Autonomous Quadrotor Flight Using a Vision System And Accommodating Frames Misalignment

Guillaume Ducard and Raffaello D'Andrea
Measurement and Control Laboratory, Mechanical and Process Engineering Department
ETH Zurich, Sonneggstrasse 3, ML K, 8092, Zurich
Email: ducard@imrt.mavt.ethz.ch; dandrea@ethz.ch

***Abstract*—In this project, a Vicon motion capture/positioning system is employed to provide the pose (ground position, ground velocity and attitude) of an autonomous quadrotor. Some markers are mounted on the vehicle so that the Vicon system can track it. The markers arrangement is stored once in the Vicon system. In practice, after a few flights the markers' configuration may be slightly altered causing offsets in the measurement of the true attitude of the quadrotor. The Vicon system itself may also measure the vehicle's attitude with some errors in the angles due to small displacements of the cameras and a long period of use without recalibration of the whole system. Since it would be very tedious and cumbersome to recalibrate the whole vision system and making sure that the markers arrangement is still valid before every flight, a simple method has been developed to accommodate any offset in the attitude measurements. Better flight performance is obtained using the method presented.**

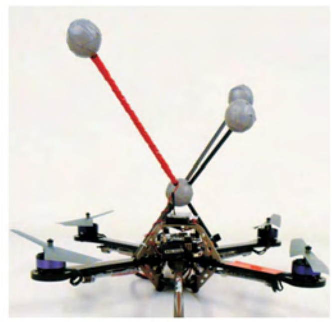

Fig. 1. X3D quadrotor with its four markers used for visual motion capture by a Vicon system

I. Introduction

Quadrotors have become an extremely popular platform for universities to test new flight control and guidance algorithms. For some industrial companies, quadrotors are also a very appealing vehicle for autonomous flight in order to perform delicate activities such as aerial filming, inspection of buildings and constructions, monitoring of power lines, *etc.*

This paper deals with practical attitude measurement errors that occur when using an vision based motion capture/positioning system and markers that are mounted on a rigid body. In this work, the indoor autonomous flight of a quadrotor is considered. In order to localize the vehicle and measure its orientation, the vision system (Vicon) requires that some markers with reflecting infrared tape are mounted on the vehicle, see Fig. 1. However, if the spatial arrangement of the markers is modified relative to the vehicle (after a crash, hand manipulation, *etc.*), it introduces misalignments between the body-frame seen by the vision system and the actual vehicle body-frame. In other words, when the vehicle is in perfect hover, meaning that the roll and pitch angles are zero, the vision system measures nonzero roll and pitch angles that may be offset by a few degrees. These measurement offsets have detrimental effects on the performance of the flight controller and introduce undesired behavior of the vehicle. One concrete example of this phenomena occurs when the vehicle hovers and is commanded to rotate about its yaw axis only. It is observed that the vehicle starts circling around its hover position as it yaws. These undesired side motions are explained by the presence of attitude measurement offsets. Therefore, this paper suggests a method to identify very simply these offsets and to remove them in the flight controller.

In addition, in the case of autonomous flight of several vehicles simultaneously, it would be very cumbersome to recalibrate each vehicle (based on its markers' topology) before each flight. The method presented provides a means for automatic calibration at the beginning of each flight for all the vehicles in the fleet.

At the MIT [2] and at Boeing [3], a similar Vicon vision system is employed to demonstrate autonomous flight of indoor flying vehicles. To the knowledge of the authors, issues about frames misalignment in autonomous indoor flight control, using a visual motion capture system, have not been directly addressed in a paper yet.

This paper starts with the description of the ETH Flying Machine Arena testbed, www.flyingmachinearena.org. Then the flight control system is presented in order to point out which part of the the flight controller is sensitive to frame misalignments. Finally, the paper concludes on the effectiveness of the simple method presented to accommodate the attitude angles offsets. The resulting algorithms are tested in full autonomy on commercially available quadrotors (X3D quadrotor from Ascending Technologies, [1]) as shown in Fig. 1 in our flight testing facility.

II. System Description

The ETH Flying Machine Arena is a 10 x 10 x 10 m indoor testbed for the development of control strategies for the autonomous flight of small aerial vehicles. The sensing and communication infrastructure is organized similar to the MIT

Raven testbed [2]. An 8-camera Vicon motion capture system provides complete pose data for all appropriately marked vehicles in the arena at 200Hz, with a latency of about 10 ms. Robustness is achieved by having unique 4-marker patterns for every vehicle: any single marker in each configuration may be lost without loss of the vehicle. Only 2 cameras are required to observe any given marker so occlusion/distance-based sensitivity issues are largely avoided. As a result each vehicle is robustly localized for all 6 degrees of freedom every 5 ms.

