The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

3D Trajectory Control for Quadrocopter

Tim Puls and Andreas Hein, *Member, IEEE*

Abstract— **The demand for unattended aerial systems capable of fulfilling e.g. surveillance tasks in contaminated or inaccessible areas without any assistance of a human pilot is the motivation for the investigation of a 3D trajectory control. Hence, this paper deals with the development of such a control algorithm able to follow any kind of 3D trajectory within the quadrocopter's capabilities. In this paper the 3D trajectory control algorithm is described. Many simulations were done to find the optimal trajectory course. Further, the control algorithm was implemented on a flight demonstrator for validation and experimental results regarding linear and circle trajectories are provided.**

Index Terms— **UAS, UAV, trajectory control, VTOL aircraft**

I. Introduction

It is foreseen that there will be a future market for intelligent service and surveillance robots, capable of discreetly penetrating confined spaces and maneuvering in those without the assistance of a human pilot tele-operating the vehicle. Thus, the development of small unattended aerial vehicles (UAV) for outdoor and urban applications which are able to perform agile flight maneuvers is of significant importance. Such vehicles can also be used for establishing ad-hoc networks in environments where direct or remote human assistance is not feasible, e.g. in contaminated areas or in urban search and rescue operations for locating earthquake victims. Especially the abilities of hovering above a given fixed position and maneuvering with high agility at low speed are essential for the mentioned applications. For this reason, it was decided to investigate four rotor vertical takeoff and landing (VTOL) helicopters instead of fixed-wing aircrafts.

A. Problem description

One of the main requirements of such systems is the ability to operate without any assistance of a human pilot. The UAV must be able to start, hover [15], navigate [16] and land [17] fully autonomously. However, in the recent past showed that these core capabilities are not sufficient to cover all kind of flight requirements.

For example, it would be easier for an operator if he could handle any kind of aerial vehicle in the same way without distinguishing between fix-wing aircrafts and quadrocopters. Thus, the quadrocopter described in this paper must be able to follow the same flight schedules as a fix-wing aircraft including flying arcs, circles or any other kind of 3D trajectory.

Another requirement is the capability to follow a vehicle or to fly in front of a vehicle with a given distance and velocity. For that task, a VTOL must also be able to follow any kind of trajectories in x, y and z. With such a control algorithm a VTOL would even be able to land on moving objects, which is another capability a fully autonomous VTOL should have.

Furthermore for observation tasks, it is often useful to fly around the target object to gather information from any direction. For example a camera underneath the VTOL must always be oriented toward an object while the VTOL circles around it. That means the VTOL must yaw while flying the circle (geocentric circle).

II. State Of The Art

There are publications regarding modeling and control of quadrocopters such as [1], [2] and [3] but the authors mainly provide different types of attitude controllers and primarily not position controllers. In the recent past researchers investigate position control of VTOL for indoor applications. For example, in [11] a position controller based on monocular vision has been implemented and tested indoors. The quadrocopter was able to hover above a position within a range of around 1m.

In [6] the ability of combining the inertial measurement unit (IMU) with global positioning system (GPS) to enable the acquisition of accurate position information has been described. Additionally, the authors describe the stabilization of VTOL systems with and without GPS. Navigation between different waypoints is undisclosed.

[12] deals with trajectory planning of a 4 rotor helicopter in GPS-denied environments. A belief roadmap algorithm was used to plan trajectories in indoor environments.

In [13] a semi-autonomous "position hold" algorithm and waypoint navigator is presented. The procedure does not consider disturbances, e.g. that the VTOL could be pushed away from its trajectory. In [16], a waypoint navigation algorithm is presented, which is able to fly linear trajectories but is not able to follow circles or arcs.

In [10], a 2D trajectory planning and control algorithm is presented. The trajectory is divided into a sequence of n desired waypoints. Between each waypoint the trajectory is linear. Following 3D trajectories is not mentioned.

In [5], a direct method to generate time-optimal trajectories using B-spline functions has been described, and in [17], a prototype of an autonomous controller based on the LQR method has been introduced capable of following trajectories. The focus concentrates on the optimization of the speed profile independently of the trajectory.

Manuscript received March 10, 2010.

Both authors are with OFFIS, Escherweg 2, 26121 Germany (phone: +49 441 2176970; e-mail: tim.puls@offis.de).

