2015 XXV International Conference on
Information, Communication and Automation Technologies (ICAT)
October 29 – October 31, 2015, Sarajevo, Bosnia and Herzegovina

Backstepping Trajectory Tracking Control of a Quadrotor with Disturbance Rejection

Ramy Rashad, Ahmed Aboudonia and Ayman El-Badawy

***Abstract*—This paper presents a trajectory tracking controller based on the backstepping technique. To avoid the increasing complexity of analytically calculating the derivatives of the virtual control signals in standard backstepping control, a command filtered backstepping approach is utilized. The command filtered backstepping controller is modified to include integral action to increase robustness against external disturbances and unmodeled dynamics. The stability proof of the command filtered integral backstepping approach is presented based on Lyapunov's theorem. The controller is implemented on a quadrotor unmanned aerial vehicle in simulation and compared to a standard integral backstepping controller. Simulation results show that the presented controller yields an improvement in the tracking performance of the quadrotor in the presence of constant disturbances and unmodeled actuator dynamics with lower control effort.**

Keywords*—*quadrotor, integral backstepping, command filter

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have attracted many researchers with different specialties during the last decade. One of the most popular vehicles is the quadrotor helicopter. Quadrotor helicopters have the advantage of hovering, vertical landing and taking-off with high maneuverability. Contrary to fixed wing aircraft control, it is difficult to design a decoupled control law that stabilizes the quadrotor dynamics due to the strong coupling between the roll, pitch and yaw nonlinear dynamics in addition to the rotor dynamics [1]. Therefore many researchers have focused on designing multiple-input multiple-output nonlinear controllers that guarantee the quadrotor's stability and improved performance during flight.

In the literature, several nonlinear controllers have been designed and implemented on quadrotors. One method that have been commonly used by researchers is backstepping control [2]. In [3], the authors designed a backstepping controller to stabilize the quadrotor dynamics. The same authors extended their work in [4] to include the motor dynamics in addition to experimental results. They also presented a backstepping controller running parallel with a sliding mode observer to estimate the external disturbances and quadrotor velocities in [5]. Other works utilizing backstepping control include [6] and [7]. In [8], the authors presented a backstepping controller developed on the Lagrangian form of the quadrotor dynamics, not the state space form in contrast to the previous approaches. One of the drawbacks of backstepping control, is that high order time derivatives of the virtual control signals are required for the controller design. In some applications, the analytical derivation of these signals might be tedious. In [9], a sliding mode observer was utilized to estimate the virtual control signals and their derivatives required for the backstepping controller. In [10], the authors proposed a command filtered backstepping approach to handle this issue that includes using a command filter to generate the required signals and their time derivatives. In [11], the author implemented the command filtered backstepping controller for a quadrotor UAV.

Another drawback of backstepping control is the lack of robustness against external disturbances and modeling uncertainties. To handle modeling uncertainties, adaptive backstepping control have been used as in [12] and [13]. To reject disturbances, the authors in [14] studied augmenting the backstepping technique with integral control to increase its robustness properties. The integral backstepping technique have been used for quadrotor control in [15], [16] and [17].

In this paper, an integral backstepping controller with command filtered compensation is proposed to generate the time derivatives of the virtual control signals. The presented controller has not been exploited previously by other researchers as far as the authors know. The result of this approach is compared to an integral backstepping controller through various simulation studies. Both approaches are applied to the quadrotor's nonlinear model with gyroscopic effects and constant external disturbances.

The paper is organized as follows: the quadrotor's nonlinear model and state space equations are presented in section 2. Then in section 3, the integral backstepping controller is derived. In section 4, the command filtered integral backstepping controller is presented along with its stability analysis. Finally, the simulation results are shown in section 5 followed by the conclusion in section 6.

II. DYNAMIC MODEL

A. Quadrotor Nonlinear Model

The equations of motion of quadrotors have been derived and analyzed by several researchers in the literature [18], [19]. The quadrotor's dynamic equations of motion (EoM) are summarized in the following equations

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} (s_\phi s_\psi + c_\phi s_\theta c_\psi)\frac{U_1}{m} + F_{dx} \\ (-s_\phi c_\psi + c_\phi s_\theta s_\psi)\frac{U_1}{m} + F_{dy} \\ (c_\phi c_\theta)\frac{U_1}{m} - g + F_{dz} \end{bmatrix} \tag{1}$$

R. Rashad and A. Aboudonia are with the Mechatronics Department, Faculty of Engineering and Materials Science, German University in Cairo, Cairo, Egypt (email: ramy.abdelmonem@guc.edu.eg, ahmed.aboudonia@guc.edu.eg).

A. El-Badawy is with the Mechanical Engineering Department, Al-Azhar University and German University in Cairo, Cairo, Egypt (email: ayman.elbadawy@guc.edu.eg).

