Robust Backstepping Control of a Quadrotor UAV Using Extended Kalman Bucy Filter

Raziyeh Babaei¹, Amir Farhad Ehyaei¹

¹Department of Electrical Engineering, Imam Khomeini International University, Qazvin, Iran

*Corresponding Author's E-mail: f.ehyaiei@eng.ikiu.ac.ir

Abstract

In this paper, to overcome the problem of hovering a quadrotor system, under aerodynamic effects, an optimal recursive backstepping controller is designed. One of the main achievements of this study is to propose a nonlinear efficient observer based on extended kalman bucy filter (EKBF) to estimate the unmeasured states of the system. Our control method is robust against perturbations and ensures the stability of the system by using Lyapunov theory. In order to improve the system response, the controller coefficients are also optimized by using the genetic algorithm. Finally, simulation results are included to show that the performance of the quadrotor UAV with the proposed controller and observer is quite satisfactory. The results demonstrate better robustness against the aerodynamic effects in comparison with previous works.

Keywords: Quadrotor, backstepping, Lyapunov theory Control, Genetic Algorithm, Extended Kalman Bucy Filter, Aerodynamic Effects

1. Introduction

Recently, the Vertical Take-Off and Landing (VTOL) Unmanned Aerial Vehicles (UAVs), such as quadrotor helicopters have been widely used in many fields. A quadrotor has more privilege, including: good maneuverability, small dimension and strong subterfuge. These features make quadrotor unique among the other Unmanned Aerial Vehicles. A quadrotor is a dynamic vehicle with four input forces and six output coordinators which is highly coupled and unstable [1-6]. The main difference equations such as rotor aerodynamic, the attitude dynamics and position dynamics are displayed in [7]. [8] presents landing control of a four-rotor system by designing a controller based on the Lyapunov theory. In [9] Bouaballah et al, used PID control and LQ regulation methods to control the system and compression of them on the quadrotor performance. Salih et al, presented a PID controller to the set point flight of a quadrotor helicopter [10]. As a kind of nonlinear control method, backstepping control was implemented by several works. In [7], Ashfaq and Wang proposed a backstepping-based PID controller for a quadrotor under the condition of hovering. Bouchoucha et al, improved an integral backstepping controller for attitude tracking control of a quadrotor helicopter [11]. In [12-13] authors presented a full-state backstepping and sliding mode technique based on Lyapunov stability theory. There are also other nonlinear control methods, such as feedback linearization, that have been attempted in the control of quadrotor systems. In [14] a robust second order sliding mode controller has been implemented for trajectory tracking of a quadrotor with uncertain parameters based on high order sliding mode control method (HOSMC). In [15] the author utilizes the sliding mode control of a class of under actuated systems and he considered the quadrotor as a sample application. [16] presents a new robust backstopping-based controller that impulse integral sliding modes for the under actuated dynamic model of a quadrotor in
In order to achieve smooth stability, including wind gust and sideslip aerodynamics. In a specific problem of a continuous-time stochastic system, in order to improve the results, state estimation with a nonlinear filter such as Extended Kalman-Bucy Filter (EKBF) is used which constitutes the basis for [17]. In [18] the Extended Kalman-Bucy Filter is designed for suitably accurate and robust tracking of targets in an air combat scenario. Recently, for obtaining better results in comparison with classical methods, modern techniques such as Genetic Algorithm (GA), is applied with the aim of parameter optimization of controllers. In this regard, a comparative study is presented in [19]. In [20-22] the effectiveness of this technique has been shown in position control and path generation of robotic systems. In [23] parameter optimization was obtained for a small helicopter based on GA focusing on the stability of the system. To the best of our knowledge, the idea of using continuous kalman filter to synthesis a robust controller such as recursive backstepping associated with the capability of robustness against disturbances has not been applied to control a quadrotor UAV yet. Also a few researches considered utilization of an EKBF and impact of aerodynamics forces on modeling and control of a quadrotor helicopter. Therefore, in this paper we utilize the extended kalman bucy filter along with the nonlinear dynamics of the quadrotor as an efficient solution to estimate unmeasured states of the quadrotor dynamics. Finally, a nonlinear recursive backstepping controller is developed with optimized control parameters adjusted by genetic algorithm for decreasing the impact of aerodynamic effects on modeling and control of the system and the locally asymptotic stability of the system is proved based on lyapunov theory.

