to give output to the system to be controlled, input for neural network taken as the output of P, I, D controllers and For training target, data taken as the output of PID controller.

The capability of neural networks at fitting functions is well known. A fairly simple neural network has ability to fit in any practical function.

Fig. 4: Proposed structure of neural network

Two-layer feed forward network used. Levenberg-Marquardt back propagation method used to train proposed neural network. The Levenberg-Marquardt algorithm is a variation of Newton’s method that was designed for minimizing function that are sums of squares of other nonlinear functions this is very well suited to neural network training where the performance index is the mean square error.

1) The Levenberg-Marquardt algorithm:

\[
x_{k+1} = x_k - (J^T J + \mu I)^{-1} J^T e
\]

This algorithm has the very useful features that as \( \mu \) is increased it approaches the steepest descent algorithm with small learning rate.

When the scalar \( \mu \) is decreased to zero the algorithm becomes Gauss-Newton. When is large, this becomes gradient descent with a small step size.

The algorithm provides a nice compromise between the speed of Newton’s method and the guaranteed convergence of steepest descent [7–8].

IV. SIMULATION RESULTS

A. Twin Rotor MIMO system

In this section simulation results of highly nonlinear TRMS, considering following experiments -

1) 1 DOF pitch rotor control
2) 1 DOF yaw rotor control
3) 2 DOF control

Fig. 5 shows the step response of the twin rotor MIMO system in vertical plane for reference input 0.8 using a
hybrid neural network controller. Fig. 6 shows the step response of the twin rotor MIMO system in horizontal plane for reference input 0.5 using a hybrid neural network controller. Fig. 6 (a) and (b) shows the step response of the twin rotor MIMO system in vertical and horizontal for reference input 0.8 & 0.5 respectively using a hybrid neural network controller.

B. Experimental Validation
The designed hybrid neural network controller has been tested on real time set up of twin rotor MIMO system (TRMS). The responses, as shown in Fig. 6(a) and 6(b) depict efficacy of the controllers.

In this paper, a hybrid neural network technique for controlling TRMS has been developed and implemented. The simulation results show the new approach to control the nonlinear problem for the positioning and tracking performances in real time.

REFERENCES