978-1-4673-8146-8/15/$31.00 ©2015 IEEE

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}\frac{1}{I_x}[(I_y - I_z)qr - I_{p_z}q\Omega_T + U_2]\\ \frac{1}{I_y}[(I_z - I_x)pr + I_{p_z}p\Omega_T + U_3]\\ \frac{1}{I_z}[(I_x - I_y)pq + U_4]\end{bmatrix}, \quad (2)$$

where x, y, z represent the quadrotor's Cartesian position, ϕ, θ, ψ are the roll, pitch and yaw angles respectively, p, q, r are the body angular rates, m, I_x, I_y, I_z are the quadrotor's mass and moments of inertia respectively, I_p is the propeller's moment of inertia about its axis of rotation, Ω_T is the algebraic sum of the four rotors, U_1, U_2, U_3, U_4 represent the aerodynamic forces and moments generated by the propellers. The terms F_{dx}, F_{dy} and F_{dz} denote the external disturbances applied on the quadrotor's body in the inertial frame. Finally the terms $c_{(\cdot)}$ and $s_{(\cdot)}$ represent the cosine and sine functions respectively.

The relation between the Euler rates and the body angular rates [20] is expressed as

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & s_\phi t_\theta & c_\phi t_\theta\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi/c_\theta & c_\phi/c_\theta\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix}, \quad (3)$$

The relation between the propellers angular speeds and the generated aerodynamic forces and moments due to the propellers is expressed as

$$\begin{bmatrix}U_1\\ U_2\\ U_3\\ U_4\end{bmatrix} = \begin{bmatrix}b & b & b & b\\ bl_a & 0 & -bl_a & 0\\ 0 & bl_a & 0 & -bl_a\\ d & d & d & d\end{bmatrix}\begin{bmatrix}\omega_1^2\\ \omega_2^2\\ \omega_3^2\\ \omega_4^2\end{bmatrix}, \quad (4)$$

where b is the propeller thrust coefficient, d is the propeller drag coefficient, l_a is the distance between the quadrotor center line and a propeller's center line.

Most quadrotor systems are constructed using brushless DC motors driven by electric speed controllers (ESCs). The main function of each ESC is electronic commutation in addition to controlling the motor's speed. As a result of a system identification experiment done by [21], it is suggested that the closed loop motor control system can be modeled by a first order differential equation

$$\dot{\omega}_i^{act} = K_m(\omega_i^{des} - \omega_i^{act}), \quad (5)$$

where ω_i^{des} is the commanded rotor speed, ω_i^{act} is the actual rotor speed and K_m is the motor gain. In this work, the motor dynamics are incorporated in this manner.

B. State Space Formulation

The dynamic model of the quadrotor consists of a translational subsystem S_T and a rotational subsystem S_R controlled by four inputs $u = [U_1, U_2, U_3, U_4]^\top$. The equations of motion can be written in state space form according to the following states:

$$x_1 = \begin{bmatrix}\phi\\ \theta\\ \psi\end{bmatrix}, x_2 = \begin{bmatrix}p\\ q\\ r\end{bmatrix}, x_3 = \begin{bmatrix}x\\ y\\ z\end{bmatrix}, x_4 = \begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix}. \quad (6)$$

The state space equations are then given by

$$S_R : \begin{cases}\dot{x}_1 = \underbrace{\begin{bmatrix}1 & s_\phi t_\theta & c_\phi t_\theta\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi/c_\theta & c_\phi/c_\theta\end{bmatrix}}_{g_0} x_2\\ \dot{x}_2 = \underbrace{\begin{bmatrix}c_{f1}qr - c_{f2}q\Omega_T\\ c_{f3}pr + c_{f4}p\Omega_T\\ c_{f5}pq\end{bmatrix}}_{f_1} + \underbrace{\begin{bmatrix}\frac{1}{I_x} & 0 & 0\\ 0 & \frac{1}{I_y} & 0\\ 0 & 0 & \frac{1}{I_z}\end{bmatrix}}_{g_1}\begin{bmatrix}U_2\\ U_3\\ U_4\end{bmatrix}\end{cases}$$

$$S_T : \begin{cases}\dot{x}_3 = x_4\\ \dot{x}_4 = \underbrace{\begin{bmatrix}0\\ 0\\ -g\end{bmatrix}}_{f_2} + \underbrace{\begin{bmatrix}\frac{U_1}{m}s_\psi & \frac{U_1}{m}c_\psi & 0\\ -\frac{U_1}{m}c_\psi & \frac{U_1}{m}s_\psi & 0\\ 0 & 0 & \frac{c_\phi c_\theta}{m}\end{bmatrix}}_{g_2}\begin{bmatrix}s_\phi\\ c_\phi s_\theta\\ U_1\end{bmatrix}\end{cases} \quad (7)$$

where $c_{f1}, c_{f2}, c_{f3}, c_{f4}, c_{f5}$ are constant coefficients given by

$$\begin{aligned} c_{f1} &= \frac{I_y - I_z}{I_x} & c_{f2} &= \frac{I_{p_z}}{I_x}\\ c_{f3} &= \frac{I_z - I_x}{I_y} & c_{f4} &= \frac{I_{p_z}}{I_y}\\ c_{f5} &= \frac{I_x - I_y}{I_z} \end{aligned} \quad (8)$$

In the quadrotor dynamics, the roll and pitch angles can be considered as virtual inputs to control the horizontal translational dynamics. In the following sections, a nonlinear controller will be developed based on the integral backstepping approach and then modified with a command filter to remedy the drawbacks of the proposed controller. The controller's objective will be to ensure the output states $[x(t), y(t), z(t), \psi(t)]$ track the desired trajectory given by $[x_d(t), y_d(t), z_d(t), \psi_d(t)]$ in the presence of constant external disturbances.