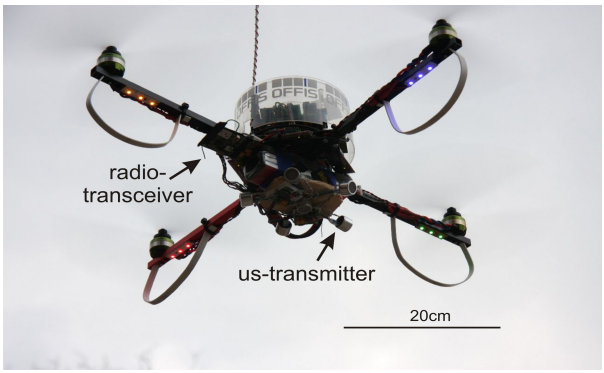

Fig. 1: The experimental platform: Quadrocopter equipped with radio-transceiver and ultra sonic transmitters.

III. Target System

Fig. 1 shows a photograph of the 4 rotor helicopter presented in real flight. It was used to test and verify the developed algorithm.

A. Sensors

The system is equipped with sensors to measure and calculate the current pose of the VTOL. Three gyroscopes for the angles φ, θ, ψ, three accelerometers for *x, y, z* (cf. Fig. 2) and three magnetic field sensors are attached to the system. These sensors are primarily used to detect the attitude of the system. Due to their inherent inaccuracy and drift-behavior these sensors are insufficient to calculate the position in all three dimensions dependably over time, so that additional sensors are necessary. Hence, a barometer is employed to correct the calculated height and a GPS receiver is applied to detect the position. This paper does not concentrate on the question of how to detect and calculate a precise geodesic position via GPS (see [6], [7]). Due to real-time constraints, raw GPS data are considered for test purposes.

B. Attitude and Position Controller

The attitude controller developed in [3],[4] for the angles φ (roll), θ (pitch) and ψ (yaw) was implemented on the target system. Due to the under-actuation of the quadrocopter, every position change leads to an adjustment of the roll and pitch angles φ and θ.

IV. Theoretical Foundations

An example of the mathematical modeling of quadrocopters is shown in Fig. 2, with the initial body-fixed frame in the geometric center. The orientation of the body can be described by a rotation *R*: body-fixed *f* → inertial *g* [4].

By means of this model, the three rotational and the three translational differential equations of the system can be derived.

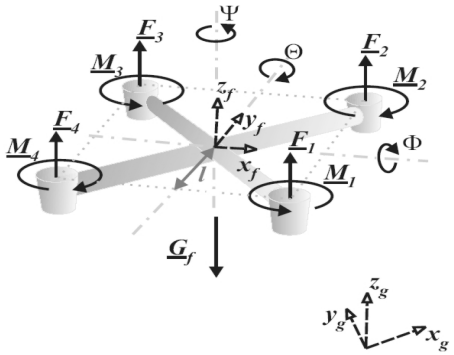

Fig. 2: System with force and torque Control

The following equations 1 (rotation) and 2 (translation) are the basis for the development of the position controller.

$$\begin{aligned} I_{xx}\cdot\ddot{\varphi} &= l\cdot(u_2 - c_{D4}\cdot\dot{\varphi}^2) - (I_{xx}-I_{yy})\dot{\theta}\dot{\psi} \\ &\quad +\dot{\theta}\cdot I_z^{rot}\cdot\Omega_z \\ I_{yy}\cdot\ddot{\theta} &= l\cdot(u_3 - c_{D5}\cdot\dot{\theta}^2) - (I_{yy}-I_{zz})\dot{\psi}\dot{\varphi} \\ &\quad +\dot{\varphi}\cdot I_z^{rot}\cdot\Omega_z \\ I_{zz}\cdot\ddot{\psi} &= u_4 - c_{D5}\cdot\dot{\psi}^2 - (I_{yy}-I_{xx})\dot{\varphi}\dot{\theta} \end{aligned} \tag{1}$$

$$\begin{aligned} m\cdot\ddot{x}_g &= u_1\cdot(\cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi) \\ &\quad -c_{D1}\cdot\dot{x}_g^2 - (w_y\dot{z}_g - w_z\dot{y}_g) \\ m\cdot\ddot{y}_g &= u_1\cdot(\cos\varphi\sin\theta\cos\psi - \sin\varphi\sin\psi) \\ &\quad -c_{D2}\cdot\dot{y}_g^2 - (w_z\dot{x}_g - w_x\dot{z}_g) \\ m\cdot\ddot{z}_g &= u_1\cdot(\cos\varphi\cos\theta) - mg \\ &\quad -c_{D3}\cdot\dot{z}_g^2 - (w_x\dot{y}_g - w_y\dot{x}_g) \end{aligned} \tag{2}$$

where *I* is the inertial tensor, *l* the distance between rotor and center, *m* the mass of the system, u_i the inputs and c_{Di} the coefficients of drag.