This paper is organized as follows: Section 2 presents the dynamic model of a quadrotor aircraft. In section 3 extended kalman bucy filter is described. Section 4 proposes the developed recursive backstepping control technique for two subsystems of the quadrotor separately which ensures the locally asymptotic stability of the system. In the remaining of the section we improve the recursive backstepping controller to ensure tracking of desired trajectories by using Genetic Algorithm. Finally the simulation results have been presented in section 5 and some concluding remarks (section 6) end the paper.

2. Dynamical model of quadrotor

Quadrotor is an extremely nonlinear, multivariable, strongly coupled, and under actuated system (six degrees of freedom with only four actuators) as shown in Fig.1. $F_i$ $(i = 1, 2, 3, 4)$ in the figure is the trust force produced by rotors. Simultaneous increase or decrease in speed of all rotors, will generate the vertical motion of the quadrotor. To analyze the dynamics of this system, consider two reference frames. The earth-fixed frame and the body-fixed reference frame denoted by $E = \{E_x, E_y, E_z\}$ and $B = \{B_x, B_y, B_z\}$ respectively. The position $\zeta = [x, y, z]^T$ and the angle $\Theta = [\phi, \theta, \psi]^T$ are defined in the reference frame E. These three Euler Angels are named roll angle (-$\pi/2 < \phi < \pi/2$), pitch angle (-$\pi/2 < \theta < \pi/2$) and yaw angle (-$\pi < \psi < \pi$).

![Quadrotor helicopter schematic](image-url)
The rotation matrix from B to E is:

\[
R(\Omega) = \begin{bmatrix}
C_\phi C_\theta & C_\phi S_\theta S_\phi - S_\phi C_\theta & C_\phi S_\theta C_\phi + S_\phi S_\theta \\
S_\phi C_\theta & S_\phi S_\theta S_\phi + C_\phi C_\theta & S_\phi S_\theta C_\phi - C_\phi S_\theta \\
- S_\theta & C_\theta S_\phi & C_\theta C_\phi
\end{bmatrix}
\]  

(1)

2.1. Model Due to Aerodynamic Effects

The aerodynamic forces and moments are established by synthesizing momentum with blade element theory according to results of [24-25]. The power exerted on each motor produces a torque, \( M_{Qi} \), which results in a thrust \( F_i \) that can be derived as:

\[
\begin{bmatrix}
F_i \\
M_{Qi}
\end{bmatrix} = \begin{bmatrix}
C_f \rho A r^2 \Omega^2 r \\
C_Q \rho A r^2 \Omega^2 r
\end{bmatrix}
\]

(2)

where aerodynamic coefficients are denoted by thrust coefficient \( C_f \) and torque coefficient \( C_Q \). Also, \( A \) is the area of propeller disk, \( \rho = 1.293 \text{ kg/m}^3 \) is the air density, \( r \) is the radius of the blade and \( \Omega \) is the angular velocity of the quadrotor propeller. The generated moment about the roll axis, \( M_{R_i} \), and drag force on the rotor, \( D_i \), can also be given as:

\[
\begin{bmatrix}
M_{R_i} \\
D_i
\end{bmatrix} = \begin{bmatrix}
C_k \rho A r^2 \Omega^2 \\
C_D \rho A r^2 \Omega^2
\end{bmatrix}
\]

(3)

where aerodynamic coefficients are denoted by drag coefficient \( C_D \) and roll coefficient \( C_R \).

2.2. State Space Model of Quadrotor

The full quadrotor dynamic model without aerodynamics effects, regarding the \( x, y, z \) motions as outcome of pitch, roll or yaw rotations can be written as:

\[
\begin{align*}
\dot{\phi} &= \dot{\theta} \left( \frac{I_z - I_x}{I_z} \right) - \frac{j_z}{I_z} \dot{\Omega} + \frac{j_z}{I_z} U_2 \\
\dot{\theta} &= \dot{\phi} \left( \frac{I_z - I_x}{I_z} \right) - \frac{j_z}{I_z} \dot{\Omega} + \frac{j_z}{I_z} U_4 \\
\dot{\psi} &= \dot{\phi} \left( \frac{I_z - I_x}{I_z} \right) + \frac{1}{I_z} U_4 \\
\dot{z} &= -g + (\cos \phi \cos \theta) \frac{1}{m} U_1 \\
x &= (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \frac{1}{m} U_1 \\
y &= (\cos \phi \sin \theta \sin \psi - \sin \phi \sin \psi) \frac{1}{m} U_1
\end{align*}
\]