III. Integral Backstepping Control

The control system design of the rotational subsystem S_R and the translational subsystem S_T can be achieved in four steps as follows [15]:

Step 1: First we consider the virtual subsystem

$$\dot{x}_1 = g_0(x_1)\upsilon_1, \quad (9)$$

which is driven by υ_1, the first virtual control input to be designed. Then we define the following two backstepping variables

$$\begin{aligned} z_1 &= x_1^d - x_1\\ \xi_1 &= \int_0^t z_1(\tau)d\tau \end{aligned}, \quad (10)$$

where z_1 represents the tracking error, ξ_1 represents its integration and $x_1^d = [\phi_d, \theta_d, \psi_d]^\top$. The desired roll ϕ_d and pitch

θ_d angles are calculated later. We then proceed by choosing the following quadratic Lyapunov function

$$V_1 = \frac{1}{2} z_1^\top z_1 + \frac{1}{2} \xi_1^\top \Lambda_1 \xi_1, \tag{11}$$

where $\Lambda_1 = \Lambda_1^\top \in \mathbb{R}^{3\times 3}$, the time derivative of V_1 is given by

$$\begin{aligned} \dot{V}_1 &= z_1^\top \dot{z}_1 + \xi_1^\top \Lambda_1 \dot{\xi}_1 \\ &= z_1^\top (\Lambda_1 \xi_1 + \dot{x}_1^d - g_0 \upsilon_1). \end{aligned} \tag{12}$$

By choosing the virtual control input to be

$$\upsilon_1 = g_0^{-1}(\Lambda_1 \xi_1 + \dot{x}_1^d + A_1 z_1), \tag{13}$$

where $A_1 = A_1^\top \in \mathbb{R}^{3\times 3}$ and then we end up with

$$\dot{V}_1 = -z_1^\top A_1 z_1 \leq 0. \tag{14}$$

Step 2: Moving on to the second step, we consider the subsystem

$$\dot{x}_2 = f_1(x_2) + g_1 \upsilon_2, \tag{15}$$

where υ_2 represents the control inputs

$$\upsilon_2 = [U_2, U_3, U_4]^\top. \tag{16}$$

The next backstepping variable is chosen to be

$$z_2 = \upsilon_1 - x_2. \tag{17}$$

This equation can be rewritten using (13) as

$$g_0 z_2 = \Lambda_1 \xi_1 + A_1 z_1 + \underbrace{\dot{x}_1^d - g_0 x_2}_{\dot{z}_1}, \tag{18}$$

hence we can write

$$\dot{z}_1 = g_0 z_2 - \Lambda_1 \xi_1 - A_1 z_1. \tag{19}$$

To design the control laws, we consider the augmented Lyapunov function

$$V_2 = \frac{1}{2} z_1^\top z_1 + \frac{1}{2} \xi_1^\top \Lambda_1 \xi_1 + \frac{1}{2} z_2^\top z_2, \tag{20}$$

with its time derivative given by

$$\begin{aligned} \dot{V}_2 &= z_1^\top \dot{z}_1 + \xi_1^\top \Lambda_1 \dot{\xi}_1 + z_2^\top \dot{z}_2 \\ &= -z_1^\top A_1 z_1 + z_2^\top (\dot{\upsilon}_1 + g_0^\top z_1 - f_1 - g_1 \upsilon_2). \end{aligned} \tag{21}$$

To stabilize z_2, the control input υ_2 is designed to be

$$\upsilon_2 = g_1^{-1}(-f_1 + g_0^\top z_1 + \dot{\upsilon}_1 + A_2 z_2), \tag{22}$$

where $A_2 = A_2^\top \in \mathbb{R}^{3\times 3}$ which yields

$$\dot{V}_2 = -z_1^\top A_1 z_1 - z_2^\top A_2 z_2 \leq 0, \tag{23}$$

which proves the asymptotic stability of the rotational subsystem S_R. The coming two steps consider the stabilization of the translational subsystem S_T.

Step 3: First we consider the virtual system

$$\dot{x}_3 = \upsilon_3, \tag{24}$$

and define the following backstepping variables

$$\begin{aligned} z_3 &= x_3^d - x_3 \\ \xi_2 &= \int_0^t z_3(\tau) d\tau, \end{aligned} \tag{25}$$

where z_3 represents the position tracking error, ξ_2 represents its integration and $x_3^d = [x_d, y_d, z_d]^\top$. We then proceed by introducing the following Lyapunov function

$$V_3 = \frac{1}{2} z_3^\top z_3 + \frac{1}{2} \xi_2^\top \Lambda_2 \xi_2, \tag{26}$$

where $\Lambda_2 = \Lambda_2^\top \in \mathbb{R}^{3\times 3}$ and the time derivative of V_3 is given by

$$\begin{aligned} \dot{V}_3 &= z_3^\top \dot{z}_3 + \xi_2^\top \Lambda_2 \dot{\xi}_2 \\ &= z_3^\top (\Lambda_2 \xi_2 + \dot{x}_3^d - \upsilon_3). \end{aligned} \tag{27}$$