V. Position Filtering

Due to the inaccuracy of the accelerometers and gyroscopes, it is necessary to correct the onboard-calculated position and velocity. For that purpose a Kalman filter is used to correct these values by means of the GPS data. The frequency of the onboard calculations is 500Hz and the frequency of the GPS data is 5Hz.

However, due to the signal propagation delay of the GPS-signals (t_g) and the calculation time (t_c), the GPS data does not describe the actual state S_t of the system but the state S_{t-t_p} in the past, with $t_p = t_g + t_c$ (cf. Fig. 3). In the described system, the delay is approx. 500ms.

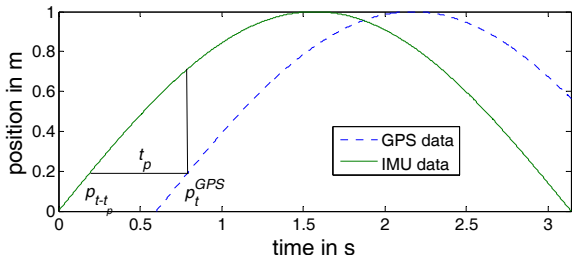

Fig. 3: Signal propagation delay of GPS data during a translation.

Thus, the GPS position p_t^{GPS} and velocity v_t^{GPS} have to be filtered onboard via the IMU-calculated position p_{t-t_p} and velocity v_{t-t_p} of the past.

$$p_{t+1} = p_t + K \cdot \left(p_t^{gps} - p_{t-t_p}\right),$$

where K is the Kalman gain (v_{t+1} analogously). The states of the past have to be updated in the same way. Otherwise, the filter would begin to resonate.

VI. 3D Trajectory Control Algorithm

This section focuses on the 3D trajectory control algorithm which is adapted from the 2D position control described in [15] and [16]. The trajectory algorithm is calculated with 100Hz, the attitude and altitude control with 500Hz.

A basic prerequisite of the control algorithm is the ability to maintain a given state S_d of the UAV defined by

$$S_d = \{x_d, y_d, z_d, \dot{x}_d, \dot{y}_d, \dot{z}_d, \ddot{x}_d, \ddot{y}_d, \ddot{z}_d\}.$$

For that purpose, the following position, velocity and acceleration controllers were used. Afterwards, it is described how the trajectories are calculated followed by an investigation on the state transitions.

A. Position Control

Using the current state S_c of the VTOL and S_d, the bearing ω and 2D distance d toward the target position are calculated. These two values are directly computed from longitudes and latitudes, cf. [14]. The distance d can be divided into its components:

$$d_x = \cos(\omega) \cdot d, d_y = \sin(\omega) \cdot d.$$

Due to the fact that the current yaw angle ψ_c must be considered ω is corrected by $\omega_\psi = \psi_c - \omega$.

At first, the desired angles θ_d and φ_d for position and the thrust thr for the altitude are calculated by an adapted PI controller with the parameters k_p^{xy} and k_p^z as well as k_i^x, k_i^y and k_i^z.

$$\begin{aligned}\varphi_d^{PI} &= \sin(\omega_\psi) \cdot k_p^{xy} \cdot d + \sin(\psi_c) \cdot I_x - \cos(\psi_c) \cdot I_y \\ \theta_d^{PI} &= \cos(\omega_\psi) \cdot k_p^{xy} \cdot d + \cos(\psi_c) \cdot I_x + \sin(\psi_c) \cdot I_y \\ thr^{PI} &= k_p^z \cdot e_z + I_z,\end{aligned}$$

with $e_z = z_d - z_c$.

The integral part of the controller is used to eliminate a steady state deviation from the desired position. In contrast to a classic I-controller, it uses two sets of parameters k_{ih} and k_{il} dependent on the current movement direction. Thus, the integral part I_x is:

$$I_x = \int g(t)dt, g(t) = \begin{cases} k_{ih} \cdot d_x, & \text{if } d_x \cdot \dot{x}_c < 0 \\ k_{il} \cdot d_x, & \text{else.} \end{cases}$$

I_y and I_z are derived analogously. The constant k_{ih} is chosen bigger than k_{il}, so that the integral part can react rapidly to the quadrocopter being pushed off position by wind. Then, the lower factor k_{il} ensures that the correct position is approached slowly with limited overshooting.

