Nonlinear Control of a Quadrotor Micro-UAV using Feedback-Linearization

Holger Voos
University of Applied Sciences Ravensburg-Weingarten, D-88241 Weingarten,
P.O.-Box 1261, Germany, voos@hs-weingarten.de

Abstract— **Four-rotor micro aerial robots, so called quadrotor UAVs, are one of the most preferred type of unmanned aerial vehicles for near-area surveillance and exploration both in military and commercial in- and outdoor applications. The reason is the very easy construction and steering principle using four rotors in a cross configuration. However, stabilizing control and guidance of these vehicles is a difficult task because of the nonlinear dynamic behavior. In addition, the small payload and the reduced processing power of the onboard electronics are further limitations for any control system implementation. This paper describes the development of a nonlinear vehicle control system based on a decomposition into a nested structure and feedback linearization which can be implemented on an embedded microcontroller. Some first simulation results underline the performance of this new control approach for the current realization.**

I. INTRODUCTION

Unmanned flying robots or vehicles (UAVs) are gaining increasing interest because of a wide area of possible applications. While the UAV market has first been driven by military applications and large expensive UAVs, recent results in miniaturization, mechatronics and microelectronics also offer an enormous potential for small and inexpensive Micro-UAVs for commercial use. These Micro-UAVs would be able to fly either in- or outdoor, leading to completely new applications. However, indoor flight comes up with some very challenging requirements in terms of size, weight and maneuverability of the vehicle that rule out most of the aircraft types, see [1] for an excellent overview. One type of aerial vehicle with a strong potential also for indoor flight is the rotorcraft and the special class of four-rotor aerial vehciles, also called quadrotor. This vehicle, shown in Fig. 1, has been chosen by many researchers as a very promising vehicle, see e.g. [1], [2], [3] and [4].

The quadrotor is a mechatronic system with four propellers in a cross configuration. While the front and the rear motor rotate clockwise, the left and the right motor rotate counterclockwise which nearly cancels gyroscopic effects and aerodynamic torques in trimmed flight. One additional advantage of the quadrotor compared to a conventional helicopter is the simplified rotor mechanics. By varying the speed of the single motors, the lift force can be changed and vertical and/or lateral motion can be created. Pitch movement is generated by a difference between the speed of the front and the rear motor while roll movement results from differences between the speed of the left and right rotor, respectively. Yaw rotation results from the difference in the counter-torque between each pair (front-rear and left-right) of rotors. The overall thrust is the sum of the thrusts generated by the four single rotors.

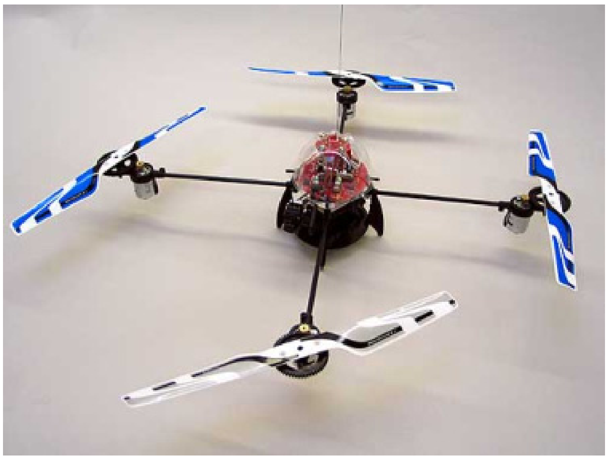
Fig. 1. A commercially available quadrotor.

Besides the choice of a suitable aircraft type, combined in- and outdoor flight also requires a more advanced onboard automation system. Inside a building, not much space for maneuvering is available but many obstacles exist and there is a high possibility that any wireless data link will fail. Therefore, a very accurate stabilization of the platform, a highly precise navigation with collision avoidance functionality and the onboard implementation of more cognitive functions in order to guarantee a higher degree of autonomy is necessary, see [5]. In addition to this functional complexity, the algorithms also have to be implemented in the embedded hardware and have to fulfil realtime requirements while limited memory and onboard processing capacity have to be considered.