The vehicle pose data is distributed via UDP to a cluster of off-the-shelf machines that execute estimation and control algorithms. These processes can be deployed arbitrarily in either a centralized or a distributed manner, both physically (different/same machines) and computationally. The UDP network is designed to handle traffic rates of far above the currently-used traffic load while at the same time the high-refresh rate of the system allows for some packet losses without serious degradation in performance. All packets are sequence-numbered allowing for continuous monitoring of network performance. Currently packet drops on the UDP network are virtually nonexistent. An additional benefit of this architecture is that all data can be recorded and played back transparently for debugging and diagnostics. For running the system in simulation, the wireless and Vicon data bridges are simply replaced with a simulator process, with all of the other components remaining completely unaffected and unaware of any difference.

The flying vehicles currently used in the Arena are commercially available quadrotors designed and manufactured by Ascending Technologies [1]. Each vehicle is individually equipped with two dedicated wireless links: a 72 MHz FM PPM command channel (50 Hz 8-channel commands) and a 2.4 GHz ZigBee bidirectional radio. The PPM channel is used exclusively for controlling the vehicle to constrain the amount of variable latency in the system. The ZigBee links are used for health diagnostics (for example, battery voltage) and debugging.

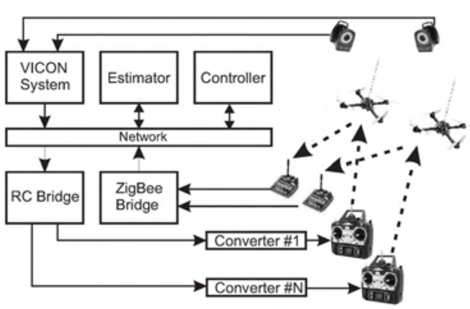

Fig. 2. Flying machine arena overview

III. Flight Controllers

A. Direction Cosine Matrix

The attitude transformation matrix (also called direction cosine matrix) is necessary to transform vectors and point coordinates from the quadrotor's body-fixed frame (b) to the navigation frame (n) and vice versa, see Fig. 3. The direction cosine matrix $\boldsymbol{C_n^b}$ transforms the vector $\boldsymbol{A}$ expressed in the navigation frame $\boldsymbol{A}^n$ into a vector expressed in the quadrotor's body fixed frame $\boldsymbol{A}^b = \boldsymbol{C_n^b}\boldsymbol{A}^n$, with the matrix $\boldsymbol{C_n^b}$ as defined as

$$\boldsymbol{C_n^b} = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix},$$

where roll, pitch and yaw angles are denoted by ϕ, θ, and ψ respectively.

The following sections present the flight control system designed for an X3D quadrotor. It is shown in Fig. 4.

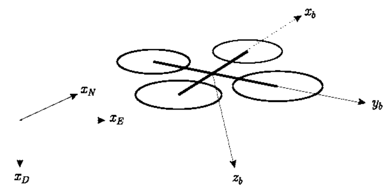

Fig. 3. Quadrotor configuration

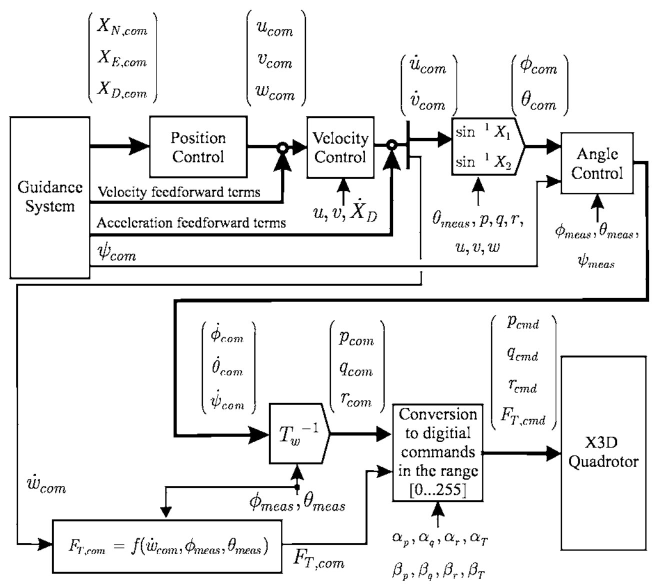

Fig. 4. Flight control system

B. Commands Sent to the X3D

The X3D quadrotor is delivered with an onboard controller for the turn rates. The actual commands that are sent to the vehicle are: roll-, pitch-, yaw-rate and thrust. They are denoted p_{cmd}, q_{cmd}, r_{cmd}, and T_{cmd} respectively, and are an integer number between 0 and 255.