Furthermore, all integrals (I_x, I_y and I_z) are limited to prevent integral windup.

B. Velocity and Acceleration Control

The velocity and acceleration control are the most important terms to stabilize the VTOL, because a quadrocopter is an under-actuated system with almost no internal damping characteristics. Hence, two different kinds of damping are added to the controller: The velocity-damping VD and the acceleration-damping AD.

Thus φ_d^{PI}, θ_d^{PI} and thr^{PI} are adapted by:

$$\begin{aligned}\varphi_d &= \varphi_d^{PI} + \left(\sin(\psi_c) \cdot VD_x - \cos(\psi_c) \cdot VD_y\right) \\ &\quad + \left(\sin(\psi_c) \cdot AD_x - \cos(\psi_c) \cdot AD_y\right) \\ \theta_d &= \theta_d^{PI} + \left(\cos(\psi_c) \cdot VD_x + \sin(\psi_c) \cdot VD_y\right) \\ &\quad + \left(\cos(\psi_c) \cdot AD_x + \sin(\psi_c) \cdot AD_y\right) \\ thr &= thr^{PI} + VD_z + AD_z\end{aligned}$$

with $VD_x = k_v(\dot{x}_d - \dot{x}_c)$ and $AD_x = k_a(\ddot{x}_d - \ddot{x}_c)$.
VD_y, AD_y and VD_z, AD_z analogously.

As soon as φ_c and θ_c are unequal to zero the thrust vector is not pointing exactly up so that thr has to be increased to achieve the same lift. For that reason thr is adjusted in the following way.

$$thr^{new} = thr \cdot \frac{1}{\cos\varphi_c \cdot \cos\theta_c}.$$

Due to the quadrocopter's physical limitations, the total tilt angle $\alpha = \left|\begin{pmatrix}\varphi_d \\ \theta_d\end{pmatrix}\right|$ is limited to a maximum tilt angle $\alpha_{\max}$ while keeping the tilt direction:

$$\begin{aligned}\tilde{\varphi}_d &= \varphi_d \cdot min(1, \alpha_{max}/\alpha) \\ \tilde{\theta}_d &= \theta_d \cdot min(1, \alpha_{max}/\alpha)\end{aligned}$$

Furthermore, the thrust is limited to remain in the interval $[thr^{\min}, thr^{\max}]$.

In case of an unwanted descending of the VTOL, e.g. due to wind disturbances, it must be assured that the altitude control is considered preferentially. Thus, the calculated desired angles are updated by:

$$\begin{pmatrix}\tilde{\varphi}_d^{new} \\ \tilde{\theta}_d^{new}\end{pmatrix} = \begin{pmatrix}\tilde{\varphi}_d \\ \tilde{\theta}_d\end{pmatrix} \cdot z_{loss},$$

where $z_{loss} = \begin{cases} \frac{1}{(e_z - (e_z^{err} - 1))}, & e_z > e_z^{err} \\ 1, & \text{else} \end{cases}$, is a equation to antagonize the height loss problem [15] with a constant $e_z^{err} > 0$.

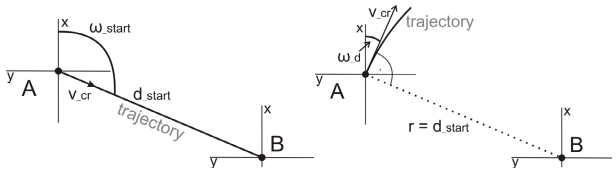

Fig. 4: Different trajectories - left: linear and right: circle.

C. Trajectory Calculation

For the described controller, it is necessary to continuously calculate the next desired state. That means if the UAV should fly from position A to B the controller has to calculate all states in between step by step.

For that calculation the algorithm needs a given cruising speed v_{cr} and a sink or climb rate $\dot{z}_d$. Out of these velocities, each value of the next desired state S_d is calculated by:

$$\dot{x}_d = v_{cr} \cdot \cos(\omega_d), \qquad x_d = \int \dot{x}_d \, dt, \qquad \ddot{x}_d = \frac{d\dot{x}_d}{dt}$$

$$\dot{y}_d = v_{cr} \cdot \sin(\omega_d), \qquad y_d = \int \dot{y}_d \, dt, \qquad \ddot{y}_d = \frac{d\dot{y}_d}{dt}$$

$$z_d = \int \dot{z}_d \, dt, \qquad \ddot{z}_d = \frac{d\dot{z}_d}{dt}$$

where ω_d is the direction of the trajectory at the current step.