In this paper, we address the first problem of accurate stabilization of the quadrotor UAV since the fulfillment of that task is a precondition for further implementation of other functionalities in the vehicle. In spite of the four actuators, the quadrotor is a dynamically unstable system that has to be stabilized by a suitable control system. Unfortunately, the dynamic behavior is nonlinear leading to more complex control algorithms. There are some contributions in the literature that are concerned with control system design for small quadrotor vehicles, see e.g. [1], [2], [3], [4] and [5] to mention only a few. Many of the proposed control systems are based on a linearized model and conventional PID- or state space control while other approaches apply sliding-mode, H$_\infty$ or SDRE control [4], [5]. This paper proposes a control system approach that is first based upon a decomposition of the overall control functionality into a nested structure of velocity and attitude control. The attitude control problem is then solved via feedback linearization which directly takes the nonlinear dynamics of the vehicle into account. For velocity control, a proportional controller with a nonlinear transformation is applied. First simulation results are promising for the current realization and implementation of the algorithms.

II. DYNAMIC MODEL OF THE QUADROTOR

The general dynamic model of a quadrotor has been presented in a number of papers and will not be discussed here in all details again. For further considerations of modelling, we refer to [1], [3] or [4]. We consider an inertial frame and a body fixed frame whose origin is in the center of mass of the quadrotor as shown in Fig. 2.

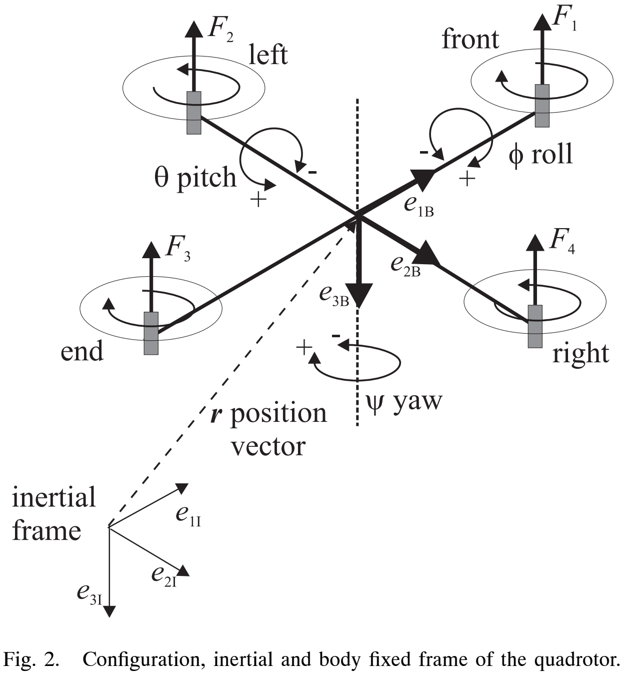

Fig. 2. Configuration, inertial and body fixed frame of the quadrotor.

The orientation of the quadrotor is given by the three Euler angles, namely yaw angle ψ, pitch angle θ and roll angle ϕ that together form the vector $\boldsymbol{\Omega}^T = (\phi, \theta, \psi)$. The position of the vehicle in the inertial frame is given by the vector $\boldsymbol{r}^T = (x, y, z)$. The transformation of vectors from the body fixed frame to the inertial frame is given by the rotation matrix $\boldsymbol{R}$ where c_θ for example denotes $\cos\theta$ and s_θ denotes $\sin\theta$:

$$\boldsymbol{R} = \begin{pmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{pmatrix} \tag{1}$$

Since the thrust force generated by rotor $i, i = 1, 2, 3, 4$ is $F_i = b \cdot \omega_i^2$ where b is the thrust factor and ω_i is the speed of rotor i, we obtain a first set of differential equations that describe the acceleration of the quadrotor:

$$\ddot{\boldsymbol{r}} = g \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} - \boldsymbol{R} \cdot b/m \sum_{i=1}^{4} \omega_i^2 \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{2}$$

With the inertia matrix $\boldsymbol{I}$ (which is a diagonal matrix with the inertias I_x, I_y and I_z on the main diagonal), the rotor inertia J_R, the vector $\boldsymbol{M}$ that describes the torque applied to the vehicle's body and the vector $\boldsymbol{M}_G$ of the gyroscopic torques we obtain a second set of differential equations:

$$\boldsymbol{I}\ddot{\boldsymbol{\Omega}} = -\left(\dot{\boldsymbol{\Omega}} \times \boldsymbol{I}\dot{\boldsymbol{\Omega}}\right) - \boldsymbol{M}_G + \boldsymbol{M} \tag{3}$$

The vector $\boldsymbol{M}$ is defined as (see Fig. 2)

$$\boldsymbol{M} = \begin{pmatrix} Lb(\omega_2^2 - \omega_4^2) \\ Lb(\omega_1^2 - \omega_3^2) \\ d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2) \end{pmatrix} \tag{4}$$

with the drag factor d and the length L of the lever. The gyroscopic torques caused by rotations of the vehicle with rotating rotors are

$$\boldsymbol{M}_G = I_R(\dot{\boldsymbol{\Omega}} \times \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}) \cdot (\omega_1 - \omega_2 + \omega_3 - \omega_4) \tag{5}$$

