esa
GNC 2017
29 May - 2 June 2017, Salzburg, Austria
European Space Agency

COORDINATED ORBIT AND ATTITUDE CONTROL OF A SATELLITE FORMATION IN A SATELLITE SIMULATOR TESTBED

S. Wehrmann[1], M. Schlotterer[1]

*[1]German Aerospace Center, Institute of Space Systems, Robert-Hooke-Str. 7
28359 Bremen, Germany, +49-421-244201118, markus.schlotterer@dlr.de*

ABSTRACT

This paper presents coordinated orbit and attitude control of a satellite formation on an air bearing table. The investigated application is the use of the formation as a telescope. The coordination and exact control of the relative position and attitude of the satellites towards each other is a challenging formation flying problem. The satellite relative dynamics are investigated in a hardware-in-the-loop environment.

For the validation of the formation control strategy, the Test Environment for Applications of Multiple Spacecraft (TEAMS), a test facility for satellite formations and swarms based on air cushion vehicles, at the Institute of Space Systems of the DLR in Bremen, Germany, is used. Two 5-DoF vehicles are floating on a granite table with a total experiment area of 5m x 4m. Each vehicle has a thruster system, a reaction wheel system and its own onboard computer running the control algorithms.

A laser, attached to the first vehicle, is pointed at a fixed target on the other vehicle to demonstrate the telescope application. The Clohessy-Wiltshire equations are used for the guidance and feedforward control of the satellites position in orbit. A second controller is used to calculate the correct pointing angles for the laser and the target. The control law is based on the relative distance and velocity of the two satellites. To achieve the optimal control of the linear system, LQR controllers are used together with feedforward for optimal tracking and to cover the nonlinearities. The characteristics of the facility and its devices are introduced, the control algorithms for both satellites are explained and simulation and HIL test results are presented.

1. INTRODUCTION

In many space applications, modern approaches show a trend of moving away from the classical single satellite missions, where one satellite includes a complete set of sensors and instruments, towards fractionated and distributed sensor missions. There multiple satellites are carrying different types of sensors or different elements of a larger single instrument and act in a formation. Such missions promise an increase of imaging quality, an increase of service quality and in many cases a decrease of deployment costs for satellites that can become smaller and less complex due to mission requirements.

Satellite formation flying is an enabling technology for this kind of missions. It allows multiple satellites to cooperate and act in some sense as a single physically larger spacecraft. A potential application for the latter case is forming a telescope from at least two spacecraft. This requires a tight control of both attitude and position - denoted as coordinated pointing - to ensure that the instrument can operate as expected. Developing and testing of this control system is a challenging task. For first steps simulations and later on-ground physical demonstrators can be used.

This paper focuses on the realization of the coordinated attitude and position control for a formation of two satellites using the ground-based Test Environment for Applications of Multiple Spacecraft (TEAMS) within a Master thesis project. In section 2 the test set-up is introduced and special characteristics are provided. Section 3 describes the controller structure and general controller design while in section 4 the guidance laws for attitude and position control are presented. Finally, section 5 shows simulation and experiment results for the coordinated attitude and position control.

2. THE TEST ENVIRONMENT FOR APPLICATIONS OF MULTIPLE SPACECRAFT (TEAMS)

The Test Environment for Applications of Multiple Spacecraft (TEAMS) at the Institute of Space Systems of DLR (German Aerospace Center) has the objective to emulate the force- and torque-free dynamics of in-orbit satellites. The satellites are represented by air cushion vehicles floating on a granite table with a total experiment area of 4m by 5m. Two types of air cushion vehicles represent the spacecraft. The two bigger ones called TEAMS_5D have an actuated linear stage to simulate vertical movement and a rotatable upper platform to simulate attitude dynamics (attitude platform). These vehicles can emulate five degrees of freedom. The four smaller vehicles are called TEAMS_3D and are able to emulate three degrees of freedom. They are mainly used for swarm simulations. Furthermore, the facility is equipped with an infrared tracking system which provides position and attitude of the vehicles at an update rate of 60 Hz.

For the test described in this paper the two TEAMS_5D vehicles are used. The following section provides more information about the vehicle and its properties.

2.1. TEAMS 5 DoF Vehicle

The vehicles used for experiments described in this paper have five degrees of freedom (TEAMS_5D). They consist of two different platforms: The lower platform glides over the granite table on air cushions, while the upper platform sits on top of a spherical air bearing, which is connected to the lower platform. In the following chapters the upper platform will be named as ”attitude platform” and the lower platform will be named as ”position platform”.