(4)

Table 1 in section 5, summarizes different parameters of the prototype quadrotor. The inputs of the system posed on \( U_1, U_2, U_3, U_4 \) and \( \Omega \) as a disturbance are given by:
\[
\begin{align*}
U_1 &= b(\Omega^2 + \Omega_z^2 + \Omega_z^2 + \Omega^2) \\
U_2 &= b(\Omega_2^2 - \Omega_z^2) \\
U_3 &= b(\Omega_2^2 - \Omega_z^2) \\
U_4 &= d(\Omega_2^2 + \Omega_2^2 - \Omega_z^2) \\
\Omega &= -\Omega + \Omega_z + \Omega_3
\end{align*}
\] (5)

The model (4) can be rewritten in a state-space form \( \dot{X} = f(X, U) \) by introducing \( X = [x_1, \ldots, x_{12}] \) as state vector of the system:
\[
\begin{align*}
x_1 &= \phi \\
x_2 &= \dot{x}_1 = \dot{\phi} \\
x_3 &= \theta \\
x_4 &= \ddot{x}_3 = \dot{\theta} \\
x_5 &= \psi \\
x_6 &= \dot{x}_5 = \dot{\psi} \\
x_7 &= x \\
x_8 &= \dot{x}_7 = \dot{x} \\
x_9 &= y \\
x_{10} &= \dot{x}_9 = \dot{y} \\
x_{11} &= z \\
x_{12} &= \dot{x}_{11} = \dot{z}
\end{align*}
\] (6)

In the remaining of this section translational and rotational dynamic equations with aerodynamics effects according to Newton-Euler method are derived. Firstly, the translational dynamic equation is given by:
\[
F_{\text{total}} = m \ddot{z}
\] (7)

where, \( F_{\text{total}} \) is the external force which can be defined as:
\[
F_{\text{total}} = F_{\text{rotor}} - F_{\text{aero}} - F_G
\] (8)

in which \( F_G = mG \) is the gravitational force, \( G = [0, 0, g]^T \) and \( F_{\text{rotor}} \) indicate the aerodynamic forces of the rotor and \( F_{\text{aero}} \) is the air resistance:
\[
F_{\text{rotor}} = R(\Omega)(\sum_{i=1}^{4} F_i - \sum_{i=1}^{4} D_i)
\]
\[
F_{\text{aero}} = \frac{1}{2} \rho AC(U^B)^2
\] (9)

Where \( C \) is aerodynamic force coefficient, \( C = \text{diag}[C_x, C_y, C_z] \). From eq. (9) the translational dynamic equations are defined [33]:
\[
\begin{align*}
\ddot{z} &= g + \frac{1}{m}[((\cos \phi \cos \theta)\sum_{i=1}^{4} F_i - \frac{1}{2} \rho AC_z(U_z^B)^2] \\
\dot{x} &= \frac{1}{m}[(\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi)\sum_{i=1}^{4} F_i - \\
&\quad \sum_{i=1}^{4} D_u - \frac{1}{2} \rho AC_x(U_x^B)^2] \\
\dot{y} &= \frac{1}{m}[(\cos \phi \sin \theta \sin \psi - \sin \phi \sin \psi)\sum_{i=1}^{4} F_i - \\
&\quad \sum_{i=1}^{4} D_u - \frac{1}{2} \rho AC_y(U_y^B)^2]
\end{align*}
\] (10)

Next the rotational dynamic equations are obtained [33]:
\[ M_{total} = \Theta + \hat{\Theta} \times (I \hat{\Theta}) \]
\[ M_{total} = M_c + M_g + M_R \]  

(11)