The four rotational velocities ω_i of the rotors are the input variables of the real vehicle, but with regard to the obtained model a transformation of the inputs is suitable. Therefore, we define the new artificial input variables as follows:

$$\begin{aligned} u_1 &= b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ u_2 &= b(\omega_2^2 - \omega_4^2) \\ u_3 &= b(\omega_1^2 - \omega_3^2) \\ u_4 &= d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2) \end{aligned} \tag{6}$$

However, also the gyroscopic torques depend on the rotational velocities of the rotors and hence on the vector $\boldsymbol{u}^T = (u_1, u_2, u_3, u_4)$ of the transformed input variables. With

$$\omega_1 - \omega_2 + \omega_3 - \omega_4 = g(\boldsymbol{u}) \tag{7}$$

evaluation of (2) and (3) yields the overall dynamic model in the following form

$$\begin{aligned} \ddot{x} &= -(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \cdot \frac{u_1}{m} \\ \ddot{y} &= -(\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \cdot \frac{u_1}{m} \\ \ddot{z} &= g - (\cos\phi \cos\theta) \cdot \frac{u_1}{m} \\ \ddot{\phi} &= \dot{\theta}\dot{\psi}(\frac{I_y - I_z}{I_x}) - \frac{I_R}{I_x}\dot{\theta} g(\boldsymbol{u}) + \frac{L}{I_x} u_2 \\ \ddot{\theta} &= \dot{\phi}\dot{\psi}(\frac{I_z - I_x}{I_y}) + \frac{I_R}{I_y}\dot{\phi} g(\boldsymbol{u}) + \frac{L}{I_y} u_3 \\ \ddot{\psi} &= \dot{\phi}\dot{\theta}(\frac{I_x - I_y}{I_z}) + \frac{1}{I_z} u_4 \end{aligned} \tag{8}$$

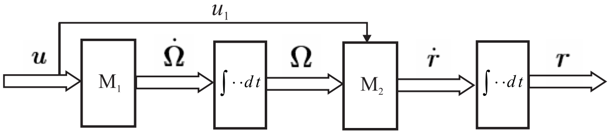

Fig. 3. Overall model of the quadrotor dynamics.

This model can be rewritten in state variable form $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ where $\boldsymbol{x} \in \mathbb{R}^9$ is the vector of state variables

$$\boldsymbol{x}^T = (\dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, \dot{\phi}, \dot{\theta}, \dot{\psi}) \tag{9}$$

Using (7) and (8) we obtain

$$\dot{\boldsymbol{x}} = \begin{pmatrix} -(\cos x_4 \sin x_5 \cos x_6 + \sin x_4 \sin x_6) \cdot u_1/m \\ -(\cos x_4 \sin x_5 \sin x_6 - \sin x_4 \cos x_6) \cdot u_1/m \\ g - (\cos x_4 \cos x_5) \cdot u_1/m \\ x_7 \\ x_8 \\ x_9 \\ x_8 x_9 I_1 - \frac{I_R}{I_x} x_8 g(\boldsymbol{u}) + \frac{L}{I_x} u_2 \\ x_7 x_9 I_2 + \frac{I_R}{I_y} x_7 g(\boldsymbol{u}) + \frac{L}{I_y} u_3 \\ x_7 x_8 I_3 + \frac{1}{I_z} u_4 \end{pmatrix} \tag{10}$$

with $I_1 = (I_y - I_z)/I_x$, $I_2 = (I_z - I_x)/I_y$ and $I_3 = (I_x - I_y)/I_z$. Here it has also been taken into account that the reference variable of the quadrotor is a desired velocity vector and not a position vector.

