Photogrammetric Week '13 Dieter Fritsch (Ed.) Wichmann/VDE Verlag, Belin & Offenbach, 2013

Another Step towards Measuring the World from the Air: Model-based 3D Real-time Simulation of Micro-UAV

THOMAS LINKUGEL, Arnsberg and ANDREAS SCHILLING, Tübingen

ABSTRACT

Unmanned Aircraft Systems are increasingly used by commercial and scientific users. Due to new areas of application the demand for a fail-safe operation of unmanned aerial platforms grows steadily. An important requirement is the reliability of the algorithms for position/attitude recognition and control of the aircraft platform. Until now, this evidence could be provided exclusively by time-consuming and costly field tests.
In this paper a new simulation system is presented, in which a micro-UAS is transferred into a virtual reality. The solution developed here is based on an accurate modeling of the aircraft platform, the integrated sensors and the platform-specific dynamic flight characteristics and includes the on-board control algorithms that are tested "software in the loop" inside the simulation. It is shown, how this real-time simulation can be used to test and optimize algorithms but also how the system has been employed in the development of new flight systems. Finally an outlook on analysis-by-synthesis type measuring is given and the possibilities opened up by the simulation system for such measuring systems are explained.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAV) are nowadays increasingly being used by civil and scientific users in different fields of application. In civilian use, the safety and reliability of the whole system is an essential requirement for the operation of those flight platforms. The resilience of the geometrical structure and the materials used must be proofed by different test methods already in the phase of development. The reliability of the control software, however, could so far only be tested with a finished prototype and only through time-consuming and expensive field tests. The generation of reference data (ground truth) for the evaluation of algorithms for position and orientation estimation and controlling is a challenging and a very time-consuming work. Related work has been performed in the field of robotics, where the autonomy of micro-UAV is also a current research topic. A main task is the development of algorithms for simultaneous localization and mapping (SLAM) of the flight system environment [Grzonka et al. 2009, Morris et al. 2010]. For the analysis and verification of these algorithms, one important problem is generating reference data. In general, reference data can only be obtained under certain environmental conditions and by the use of very sophisticated motion capture systems [Mellinger et al. 2010, Vicon et al. 2010].
In this report, a novel "software in the loop" (SiL) simulation system is presented, by which a VTOL (Vertical Take-Off and Landing) micro-UAV is transformed into virtual reality. The solution developed here is based on a precise modeling of the flight platform, the integrated sensors and the platform-specific parameters. We show that with this model based 3D real-time simulation it is possible to test the reliability of algorithms under different conditions and to analyze the flight control and optimize data filtering. For this purpose, reproducible reference data from the simulated system are extracted and compared with the data from the algorithms for position and orientation estimation. Furthermore, we present the development and testing of a complete flight system with 6 motors / rotors (hexacopter) up to prototype maturity solely based on this simulation.

2. SYSTEM OVERVIEW AND CONCEPT

The basic idea of the simulation shown here is the precise modeling of all forces and moments of the flight platform and its flight dynamic characteristics. Based on this model the physical behavior of the system is simulated in order to produce accurate and realistic sensor data. In this virtual replica of the flight platform, the realistic integration of existing onboard control algorithms for the UAV is possible. This allows for early and cost-efficient software validation, as well as evaluation under repeatable and determinable conditions (software in the loop).

The SiL idea plays an essential role in the overall implementation characterized by a portable structure. The complete software portability of algorithms between simulation and real embedded system on the flight platform is a fundamental element of the simulation.

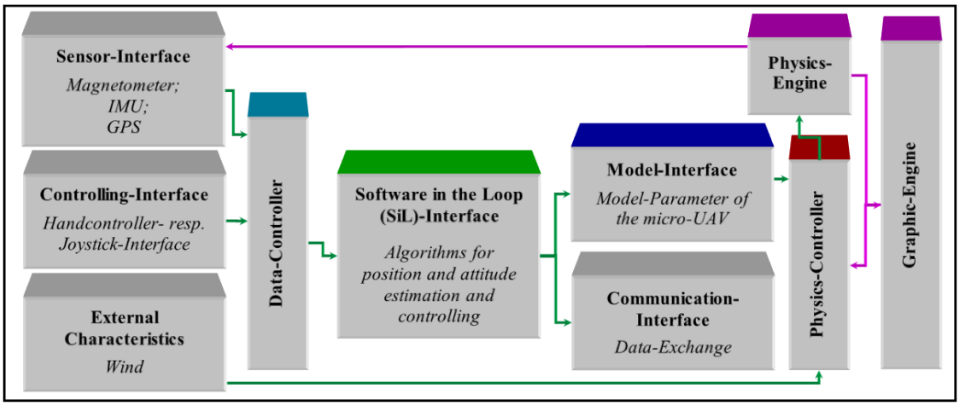


Figure 1: Concept of model-based 3D real-time simulation.