Due to the air cushions and the spherical air bearing, each vehicle has five degrees of freedom (two translational and three rotational). Both platforms have two independent storage systems for compressed air. The system on the position platform supplies air for the spherical air bearing and the air pads. To generate torques and forces on the vehicle, the storage system on the attitude platform

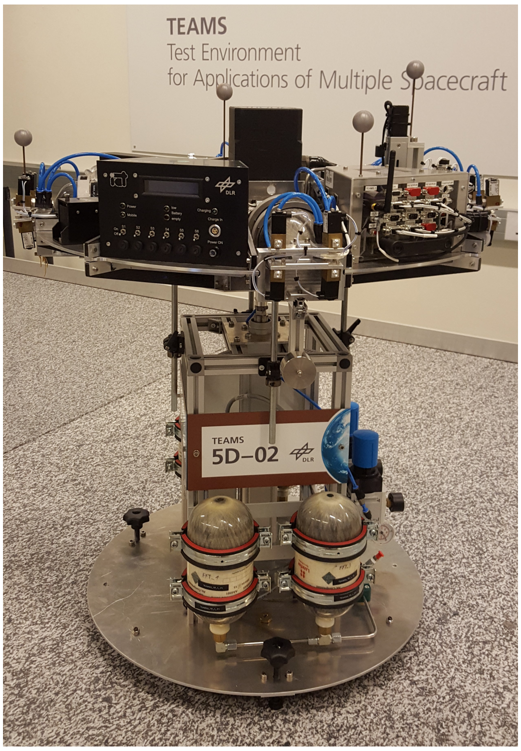

Figure 1: *TEAMS_5D vehicle*

supplies air as propellant gas for 16 cold gas thrusters. There are also three reaction wheels for torque actuation and a balancing system, where small weights are moved by step motors to place the center of mass directly in the center of rotation of the spherical air bearing. An on-board-computer, an IMU, a battery and a power supply system are mounted on top of the attitude platform, too. The on-board computer executes the control algorithms in real time. The different models for the controllers can be developed in the Matlab/Simulink environment. The Matlab/Simulink coder generates C code from the models, which can be compiled and uploaded to the on-board-computer. During the tests the external mode of Matlab/Simulink is used to receive data from the model running on the on-board computer. Infrared targets mounted on the attitude platform are tracked by the laboratory's position and attitude tracking system.

2.2. Thrust System

Each of the TEAMS_5D vehicles carries 16 proportional thrusters on the attitude platform. For creating the forces and torques required by the controller their transformation into individual thrust commands is determined by a real time convex optimizer. Let $\bar{T}$ be the thruster vector containing the individual thrust commands of all 16 thrusters. Each thrust value in the vector is expressed as a ratio of the maximum thrust. Thus each element must be between 0 and 1. Furthermore $\boldsymbol{A}$ is defined as the thruster mapping matrix representing the geometry of the thruster system including thrust direction and lever arms. The augmented force/torque vector $\bar{F} = [\vec{F}, \vec{T}]$ can then be computed as

$$\bar{F} = \boldsymbol{A}\bar{T}. \tag{1}$$

The calculation of the thruster vector $\bar{T}$ for a given $\bar{F}$ can be expressed as a convex optimization problem:

$$\min \sum \bar{T} + \lambda \|\boldsymbol{A}\bar{T} - \bar{F}\|_1 \tag{2}$$

subject to

$$\bar{T} \geq 0 \tag{3}$$
$$\bar{T} \leq 1 \tag{4}$$
$$\boldsymbol{C}\boldsymbol{A}\bar{T} = 0 \tag{5}$$

and

$$\boldsymbol{C} = \begin{bmatrix} -F_2 & F_1 & 0 & 0 & 0 & 0 \\ -F_3 & 0 & F_1 & 0 & 0 & 0 \\ \vdots & 0 & \cdots & \ddots & \ddots & \ddots \\ 0 & -F_3 & F_2 & 0 & 0 & 0 \\ 0 & -F_4 & 0 & F_2 & 0 & 0 \\ \vdots & \vdots & 0 & \cdots & \ddots & \ddots \end{bmatrix} \tag{6}$$

The optimizer searches for the minimum of the sum of all elements of T and thus minimizes the air consumption of the thruster system. The constraint (5) guarantees that $\boldsymbol{A}\bar{T} \sim \bar{F}$, even if the thruster system can not fulfill the commanded force/torque. In that case, the second term of the cost function causes $\boldsymbol{A}\bar{T}$ to be as near to $\bar{F}$ as possible, if λ is large enough, while still minimizing $\sum \bar{T}$. If the thruster system can fullfil the commanded force/torque, the second term of the cost function can be minimized to 0 and the optimizer only minimizes the air consumption.