It becomes obvious that the state space model can be decomposed into one subset of differential equations that describes the dynamics of the attitude (i.e. the angles) and one subset that describes the translation of the UAV. From (10) we obtain the first subset of differential equations, called submodel M_1, that describes the quadrotor's angular rates as

$$\begin{pmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \end{pmatrix} = \begin{pmatrix} x_8 x_9 I_1 - \frac{I_R}{I_x} x_8 g(\boldsymbol{u}) + \frac{L}{I_x} u_2 \\ x_7 x_9 I_2 + \frac{I_R}{I_y} x_7 g(\boldsymbol{u}) + \frac{L}{I_y} u_3 \\ x_7 x_8 I_3 + \frac{1}{I_z} u_4 \end{pmatrix} \tag{11}$$

The Euler angles of the quadrotor, i.e. the state variables x_4, x_5 and x_6 can then be obtained by pure integration. These Euler angles as well as the variable u_1 are the input variables of a submodel M_2 that describes the velocities of the quadrotor:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} -\cos x_4 \sin x_5 \cos x_6 - \sin x_4 \sin x_6 \\ -\cos x_4 \sin x_5 \sin x_6 + \sin x_4 \cos x_6 \\ g - (\cos x_4 \cos x_5) \end{pmatrix} \cdot \frac{u_1}{m} \tag{12}$$

Further integration delivers the position of the vehicle. The overall structure of the overall model including the submodels is shown in Fig. 3. The derived dynamic model has been implemented in MATLAB/Simulink, in addition an experimental platform has been designed and the parameters of this vehicle have been identified via experiments. The presented model then serves as the basis for the development of the control system.

III. VEHICLE CONTROLLER DESIGN

From a control engineering point of view, a UAV system contains two main control loops [5]. The first main and underlying control loop is the vehicle control loop. This control loop is responsible for the generation and stabilization of a currently required movement of the UAV. The second main loop is the mission control loop that comprises the stabilized vehicle as a platform for mission related sensors and actuators and the mission control system. The mission control loop computes the desired flight path, e.g. given by waypoints, and commands current required movements to the vehicle control loop. The remaining question comprises the type of commands that will be given to the vehicle control loop. Direct position control as proposed in some papers (see e.g. [2], [3]) is most often not necessary for vehicle guidance and position measurement or estimation is most often not accurate enough for direct feedback control of the position.

For that reason we assume in this approach that the mission control system commands a desired velocity vector to the vehicle control system. This required velocity vector then has to be established and stabilized. In order to obtain the necessary measurements for this velocity control, the vehicle control loop must be equipped with a suitable inertial measurement system (IMU). This IMU delivers the accelerations and angular rates that can be used to further estimate velocities and Euler angles with the help of a Kalman filter. The default command from the mission system is the zero velocity vector, i.e. the quadrotor UAV should hover at the current position. In this paper the main challenge and focus is on the vehicle control loop, i.e. the control of a required velocity vector of the UAV.

The decomposed model structure as shown in Fig. 3 already suggests a nested structure for vehicle control. In order to achieve and maintain a desired velocity vector, first the necessary attitude of the UAV has to be stabilized. Therefore, we propose a decomposition of the control system in an outer-loop velocity control and an inner-loop attitude control system. In this structure, the inner attitude control loop has to be much faster than the outer loop and stabilizes the desired angles that are commanded by the outer loop. This nested structure is shown in Fig. 4.

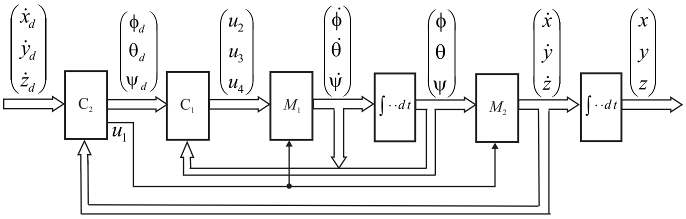

Fig. 4. Nested structure of the UAV vehicle control.

First we consider the inner control loop with controller C_1, the attitude control loop, that has to stabilize the desired roll, pitch and yaw angle, i.e. the desired vector $\boldsymbol{\Omega}_d^T = (\phi_d, \theta_d, \psi_d) = (x_{4,d}, x_{5,d}, x_{6,d})$. The corresponding dynamic model comprises the last six equations of the state variable model (10) which is a series of the nonlinear submodel M_1 and an integrator. Then we derive the outer-loop controller C_2 to stabilize a desired velocity vector.

A. Attitude Control System

For the design of the attitude control C_1 we first neglect the gyroscopic terms in the relevant submodel M_1 which are comparatively small if the rotor inertias are small (as it is the case here). However, we show later that the derived controller nevertheless also stabilizes the model with gyroscopic terms. We obtain the simplified submodel M_1 as

$$\begin{pmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \end{pmatrix} = \begin{pmatrix} x_8 x_9 I_1 + \frac{L}{I_x} u_2 \\ x_7 x_9 I_2 + \frac{L}{I_y} u_3 \\ x_7 x_8 I_3 + \frac{1}{I_z} u_4 \end{pmatrix} \tag{13}$$