The stabilization of z_3 is achieved by choosing the virtual control input υ_3 to be

$$\upsilon_3 = \Lambda_2 \xi_2 + \dot{x}_3^d + A_3 z_3, \tag{28}$$

where $A_3 = A_3^\top \in \mathbb{R}^{3\times 3}$ which yields

$$\dot{V}_3 = -z_3^\top A_3 z_3 \leq 0. \tag{29}$$

Step 4: We now consider the subsystem

$$\dot{x}_4 = f_2 + g_2(x_1, U_1) \upsilon_4, \tag{30}$$

where υ_4 represents the control inputs

$$\upsilon_4 = [s_{\phi_d}, c_{\phi_d} s_{\theta_d}, U_1]^\top, \tag{31}$$

we then define the last backstepping variable to be

$$z_4 = \upsilon_3 - x_4, \tag{32}$$

which can be rewritten using (28) as

$$z_4 = \Lambda_2 \xi_2 + A_3 z_3 + \underbrace{\dot{x}_3^d - x_4}_{\dot{z}_3}, \tag{33}$$

from which we can write the time derivative of z_3 as

$$\dot{z}_3 = z_4 - \Lambda_2 \xi_2 - A_3 z_3. \tag{34}$$

Then we proceed by introducing the following augmented Lyapunov function

$$V_4 = \frac{1}{2} z_3^\top z_3 + \frac{1}{2} \xi_2^\top \Lambda_2 \xi_2 + \frac{1}{2} z_4^\top z_4, \tag{35}$$

with its time derivative given by

$$\begin{aligned} \dot{V}_4 &= z_3^\top \dot{z}_3 + \xi_2^\top \Lambda_2 \dot{\xi}_2 + z_4^\top \dot{z}_4 \\ &= -z_3^\top A_3 z_3 + z_4^\top (\dot{\upsilon}_3 + z_3 - f_2 - g_2 \upsilon_4). \end{aligned} \tag{36}$$

The stabilization of z_4 can be obtained by choosing

$$\begin{aligned} \upsilon_4 &= g_2^{-1}(-f_2 + z_3 + \dot{\upsilon}_3 + A_4 z_4) \\ &= [\upsilon_{4_1}, \upsilon_{4_2}, \upsilon_{4_3}]^\top, \end{aligned} \tag{37}$$

which leads to

$$\dot{V}_4 = -z_3^\top A_3 z_3 - z_4^\top A_4 z_4 \le 0. \tag{38}$$

Thus the whole system is asymptotically stable using the control laws (13),(22),(28) and (37). The desired roll and pitch angles, required to calculate z_1 (cf. equation 10) are calculated by

$$\begin{aligned} \phi_d &= \arcsin(\upsilon_{4_1}) \\ \theta_d &= \arcsin(\frac{\upsilon_{4_2}}{c_{\phi_d}}). \end{aligned} \tag{39}$$

A drawback of the backstepping control approach in general is that the derivatives of the virtual control signals are required. In the presented controller above, the required time derivatives are $\dot{x}_1^d$, $\dot{\upsilon}_1$ and $\dot{\upsilon}_3$. Fortunately the signal $\dot{\upsilon}_3$ can be analytically derived, using the system dynamics, which is given by

$$\dot{\upsilon}_3 = \ddot{x}_3^d + A_3(\dot{x}_3^d - x_4) + \Lambda_2 z_3. \tag{40}$$

However, the analytic derivation of $\dot{x}_1^d$ and $\dot{\upsilon}_1$ will be tedious and impractical for implementation. This problem will be addressed in the following section.

IV. Command Filtered Integral Backstepping Control

In this section the integral backstepping controller presented in the previous section will be modified to remedy its drawbacks. The approach followed, proposed by [10], was to use a filtering approach and compensate the tracking errors that occur when using the filtered command signals.

The first signal to be filtered is the virtual control input υ_1 which is filtered using a first order command filter given by

$$\dot{\upsilon}_{1,f} = \Gamma(\upsilon_1 - \upsilon_{1,f}), \tag{41}$$

where $\Gamma = diag(\gamma_1, \gamma_2, \gamma_3)$ represents the filter time constants and the (f) subscript denotes the filtered signal.

The second signal to be filtered is the desired angles vector x_1^d which is filtered using a linear tracking differentiator, proposed by [22], given by

$$x_{1,f}^d = \alpha_1 \qquad \dot{x}_{1,f}^d = \alpha_2, \tag{42}$$

where α_1, α_2 are the states of the following second order system

$$\begin{aligned} \dot{\alpha}_1 &= \alpha_2 \\ \dot{\alpha}_2 &= -F_2\alpha_2 - F_1(\alpha_1 - x_1^d), \end{aligned} \tag{43}$$

while $F_1, F_2 \in \mathbb{R}^{3\times 3}$ are the tracking differentiator constants. The backstepping controller is designed in four steps similar to the previous controller.