C. Measurement Data Acquisition

The Vicon system measures the position $\boldsymbol{X^n_{meas}} = [X_N \ X_E \ X_D]$ and the orientation (ϕ_{meas}, θ_{meas}, ψ_{meas})

of the vehicle. The velocity of the vehicle is simply obtained by low-pass filtering the time differentiated position data $\dot{\boldsymbol{X}}^{\boldsymbol{n}}_{\boldsymbol{meas}} = [\dot{X}_N \; \dot{X}_E \; \dot{X}_D]^T$. The body-rates measured onbard the vehicle p_{meas}, q_{meas}, r_{meas} are accessible and sent back to the control computer via a XBee radio link.

D. Reference Trajectory

The guidance system provides the desired North, East, Down position coordinates $\boldsymbol{X}^{\boldsymbol{n}}_{\boldsymbol{com}} = [X_{N,com} \quad X_{E,com} \quad X_{D,com}]^T$ and generates velocity and acceleration feedforward signals. The guidance system also generates the desired yaw angle ψ_{com}.

E. Position and Velocity Control

The position error vector in the navigation frame is $\boldsymbol{\Delta X}^n = \boldsymbol{X}^{\boldsymbol{n}}_{\boldsymbol{com}} - \boldsymbol{X}^{\boldsymbol{n}}_{\boldsymbol{meas}}$, and is transformed into a position error vector in the body frame as follows:

$$\boldsymbol{\Delta X}^b = \boldsymbol{C}^b_n \boldsymbol{\Delta X}^n \tag{1}$$

The matrix $\boldsymbol{K_X}$ represents parallel position controllers that are designed to drive the error vector $\boldsymbol{\Delta X}^b$ to zero. The desired body accelerations are constructed as:

$$[\dot{u}_{com} \; \dot{v}_{com} \; \dot{w}_{com}]^T = \boldsymbol{K_V} \left[\boldsymbol{K_X} \boldsymbol{\Delta X}^b - [u \quad v \; w]^T \right], \tag{2}$$

where the 3×3 matrix $\boldsymbol{K_V}$ corresponds to parallel velocity controllers.

F. Generation of the Angle Commands

The desired angles are generated as follows:

$$\begin{aligned} \phi_{com} &= \arcsin(X_1) \;, \\ \text{with } X_1 &= \frac{\dot{v}_{com} + ru - pw}{g\cos\theta_m} \approx \frac{\dot{v}_{com}}{g\cos\theta_m} \; \in \; [-1;1], \\ \theta_{com} &= \arcsin(X_2) \;, \\ \text{with } X_2 &= \frac{\dot{u}_{com} + qw - rv}{-g} \approx \frac{\dot{u}_{com}}{-g} \; \in \; [-1;1], \end{aligned}$$

and finally, the angle commands are saturated

$$-\phi_{c,max} < \phi_{com} < \phi_{c,max} \;, \quad -\theta_{c,max} < \theta_{com} < \theta_{c,max} \;.$$

An appropriate selection of all of the controller gains is described in [4], where more details about the flight controller are available.

G. Generation of the Angle-rate Commands

The turn-rate command vector is computed as follows:

$$\begin{bmatrix} p_{com} \\ q_{com} \\ r_{com} \end{bmatrix} = T_w^{-1}(\phi_m, \theta_m) \begin{bmatrix} K_\phi(\phi_{com} - \phi_m) \\ K_\theta(\theta_{com} - \theta_m) \\ K_\psi(\psi_{com} - \psi_m) \end{bmatrix} \;, \tag{3}$$

where the proportional gain for each axis are K_ϕ, K_θ and K_ψ. We recall that the body-axis angle rates are p, q, and r and that the transformation matrix T_w^{-1} is expressed as follows:

$$T_w^{-1}(\phi_m, \theta_m) = \begin{bmatrix} 1 & 0 & -\sin\theta_m \\ 0 & \cos\phi_m & \cos\theta_m \sin\phi_m \\ 0 & -\sin\phi_m & \cos\theta_m \cos\phi_m \end{bmatrix} \;. \tag{4}$$