Where \( M_g = \Theta \delta \omega, (-1)^{i+1} \sum_{i=1}^{4} M_{gi} \) is the gyroscopic torque of the rotors. \( M_R = (-1)^{i+1} \sum_{i=1}^{4} M_{Ri} \), is the rolling moment and \( M_c \) is the control moment generated by the rotors:

\[ M_c = \begin{bmatrix} l(-F_2 + F_4) \\
 l(-F_1 + F_3) \\
 (-1)^{i+1} \sum_{i=1}^{4} M_{\phi} \end{bmatrix} \]  

(12)

Now, from Eqs. (11–12) we have

\[
\begin{align*}
\dot{\phi} &= \dot{\theta} \psi \left( \frac{l_x - l_z}{l_z} \right) - j \frac{l_z}{l_x} \dot{\theta} \omega + \frac{l}{l_x} (-1)^{i+1} \sum_{i=1}^{4} M_{Ri} \\
\dot{\theta} &= \dot{\phi} \psi \left( \frac{l_z - l_x}{l_x} \right) - j \frac{l_x}{l_z} \dot{\phi} \omega + \frac{l}{l_z} (-1)^{i+1} \sum_{i=1}^{4} M_{Ri} \\
\dot{\psi} &= \dot{\phi} \dot{\theta} \left( \frac{l_y}{l_z} \right) + \frac{j}{l_z} \dot{\phi} \dot{\theta} \omega + \frac{l}{l_z} (-1)^{i+1} \sum_{i=1}^{4} M_{\theta} \end{align*}
\]  

(13)

finally, the general state space equation of quadrotor using state variables in eq. (6) can be written as:

\[
f (X, U) = \begin{bmatrix} x_2 \\
 x_4 x_6 a_1 + x_4 a_2 \omega + b_1 U_2 \\
 x_4 \\
 x_2 x_6 a_3 + x_2 a_4 \omega + b_2 U_3 \\
 x_6 \\
 x_4 x_2 a_5 + b_3 U_4 \\
 x_8 \\
 -g + (\cos x_1 \cos x_3) \frac{1}{m} U_1 \\
 x_{10} \\
 u_x \frac{1}{m} U_1 \\
 x_{12} \\
 u_y \frac{1}{m} U_1 \end{bmatrix}
\]  

(14)

Where:
\[ a_i = (I_y - I_z)/I_x \]
\[ a_2 = -j_k/I_x \quad b_1 = 1/I_x \]
\[ a_3 = (I_z - I_y)/I_y \quad b_2 = 1/I_y \]
\[ a_4 = -j_k/I_y \quad b_3 = 1/I_z \]
\[ a_5 = (I_z - I_y)/I_z \]

and:
\[ u_x = (\cos x_1 \sin x_3 \cos x_5 + \sin x_1 \sin x_5) \]
\[ u_y = (\cos x_1 \sin x_3 \sin x_5 - \sin x_1 \sin x_5) \]

The variables \( u_x \) and \( u_y \) can be considered as virtual commands which rotate the thrust vector \( U_4 \) in such a way that the desired \( x - y \) translational motion is achieved.

### 3. Extended Kalman Bucy Filter

The Kalman filter provides a means to infer missing information from indirect (and noisy) measurements. It also optimizes the (minimum variance) state estimation when the dynamic system is linear. The EKBF is an optimal recursive estimation algorithm for computing the states of a nonlinear stochastic system with uncorrelated Gaussian process and measurement noise. The objective trajectory is given by the variables \( x, y, h \), which shall be estimated. According to eq.(6), the states vector is then defined as:

\[ X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}] \]

Let the state and output equations of a Gaussian stochastic system be described by the following stochastic differential equations [17,18]:

\[ \dot{x} = f(x(t), u(t), t) + \omega(t), \quad \omega(t) \sim N(0, Q(t)) \]
\[ y(t) = h(x(t), t) + v(t), \quad v(t) \sim N(0, R(t)) \]

where \( x(t) \) is state vector and \( y(t) \) is the measurements vector. \( \omega(t) \) is the state noise and the vector \( v(t) \) is the measurement noise. The state noise and the measurement noise are mutually independent and independent of the initial state \( x_0 \) whose probability density \( p(x_0) \) is determined. Then the initial conditions are:

\[ \hat{x}(t) = E[x(t_0)], \quad P(t_0) = \text{var}[x(t_0)] \]

in which \( \hat{x}(t) \) is the estimate of \( X \) vector.