Now we apply a feedback linearization in order to obtain a linear system:

$$\begin{aligned} u_2 &= f_2(x_7, x_8, x_9) + u_2^* \\ u_3 &= f_3(x_7, x_8, x_9) + u_3^* \\ u_4 &= f_4(x_7, x_8, x_9) + u_4^* \end{aligned} \tag{14}$$

with the new input variables u_2^*, u_3^*, u_4^*. In order to obtain a linear system, the following conditions must be fulfilled:

$$\begin{aligned} x_8 x_9 I_1 + \frac{L}{I_x} f_2(x_7, x_8, x_9) &= K_2 \cdot x_7 \\ x_7 x_9 I_2 + \frac{L}{I_y} f_3(x_7, x_8, x_9) &= K_3 \cdot x_8 \\ x_7 x_8 I_3 + \frac{1}{I_z} f_4(x_7, x_8, x_9) &= K_4 \cdot x_9 \end{aligned} \tag{15}$$

with the so far undetermined constant parameters K_2, K_3, K_4. Evaluation of (15) yields the nonlinear feedback for linearization

$$\begin{aligned} f_2(x_7, x_8, x_9) &= \frac{I_x}{L} (K_2 x_7 - x_8 x_9 I_1) \\ f_3(x_7, x_8, x_9) &= \frac{I_y}{L} (K_3 x_8 - x_7 x_9 I_2) \\ f_4(x_7, x_8, x_9) &= I_z (K_4 x_9 - x_7 x_8 I_3) \end{aligned} \tag{16}$$

Using this feedback (13) turns into the linear and decoupled system

$$\begin{pmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \end{pmatrix} = \begin{pmatrix} K_2 x_7 + \frac{L}{I_x} u_2^* \\ K_3 x_8 + \frac{L}{I_y} u_3^* \\ K_4 x_9 + \frac{1}{I_z} u_4^* \end{pmatrix} \tag{17}$$

It can be shown that the resulting linearized closed-loop system is stable even if we consider the gyroscopic terms in (11). For that purpose we consider $u_2^* = u_3^* = u_4^* = 0$ and the operating point $x_7 = x_8 = x_9 = 0$. We define the Lyapunov function $V(x_7, x_8, x_9)$ which is is C^1 and positive defined around the operating point:

$$V(x_7, x_8, x_9) = 0.5 \cdot (x_7^2 + x_8^2 + x_9^2) \tag{18}$$

Now we calculate the first derivative of V using the model (11) also including gyroscopic terms and the derived feedback (14), (16). In addition we assume a perfect cross configuration of the quadrotor with $I_x = I_y$ which results in $I_1 = -I_2$ and $I_3 = 0$. The derivative of the Lyapunov function then finally can be calculated as

$$\begin{aligned} \dot{V} &= x_7 \dot{x}_7 + x_8 \dot{x}_8 + x_9 \dot{x}_9 \\ &= K_2 \cdot x_7^2 + K_3 \cdot x_8^2 + K_4 \cdot x_9^2 \end{aligned} \tag{19}$$

which is also independent from the gyroscopic terms. This derivative is negative defined if $K_2, K_3, K_4 < 0$ and this guarantees that the operating point of the feedback linearized system is asymptotically stable.

Taking into account that $\dot{x}_4 = x_7$, $\dot{x}_5 = x_8$, $\dot{x}_6 = x_9$ (see (10) it becomes obvious that the dynamics of the angles using the linearized dynamics and neglecting the gyroscopic terms again are described by linear decoupled differential equations of second-order, respectively. See e.g. x_4:

$$\ddot{x}_4 = K_2 \dot{x}_4 + L/I_x u_2^* \tag{20}$$

If x_{4d} is the desired angle, application of a linear controller $u_2^* = w_2 \cdot (x_{4d} - x_4)$ with constant parameter w_2 leads to a closed-loop system of second order with the transfer function

$$F(s) = \frac{X_4(s)}{X_{4d}(s)} = \frac{w_2}{I_x/L \cdot s^2 - K_2 I_x/L \cdot s + w_2} \tag{21}$$

The same considerations hold for the other angles with linear controllers $u_3^* = w_3 \cdot (x_{5d} - x_5)$ and $u_4^* = w_4 \cdot (x_{6d} - x_6)$, respectively. The dynamics of these closed-loop systems can now be easily defined by adjustment of the pairs of parameters $(K_2, w_2), (K_3, w_3), (K_4, w_4)$, respectively, with the only limitation that the parameters K_2, K_3, K_4 must be negative. For instance, a choice of $K_2 = -80$ leads to a settling time of approximately $T_s \approx 0.1$ sec, a choice $w_2 = (K_2/2)^2 \cdot I_x/L$ leads to zero overshoot.