For example for a linear trajectory (from position A to B, same altitude), ω_d is always the start bearing ω_{start}.

For a circle (with B as center of circle), $\omega_d = \omega_{circle} + (\pi \cdot dir)$ with dir is the direction of the circle[1] and

$$\omega_{circle} = \pi + \omega_{start} + \int \frac{v_{cr}}{r} dt \cdot dir$$

A key challenge is the smooth change of the cruising speed v_{cr} to get an optimal trajectory. Optimal means that the position error between S_c and S_d during the whole flight is minimal.

Fig. 5 gives an overview of several possible velocity courses if the cruising speed v_{cr} changes from $0\frac{m}{s}$ *to* $5\frac{m}{s}$. The alteration time t_{alter} is set *6s*.

VII. Simulations

In this section several simulation results are provided. All simulations were done in Matlab/Simulink with the differential equations 1 and 2 of section IV.

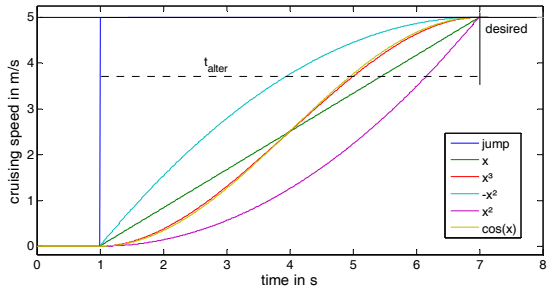

Fig. 5: Different velocity ramp functions.

[1] $dir = 1$ clockwise, $dir = -1$ counterclockwise,

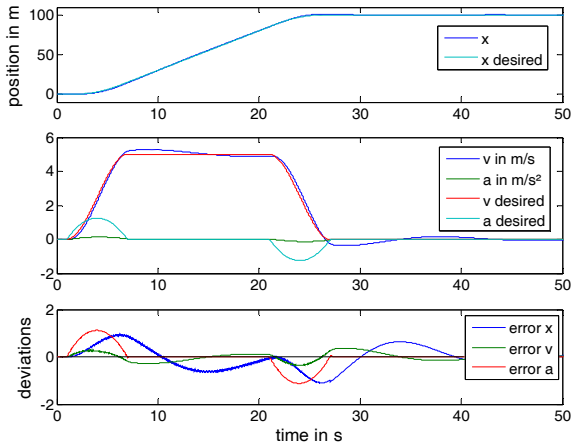

Fig. 6: 100m 1D-flight with 6s-cos(x)-ramp, $v_{cr} = 5\frac{m}{s}$

In Fig. 6, a simple 1D-trajectory of a 100m flight can be seen. As velocity ramp, the cos(x)-function was used with an rise time t_{alter} of 6s and an crusing speed v_{cr} of $5\frac{m}{s}$.

During this simulation, the average position error is 0.33m and the average speed error is 0.12m/s.

Fig. 7 shows the position error during the same 100m flight of Fig. 6 with the different ramp types of Fig. 5.

It can be seen that the different ramp functions (except the jump function) do not cause a significantly difference regarding the position error. But in Fig. 8 it can be seen that the discontinuous courses of the ramp functions "jump", "x", "-x²" and "x²" lead to unwanted jumps in the corresponding desired angle θ_d. Thus a continuous function should be used such as "cos(x)" or "x³".

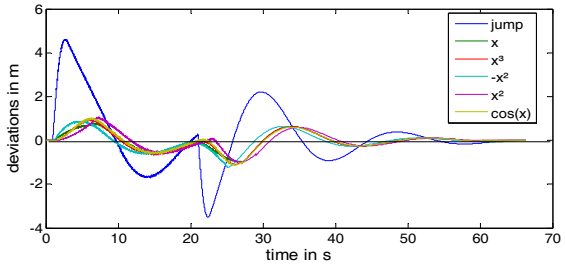

Fig. 7: Position error during linear flight with different ramps of Fig. 5

jump	x	x^3	$-x^2$	x^2	cos(x)
1.06m	0.32m	0.33m	0.33m	0.33m	0.32m

Table 1: Average absolute deviation of the position.