H. Thrust Command $F_{T,com}$

The vehicle utilizes its four propellers to generate a total thrust force F_T. The commanded thrust force $F_{T,com}$ is generated by the flight controller with

$$\begin{aligned} F_{T,com} &= m(\dot{w}_{com} + pv - qu - g\cos\phi_m \cos\theta_m) \;, \\ &\approx m(\dot{w}_{com} - g\cos\phi_m \cos\theta_m) \end{aligned} \tag{5}$$

where $F_{T,com}$ is saturated between 0 and $F_{T,max}$.

IV. Disturbance Discussion Due To Angle Offsets

A. Effects and Determination of Roll and Pitch Angle Offsets

Roll and pitch angle offsets have an influence in four locations in the flight control system, namely in the control variables X_1 and $F_{T,com}$ and in the transformation matrices C_n^b and T_w^{-1}. In the introduction of this paper we mentioned that the vehicle exhibits an undesired behavior that we attributed to offsets on the measured Euler angles. Indeed, when the vehicle is perfectly at hover and then commanded to yaw on the spot, it starts to have undesired side motion in the form of circles around the hover location. If we have a closer look at the last column of the matrix T_w^{-1}, we can see that a commanded yaw rate $\dot{\psi}_{com} = K_\psi(\psi_{com} - \psi_m)$ will affect the commanded roll rate p_{com} or pitch rate $q_c om$ if there is an offset in the pitch angle θ_m or roll angle ϕ_m, respectively, see Fig. 6. It will yield undesired side motion. If there is no angle offsets, then during hover $\phi_m = \theta_m = 0$, and therefore, the yaw rate command would not affect the other two axis turn-rates, see Fig. 7.

The determination of the roll and pitch angle offsets is done right after the take off of the vehicle. The flight controller described in Section III stabilizes the vehicle at hover for a few seconds during which the roll and pitch angles are measured and averaged, yielding $\overline{\phi}_m$ and $\overline{\theta}_m$, respectively. Since the vehicle is at hover, the true angles are zero and therefore the value of the offsets are $\phi_{\text{m,offset}} = \overline{\phi}_m$ and $\theta_{\text{m,offset}} = \overline{\theta}_m$ respectively.

B. Determination of the Yaw Angle Offset

The following method evaluates the offset on the yaw angle, it is denoted by ψ_{offset}.

First, the vehicle hovers at a certain location A as shown in Fig. 5. The vehicle is controlled with

$$\psi_{com} = 0, \quad \phi_{com} = 0 \quad \text{and } \theta_{com}(t) \;, \tag{6}$$

where the pitch angle command $\theta_{com}(t)$ makes the vehicle move slowly forward along its $x-$body axis. The Vicon system records the trajectory of the quadrotor. After the vehicle has flown a sufficient distance (the longest possible in our indoor airspace), the vehicle slows down and stops using pitch angle commands θ_{com} only.

The experiment is declared valid if the yaw and roll angle measurement data are on average zero. These conditions are there to check if the vehicle has flown sufficiently straight and are formulated as follows:

$$\left| \frac{\sum_{k=1}^{n} \psi_{meas}(k)}{n} \right| < \epsilon_\psi \ , \qquad \left| \frac{\sum_{k=1}^{n} \phi_{meas}(k)}{n} \right| < \epsilon_\phi \ . \quad (7)$$

As the vehicle flies from point A to point B as shown in Fig. 5, n measurement data of the North and East positions of the vehicle are recorded. In order to determine the yaw angle offset accurately, a least-squares formulation is employed as follows:

$$y_k = \phi_k^T \Theta, \quad (8)$$

where $y_k = X_{E,meas}(k)$, $\phi_k^T = [X_{N,meas}(k) \quad 1]$ and $\Theta^T = [\psi_{\text{offset}} \quad X_{E,0}]$. The estimated values $\hat{\psi}_{\text{offset}}$ and $\hat{X}_{E,0}$ are given by $\hat{X}_{E,0} = \overline{X}_E - \frac{S_{xy}}{S_{xx}} \overline{X}_N$, and $\hat{\psi}_{\text{offset}} = \tan^{-1}\left(\frac{S_{xy}}{S_{xx}}\right)$,

$$\text{where } \overline{X}_N = \frac{\sum_{k=1}^{n} X_N(k)}{n}, \quad \overline{X}_E = \frac{\sum_{k=1}^{n} X_E(k)}{n},$$
$$S_{xx} = \sum_{k=1}^{n} (X_N(k) - \overline{X}_N)^2, \quad (9)$$
$$S_{xy} = \sum_{k=1}^{n} (X_E(k) - \overline{X}_E)(X_N(k) - \overline{X}_N).$$

Note that a recursive least-squares algorithm can also be employed yielding the same results but removing the need for storing many measurement data.

