Adaptive Flight Control for Quadrotor UAVs with Dynamic Inversion and Neural Networks*

1Tian Xiang, 1Fan Jiang, 1Qi Hao, and 2Wang Cong

Abstract—In this paper, we develop an adaptive nonlinear controller based on dynamic inversion and neural network for quadrotor UAVs in the presence of uncertainties in UAV and actuator dynamics. The basic control law is first designed by using conventional PID control, and then the nonlinear dynamic inversion control is provided for the purpose of stabilization and robustness. The neural network is used to eliminate the inversion error due to parameter uncertainty, disturbance, etc. The simulation results demonstrate that NN shows better performance than PID control in the presence of the external disturbances which can improve the robustness of the whole system and achieve accurate attitude and trajectory control.

I. INTRODUCTION

Recently, quadrotor UAV has become increasingly popular due to its simple structure, low cost, high maneuverability, vertical take-off and landing (VTOL) ability, and capability of hovering and easy maintenance [1]. With the development of MEMS sensors, artificial intelligence (AI) technology and computer vision technology, quadrotors have witnessed immense growth in the field of search and rescue operations, surveillance, scientific research, aerial surveying of crops, and forest fire detection, etc.

The quadrotor drone exhibits some basic problems which includes under-actuation, strong coupling, and non-linearity. Besides, there also exists various uncertainties such as external disturbances and uncertain parameters. Therefore, the autonomous control of a quadrotor UAV has been a core subject of much research. PID control algorithm is the most commonly used approach for the UAV controller design thanks to its simplicity and stability. However, classic PID has drawbacks in handling uncertainties and external disturbances. As a result, algorithms like fuzzy adaptive PID [2], linear quadratic regulator (LQR) [3-4], backstepping and sliding-mode approach [5], H* with feedback linearization [6], and model predictive control [7-8] were introduced in flight control, and have established a solid foundation for subsequent research.

Dynamic inversion [9] is effective in the control of both linear and nonlinear systems which can be applicable to different flying vehicles and well adapted to the changes of those models. But it also has some limitations, such as the need to accurately linearize the model and may inevitably introduce the modeling errors. Thus the dynamic inversion method needs to be designed in combination with other control laws to improve its robustness and stability. The inclusion of dynamic neural network [10–12] is capable of eliminating the dynamic inversion errors caused by the error in the linearized inversion model as well as the parametric uncertainties, which in turn improves the controller stability and tracking performance.

In this paper, we present a neural network based dynamic inversion controller to handle the uncertainties in the control system. This method can guarantee the convergence of control errors. According to the adjustment of the weights of Sigma-Pi neural network, external disturbance and noises can be compensated. The robustness and performance is verified through simulations. Compared with the neural network discussed above, our scheme uses random optimization of weights, which is easy to implement in real time within the UAV and shows good performance according to the simulation results.

The rest of this paper is organized as follows. Section II describes the basic dynamic model of the quadrotor. Section III presents the architecture of the NN augmented dynamic inversion model as applied to a quadrotor. Section IV provides result and discussions. Section V concludes this work and outlines future work.

II. QUADROTOR DYNAMICS

A. The Quadrotor UAV

A quadrotor is a multirotor helicopter that produces lift with four fixed-pitched rotors in a plane. Fig. 1 shows a basic model of an unmanned quadrotor. It generally uses two pairs of identical propellers: the front and rear motors rotate counter clockwise (CCW) while the other two rotate clockwise (CW). By changing the speed of each rotor it is possible to specifically generate a desired total thrust. Lifting and landing movement is achieved by increasing or decreasing all the four motors. Roll movement is obtained by varying the speed of the left and right motors. Pitch movement is obtained similarly using the front and rear motors. And yaw movement is controlled by increasing (decreasing) the speed of the front and rear motors and decreasing (increasing) the speed of the lateral motors. The quadrotor is under-actuated since it has six degrees of freedom (three rotational and three translational) but only four actuators. Therefore, only four of the six DOF can be controlled and stabilized.