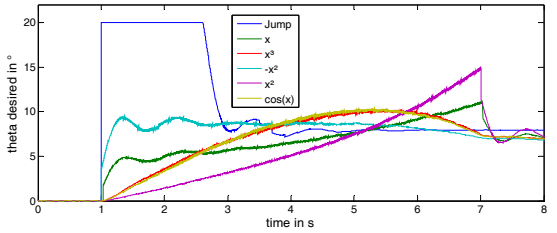

Fig. 8: Corresponding calculated angles θ_d to Fig. 5.

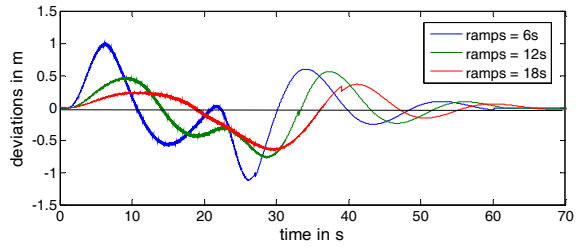

Fig. 9: Position error during 100m flight with different ramp times t_{alter}.

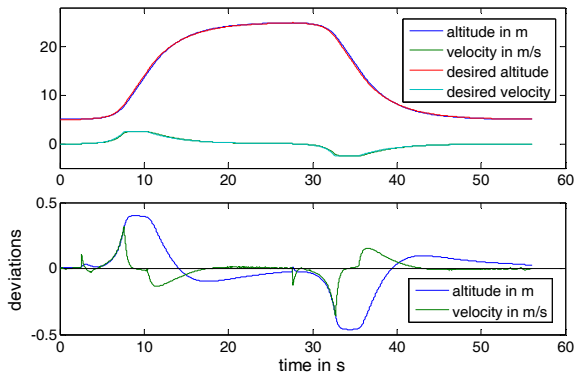

Fig. 10: Altitude change with x²-function and different start and stop ramp time, $v_{cr} = 3\frac{m}{s}$.

In Fig. 9, the effect of different alternation times t_{alter} of a cos(x)-ramp can be seen. It is apparent that a longer alternation time leads to a lower average error between S_c and S_d. For this reason, it is desirable to choose a longer alternation time if possible.

Fig. 10 shows the result of a altitude change using a x²-ramp and different alternation times. The average altitude error is 0.17m and the average velocity error is 0.04m/s.

Fig. 11 and Fig. 12 show the result of a geocentric circle trajectory with cos(x)-ramp and a cruising speed of $v_{cr} = 5\frac{m}{s}$. The radius is 25m.

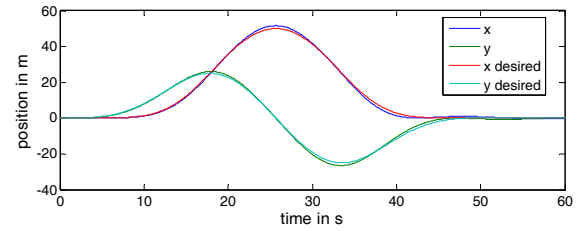

Fig. 11: Geocentric circle with 25m radius, 18s-cos(x)-ramp, $v_{cr} = 5\frac{m}{s}$.

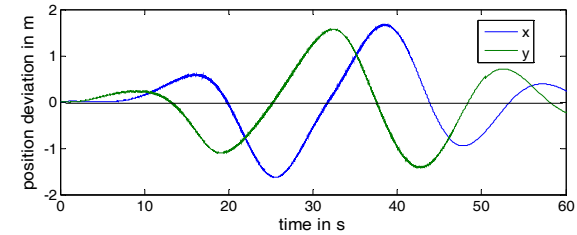

Fig. 12: Position error during geocentric circle.

The position error is greater than the position error of a linear trajectory. This is caused by the fact that the direction is changing continuously and by the tangential velocity vector, which does not point exactly to the next desired position. If the trajectory is known a priori this misalignment could be taken into account to improve the accuracy. In addition, because of the geocentric circle, the continuous adjustment of the ψ-angle causes an additional position deviation during the circle flight.

VIII. Experiments

This section provides the results of outdoor flight experiments.

In Fig. 13, the experimental result of a linear 3D trajectory is presented. The VTOL was supposed to fly bidirectional between two waypoints. These waypoints were approx. 150m apart with an altitude difference of 90m.

Table 2 shows the average absolute deviations during the trajectory.