B. Velocity Control System C_2

If the inner-loop attitude control is sufficiently fast as previously designed, we can assume that desired values of the roll, pitch and yaw angle, i.e. x_{4d}, x_{5d}, x_{6d}, are achieved very fast compared with the outer velocity control loop. Therefore the closed inner attitude control loop can be approximately considered as a static block that just transfers the desired values of roll, pitch and yaw angle to the next model M_2. According to (12), we can then describe model M_2 by the following set of nonlinear differential equations:

$$\begin{aligned} \dot{x}_1 &= -(\cos x_{4d} \sin x_{5d} \cos x_{6d} + \sin x_{4d} \sin x_{6d}) \cdot u_1/m \\ \dot{x}_2 &= -(\cos x_{4d} \sin x_{5d} \sin x_{6d} - \sin x_{4d} \cos x_{6d}) \cdot u_1/m \\ \dot{x}_3 &= g - \cos x_{4d} \cos x_{5d} \cdot u_1/m \end{aligned} \tag{22}$$

where all x_{4d}, x_{5d}, x_{6d} and u_1 are input variables. Equation (22) can be interpreted in a way that all differential equations are of the form

$$\begin{aligned} \dot{x}_1 &= \tilde{u}_1 = f_1(x_{4d}, x_{5d}, x_{6d}, u_1) \\ \dot{x}_2 &= \tilde{u}_2 = f_2(x_{4d}, x_{5d}, x_{6d}, u_1) \\ \dot{x}_3 &= \tilde{u}_3 = f_3(x_{4d}, x_{5d}, x_{6d}, u_1) \end{aligned} \tag{23}$$

with the new input variables $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$ that depend on the other four input variables in a nonlinear form. However, regarding these new input variables, the control task is very simple because it comprises the control of three independent systems of first order which might be solved by pure proportional controllers, respectively:

$$\begin{aligned} \tilde{u}_1 &= k_1 \cdot (x_{1d} - x_1) \\ \tilde{u}_2 &= k_2 \cdot (x_{2d} - x_2) \\ \tilde{u}_3 &= k_3 \cdot (x_{3d} - x_3) \end{aligned} \tag{24}$$

Herein the controller parameters k_1, k_2 and k_3 could be chosen in a way that the outer loop is sufficiently fast but not too fast with respect to the inner loop attitude control. In a next step, these transformed input variables $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$ must be used to obtain the real input variables x_{4d}, x_{5d}, x_{6d} and u_1 by evaluating (22).

First of all it becomes obvious that any desired velocity vector can be achieved without any yaw rotation and therefore we can set $x_{6d} = \psi_d = 0$ which simplifies (22):

$$\begin{aligned} \tilde{u}_1 &= -\cos x_{4d} \sin x_{5d} \cdot u_1/m \\ \tilde{u}_2 &= \sin x_{4d} \cdot u_1/m \\ \tilde{u}_3 &= g - \cos x_{4d} \cos x_{5d} \cdot u_1/m \end{aligned} \tag{25}$$

These nonlinear static equations however can be solved analytically by first applying the following substitution:

$$\begin{aligned} \alpha = \sin x_{4d} \quad &\Rightarrow \quad \cos x_{4d} = \pm\sqrt{1-\alpha^2} \\ \beta = \sin x_{5d} \quad &\Rightarrow \quad \cos x_{5d} = \pm\sqrt{1-\beta^2} \end{aligned} \tag{26}$$

Insertion in (25) yields

$$\begin{aligned} \tilde{u}_1 &= \mp\sqrt{1-\alpha^2} \cdot \beta \frac{u_1}{m} \\ \tilde{u}_2 &= \alpha \frac{u_1}{m} \\ \tilde{u}_3 &= g - \left(\mp\sqrt{1-\alpha^2} \cdot \pm\sqrt{1-\beta^2} \cdot \frac{u_1}{m} \right) \end{aligned} \tag{27}$$

If $\tilde{u}_1 \neq 0$ we obtain the following solution:

$$\begin{aligned} \beta &= \pm \left[\left(\frac{g - \tilde{u}_3}{\tilde{u}_1} \right)^2 + 1 \right]^{-\frac{1}{2}} \\ u_1 &= \pm m \cdot \sqrt{\frac{\tilde{u}_1^2}{\beta^2} + \tilde{u}_2^2} \\ \alpha &= \tilde{u}_2 \cdot \frac{m}{u_1} \end{aligned} \tag{28}$$