B. Flight Dynamics

Consider an earth-fixed inertial frame $E=\{O_{x,y,z}\}$ and a body-fixed frame $B=\{O_{x,y,z}\}$ whose origin $O$ is at the center...
of mass of the quadrotor, as shown in Fig. 1. Defined \( p = (x, y, z) \) to represent the position of the quadrotor with respect to the inertial frame and \( \Theta = (\Phi, \theta, \Psi) \) to represent the attitude of the quadrotor. Furthermore, \( V = (u, v, w) \) describes the linear velocity of the quadrotor in the body frame and \( \omega = (p, q, r) \) representing the rotational velocity of the quadrotor within the body frame.

According to Newton’s law, the kinematics and dynamics equations can be described as

\[
\sum F = \frac{dP}{dt} = m\dot{V}_{\text{b}} + \omega_{\text{b}} \times (mV_{\text{b}}^2)
\]

\[
\sum M = \frac{dL}{dt} = I\dot{\omega}_{\text{b}} + \omega_{\text{b}} \times (I\omega_{\text{b}})
\]

where \( [\omega_{\text{b}} \times ] \) is a skew-symmetric matrix and defined as follows:

\[
[\omega_{\text{b}} \times] = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}
\]

(2)

Separately, for translational motion, the combined force can be denoted by (in body fixed frame)

\[
\sum F^b = F^b_T - F^b_D - C_{\text{mg}} F^b_U
\]

(3)

where

\[
F^b_T = \begin{bmatrix} 0 \\ 0 \\ c_f (\sigma_1^2 + \sigma_2^2 + \sigma_3^2 + \sigma_4^2) \end{bmatrix}, F^b_D = \begin{bmatrix} k_d x \\ k_d y \\ k_d z \end{bmatrix}, F^b_U = \begin{bmatrix} mg^2 \end{bmatrix}
\]

Thus the translational motion equation can be described as

\[
V^b_{\text{b}} = \left( \frac{1}{m} \right) (F^b_T - F^b_D) - g^b - \omega_{\text{b}}^2 \times V^b_{\text{b}} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}
\]

(4)

\[
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = C_{\text{ub}} V^b_{\text{b}}
\]

(5)

where \( m \) denotes the quadrotor mass, and \( g \) is the gravity acceleration, and

\[
C_{\text{ub}} = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ -\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi & \cos \phi \cos \psi + \sin \phi \sin \psi & \sin \phi \cos \theta \\ \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \cos \phi \cos \psi - \sin \phi \sin \psi & \cos \phi \cos \theta \end{bmatrix}
\]

For angular motion, the moment can be denoted by (in body fixed frame)

\[
\sum M^b = M^b_T - M^b_D
\]

(6)

Define the thrust coefficient and the drag coefficient as \( c_f \) and \( c_d \), if the distance from the rotors to the centre of mass is denoted by \( d \), then the control torques generated by the four rotors are

\[
\tau = \begin{bmatrix} dc_f (\sigma_2^2 - \sigma_3^2) \\ dc_f (\sigma_3^2 - \sigma_1^2) \\ c_d (\sigma_2^2 + \sigma_3^2 - \sigma_1^2 - \sigma_2^2) \end{bmatrix}
\]

(7)

Considering the gyroscopic effects and disturbances, then the torques are

\[
M^b_T = \begin{bmatrix} dc_f (\sigma_2^2 - c_d \sigma_3^2 + J_s \Omega \left( \frac{\pi}{30} \right) (\sigma_1 - \sigma_2 + \sigma_3 - \sigma_4) \\ -dc_f (\sigma_3^2 + c_d \sigma_2^2 + J_s \Omega \left( \frac{\pi}{30} \right) (\sigma_1 - \sigma_2 + \sigma_3 - \sigma_4) \\ -c_d \sigma_1^2 + c_d \sigma_2^2 - c_d \sigma_3^2 + c_d \sigma_4^2 \end{bmatrix}
\]

(8)

\[
M^b_D = \begin{bmatrix} k_\phi \phi \\ k_\psi \psi \\ -k_\theta \theta \end{bmatrix}
\]

(9)

Thus the angular motion equation can be described as

\[
\omega_{\text{b}} = I^{-1} \left[ M^b_T - M^b_D - \omega_{\text{b}}^2 \times (I\omega_{\text{b}}) \right] = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}
\]