The initial point for the development of this model-based 3D real-time simulation was the quadrocopter AR100B from Air-Robot®. In the first step, the model parameters of the flight platform such as geometry, weight, thrust forces, etc. were determined and transferred to a specially developed mathematical model used for the calculation of the physical behavior of the flight platform in the simulation. This model is stored in the Model Interface (Figure 1, blue box). On this basis, a geometric model of the flight platform was created and a relationship between graphical elements (such as motors, rotors, etc.) and the associated physical properties were implemented. These relations are stored in the Physics-Controller (Figure 1, red box). The geometric model is equipped with sensors, such as an inertial measurement unit (IMU), a magnetic field sensor or a Global Positioning System (GPS) receiver. In the next step, the control unit of the flight platform is embedded. For this purpose we have defined a SiL-Interface (Figure 1, green box), which fully integrates the algorithms for position and orientation estimation and flight control in the simulation. Thus, any extension or modification within this interface is directly portable to the real UAV. Finally the previously calculated motor control values of the SiL module are transferred to the deposited mathematical model which produces the exact physical forces and moments of the flight platform. These values interact with the physics controller that provides the interface to Webots® respectively in some cases directly to the Physics-Engine. Webots® is a commercial simulation development environment, which provides the necessary interfaces between the physics and the Graphics-Engine (see Figure 1, magenta boxes). It also provides a very convenient interface to integrate realistic 3D-CAD models. Webots® works internally with the open source libraries Open Dynamics Engine (ODE) [ODE 2013] and Ogre3D [OGRE 2013], which are in this case also accessed directly to extract reference data or to implement sensors which are not supported. For further details of the simulation development environment one may refer to [MICHEL 2004]. In addition to the sensor inputs, external control inputs are connected to the SiL-simulation via a specially developed software interface, the Data-Controller (Fig. 1, cyan box). In this way it is possible to include security and service functions of the control unit (e.g. commands to hold the position or emergency landing when battery low) as well as manual control for a realistic simulation of the micro-UAV. An interface for external factors as e.g. wind can be used to control the simulation parameters via the Physics-Controller. The structure described here is shown in Figure 1.

3. SYSTEM DESCRIPTION AND MATHEMATICAL MODEL

In order to simulate the micro-UAV as accurate as possible, all the system parameters such as geometry, forces and torques, etc. had to be described and a mathematical model had to be derived. As already mentioned, the starting point was the quadrocopter AR100B (Figure 2, left).

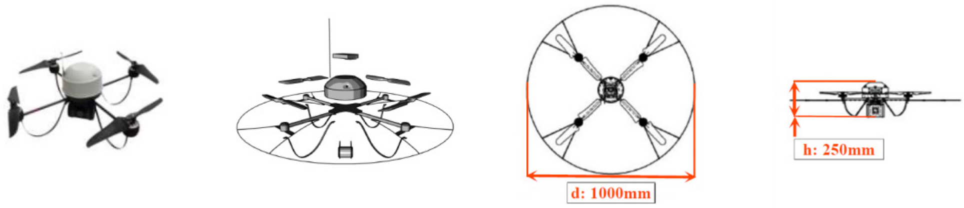


Figure 2: micro-Unmanned Aircraft System, from left to right: Real picture, explosion drawing, top view, front view.

3.1. System AR100B

The quadrocopter shown in Figure 2 has a maximum outer diameter of d = 1000mm and a building height of h = 250mm. The total weight including power supply (battery) is m = 1300g.

3.2. Mathematical Description

In this section, we present the precise characterization and development of the mathematical model, stored in the block *Model-Interface* (see Figure 1). The coordinate systems and basic physical parameters of the model are summarized in Figure 3.

Coordinate systems

To model the flight platform, two coordinate systems are defined. The body coordinate system ($\mathcal{b}$ (body)-frame) is defined by the three axes x^b, y^b and z^b. These are fixed relative to the quadrocopter and point to the longitudinal direction (x^b), to the right (y^b) and the upward direction (z^b) of the platform. The origin is the center of mass of the UAV. The world coordinate system ($\mathcal{w}$ (world)-frame) is described by the three axes x^w, y^w and z^w. Here, the x^w-axis points to the north and the y^w-axis to the east. The z^w-axis points upwards. The three Euler angles roll (ϕ), pitch (θ) and yaw (ψ) describe the rotation of the flight platform to their respective axis in $\mathcal{b}$ -frame. The transformation from one coordinate system into the other is calculated by three successive rotations:

- Rotate about the reference z-axis by the yaw angle ψ
- Rotate about the reference y-axis by the pitch angle θ
- Rotate about the reference x-axis by the roll angle ϕ

The rotation matrix for transformation of the $\mathcal{b}$ -frame to the $\mathcal{w}$ -frame thus can be expressed as a function of the Euler angles:

$$\boldsymbol{C}_b^w = \begin{pmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix} \qquad (1)$$

Where $c\theta$ respectively $s\psi$ representing the $\cos\theta$- bzw. $\sin\psi$ Function. For the derivation of this relationship reference is made to [TITTERTON et al.1996].