Since u_1 is always positive (see (6)), we obtain the unique solution

$$u_1 = +m \cdot \sqrt{\frac{\tilde{u}_1^2}{\beta^2} + \tilde{u}_2^2} \tag{29}$$

Then α is also unique and hence $x_{4d} = \arcsin \alpha$ is uniquely obtained in the interval $[\pm\,\pi/2]$. In the same way, we achieve $x_{5d} = \arcsin \beta$, but β and thus x_{5d} could be positive or negative. However, let us consider

$$\tilde{u}_1 = -\cos x_{4d} \sin x_{5d} \cdot u_1/m \tag{30}$$

Since the first cosinus term is always positive in $[\pm\,\pi/2]$ and the last term is also always positive, we see that x_{5d} must be negative if $\tilde{u}_1$ is positive and vice versa. All further special cases (e.g. $\tilde{u}_1 = 0$) can all be solved analytically too after careful consideration.

C. Overall Control System

The overall control system consist of the derived inner attitude and the outer velocity control loop, forming the structure as shown in Fig. 4. The two control systems determine the four input variables u_1, u_2, u_3 and u_4 and (6) is used to calculate the required angular rates of the rotors, namely $\omega_1, \omega_2, \omega_3$ and ω_4. The main advantage of the overall control system is the fact that the feedback linearization and the controllers are comparatively easy to be implemented, while taking the full nonlinear behavior of the vehicle into account. That leads to a fast computation even on standard embedded micro-controller systems. However, the overall control algorithm requires the measurement of all state variables, i.e. all velocities, Euler angles and rates of the Euler angles. These measurements must be provided by the inertial measurement unit with sufficient accuracy.

IV. SIMULATION RESULTS

In order to implement the derived control system, a simulation model as well as an experimental setup has been developed. The experimental setup is based upon the commercially available quadrotor manufactured by Draganfly Innovations, Inc. (see also http://www.rctoys.com). The original electronics is removed and replaced by our own hardware. In addition, a commercially availabe Micro IMU from Xsens (www.xsense.com) is implemented. The parameters of the experimental setup are determined following the experiments given in [2] and led to the identified parameters given in Table I. The quadrotor model (8) using the parameters of Table I is then implemented in MATLAB/Simulink for a simulation. The parameters of the velocity controllers are chosen as $k_1 = k_2 = k_3 = 5$ while the design parameters of the inner loop attitude controller are $K_2 = K_3 = K_4 = -80$ and $w_1 = w_2 = w_3 = 38$, see also the previous section.

In a first simulation, we assume an initial deviation of the angles $\mathbf{\Omega}^T = (\phi = 30°, \theta = -20°, \psi = 10°)$ where the control goal is to stabilize a hovering position, i.e. $\boldsymbol{v}_d = \mathbf{0}$. The obtained control result is shown in Fig. 5 as a time plot of all angles of the quadrotor. There is a very short

TABLE I

IDENTIFIED PARAMETERS OF THE QUADROTOR

Parameter	Value	Unit
g	9.81	m/s^2
m	0.5	kg
L	0.2	m
$I_x = I_y$	$4.85 \cdot 10^{-3}$	$kg \cdot m^2$
I_z	$8.81 \cdot 10^{-3}$	$kg \cdot m^2$
I_R	$3.36 \cdot 10^{-5}$	$kg \cdot m^2$
b	$2.92 \cdot 10^{-6}$	$kg \cdot m$
d	$1.12 \cdot 10^{-7}$	$kg \cdot m^2$

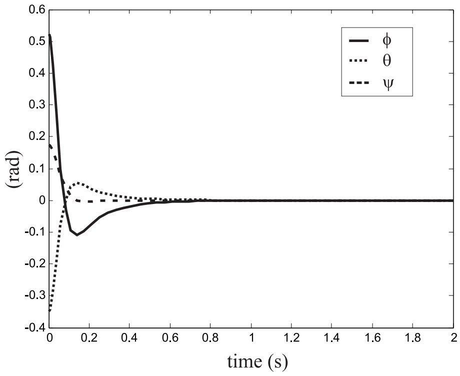

Fig. 5. Angles during attitude control.

transition phase with small under- and overshoots and the hovering state is reached after $t \approx 0.6$ sec.