When the system is linear with assumption of Gaussian probability densities for the states and noises, the functions \( f(x(t), t) \) and \( h(x(t), t) \) are linear with respect to the states, and the solution for the estimation problem is the Kalman-Bucy Filter. However, if the system is nonlinear, the probability densities may be approximated by Gaussian densities such that their moments can be recursively evaluated by a set of approximately linear equations which define the Extended Kalman Bucy Filter [17,18]. In prediction step, the mean \( \hat{x}(t) \) and covariance matrix \( P(t) \) for a continuous interval satisfy the following ordinary differential equations:
\[
\dot{x} = f(\hat{x}(t), u(t)) + K(t)(y(t) - h(\hat{x}(t))) \\
\dot{P}(t) = F(t)P(t) + P(t)F(t)^T - K(t)H(t)P(t) + Q(t) \\
K(t) = P(t)H(t)R(t)^{-1}
\]  
(20)

where \( F(\hat{x}(t), t) \) is the Jacobean matrix of \( f(x(t), t) \) evaluated at \( \hat{x}(t) \) and \( H(\hat{x}(t), t) \) is the Jacobean matrix of \( h(x(t), t) \) evaluated at \( \hat{x}(t) \):

\[
F(t) = \frac{\partial f(x(t), u(t), t)}{\partial x} \bigg|_{\hat{x}(t), u(t)}
\]

\[
H(t) = \frac{\partial h}{\partial \hat{x}} \bigg|_{\hat{x}(t)}
\]

Therefore, according to eq.(21) the matrices \( H(\hat{x}(t), t) \) and \( F(\hat{x}(t), t) \) can be written in this paper as:

\[
F(\hat{x}(t), t) = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & a_1, \hat{x}_1 & 0 & a_2, \hat{x}_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & a_1, \hat{x}_1 & 0 & 0 & a_2, \hat{x}_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & a_1, \hat{x}_1 & 0 & a_2, \hat{x}_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
H(\hat{x}(t), t) = [I_{12 \times 12}]
\]

and \( Q(t) \) is noise power measurement and \( R(t) \) is the covariance matrix of the state noise measurement that are given by:

\[
R = \text{diag } (\text{var(v)})
\]

\[
Q = \text{diag } (\text{var(\omega)})
\]

(23)

In Extended Kalman Bucy Filter choosing the value of \( R \) and \( Q \) matrices influence on the system states. In this regard, choice of the noise power is very substantial such that it could be very effective in reducing the settling time of the system.

4. Recursive Backstepping Controller Design

This method is used for special model of nonlinear systems that is called explicit feedback system. This system is presented by below equations:

\[
F = [f_1, \ldots, f_{n-1}]^T, G = [g_1, \ldots, g_{n-1}]^T
\]
In [26], using model of eq.(24) a recursive algorithm is provided to determine the Lyapunov function and control input. In the first step of the algorithm, given the Lyapunov function \( V(z_1) \) and control input \( z_3 \) for first term, Lyapunov function and control input are obtained for two states of \( z_1, z_2 \). In the next step, \( z_3 \) is supposed as control input and its value is obtained based on previous states and Lyapunov function is calculated. These series is done so that in the final step Lyapunov function of the total system and input control, \( u \), are determined.

### 4.1. Extended Recursive Backstepping Method

Suppose the eq.(25) is the given nonlinear control system model

\[
\begin{align*}
\dot{X} &= F(X) + G(X)\eta \\
\eta &= f_0(X, \eta) + g_0(X, \eta)u
\end{align*}
\]

(25)

Where \( \eta \in \mathbb{R}, X = [x_1, \ldots, x_{n-1}] \in \mathbb{R}^{n-1} \), now consider Scalar function of \( V(x) \) as below:

\[
V(x_1, \ldots, x_{n-1}) = \frac{1}{2} \sum_{i=1}^{n-1} x_i^2
\]

(26)

Scalar function \( \eta = \phi_i(x_1, \ldots, x_{n-1}) \) is defined per \( i = 1, 2, \ldots, n-1 \) so that \( V(x) \) Function of eq.(28) becomes positive-definite and its derivative becomes negative-definite. In this case, feedback control function and Lyapunov function of the system is obtained using the following relations:

\[
u = \frac{1}{g_0(X, \eta)} \left\{ \sum_{i=1}^{n-1} x_i \frac{\partial \phi_i}{\partial x_i} [f_j(X) + g_j(X)\eta] - \sum_{i=1}^{n-1} x_i \psi_i(X) - \sum_{i=1}^{n-1} k_i [\eta - \phi_i(X)] - f_0(X, \eta) \right\}
\]