Step 1: The first step includes defining the new backstepping variables

$$\begin{aligned} \tilde{z}_1 &= x_{1,f}^d - x_1 \\ \tilde{\xi}_1 &= \int_0^t \tilde{z}_1(\tau) d\tau, \end{aligned} \tag{44}$$

which use the filtered desired signal $x_{1,f}^d$ instead of x_1^d. We then introduce the following Lyapunov function

$$\tilde{V}_1 = \frac{1}{2}\tilde{z}_1^\top \tilde{z}_1 + \frac{1}{2}\tilde{\xi}_1^\top \Lambda_1 \tilde{\xi}_1, \tag{45}$$

and similar to the procedure followed in the previous section, the first backstepping variable $\tilde{z}_1$ can be stabilized by

$$\upsilon_1 = g_0^{-1}(\Lambda_1\tilde{\xi}_1 + \dot{x}_{1,f}^d + A_1\tilde{z}_1), \tag{46}$$

which yields

$$\dot{\tilde{V}}_1 = -\tilde{z}_1^\top A_1 \tilde{z}_1 \le 0. \tag{47}$$

Step 2: The second step involves defining the following backstepping variable

$$\tilde{z}_2 = \upsilon_{1,f} - x_2, \tag{48}$$

which utilizes the filtered signal of the first virtual control signal υ_1. Using (44) and (46), one can compute the time derivative of the first backstepsping variable $\tilde{z}_1$ as

$$\begin{aligned} \dot{\tilde{z}}_1 &= \dot{x}_{1,f}^d - g_0(x_2 - \upsilon_{1,f} + \upsilon_{1,f} - \upsilon_1 + \upsilon_1) \\ &= \dot{x}_{1,f}^d - g_0\upsilon_1 + g_0(\upsilon_{1,f} - x_2) + g_0(\upsilon_1 - \upsilon_{1,f}) \\ &= -\Lambda_1\tilde{\xi}_1 - A_1\tilde{z}_1 + g_0\tilde{z}_2 + g_0(\upsilon_1 - \upsilon_{1,f}). \end{aligned} \tag{49}$$

To retain the stability properties of the backstepping controller, the tracking error $\tilde{z}_1$ needs to be compensated to counteract the effect of the command filter. This is done by defining the new backstepping variable

$$\bar{z}_1 = \tilde{z}_1 - \varepsilon, \tag{50}$$

where ε represents the filtered unachieved portion of υ_1 [10] to be designed later. We then proceed with the controller design by using the following Lyapunov function

$$\tilde{V}_2 = \frac{1}{2}\bar{z}_1^\top \bar{z}_1 + \frac{1}{2}\tilde{z}_2^\top \tilde{z}_2, \tag{51}$$

with its time derivative

$$\begin{aligned} \dot{\tilde{V}}_2 &= \bar{z}_1^\top(\dot{\tilde{z}}_1 - \dot{\varepsilon}) + \tilde{z}_2^\top \dot{\tilde{z}}_2 \\ &= \bar{z}_1^\top(-\Lambda_1\tilde{\xi}_1 - A_1\tilde{z}_1 + g_0\tilde{z}_2 + g_0(\upsilon_1 - \upsilon_{1,f}) - \dot{\varepsilon}) \\ &+ \tilde{z}_2^\top(\dot{\upsilon}_{1,f} - f_1 - g_1\upsilon_2). \end{aligned} \tag{52}$$

The stability of the rotational subsystem S_R can be achieved by choosing

$$\begin{aligned} \upsilon_2 &= g_1^{-1}(-f_1 + g_0^\top \bar{z}_1 + \dot{\upsilon}_{1,f} + A_2\tilde{z}_2) \\ \dot{\varepsilon} &= -\Lambda_1\tilde{\xi}_1 - A_1\varepsilon + g_0(\upsilon_1 - \upsilon_{1,f}), \end{aligned} \tag{53}$$

yielding the Lyapunov function's time derivative to be

$$\dot{\tilde{V}}_2 = -\bar{z}_1^\top A_1 \bar{z}_1 - \tilde{z}_2^\top A_2 \tilde{z}_2 \le 0, \tag{54}$$

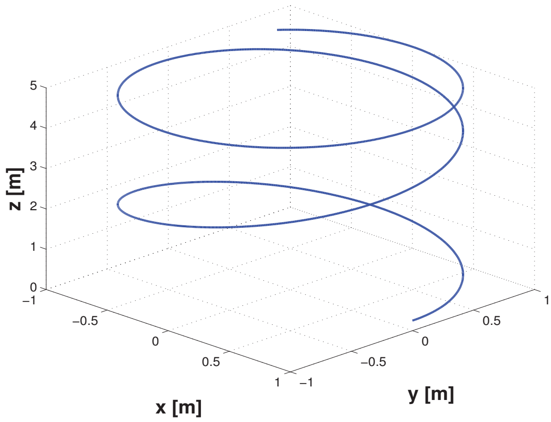

Fig. 1. Desired trajectory

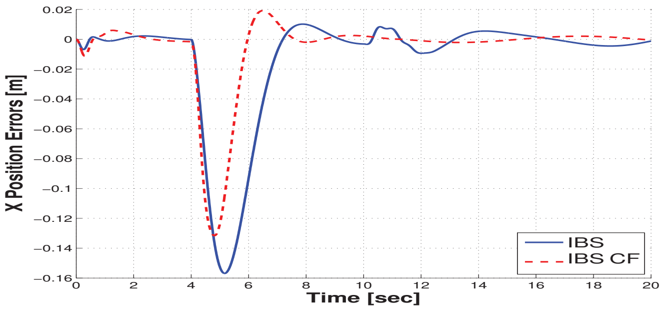

Fig. 2. Position tracking error in x

which ensures the overall stability of the rotational subsystem using Lyapunov's stability theorem. The stabilization of the translational subsystem S_T is achieved using the same control laws developed in the previous section unchanged. Therefore finally, the integral backstepping controller with command filtered compensation can be summarized in (28), (37) (41),(43),(46) and (53).