In a second simulation, we choose the initial velocity $\dot{x} = 0.8\mathrm{m/sec}$ while the other two velocities are both zero. The desired state which has to be achieved by the control action is again the hovering state where all angles and all velocities are zero. The simulation result for the compensation of the initial velocity deviation is shown in Fig. 6 as a time plot of $\dot{x}(t)$. It becomes obvious that the velocity controller perfectly compensates the deviation while the overall velocity control is a little bit slower than pure inner loop attitude control. In Fig. 7, the angular velocities during that simulation are presented.

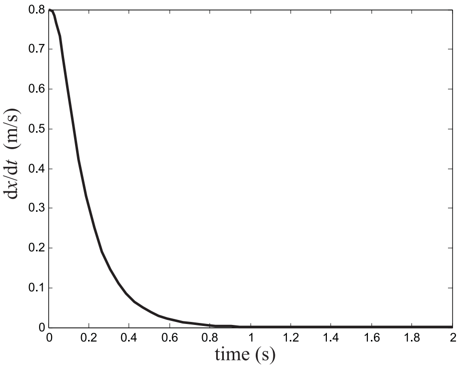

Fig. 6. Velocity in x-direction during velocity compensation.

A last simulation presented here covers the control of a given desired velocity vector. Therefore we assume that the quadrotor starts in the hovering state (i.e. all velocities are zero). The control task now comprises the stabilization of a desired velocity vector with $\dot{x}_d = \dot{y}_d = \dot{z}_d = 0.5$ m/sec and to generate a linear movement. The plot of all velocities during the control action is presented in Fig. 8. Again, the desired state is achieved after a short transition phase and the quadrotor is moving with constant velocity. During that constant flight the angles are also kept constant and hence the angular rates are all zero after the initial transition. Currently the control algorithms are implemented in the experimental setup for further testing.

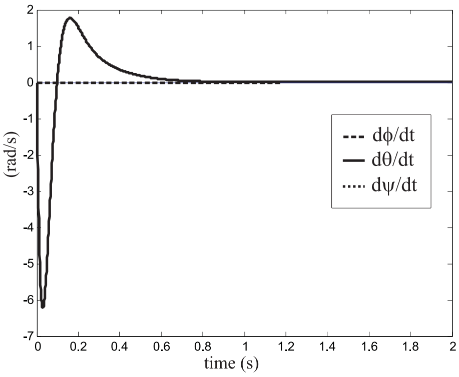

Fig. 7. Angular velocities during velocity compensation.

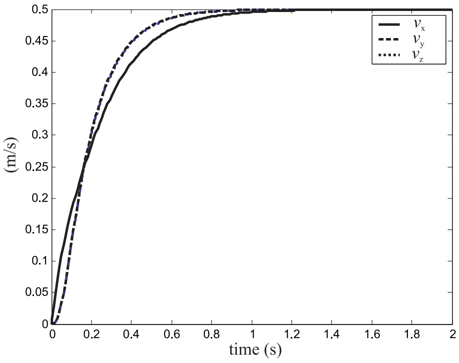

Fig. 8. Velocities during velocity control.

V. CONCLUSIONS AND FUTURE WORKS

This paper presents a vehicle control system for a quadrotor Micro-UAV based on a combined control strategy including feedback linearization to cope with the nonlinear dynamic behavior of the vehicle. Both an inner-loop attitude controller and an outer-loop velocity controller have been developed during the proposed work. The dynamic model of the quadrotor is derived and implemented in a Matlab/Simulink simulation model. With the help of that simulation, the nonlinear vehicle control system is tested and its efficiency demonstrated. In our ongoing work we are currently implementing the proposed control system in the real quadrotor UAV.

REFERENCES

[1] S. Bouabdallah, R. Siegwart, "Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor", *in Proc. of the IEEE International Conference on Robotics and Automation*, 2005, pp. 2247 2252.

[2] A. Tayebi, S. McGilvray, "Attitude Stabilization of a VTOL Quadrotor Aircraft", *in IEEE Transactions on control systems technology*, 2006, Vol. 14, 2006, pp. 562 - 571.

[3] P. Castillo, A. Dzul, R. Lozano, Real-time stabilization and tracking of a four-rotor mini rotorcraft, *IEEE Trans. on Control Systems Technology*, VOL.12, No. 4, July 2004, pp. 510 - 516.

[4] H. Voos, "Nonlinear State-Dependent Riccati Equation Control of a Quadrotor UAV", *in Proc. of the IEEE Conference on Control Applications*, Munich, Germany, 2006.

[5] H. Voos, Nonlinear and Neural Network-based Control of a Small Four-Rotor Aerial Robot, *in Proc. of the IEEE/ASME Int. Conference on Advanced Intelligent Mechatronics*, Zurich, CH, 2007.