(27)

\[\forall k_i > 0 : i = 1, 2, \ldots, n-1\]

\[
V_t(X, \eta) = \frac{1}{2} \sum_{i=1}^{n-1} x_i^2 + \frac{1}{2} \sum_{i=1}^{n-1} \frac{\partial \phi_i}{\partial x_i} [\eta - \phi_i(X)]^2
\]

(28)

In these equations \( F = [f_1, \ldots, f_{n-1}]^T, G = [g_1, \ldots, g_{n-1}]^T \). In addition, the stability of the controlled system is asymptotic and in a broad (total) concept.

### 4.2.1. Recursive Backstepping Control of the Rotations Subsystem

Let \( \hat{X} \) be the estimation of state vector (17):

\[
\hat{X} = [\hat{x}_1, \ldots, \hat{x}_{12}]^T = [\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}, \dot{z}, \ddot{z}, \dddot{z}, \ddot{x}, \dddot{x}, \dddot{y}]
\]

(29)
in backstepping method, the control law is synthesized to force the system to follow the desired trajectory [27-30]. Due to its complete independence from the other subsystem, firstly consider the control input for angular rotations subsystem and then the position control input is derived. Using the backstepping approach, one can synthesize the control law forcing the system to follow the desired trajectory. Refer to [31] and [32] for more details. For the first step we consider the tracking error:

\[ z_1 = x_{id} - \hat{x}_1 \]  

and we use the Lyapunov theorem by considering the Lyapunov function \( z_1 \) positive definite and its time derivative negative semi-definite:

\[ V(z_1) = \frac{1}{2} z_1^2 \]
\[ \dot{V}(z_1) = z_1 (\dot{x}_{id} - \dot{x}_1) \]  

The stabilization of \( z_1 \) can be obtained by introducing a virtual control input \( \dot{x}_2 \):

\[ \dot{x}_2 = \dot{x}_{id} + \alpha_1 z_1 \quad \text{with:} \quad \alpha_1 > 0 \]

The eq. (31) is then:

\[ \dot{V}(z_1) = -\alpha_1 \dot{z}_1^2 \]  

let us proceed to a variable change by making:

\[ z_2 = \dot{x}_2 - \dot{x}_{id} - \alpha_1 z_1 \]  

For the second step we consider the augmented Lyapunov function:

\[ V(z_1, z_2) = \frac{1}{2} (z_1^2 + z_2^2) \]

and its time derivative is then:

\[ \dot{V}(z_1, z_2) = z_2 (a_1 \dot{x}_4 \dot{x}_6 + a_2 \dot{x}_4 \Omega + b_1 U_2) - z_2 (\dot{x}_{id} - \alpha_1 (z_2 + \alpha_1 z_1)) - z_2 - \alpha_1 z_1^2 \]

The control input \( U_2 \) is then extracted \((\dot{x}_{1,2,3d} = 0)\), satisfying:

\[ U_2 = \frac{1}{b_1} (z_1 - a_1 \dot{x}_4 \dot{x}_6 - a_2 \dot{x}_4 \Omega - a_2 (z_2 + \alpha_1 z_1)) - \alpha_2 z_2 \]  

The term \( \alpha_2 z_2 > 0 \) with \( \alpha_2 > 0 \) is added to stabilize \( z_1 \). Similar steps are followed to find \( U_3 \) and \( U_4 \):

\[ U_3 = \frac{1}{b_2} (z_3 - a_3 \dot{x}_5 \dot{x}_6 - a_3 \dot{x}_5 \Omega - a_3 (z_4 + \alpha_2 z_3)) \]
\[ U_4 = \frac{1}{b_3} (z_5 - a_5 \dot{x}_5 \dot{x}_6 - a_5 (z_6 + \alpha_5 z_5)) \]