V. Simulation Results

The integral backstepping (IBS) controller and the integral backstepping with command filtered compensation (IBS CF) controller have been implemented with the quadrotor nonlinear

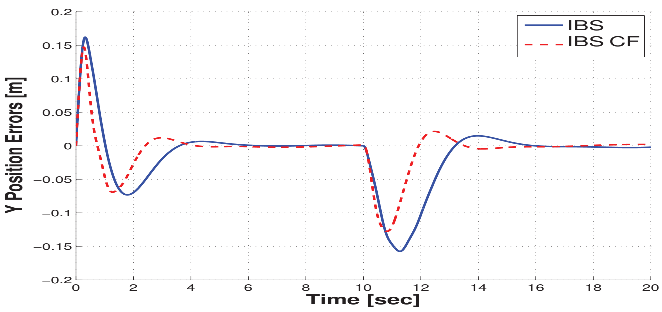

Fig. 3. Position tracking error in y

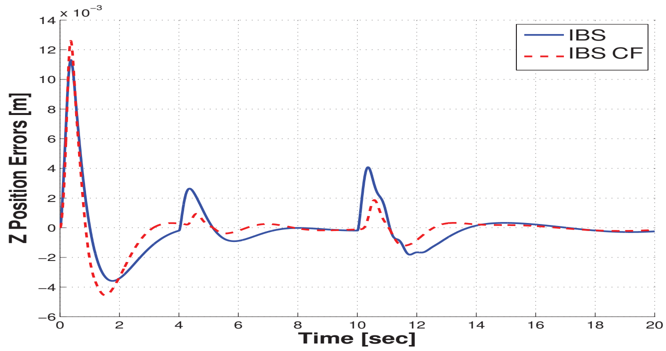

Fig. 4. Position tracking error in z

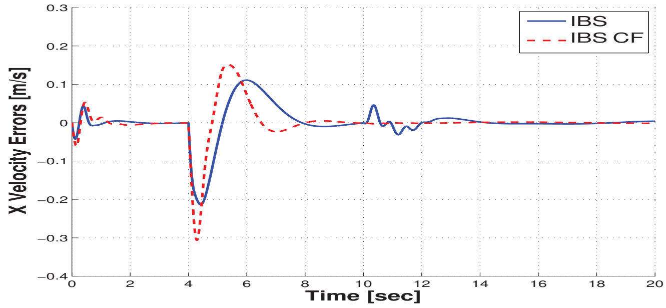

Fig. 5. Velocity tracking error in x

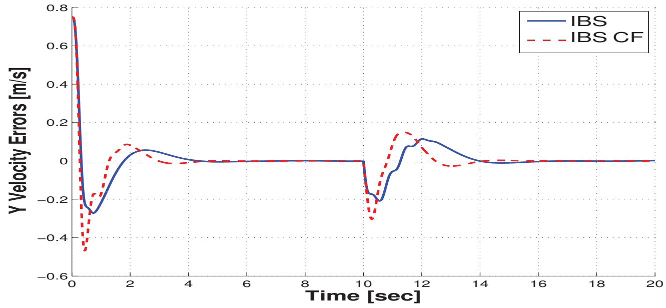

Fig. 6. Velocity tracking error in y

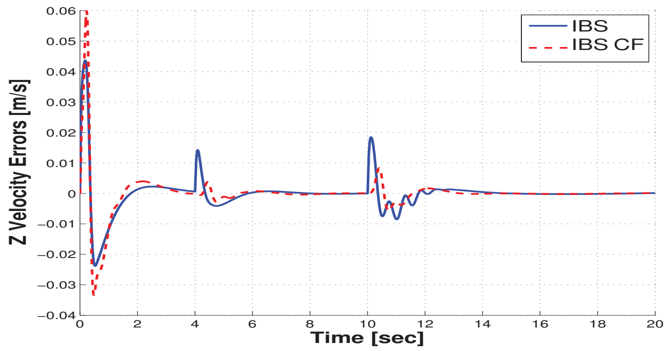

Fig. 7. Velocity tracking error in z

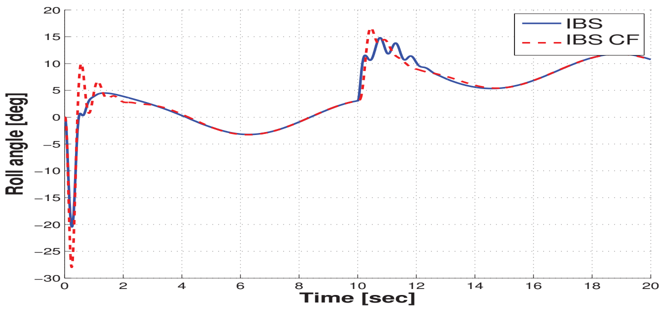

Fig. 8. Roll angle (ϕ)