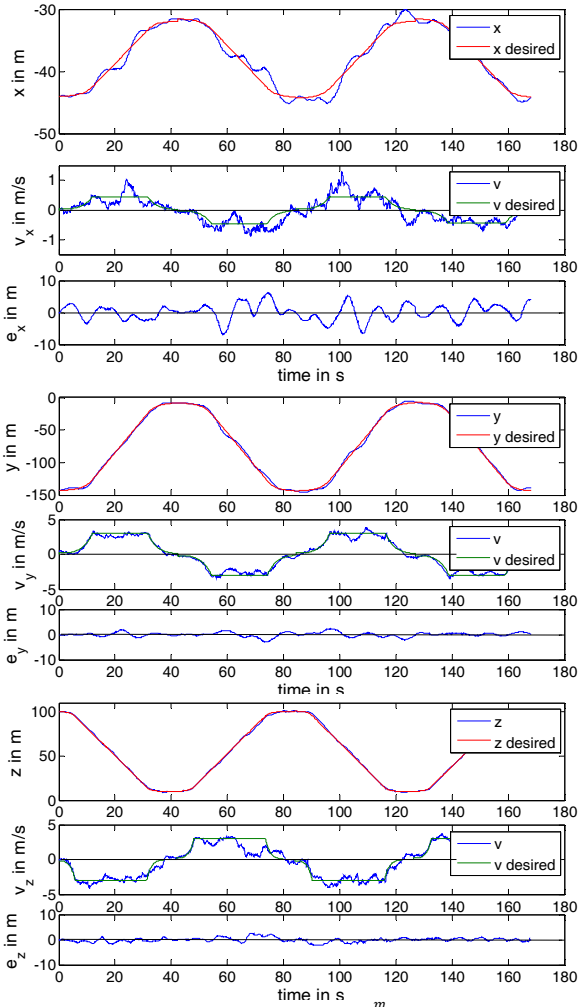

Fig. 13: 3D linear trajectory, 6s-x²-ramp, $v_{cr} = 3\frac{m}{s}$.

	x	y	z
position error	1.99m	0.6m	0.63m
velocity error	0.27m/s	0.15m/s	0.47m/s

Table 2: average absolute error of the position during trajectory.

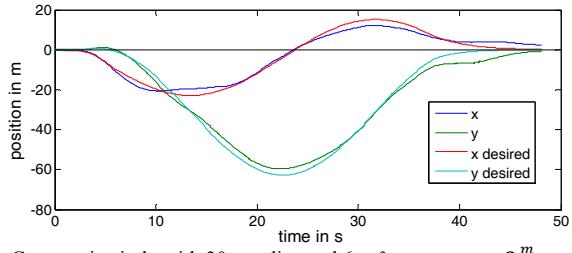

Fig. 14: Geocentric circle with 30m radius and 6s-x²-ramp, $v_{cr} = 3\frac{m}{s}$.

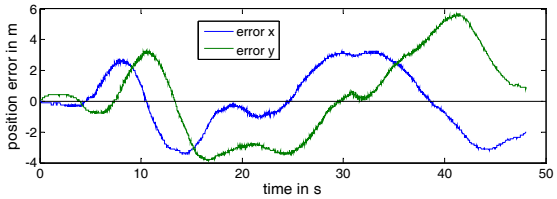

Fig. 15: Position error during geocentric circle.

It can be seen that the experimental results are similar to the simulations. Only in x-direction, the position error is notably different. This is caused by the windy condition outdoors.

At last, Fig. 14 and Fig. 15 show the experimental results of a geocentric circle flight. The radius was 30m with an x²-ramp and an alternation time of *6s*.

Again the experimental results closely match the simulations and show the ability of the algorithm to follow non linear trajectories such as a circle.

IX. Conclusion

In this paper a control algorithm for a quadrocopter was presented which is able to fly any kind of 3D trajectories in x, y and z. A quadrocopter is now able to follow or fly in front of vehicles if the position and velocity of the vehicle is known. In addition, with this algorithm a quadrocopter can be handled like conventional fix-wing airplanes while planning trajectories, which brings a huge benefit for the operators of such UAVs.

X. Outlook

The experiments showed further room for improvement of the trajectory control system.

As written in section VII, it would be benefiting to adjust the velocity vector of the algorithm while flying non-linear trajectories to decrease the position error.

Together with the results presented in [8], it should be possible to land on moving objects like vehicles using this algorithm. This is one of the next experiments we plan.