(10)

\[
\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} 1 & t(\theta) & s(\theta) \\ 0 & c(\phi) & -s(\phi) & c(\theta) \\ 0 & s(\phi) & c(\phi) & c(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = H \omega_{\text{b}}^2
\]

(11)

Define the state vector \( X = [p, q, r, \phi, \theta, \psi, u, v, w, x, y, z]^T \), the input vector \( U = [\sigma_1, \sigma_2, \sigma_3, \sigma_4]^T \), and the system modeling equations can be finally derived as

\[
f(X, U) = \dot{X} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{m} \left( F^b_T - F^b_D \right) - g^b - \omega_{\text{b}}^2 \times V^b_{\text{b}} \\ H \omega_{\text{b}}^2 \\ \frac{1}{m} \left( F^b_T - F^b_D \right) - g^b - \omega_{\text{b}}^2 \times V^b_{\text{b}} \\ C_{\text{ub}} V^b_{\text{b}} \end{bmatrix}
\]

(12)
III. NN DYNAMIC INVERSION CONTROL

A. Conventional PID Control Structure

Fig. 2 gives the conventional feedback control structure, which is composed of the PID linear controller, the system model and sensor measurement. Based on this structure, one can develop algorithms to fully control a set of four states of the quadrotor. That is, either x, y, z, ψ or φ, θ, z, ψ can be fully controlled simultaneously.

![Figure 2](image)

Figure 2. Conventional PID control structure of a quadrotor

- **Attitude controller (inner loop)**

\[
U = \begin{bmatrix} U_p \\ U_y \\ U_z \end{bmatrix} = \begin{bmatrix} K_{p\phi} (\phi_0 - \phi) + K_{d\phi} (\dot{\phi}_0 - \dot{\phi}) + K_{i\phi} \int (\phi_0 - \phi) d\phi \\ K_{p\theta} (\theta_0 - \theta) + K_{d\theta} (\dot{\theta}_0 - \dot{\theta}) + K_{i\theta} \int (\theta_0 - \theta) d\theta \\ K_{p\psi} (\psi_0 - \psi) + K_{d\psi} (\dot{\psi}_0 - \dot{\psi}) + K_{i\psi} \int (\psi_0 - \psi) d\psi \end{bmatrix}
\]

(13)

The inner loop is designed to control the attitude of the quadrotor, where the φ0 and θ0 are derived as the function of x, y, z, ẋ, ẏ, ż and Ψ, and Φ, θ, Ψ are the output of the plant dynamics which are measured by sensors.

- **Position controller (outer loop)**

\[
\begin{aligned}
u_1 &= K_m (x_0 - x) + K_d (x_0 - x) + K_i \int (x_0 - x) \\
u_2 &= K_m (y_0 - y) + K_d (y_0 - y) + K_i \int (y_0 - y) \\
u_3 &= K_m (z_0 - z) + K_d (z_0 - z) + K_i \int (z_0 - z) \\
\end{aligned}
\]

(14)

\[
\begin{aligned}
\phi &= \text{atan} \left( \frac{u_1 \sin \psi - u_2 \cos \psi}{u_1^2 + u_2^2 + (u_1 + g)^2} \right) \\
\theta &= \text{atan} \left( \frac{u_1 \cos \psi + u_2 \sin \psi}{u_1 + g} \right)
\end{aligned}
\]

(15)

The outer loop is designed to control the position of the quadrotor, which is applied in the traditional PID control method.

B. Neural Network Augmented Dynamic Inversion Structure

The drawback of the PID controller is that it is susceptible to uncertainties and external disturbances, therefore to guarantee the stability and reliability of the whole system, dynamic inversion is introduced into the inner control loop to linearize the nonlinear system and improve the performance of it, as shown in Fig. 3. Moreover, to eliminate the model inversion error, neural network is added to adaptively compensate the inversion error online with a real-time pseudo control \( v_{\text{nn}} \), which yields

\[
v = v_p + v_c - v_{\text{nn}}
\]