With:

\[ \begin{align*}
  z_3 &= x_{3d} - \dot{x}_3 \\
  z_4 &= \dot{x}_4 - \dot{x}_{3d} - \alpha_3 z_3 \\
  z_5 &= x_{5d} - \dot{x}_5 \\
  z_6 &= \dot{x}_6 - \dot{x}_{5d} - \alpha_5 z_5
\end{align*} \]
4.2.2. Recursive Backstepping Control of the Translations Subsystem

The altitude control $U_1$ is obtained using the same approach described in 4.2.1.

$$U_1 = \frac{m}{\cos \hat{x}_1 \cos \hat{x}_3} \left( z_7 + g - \alpha_z (z_8 + \alpha_z z_7) - \alpha_k z_k \right)$$  \hspace{1cm} (40)

With:

$$\begin{cases}
\hat{z}_7 = x_7d - \hat{x}_7 \\
\hat{z}_8 = \hat{x}_9 - x_7d - \alpha_7 z_7
\end{cases}$$  \hspace{1cm} (41)

Then From the model (14) one can see that the motion through the axes $x$ and $y$ depends on $U_1$. In fact $U_1$ is the total thrust vector oriented to obtain the desired linear motion. If we consider $u_x$ and $u_y$ the orientations of $U_1$ responsible for the motion through $x$ and $y$ axis respectively, we can then extract from Eq.(16) the roll and pitch angles necessary to compute the controls $u_x$ and $u_y$ satisfying $\dot{V}(z_1,z_2) < 0$. The yaw control is then given as a desired angle.

$$u_x = \frac{m}{U_1} \left( z_9 - \alpha_9 (z_{10} + \alpha_9 z_9) - \alpha_{10} z_{10} \right)$$  \hspace{1cm} (42)

$$u_y = \frac{m}{U_1} \left( z_{11} - \alpha_{11} (z_{12} + \alpha_{11} z_{11}) - \alpha_{12} z_{12} \right)$$  \hspace{1cm} (43)

4.3. Optimize Recursive Backstepping Control Using Genetic Algorithm Method

Genetic algorithm is defined as a method of solving problems to which no satisfactory, explicit, solution exists. GA is started with a population of strings and thereafter generate successive populations using the following three basic operations: reproduction, crossover, and mutation [18-23]. The main feature of GA in this paper is to transform the system output into a cost function in order to find the best values of the control parameters which minimizes the amount of control efforts. This cost function is as follows:

$$J = \sum (x^T R u + u^T Q u)$$  \hspace{1cm} (44)

where $R$ and $Q$, square matrixes of 12th order described in eq.(44), are adjusted to provide the most efficient values. Finally, the parameters of the controller ($\alpha_1,\ldots,\alpha_{12}$) resulted from GA have been placed in table 2.

5. Simulation results

This section accredits the efficiency of proposed model and control scheme by numerical simulation examinations. Table 1, summarizes different system parameters of the prototype quadrotor helicopter.
Table 1: Structural parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>m</td>
<td>Mass</td>
<td>0.723</td>
<td>kg</td>
</tr>
<tr>
<td>l</td>
<td>Arm length</td>
<td>0.314</td>
<td>m</td>
</tr>
<tr>
<td>( J_r )</td>
<td>Rotor inertia</td>
<td>7.32 \times 10^{-5}</td>
<td>kgm²</td>
</tr>
<tr>
<td>( J_x )</td>
<td>X inertia</td>
<td>8.678 \times 10^{-3}</td>
<td>kgm²</td>
</tr>
<tr>
<td>( J_y )</td>
<td>Y inertia</td>
<td>8.678 \times 10^{-3}</td>
<td>kgm²</td>
</tr>
<tr>
<td>( J_z )</td>
<td>Z inertia</td>
<td>3.217 \times 10^{-2}</td>
<td>kgm²</td>
</tr>
<tr>
<td>( b )</td>
<td>Trust factor</td>
<td>5.324 \times 10^{-5}</td>
<td>N. s²</td>
</tr>
<tr>
<td>( d )</td>
<td>Drag factor</td>
<td>8.721 \times 10^{-7}</td>
<td>Nm. s²</td>
</tr>
<tr>
<td>g</td>
<td>gravity</td>
<td>9.81</td>
<td>m/s²</td>
</tr>
</tbody>
</table>

In addition, full GA-tuned parameters of the designed controller \( \alpha_1, \ldots, \alpha_6, k_1, \ldots, k_6 \) resulted from the computations of section 4.3 have been shown in table 2.