Acknowledgment

This research activity was performed in cooperation with Rheinmetall Defence Electronics GmbH, Bremen, Germany. It was partly sponsored by the go!CART (go! Competitive Aerial Robot Technologies) research cluster which is funded by the European Regional Development Fund and the state of Bremen, Germany.

References

[1] S.Bouabdallah, P.Murrieri and R.Siegwart, *Design and Control of Indoor Micro Quadrotor, IEEE-International Conference on Robotics and Automation 2004,* New Orleans, USA, pp. 4393 – 4398.

[2] M. Kemper, M. Merkel and S. Fatikow: "A Rotorcraft Micro Air Vehicle for Indoor Applications", Proc. of 11th Int. IEEE Conf. on Advanced Robotics, Coimbra, Portugal, June 30 - July 3, 2003, pp. 1215-1220.

[3] M.Kemper, S.Fatikow: *Impact of Center of Gravity in Quadrotor Helicopter Controller Design*, in: Proc. of Mechatronics 2006, 4th IFAC Symposium on Mechatronic Systems, Heidelberg, Germany, September 12th - 14th 2006, pp. 157-162.

[4] M.Kemper, *Development of an Indoor Attitude Control and Indoor Navigation System for 4-Rotor-Micro-Helicopter*, Dissertation, University of Oldenburg, Germany, 02. Feb 2007.

[5] Y. Bouktir, M. Haddad and T. Chettibi, *Trajectory planning for a quadrotor helicopter, 16th Mediterranean Conference on Control and Automation (MED08)*, Juni 2008,, Ajaccio, Corsica, France pp.1258-1263

[6] Jan Wendel, *Integrierte Navigationssysteme,* Oldenbourg Wissenschaftsverlag GmbH, March 2007

[7] U-blox AG, *Essentials of Satellite Navigation,* Compendium Doc ID: GPS-X-02007-C, April 2007
Available: http://www.u-blox.com/customersupport/docs/GPS_Compendium(GPS-X-02007).pdf

[8] T.Puls and A.Hein, *Outdoor Position Estimation and Autonomous Landing Algorithm for Quadrocopter using a Wireless Sensor Network, IEEE/ASME International Conference on Advanced Intelligent Mechatronics*,July 2010, Canada, Montreal, submitted

[9] Nanotron Technologies, Available: http://www.nanotron.com

[10] G.M.Hoffmann, S.L.Waslander and C.J.Tomlin, *Quadrotor Helicopter Trajectory Tracking Control, AIAA Guidance, Navigation and Control Conference and Exhibit*, August 2008, Honolulu, Hawaii, AIAA 2008-7410

[11] G.P.Tournier, M.Valenti and J.P.How, *Estimation and Control of a Quadrotor Vehicle Using Monocular Vision and Moiré Patterns, AIAA Guidance, Navigation and Control Conference and Exhibit,*21-24 August 2006, Keystone, Colorado, AIAA 2006-6711

[12] R.He, S.Prentice and N.Roy, *Planning in Information Space for a Quadrotor Helicopter in a GPS-denied Environment, IEEE International Conference on Robotics and Automation* (ICRA 2008), 19-23 May 2008, Pasadena Conference Center, Pasadena, CA, USA

[13] O.Meister, R.Mönikes, J.Wendel, N.Frietsch, C.Schlaile and G.F.Trommer, *Development of a GPS/INS/MAG navigation system and waypoint navigator for a VTOL UAV, SPIE Unmanned Systems Technology IX,* Volume 6561, pp. 65611D, 9-12.April 2007, Orlando, FL, USA

[14] B.Hofmann-Wellenhof, M.Wieser and K.Legat, *Navigation: Principles of Positioning and Guidance*, Springer Verlag, Wien, 1.Auflage, Oktober 2003

[15] T.Puls, H.Winkelmann, S.Eilers, M.Brucke and A.Hein, *Interaction of Altitude Control and Waypoint Navigation of a 4 Rotor Helicopter, German Workshop on Robotics* (GWR 2009), 09-10 June 2009, Braunschweig, Germany, pp.287-298

[16] T.Puls, M.Kemper, R.Küke and A.Hein, *GPS-based Position Control and Waypoint Navigation System for Quadrocopters, IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS09), October 11-15,2009, St.Louis, USA, pp.3374-3379

[17] Cowling, I.D., Whidborne, O.A. Yakimenkoand J.F. und Cooke, A.K.: *A Prototype of an Autonomous Controller for a Quadrotor UAV*. Kos, Greece, Jul 07. European Control Conference.