V. Conclusion

New flight control and guidance systems for single and multiple autonomous vehicles are tested in the ETH flying machine arena. Position and attitude of each vehicle is obtained by a visual positioning system from the Vicon company. In order to track a vehicle, the vision system needs that some markers be mounted on the vehicle. After calibration the vision system is able to compute a frame associated with the rigid body of the flying vehicle based on the markers seen. However, when the mounting position of the markers is modified, the frame computed by the vision system does not match anymore the true rigid-body frame of the quadrotor. Thus, the attitude angles are offset, which degrades the flight control performance. This paper has shown how a simple strategy enables to automatically estimate the offsets on the Euler angles at the beginning of each new flight. Consequently, the time consuming process of recalibrating the vision system when the markers' configuration has been modified can be avoided. This method is all the more time appealing when dealing with several vehicles simultaneously.

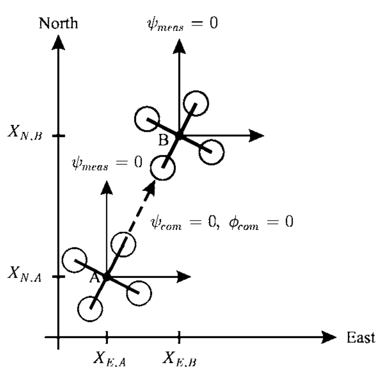

Fig. 5. Identification of the yaw angle offset

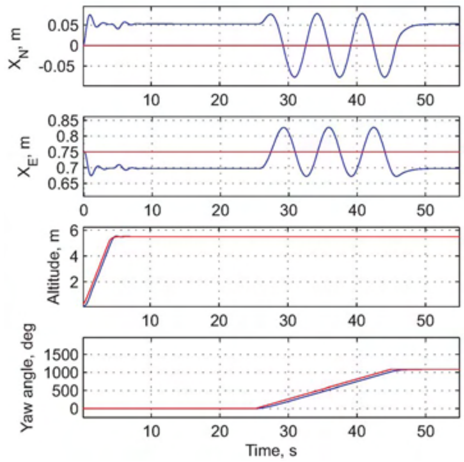

Fig. 6. Take off and hover flight with angle offsets: 3 deg on both roll and pitch angles. Simulation results

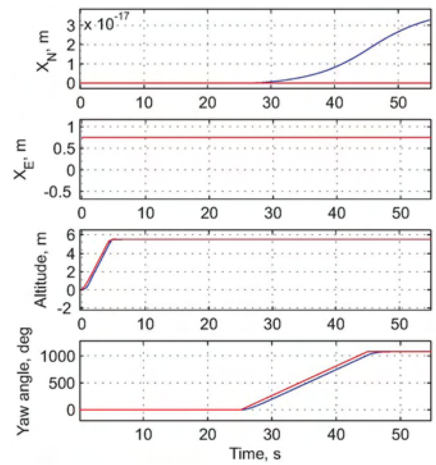

Fig. 7. Take off and hover flight with no angle offset. Simulation results

References

[1] D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger, and D. Rus, "Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1kHz," Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10-14 April, 2007.

[2] J. P. How, B. Bethke, A. Frank, D. Dale and J. Vian, "Real-time indoor autonomous vehicle test environment," IEEE Control Systems Magazine, vol. 28, no. 2, pp. 51-64, April, 2008.

[3] S. R. Bieniawski, D. Halaas and J. Vian, "Micro-Aerial Vehicle Flight in Turbulent Environments: Use of an Indoor Flight Facility for Rapid Design and Evaluation," Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, Hawaii, USA, 2008.

[4] G. Ducard, S. Lupashin and R. D'Andrea, "Simple Flight Control and Vehicle Parameter Identification Strategy for an Autonomous Quadrotor," submitted to *IEEE Conference on Control and Decision*, China, December 2009.