To solve the optimization problem on the onboard computer and in realtime the code generator for convex optimization CVXGEN [3] has been used.

2.3. System Model

For designing the control law and for simulating the motion of the vehicles, the system dynamics have to be modeled. The vehicle on the table can be modeled as a single rigid body in a first order approach. If a force (external or generated by the thrusters) affects the vehicle the displacement follows Newton's second law:

$$\vec{F} = m \cdot \vec{a}. \tag{7}$$

The translational motion can be modeled by a double integrator:

$$\begin{bmatrix} \dot{\vec{s}} \\ \dot{\vec{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \vec{s} \\ \vec{v} \end{bmatrix} + \begin{bmatrix} \vec{0} \\ \vec{a} \end{bmatrix} . \tag{8}$$

The rotational movement is caused by torques which act on the vehicle. To describe the relations between the acting torques and the angular velocity of the attitude platform, Euler's equation of a momentum-biased rigid body can be used:

$$\vec{T} - \vec{T_w} = \boldsymbol{I}\dot{\vec{\omega}} + \vec{\omega} \times (\boldsymbol{I}\vec{\omega} + \vec{h_w}), \tag{9}$$

where $\vec{T}$ is the external torque, $\vec{T_w}$ the commanded reaction wheels' torque, $\vec{h_w}$ is the stored momentum and $\boldsymbol{I}$ is the inertia tensor of the attitude platform.

The differential equation for the quaternion representing the attitude is given by [6]:

$$\dot{q} = \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} q \tag{10}$$

Due to the fact, that the thrusters and the reaction wheels have special characteristics and limitations, they have to be modeled as well. These models include the saturation and the response time of both actuators.

2.4. Parameter identification

In order to achieve an accurate control a high level of accuracy is required for the determination of parameters such as the magnitude and direction of thruster forces and the platform's inertia tensor. Procedures were devised to identify inaccuracies in the manufacturing processes of the platform's components and assembly process. The algorithms for the identification are explained in [2].

For the inertia tensor identification an open-loop manoeuvre is executed on one axis by means of its reaction wheel; this is followed by a closed-loop return to the initial attitude. The procedure is repeated on each axis. Using equation (9) the inertia tensor can be calculated.

For identification, the thruster to be identified is set to the maximum thrust for a fixed time while all other thrusters are kept turned off. In this procedure the reaction wheels are used in a closed control loop (PI velocity regulator) to keep a static attitude under thruster action. In steady state the thruster torque is equal to the the control torque created by the wheels.

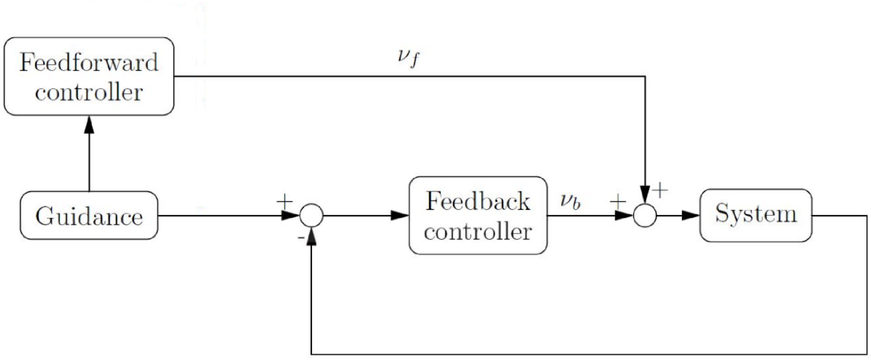

Figure 2: Feedback controller with feedforward [2]

3. CONTROL ALGORITHMS

For position and pointing a control system was designed. The overall structure for both controllers (position and attitude) is a feedback controller together with feedforward as shown in figure 3.

The feedforward control imposes the predicted forces and torques on the plant, which are needed to follow the trajectory given by the guidance. In an undisturbed ideal case the system then follows the trajectory provided by the guidance with a zero control error such that the feedback control loop has no effect. Nonlinearities of the system are included in the guidance such that the feedforward control brings the system in a state which has only small deviations from the required state given by the guidance. With these small deviation it is possible to design a linear feedback controller.