(16)

where

\[
\begin{aligned}
v_p &= \begin{bmatrix} v_{\phi} \\ v_{\theta} \\ v_{\psi} \end{bmatrix} \\
v_c &= \begin{bmatrix} K_{\phi} (z_0 - z) + K_{d\phi} (z_0 - z) + K_{i\phi} \int (z_0 - z) dz \\ K_{\theta} (\phi_0 - \phi) + K_{d\theta} (\phi_0 - \phi) + K_{i\theta} \int (\phi_0 - \phi) d\phi \\ K_{\psi} (\psi_0 - \psi) + K_{d\psi} (\psi_0 - \psi) + K_{i\psi} \int (\psi_0 - \psi) d\psi \end{bmatrix} \\
\end{aligned}
\]

\[
\begin{aligned}
v_{\text{nn}} &= \begin{bmatrix} v_{10} \\ v_{20} \\ v_{30} \end{bmatrix} = \begin{bmatrix} \frac{1}{m} \cos \theta \cos \phi \\ 0 \\ 0 \\ 0 \\ \frac{l}{I_z} \end{bmatrix}
\end{aligned}
\]

Thus we can obtain the actual control input as

\[
U = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = E^{-1} (-M + v)
\]

C. Neural Network Design

A general single-layer sigma-pi network is applied to form the neural network architecture, as shown in Fig. 4.
Figure 4. Neural network architecture

The output of the neural network can be derived as

\[ v_{ne} = \sum_{i=1}^{n} w_i \beta_i (\bar{X}, v) = w^T \beta(\bar{X}, v) \]

where \(\bar{X}\) denotes the normalized states, \(w\) represent the network weights. \(\beta\) is a vector of basis function which is composed of combination of the elements of \(\bar{X}_1, \bar{X}_2, \bar{X}_3\) by means of the Kronecker product.

\[ \beta = \text{kron}(\text{kron}(\bar{X}_1, \bar{X}_2), \bar{X}_3) \]

IV. SIMULATION RESULTS

The traditional PID controller and the proposed NN augmented dynamic inversion controller for the quadrotor UAVs are tested in the simulation. Three steps were performed for both controllers: (1) including the external disturbances and noise using PID method; (2) including the external disturbances and noise using dynamic inversion method; (3) including the external disturbances and noise using NN augmented dynamic inversion method

A. Simulink model

The simulink model is established as shown in Fig. 5, which is composed of the position controller, attitude controller and quadrotor system.

The nominal parameters and the initial conditions of the quadrotor for simulation are:

<table>
<thead>
<tr>
<th>Name</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass</td>
<td>m</td>
<td>1</td>
<td>kg</td>
</tr>
<tr>
<td>inertia on x axis</td>
<td>I_x</td>
<td>8.1e-3</td>
<td>kg.m^2</td>
</tr>
<tr>
<td>inertia on y axis</td>
<td>I_y</td>
<td>8.1e-3</td>
<td>kg.m^2</td>
</tr>
<tr>
<td>inertia on z axis</td>
<td>I_z</td>
<td>14.2e-3</td>
<td>kg.m^2</td>
</tr>
<tr>
<td>thrust coefficient</td>
<td>b</td>
<td>5.42e-5</td>
<td>Ns^2</td>
</tr>
<tr>
<td>drag coefficient</td>
<td>d</td>
<td>1.1e-6</td>
<td>Nms^2</td>
</tr>
<tr>
<td>distance</td>
<td>l</td>
<td>0.24</td>
<td>m</td>
</tr>
<tr>
<td>rotor inertia</td>
<td>J_r</td>
<td>104e-6</td>
<td>kg.m^2</td>
</tr>
<tr>
<td>aerodynamic friction</td>
<td>K_c</td>
<td>10e-2</td>
<td>Nms/rad</td>
</tr>
<tr>
<td>factor on attitude</td>
<td>K_c</td>
<td>10e-1</td>
<td>Ns/m</td>
</tr>
<tr>
<td>aerodynamic friction</td>
<td>K_r</td>
<td>10e-2</td>
<td>Nms/rad</td>
</tr>
<tr>
<td>factor on position</td>
<td>K_r</td>
<td>10e-1</td>
<td>Ns/m</td>
</tr>
<tr>
<td>Noise on x axis</td>
<td>n_x</td>
<td>0.05</td>
<td>m</td>
</tr>
<tr>
<td>Noise on y axis</td>
<td>n_y</td>
<td>0.05</td>
<td>m</td>
</tr>
<tr>
<td>Noise on z axis</td>
<td>n_z</td>
<td>0.1</td>
<td>m</td>
</tr>
<tr>
<td>Noise on roll angle</td>
<td>n_roll</td>
<td>0.02</td>
<td>deg</td>
</tr>
<tr>
<td>Noise on pitch angle</td>
<td>n_pitch</td>
<td>0.02</td>
<td>deg</td>
</tr>
<tr>
<td>Noise on yaw angle</td>
<td>n_yaw</td>
<td>0.1</td>
<td>deg</td>
</tr>
</tbody>
</table>