Forces and moments

The description of all the aerodynamic forces and torques acting on the flight platform is a very complex part of the system modeling. Below the physical quantities, shown in Figure 3 left, are derived and optimal approximations are found. For similar considerations on these physical properties we refer to the contributions of [MELLINGER et al. 2010] and [GUENARD et al. 2005].

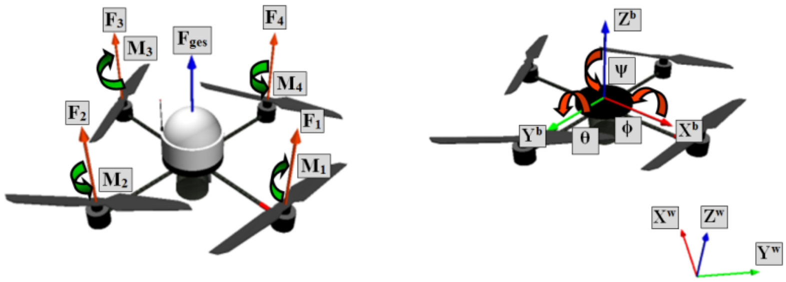


Figure 3: Forces (*left*) and Coordinate systems (*right*) of the micro-UAV.

The accelerations $\dot{\vec{v}}^w$ with respect to the $\mathscr{w}$ -frame, which are crucial for the translational motion of the flight platform, can be derived in accordance with the principle of superposition from the gravitational acceleration $\vec{g}$ and the acceleration vector of the flight platform $\vec{a}^b$ with respect to the $\mathscr{b}$ –frame:

$$\dot{\vec{v}}^w = \boldsymbol{C}_{\boldsymbol{b}}^{\boldsymbol{w}} * \vec{a}^b + \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}. \tag{2}$$

From this we get with mass m of the flight platform and thrust F_i of each motor / rotor combination in the z^b-direction:

$$\dot{\vec{v}}^w = \frac{1}{m} * \boldsymbol{C}_{\boldsymbol{b}}^{\boldsymbol{w}} * \begin{pmatrix} 0 \\ 0 \\ \sum_{i=1}^{4} F_i \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix}. \tag{3}$$

The entire thrust $\overrightarrow{F_{\text{ges}}}^b$ of the system in the $\mathscr{b}$ -frame is then:

$$\overrightarrow{F_{\text{ges}}}^b = \begin{pmatrix} 0 \\ 0 \\ \sum_{i=1}^{4} F_i \end{pmatrix} + \boldsymbol{C}_{\boldsymbol{b}}^{\boldsymbol{w}^T} * m * \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix}. \tag{4}$$

Up to now we have not considered the flow resistance $\overrightarrow{F_{C_w}}^w$ of the flight platform. This force results from the movement of the micro-UAV and counteracts the direction of movement. $\overrightarrow{F_{C_w}}^w$ is determined by the context of air density ρ, flow resistance coefficient C_w, reference area A of the flight platform and flight velocity $\vec{v}^w$.

We derive flow resistance in the $\mathscr{w}$-frame, as the flight velocity $\vec{v}^w$ is also specified with respect to the world coordinate system. We get:

$$\overrightarrow{F_{C_w}}^w = \frac{1}{2} * \rho * C_w * A * \overrightarrow{v^2}^w. \tag{5}$$

Subsequently, the torque of the flight platform should be derived. According to Euler, a torque $\vec{M}$ can be expressed with respect to the world coordinate system ($\mathscr{w}$-frame) by the time derivative of the angular momentum $\frac{d\vec{L}}{dt}$ as

$$\vec{M}^w = \frac{d\vec{L}^w}{dt}. \tag{6}$$

Transforming this into the body-fixed coordinate system we get:

$$\vec{M}^b = \vec{\omega}^b{}_F \times \vec{L}^b + \dot{\vec{L}}^b. \tag{7}$$

Then we determine the five different angular moments of the system. The first four angular moments caused by the motor / rotor combinations are given as:

$$\vec{L}^b{}_R = \boldsymbol{I^b}_{\boldsymbol{R}} * \vec{\omega}^b{}_R. \tag{8}$$