3.1. Control laws

For the attitude control we use an approach, which was discussed in [6]. The attitude control law is given by:

$$\vec{T} = 2k_p q_{e4} \begin{bmatrix} q_{e1} \\ q_{e2} \\ q_{e3} \end{bmatrix} + k_d \vec{\omega}. \tag{11}$$

This control law uses directly unit quaternions to calculate an output torque for the system. It has the structure of a general PD controller, where the error quaternion is used for the proportional feedback part, and the error of the angular velocity for the differential part. One disadvantage of this law is, that it is not globally asymptotically stable. A problematic case occurs if the rotation angle of the error quaternion is exactly π which correlates to $q_{e4} = 0$. But due to the feedforward control, it can guaranteed, that there are only control errors much smaller than π. A big advantage is, that there is no unwinding in contrast to other quaternion control laws with better performance but with unwinding.

The structure of a PD controller can also be used to generate a force to control the position:

$$\vec{f} = k_p\vec{\Delta s} + k_d\vec{\Delta v}, \tag{12}$$

where $\vec{\Delta s}$ is the position error and $\vec{\Delta v}$ is the velocity error. In the next step, the controller gains k_p and k_d for both controllers are calculated. One method to calculate these gains is the linear quadratic regulator (LQR) using the linearized model. For the attitude controller we get:

$$k_p = \sqrt{\frac{q_2}{R}},\ k_d = \sqrt{2\boldsymbol{I}\sqrt{\frac{q_2}{R}} + \frac{q_1}{R}}, \tag{13}$$

where $\boldsymbol{I}$ is the Moment of Inertia in the corresponding axis.

The position controller was designed analog. The only difference is, that in place of rotational inertia $\boldsymbol{I}$, it is necessary to use linear inertia m (mass), albeit different weights were used:

$$k_p = \sqrt{\frac{q_2}{R}},\ k_d = \sqrt{2m\sqrt{\frac{q_2}{R}} + \frac{q_1}{R}}. \tag{14}$$

q_1 and q_2 are the elements of the LQR matrix Q, which weights the control errors in the optimization process.

$$Q = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix} \tag{15}$$

R is also an element of the LQR algorithm to weight the control inputs.

3.2. Simulation of the controllers

Before applying the controller to the real system (demonstrator) a simulation is used for analysis and design. After the design phase it shall show the performance of the system. In a first step a static position is controlled. With a set point for the target position which is different to the initial state the simulations show a step response. They have been used to pick the proper values for the weighting matrices Q and R of the Linear Quadratic Regulator (LQR) design process. For simulations with a constant disturbance - as expected - a steady state error occurs. To overcome this effect a small integral term has been introduced. The gain was chosen as small as possible to improve the steady state accuracy but not compromise the stability of the system. The two plots in figure 3 show the position step response of a controller without the integral term (left) and a controller with the integral term (right). As it can be seen the steady-state control error is eliminated after about 400 s.

If we analyze the expected disturbances of about $F_{ext} = 0.01\,\mathrm{N}$ and use equation (12) to estimate the constant control error without the integral term in the controller we get for the position error 0.01 m. The large position error is observable and justifies the introduction of an integral part in the controller.

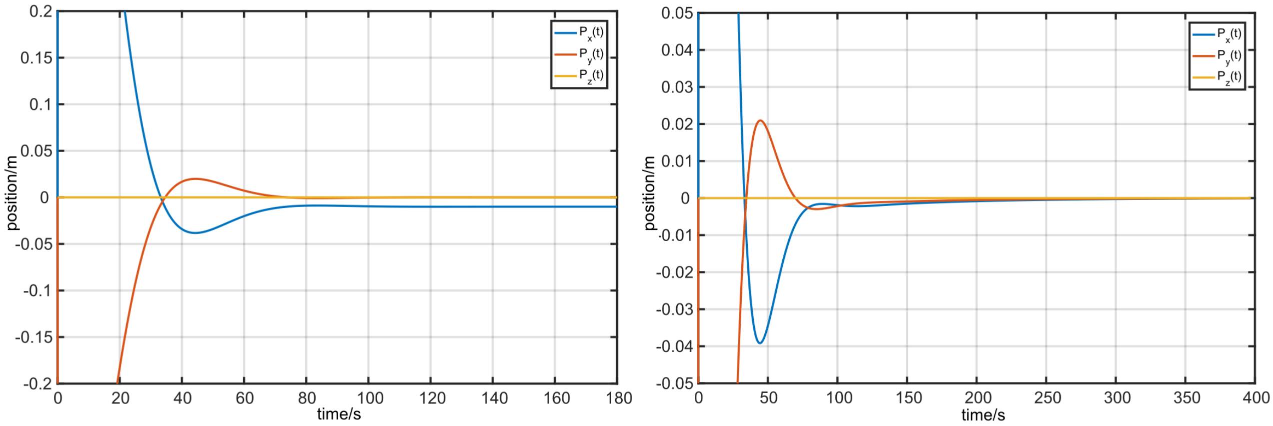

Figure 3: *Position step response with external force and without integral term (left); Position step response with external force and with integral term (right);*