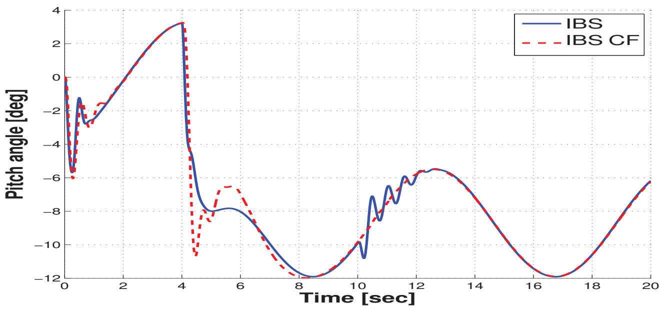

Fig. 9. Pitch angle (θ)

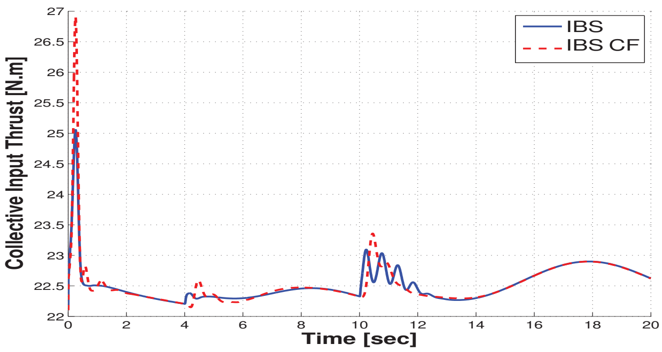

Fig. 10. Collective input thrust (U_1)

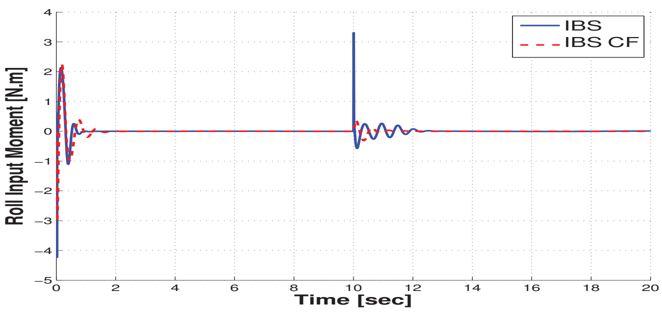

Fig. 11. Rolling input moment (U_2)

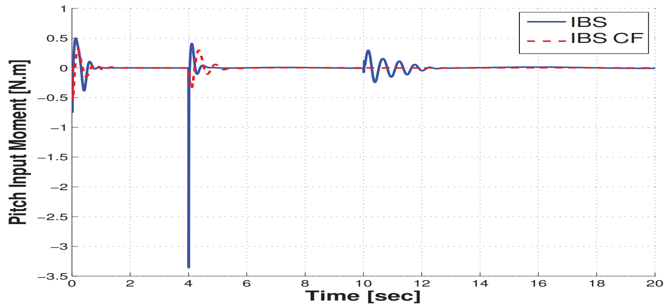

Fig. 12. Pitching input moment (U_3)

model in simulation. The results of these simulation studies, carried out using Matlab/Simulink®, are presented in this section. The quadrotor's parameters used in these simulations were obtained from a real quadrotor platform and are as follows: $m = 2.25Kg$, $g = 9.81m/s^2$, $I_x = 41.3g.m^2$, $I_y = 42.2g.m^2$, $I_z = 75.9g.m^2$ and $I_{p_z} = 0.041g.m^2$. The simulation scenario is as follows: the quadrotor starts in the starting location of the desired trajectory with zero translational velocities but in hovering condition. A constant step disturbance is applied in the x direction at t = $4sec$ and another one is applied in the y direction at t = $10sec$.

The desired trajectory chosen in simulation was a helix shown in figure (1). The equations of the desired states are given by

$$\begin{aligned} x_d(t) &= R\cos(\omega t) \\ y_d(t) &= R\sin(\omega t) \\ z_d(t) &= z_0 + a_{z2}t^2 + a_{z3}t^3 + a_{z4}t^4 + a_{z5}t^5 \\ \psi_d(t) &= 0 \end{aligned}, \tag{55}$$

where R, ω are the radius and angular frequency respectively, z_0 is the initial altitude and a_{zi} are coefficients of a fifth order polynomial. These coefficients are calculated based on the initial altitude, final altitude and reaching time to minimize the jerk of the quadrotor as proposed in [1]. The analytic derivation of the first and second order derivatives of the desired trajectory is simple and straightforward.

The simulation results are shown in figures (2-12). Figures (2) and (3) show the position tracking errors in the x and y directions. It can be observed that the position error in the y direction reaches a higher initial value compared to the x direction. This can be justified by the desired trajectory chosen which has a non-zero initial velocity in the y direction. This can be clearly observed also from the velocity tracking errors shown in figures (5) and (6). The position and velocity errors in the z direction are very similar in the two approaches due to the fact that the altitude control law is the same in both control approaches. The presented controllers guarantee the boundedness of the tracking errors in addition to the boundedness of the roll and pitch angles as shown in figures (8-9).

The results show a slight improvement in the tracking errors between the IBS controller and the IBS-CF controller. However, the main contribution of the work is the control effort required to achieve each performance as shown in the figures of the rolling moment U_2 and pitching moment U_3 shown in figures (11-12). The results show that the IBS CF controller achieves the aforementioned tracking errors with lower control effort. This can be observed clearly at the instants where the step disturbances are applied at $t = 4sec$ and $t = 10sec$. The problem with these high instantaneous control efforts is that actuator saturation may occur and cause instability in real applications. Finally the collective thrust control input U_1 shows a higher initial value in the case of IBS-CF controller as seen in figure (10). However, this slight percentage change in U_1 is much lower than the percentage change that occurs in U_2 and U_3.