In this case the inertia tensor is shown by $\boldsymbol{I^b}_{\boldsymbol{R}}$, the angular velocity of the rotor by $\vec{\omega}^b{}_R$. Each with respect to the b-frames. Similarly, the angular momentum of the air platform without the rotor/motor combinations is described by

$$\vec{L}^b{}_F = \boldsymbol{I^b}_{\boldsymbol{F}} * \vec{\omega}^b{}_F \,, \tag{9}$$

where $\boldsymbol{I^b}_{\boldsymbol{F}}$ represent the inertia tensor and $\vec{\omega}^b{}_F$ is the angular velocity of the flight platform. It follows a total angular momentum $\vec{L}^b$ of

$$\vec{L}^b = \left(\boldsymbol{I^b}_{\boldsymbol{F}} + 4 * \boldsymbol{I^b}_{\boldsymbol{R}}\right) * \vec{\omega}^b{}_F + \textstyle\sum_{i=1}^{4} \boldsymbol{I^b}_{\boldsymbol{R}} * \vec{\omega}^b{}_{R,i} \,. \tag{10}$$

In conjunction with equation (7) we get the total torque of the flight platform as

$$\begin{aligned}\vec{M}^b = {} & \vec{\omega}_F^b \times \left[\left(\boldsymbol{I^b}_{\boldsymbol{F}} + 4 * \boldsymbol{I^b}_{\boldsymbol{R}}\right) * \vec{\omega}^b{}_F + \textstyle\sum_{i=1}^{4} \boldsymbol{I^b}_{\boldsymbol{R}} * \vec{\omega}^b{}_{R,i}\right] \\ & + \left(\boldsymbol{I^b}_{\boldsymbol{F}} + 4 * \boldsymbol{I^b}_{\boldsymbol{R}}\right) * \dot{\vec{\omega}}^b{}_F + \textstyle\sum_{i=1}^{4} \boldsymbol{I^b}_{\boldsymbol{R}} * \dot{\vec{\omega}}^b{}_{R,i} \,.\end{aligned} \tag{11}$$

The inertia tensor $\boldsymbol{I^b}_{\boldsymbol{R}}$ of the rotors can be neglected, thus, equation (11) simplifies to

$$\vec{M}^b \approx \vec{\omega}_F^b \times \boldsymbol{I^b}_{\boldsymbol{F}} * \vec{\omega}_F^b + \boldsymbol{I^b}_{\boldsymbol{F}} * \dot{\vec{\omega}}_F^b \,. \tag{12}$$

This derivation of the mathematical relationships of forces and moments is based on the representation in [WENDEL 2007], which we refer to for a more detailed examination.

Model of the motor/rotor combination

In the simulation, the components of the thrust forces F_i in the z^b-direction of the rotors are approximated by a quadratic function:

$$F_i = \alpha * {n_i}^2 + \beta * n_i \,. \tag{13}$$

The coefficients α and β are determined from a function of the metrologically acquired characteristic of thrust F_i and rotating speed n_i of the rotors.
Analogously, the torques M_i are represented by a linear function:

$$M_i = \gamma * n_i \,. \tag{14}$$

The coefficient γ of the function is also determined from the metrologically acquired characteristic of torque M_i and rotating speed n_i of the rotors. As the real measured characteristics are used, we achieve a precise modeling in the simulation.

4. SIMULATION OF THE QUADROCOPTER

In this section, the real quadrocopter AR100B is compared with its simulated counterpart. We have developed two scenarios which we present graphically. In the first scenario we examine on the basis of the attitude estimation and control how well the mathematical model is compliant with the physical reality. In the second use case, the position estimation and control as well as a programmed waypoint navigation is investigated. For all of the following scenarios we have modeled and faithfully reproduced not only the flight platform but also the whole environment, i.e. indoor flying area and outdoor area in order to create a realistic visual feedback.

Estimate and control of attitude

In this application, the flight platform is simulated in the indoor flying area of AirRobot at Arnsberg. The aim is to analyze and compare the behavior of the simulated and the real flight platform without external influences such as wind. Fig. 4 shows the flight platform in the indoor flying area. The UAV is brought from the rest position (first scene) by a fixed pulse duration to the maximum roll angle position (2nd and 3rd scene). Then the system should automatically return (4th and 5th scene) to the starting position. Shots of the real system are presented in the upper strip; corresponding results of the simulation are shown in the lower strip. It can be seen that the behavior of the real flight platform is modeled very accurately by the simulation. The described experiments show that basically the developed mathematical model and the approximations made in Section 3.2 are qualitatively correct.