Table 2: Control parameters

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha_1 )</td>
<td>9.89</td>
<td>( \alpha_7 )</td>
<td>0.16</td>
</tr>
<tr>
<td>( \alpha_2 )</td>
<td>9.68</td>
<td>( \alpha_8 )</td>
<td>1.55</td>
</tr>
<tr>
<td>( \alpha_3 )</td>
<td>9.25</td>
<td>( \alpha_9 )</td>
<td>3.69</td>
</tr>
<tr>
<td>( \alpha_4 )</td>
<td>9.45</td>
<td>( \alpha_{10} )</td>
<td>7.68</td>
</tr>
<tr>
<td>( \alpha_5 )</td>
<td>3.90</td>
<td>( \alpha_{11} )</td>
<td>2.14</td>
</tr>
<tr>
<td>( \alpha_6 )</td>
<td>6.92</td>
<td>( \alpha_{12} )</td>
<td>9.71</td>
</tr>
</tbody>
</table>

Regarding to explanations in section 3, the necessary parameters for simulating the part belonging to estimation of EKBF will mention as follows.

The Number of generating white noise for both \( \omega \) and \( \nu \) is considered to \( 10^6 \) and the rate of measurement noise power is considered \( 10^{-4} \) and the rate of state noise is also considered 1, both of noises are provided linear which general format of that is as follow:

\[
\omega = wgn(10^6, 1, 10^{-4}, 'linear')
\]

\[
\nu = wgn(10^6, 1, 1, 'linear')
\]

According to values of \( \omega \) and \( \nu \), which provided by white noise, \( R \) matrix and \( Q \) matrix, respectively are considered to

\[
Q = diag \left[ 10^{-4} * I_{7x7}, 10^{-3} * I_{5x5} \right], \quad R = diag \left[ 10^{-4} * I_{7x7}, 10^{-3} * I_{5x5} \right]
\]

also the initial value for \( P \) covariance matrix is chosen as \( P = diag (x^2) \). The initial conditions for uav are \( x_i(0) = [3\pi/4]_{3x1} \) with \( i = 1, \ldots, 3 \) for rotational subsystem and \( x_i(0) = [1]_{3x1} \) with \( i = 1, \ldots, 3 \) for translational one. Also the desired positions are \( x_{id} = [3]_{3x1} \) with \( i = 1, \ldots, 3 \), and desired angles are \( x_{id} = [0]_{3x1} \) with \( i = 1, \ldots, 3 \). The values of the aerodynamic force coefficient \( [C_x, C_y, C_z] \) are set to 0.25, and stop when angles \( \phi \) and \( \theta \) and \( \psi \) are stabilized to zero values. Fig.2 shows the response of the nonlinear controller to stabilize the quadrotor during hovering.
It can be seen that the controller succeeded in controlling the roll, pitch, and yaw angles of the quadrotor in less than 2s. Fig.3 shows that the quadrotor dynamics are stabilized following the given position.

Fig.4 and Fig.5, show the estimated error results, obtained for the attitude and position stabilization of the mini aircraft controller based EKBF observer that ensures a good tracking presents.
Fig. 6 shows the stability rotor speed response of a quadrotor during hovering. And finally Fig. 7 indicates an excellent reference tracking performance even if external disturbances originated by aerodynamic forces and moments are considered.
Figure 6: Control inputs of the quadrotor helicopter

Figure 7: Global trajectory of the quadrotor
Conclusion

In this paper, an efficient recursive back stepping controller was designed that its most important strength is its robustness against aerodynamic effects and impacts on system instability. In quad rotor system modeling, aerodynamic effects were considered on the dynamics of the system and its equations were extracted by Euler -Newton method. What distinguishes proposed control method in this paper from the other recursive back stepping controller is that system state variables were estimated using an extended continuous Kalman Estimator (EKBF) and then were replaced in proposed control method that demonstrated a significant effect on the system stability process. On the other hand, the controller parameters were calculated using a genetic optimization algorithm in order to improve the system performance. The simulation results show designed controller’s ability to track the desired trajectory and reduce error and robustness against external disturbances clearances.

References