VI. Conclusion

In this paper, an integral command filtered backstepping controller is presented and utilized to control the quadrotor to track a desired trajectory. The presented controller provided the quadrotor with robustness against external disturbances and unmodeled dynamics with lower control effort compared to the standard integral backstepping controller. The two approaches are compared in simulation and the results have shown that the proposed controller yields a slight improvement in the tracking errors with a superior improvement in the control effort required.

References

[1] A. Das, K. Subbarao, and F. Lewis, "Dynamic inversion with zero-dynamics stabilisation for quadrotor control," *IET control theory & applications*, vol. 3, no. 3, pp. 303–314, 2009.

[2] H. K. Khalil, *Nonlinear systems*, vol. 3. Prentice hall New Jersey, 1996.

[3] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 3255–3260, IEEE, 2006.

[4] T. Madani and A. Benallegue, "Control of a quadrotor mini-helicopter via full state backstepping technique," in *45th IEEE Conference on Decision and Control*, pp. 1515–1520, 2006.

[5] T. Madani and A. Benallegue, "Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles," in *American Control Conference, 2007. ACC'07*, pp. 5887–5892, IEEE, 2007.

[6] P. Adigbli, C. Gr, J.-B. Mouret, and S. Doncieux, "Nonlinear attitude and position control of a micro quadrotor using sliding mode and backstepping techniques," in *3rd US-European Competition and Workshop on Micro Air Vehicle Systems (MAV07) & European Micro Air Vehicle Conference and Flight Competition (EMAV2007)*, (Toulouse,, France), Citeseer, September 2007.

[7] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *IEEE International Conference on Robotics and Automation*, pp. 2247–2252, 2005.

[8] A. Das, F. Lewis, and K. Subbarao, "Backstepping approach for controlling a quadrotor using lagrange form dynamics," *Journal of Intelligent and Robotic Systems*, vol. 56, no. 1-2, pp. 127–151, 2009.

[9] T. Madani and A. Benallegue, "Backstepping control with exact 2-sliding mode estimation for a quadrotor unmanned aerial vehicle," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 141–146, IEEE, 2007.

[10] J. Farrell, M. Polycarpou, M. Sharma, W. Dong, *et al.*, "Command filtered backstepping," *Automatic Control, IEEE Transactions on*, vol. 54, no. 6, pp. 1391–1395, 2009.

[11] Z. Zuo, "Trajectory tracking control design with command-filtered compensation for a quadrotor," *Control Theory & Applications, IET*, vol. 4, no. 11, pp. 2343–2355, 2010.

[12] I.-H. Choi and H.-C. Bang, "Adaptive command filtered backstepping tracking controller design for quadrotor unmanned aerial vehicle," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 226, no. 5, pp. 483–497, 2012.

[13] M. Huang, B. Xian, C. Diao, K. Yang, and Y. Feng, "Adaptive tracking control of underactuated quadrotor unmanned aerial vehicles via backstepping," in *American Control Conference (ACC), 2010*, pp. 2076–2081, IEEE, 2010.

[14] R. Skjetne, T. Fossen, *et al.*, "On integral control in backstepping: Analysis of different techniques," in *American Control Conference, 2004. Proceedings of the 2004*, vol. 2, pp. 1899–1904, IEEE, 2004.

[15] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *Intelligent robots and systems, 2007. IROS 2007. IEEE/RSJ international conference on*, pp. 153–158, IEEE, 2007.

[16] Z. Fang and W. Gao, "Adaptive integral backstepping control of a micro-quadrotor," in *Intelligent Control and Information Processing (ICICIP), 2011 2nd International Conference on*, vol. 2, pp. 910–915, IEEE, 2011.

[17] S. Seghour, M. Bouchoucha, and H. Osmani, "From integral backstepping to integral sliding mode attitude stabilization of a quadrotor system: Real time implementation on an embedded control system based on a dspic μc," in *Mechatronics (ICM), 2011 IEEE International Conference on*, pp. 154–161, IEEE, 2011.

[18] P. Pounds, R. Mahony, and P. Corke, "Modelling and control of a large quadrotor robot," *Control Engineering Practice*, vol. 18, no. 7, pp. 691 – 699, 2010.

[19] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *Robotics Automation Magazine, IEEE*, vol. 19, no. 3, pp. 20–32, 2012.

[20] L. R. G. Carrillo, A. E. D. López, R. Lozano, and C. Pégard, "Modeling the quad-rotor mini-rotorcraft," in *Quad Rotorcraft Control*, pp. 23–34, Springer, 2013.

[21] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *IEEE Robotics Automation Magazine*, vol. 17, no. 3, pp. 56–65, 2010.

[22] B.-Z. Guo, J.-Q. Han, and F.-B. Xi, "Linear tracking-differentiator and application to online estimation of the frequency of a sinusoidal signal with random noise perturbation," *International Journal of Systems Science*, vol. 33, no. 5, pp. 351–358, 2002.