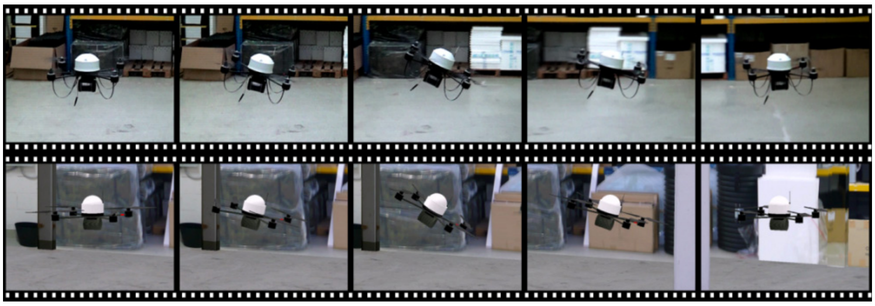


Figure 4: Graphical comparison of the attitude estimation and control of the real (top, generated from video recording) and the simulated flight platform (below, rendered in the simulation). (from left to right: 1. initial position, 2. command pulse → maximum roll angle, 3. maximum roll angle reached, 4. independent return to initial position, 5. starting position reached).

Estimate and control of position and waypoint navigation

In this scenario, the micro-UAV is analyzed in a natural environment. Objective in this use case is the evaluation of the position hold functionality of the flight platform using simulated Global Positioning System (GPS) (Fig. 5 upper strip). The flight platform is deflected from its rest position with simulated crosswind heading north with 6 m/s (*1*st *and* 2nd *image*). After activating position hold, the system slows down (3rd *image*) and produces a counteracting force by adjusting the roll angle towards the wind direction (*Scene 4*). So the flight platform keeps its position at wind speeds up to 8 m/s. After wind abates, the system continues to hold the position by returning to the idle position (5th image). An extension of this use case is waypoint navigation, where GPS coordinates can be specified, which will then be served by the flight platform. In the lower strip we show as an example the planning of waypoint navigation (left), the trajectory flown in reality (middle) and in the simulation (right).

This comparison shows qualitatively, that the simulation of external influences such as wind is effective, and the trajectory flown by the simulated Waypoint navigation approximates the real trajectory. The remarkable difference in the center of the navigation path is due to a temporary discontinuity in the real GPS signal that is not captured in the model.

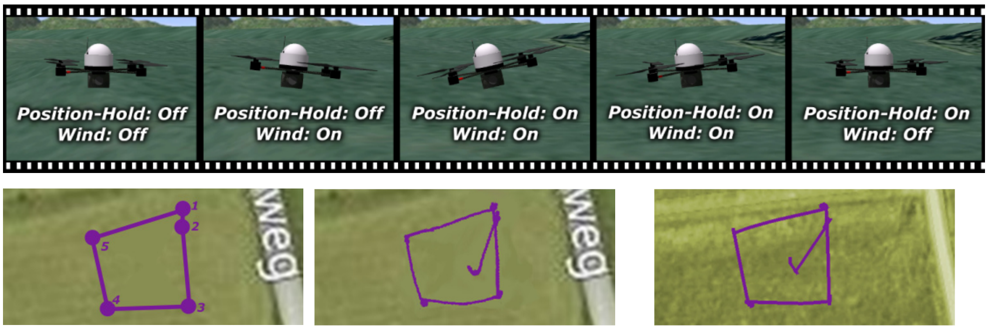


Figure 5: Graphical representation of the simulated GPS position hold (up) under influence of crosswind and the Waypoint navigation (bottom) of the flight platform.

5. SYSTEM DESIGN OF A HEXACOPTER

In the previous section it has been shown that the simulation of the quadrocopter AR100B could be successfully implemented. Here we demonstrate the design of a new flight platform using the simulation tool. This flight platform, based on the technology of the AR100B, should have a higher payload capacity with at least equivalent flight characteristics. For this purpose, a micro-UAV with six motors was designed.

In the following this new hexacopter will be called AR100-X6. The first step was the creation of the geometry of the system in the simulation. Based on this all system parameters, such as masses, forces and torques, etc. derived in analogy to Section 3 and the appropriate mathematical model is created. After successful implementation of these basic model parameters, the platform-specific filter adaption and model-based attitude- and position controller was designed. These algorithms were implemented using the SiL interface (Fig. 1, green box). Equivalent to the procedures in the previous section, the behavior of the model were investigated and the control algorithms were optimized. Only after a successful optimization phase and various reliability tests the first prototype was built. Due to the consistent implementation of the SiL idea in the simulation, the developed algorithms could be ported directly and without any changes to the real embedded system. Again we compare the simulated platform to the real prototype in five real (Fig. 6, upper strip) and simulated (lower strip) images.