B. Simulation Results

We performed simulations for different methods. In the simulation, external disturbances and noises are both included as set in Tab. 1. Fig. 6 (a) shows the quadrotor path plotted in 3-D, (b) and (c) show the motion and attitude in three axes respectively with their control errors, using PID control method. From the results, it can be seen that PID method shows poor performance in the presence of uncertainties.

Fig. 7 (a) shows the quadrotor path plotted in 3-D, (b) and (c) show the motion and attitude in three axes respectively with their control errors, using dynamic inversion control method. From the results, it can be seen that dynamic inversion method performs much better than PID method in case of disturbances and noises.
Figure 6. Quadrotor motions with external disturbances and noises adopting conventional PID method

(a) Quadrotor path in 3-D

(b) Motion in x,y,z axis and their corresponding errors

(c) Three axis attitude and their corresponding errors

Figure 7. Quadrotor motions with external disturbances and noises adopting Dynamic inversion method

The external disturbances where increasing them will obviously cause the performance deteriorate. On the contrary, the dynamic inversion algorithm is robust and well adaptive to the external disturbances. Moreover, neural network can further eliminate the model inversion error and improve the performance of the control system.

To eliminate the model inversion error, we also perform simulations utilizing the neural network augmented dynamic inversion method, which is shown in Fig. 8. Results demonstrate the effectiveness of neural network.

It can be seen that the traditional PID control is sensitive to
Besides, the comparison of the control algorithms has been listed in Table 2 to summarize the performance in different aspects.

### Table II. Comparison of Quadrotor Control Algorithms

<table>
<thead>
<tr>
<th>Name</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
</tr>
<tr>
<td>PID</td>
<td>1</td>
</tr>
<tr>
<td>LQR</td>
<td>0</td>
</tr>
<tr>
<td>$H^*$</td>
<td>2</td>
</tr>
<tr>
<td>Sliding Mode</td>
<td>2</td>
</tr>
<tr>
<td>Backstepping</td>
<td>0</td>
</tr>
<tr>
<td>Model Predictive</td>
<td>2</td>
</tr>
<tr>
<td>Control</td>
<td></td>
</tr>
<tr>
<td>Dynamic Inverse</td>
<td>2</td>
</tr>
<tr>
<td>Neural Network</td>
<td>2</td>
</tr>
</tbody>
</table>

Legend: R-robust; A-adaptive; O-optimal; I-intelligent; T-tracking ability; F-fast convergence/response; P-precision; S-simplicity; D-disturbance rejection; U-unmodeled parameter handling; 0-low to none;1-average; 2-high

From Table 2, it can be deduced that the use of hybrid control as proposed in this paper will further improve the control performance and compensate for the limitations using a single algorithm.

### V. Conclusion

In this paper, we propose a dynamic inversion controller combined with the traditional PID method for a quadrotor. Since in the classic flight control system, PID seems not suitable for quadrotors when it flies in a complex environment, therefore the neural network based dynamic inversion method is added to compensate for the nonlinearity of the system and improve the performance of UAV control. Simulation results demonstrate that the proposed technique indeed shows good performance in the presence of the external disturbances in comparison with the PID controller. In future work we hope to further optimize the neural network system so as to enhance the system robustness and stability.

### Acknowledgment

This paper is partly supported by South University of Science and Technology of China Research Committee (No. FRG-SUSTC1501A-29 and FRG-SUSTC1501A-44).

### References