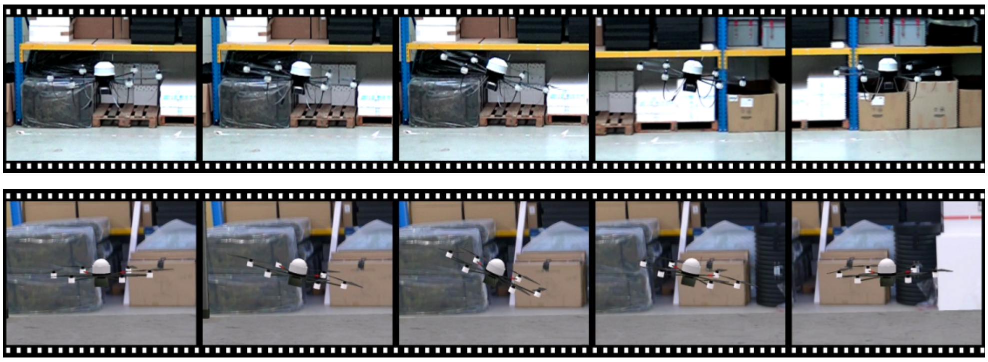


Figure 6: Graphical comparison of the real hexacopter prototype AR100-X6 (top) and simulated (bottom).

With our prototype we could demonstrate the system design purely based on simulation. So far, all test flights of the new platform showed the accuracy of the simulation and could be completed successfully and without loss of the flight platform.

6. OTHER MODELS DEVELOPED IN THE SIMULATION

As shown in section 5, this simulation can be used for the development of new platforms. In the following, two further models are shown. As the AR100-X6, presented in section 5, the AR100C and the AR200, shown in Figure 7-1a and -2a) have also been developed purely on the basis of this simulation. Again, the platform-specific filter adaption and model-based attitude- and position controller design of these new platforms have been performed completely in the simulation and have been ported to the physical

(1a) AR100C Simulated

(2a) AR200 Simulated

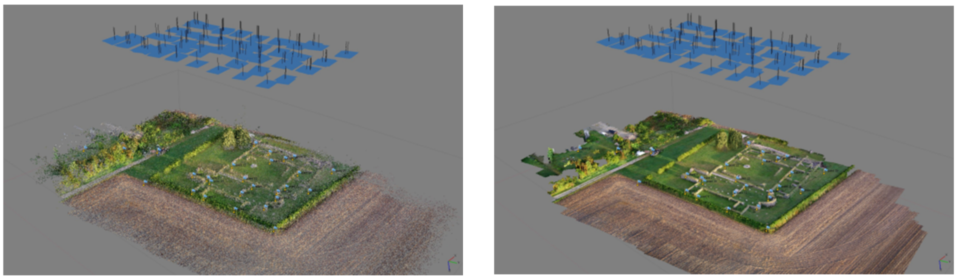
(3a) AR.Drone 2.0 Simulated

(1b) AR100C Reality

(2b) AR200 Reality

(3b) AR.Drone 2.0 Reality

Figure 7: Models developed in the Simulation and transferred to Reality.



(a) 3D-reconstruction (point-cloud) (b) 3D-reconstruction (rendered)

Figure 8: 3D-reconstruction of an archeological site captured with the newly developed AR200 (images courtesy of AirRotorMedia GmbH, 2013)

prototypes (shown in Figure 7-1b and -2b) only after a successful optimization phase and various reliability tests. The most interesting part of these new platforms is the filter adaption and controller design of the AR200. This new hexacopter with a diameter of 200cm has a payload capacity of 3000g and an overall weight of 10.000g. For such heavy VTOL an innovative and advanced controlling strategy was essential and it could be shown that the simulation based approach worked perfectly. Fig. 8 shows a 3D-reconstruction of an archeological site, acquired with the AR200 hexacopter by a commercial company. Currently, the simulation has been extended to the well-known AR.Drone 2.0 [AR.Drone 2013] produced by the French company Parrot SA® (shown in Figure 7-3b). With this pioneering implementation of a Low-Cost UAV in this simulation (shown in Figure 7-3a) it will be much easier to develop new algorithms for autonomous flight and new controlling schemes. It is now possible to fly this drone with self-built algorithms which can be completely tested and evaluated in the virtual reality and only after this phase be ported to this closed source consumer market product.

7. MEASURING THE WORLD – THE ANALYSIS-BY-SYNTHESIS WAY

The simulation tool described above can be regarded conceptually as a part of a larger framework, a framework for measuring by simulation. Measuring by simulation or analysis-by-synthesis (a-b-s) is a method, where a model is optimized in such a way that "simulated" measurements optimally fit the real measurements. The principle has been known for a long time; network adjustment in surveying or bundle adjustment in photogrammetry are examples for such methods. Taking this idea further, even brightness values in images can serve as measurements. The simulation can then not just be expressed as a simple equation, but computer graphics does a good job in creating simulated images from 3D scene descriptions and camera parameters. Using modern graphics hardware even lens distortions can be simulated with high speed. Speed is a crucial factor when it comes to numerical optimization. As literally millions of measurements contribute to the final result, even with cheap cameras very precise measurements can be made. The basic principle is shown in Fig. 9.

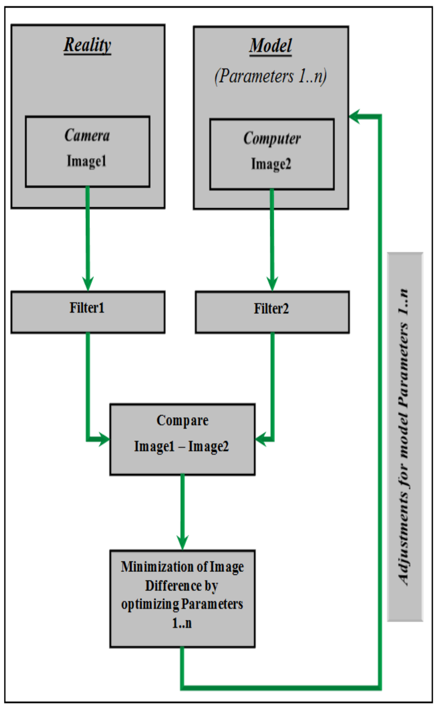


Figure 9: Analysis-by-Synthesis with image, basic principle.

The model is adjusted to the real world by changing its parameters in an iterative way, until the rendering of the model is similar to the photographs of the real scene. Filters can be used to get rid of information that is not important for the problem being solved, e.g. material properties, colors, etc. An example would be a filter that only

Figure 10: Analysis-by-synthesis texture reconstruction. Top: one of 49 input images of nut data set (left), reconstructed texture. Bottom: three of 48 input images (79x59 pixels) of Uni Tübingen dataset (left), reconstructed texture (right). Images from [LIEFERS, 2012].

preserves edges, if the brightness values are not important or cannot be calculated precisely. In order for the a-b-s principle to work well, some preconditions must be met:

1. The model must capture well, what we know a-priori about the scene. This means that known constraints should be included in the formulation of the model, resulting in a low number of parameters that have to be determined (as few as possible).
2. We need an efficient optimization algorithm. As the similarity function often has many local minima, it is important that the virtual model is already close to reality, i.e. a good estimate for the model parameters is needed as a starting point for an iterative numeric optimization.

In Fig. 10 we show an example of how such an analysis by synthesis algorithm can be used for determining textures from a series of photographs, where each of the photographs only contains an extremely oblique view of the surface with the texture [LIEFERS, 2012]. Graphical rendering employing modern hardware is used in this example for the synthesis part of the algorithm. Despite the high number of model parameters (each of the texels' color values) the problem can be solved here due to a well-fitting optimization algorithm. Other examples include increasing the precision of measurements of buildings by using shadows, or finding out the direction of lighting in historic photographs.

The UAV simulation tool fits well into this framework. The model of reality now includes the micro-UAV with its sensors, including cameras, but also including sensors for acceleration, barometric pressure, magnetic field of the earth and last not least the rotation speed of the micro-UAV's rotors (Fig. 11, compare with Fig. 9). The simulation incorporates our prior knowledge about the system into the formulation of the model. Instead of optimizing camera positions and orientations for each frame, the optimization adjusts a small number of control-parameters and parameters for external influences

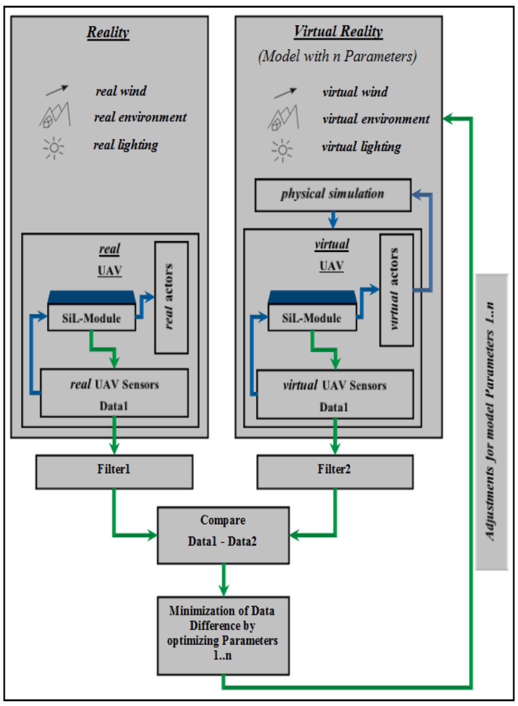


Figure 11: Analysis-by-Synthesis, model enhanced with physical simulation of acquisition platform.

(e.g. wind). In this way, only physically plausible trajectories can be produced, which enforces known constraints on the camera positions. Given the control commands of the micro-UAV, the simulation also provides a good starting point for the optimization of the trajectory. Furthermore, ambiguities that are difficult to resolve otherwise like the ones between viewing direction and translation are resolved in a natural way. Although classical sensor integration of acceleration and rotation sensors could also be used in this case, the inclusion of the simulation into the a-b-s loop not only performs this sensor integration in a very straightforward way, but also filters noisy sensor output by including knowledge about the physical properties of the platform.

A practical example application of the system would be the measurement of wind velocity and direction at arbitrary positions in space, important e.g. for predicting propagation of smoke clouds etc. External forces caused by wind are determined independent of flight maneuvers by optimizing the relevant model parameters in such a way, that virtual sensor output is similar to the measured sensor output.

8. SUMMARY AND OUTLOOK

This paper describes a model-based 3D real-time simulation of a micro-UAV. With this tool, the reliability of control algorithms could be tested under a variety of conditions. The visual feedback of the simulation together with an analysis tool [LINKUGEL et al. 2013] make software validation in an early stage of the development feasible. Algorithms for sensor filtering and control can be evaluated and optimized in real time under repeatable and configurable conditions. Furthermore it was successfully demonstrated that complete systems can be designed and tested exclusively within this simulation resulting in airworthy prototypes. This has high relevance, because the development time of prototypes is significantly shortened; resulting in lower costs and early consideration of safety-critical factors.

In the future we want to employ the model-based simulation in analysis-by-synthesis type measurement settings. It will serve as a convenient development platform for control algorithms, data fusion and filtering and a-b-s type measuring and modeling algorithms, as well as algorithms for exploration and acquisition planning, but the simulation can also be integrated into the a-b-s loop itself. Simulation is a way to include prior knowledge into the a-b-s model and can therefore improve the precision of measurements and model building. In the near future the system will be extended to support an image processing interface for the SiL component. This will allow for fast implementation and evaluation of algorithms for visual position and motion estimation, as described in [LINKUGEL ET. AL. 2010], which then can be ported directly to real systems.

9. REFERENCES

AR.Drone: AR.Drone 2.0 is a wi-fi controlled flying quadcopter built by the French company Parrot. Website: http://ardrone2.parrot.com/, accessed on 10.07.2013.

Grzonka, S., Grisetti, G., & Burgard, W.: Towards a navigation system for autonomous indoor flying. International Conference on Robotics and Automation (ICRA), 2009, pp. 2878-2883.

Guenard, N., Hamel, T., & Moreau, V.: Dynamic modeling and intuitive control strategy for an "X4-flyer". International Conference on Control and Automation (ICCA), 2005.

Liefers, F., Parys, R., & Schilling, A.: Analysis-by-Synthesis Texture Reconstruction. Proceedings of the Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012, pp. 571-578, IEEE Oct. 2012.

Linkugel, T., & Bobey, K.: Visuelle Positions- und Bewegungsbestimmung eines autonomen systems. Photogrammetrie, Laserscanning und Optische 3D-Messtechnik in Beiträge der Oldenburger 3D-Tage, 2010.

Linkugel, T., Schilling, A., & Mallot, H. (2013): Modellbasierte 3D-Echtzeit-Simulation von micro-UAS. avn. – Allgemeine Vermessungs-Nachrichten, 03/2013 (Fachbeiträge begutachtet), pp. 90-97.

Mellinger, D., Shomin, M., & Kumar, V.: Control of Quadrotors for Robust Perching and Landing. Proceedings of the International Powered Lift Conference, 2010.

Michel, O.: Webots: Professional Mobile Robot Simulation. International Journal of Advanced Robotic Systems, Vol. 1, Num. 1, 2004, pp. 39-42.

Morris, W., Dryanovski, I., & Jizhong, X.: 3D Indoor Mapping for Micro-UAVs Using Hybrid Range Finders and Multi-Volume Occupancy Grids. Science and Systems Conference (RSS), 2010.

OGRE: OGRE (Object-Oriented Graphics Rendering Engine) is a scene-oriented, flexible 3D engine written in C++. Website: http://www.ogre3d.org, accessed on 10.07.2013.

ODE: ODE is an open source, high performance library for simulating rigid body dynamics. Website: http://www.ode.org, accessed on 10.07.2013.

Titterton, D., & Weston, J.: Strapdown Inertial Navigation Technology. IET Radar, Sonar, Navigation and Avionics, Series 17, 1996.

Vicon, GRASP LAB: University of Pennsylvania uses Vicon Motion Capture to Achieve Remarkable Maneuvers for autonomous Quadrotor Helicopter. Flight Engineering Case Study, 2010.

Wendel, J.: Integrierte Navigationssysteme. Oldenbourg Verlag, München/Wien, 2007, pp. 292-296.