For the attitude the error angle is not observable assuming the expected constant disturbance torque. Thus the attitude controller does not need an integral term.

4. TRAJECTORY CONTROL

4.1. Position guidance for feedforward controller

For simulating a formation with an elliptical relative orbit on the testbed we have to introduce a trajectory controller and a guidance for each of the two vehicles. Both trajectories shall be based on a simulated relative orbit. For that purpose the Clohessy-Wiltshire equations are used (see equations (16) to (18)).

$$\ddot{x} - 2n\dot{y} - 3n^2x = u_x \tag{16}$$

$$\ddot{y} + 2n\dot{x} = u_y \tag{17}$$

$$\ddot{z} + n^2z = u_z \tag{18}$$

They describe the simplified relative motion between two satellites (of a satellite and a reference point at the origin of the frame) where x, y, z are the position coordinates in the local frame (x - radial direction, z - along the orbit's angular momentum vector) and u_x, u_y, u_z are external forces. Based on the knowledge of the state, the orbital frequency and the assumption that no external forces act on the spacecraft ($u_i = 0$) the Clohessy-Wiltshire equations can be used to compute the acceleration for the guidance and the feedforward control. The initial conditions are chosen to form a closed ellipse for both vehicles around a reference.

4.2. Attitude guidance for feedforward controller

In order to enable pointing during relative motion a guidance and feedforward function is also needed for the attitude. For each relative position of the two satellites, the attitude guidance has to calculate a quaternion for both satellites. It is assumed that the relative state of both vehicles is well known at every time.

In [2] an algorithm to point a laser in a fixed target was presented. This algorithm was picked up and enhanced to point a laser into a moving target on another satellite.

To calculate the correct attitude the closest approach of the laser ray line to the origin $\vec{\delta}^b$ and the direction of the laser ray $\vec{\rho}^b$ have to be determined. Both vectors are shown in figure 4. $\vec{\delta}^b$ and $\vec{\rho}^b$ are mutually orthogonal by construction. The circle in the figure illustrates the attitude platform. $\vec{r}_p^b$ presents the direct pointing line between the attitude platform and the target. The algorithm in [2] calculates the attitude φ by:

$$\varphi = \arcsin \frac{||\vec{\delta}^b||}{||\vec{r}_p^b||} \tag{19}$$

which lets the laser pointing directly into the target. In the illustration there is shown only one angle for better understanding. In the real model the algorithm calculates three angles for the attitude.

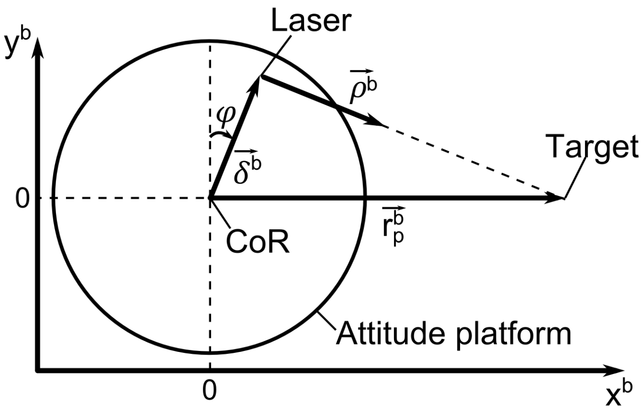

Figure 4: *Pointing correction of a laser ray to hit a fixed target*

The next step is to expand the algorithm for a movable target. The position of the target in the inertial frame has to be determined. As the position of the target on the target satellite and the attitude of the platform are known, we can calculate the exact position of the movable target depending on the attitude. The pointing vector to the target depends on the position of the target satellite in relation to the laser satellite. If the relative distance is large, the inclination of the attitude platforms has to be small to let the laser hit the target. If the relative distance is getting smaller, the inclination increases. The relation in shown in the following figure.

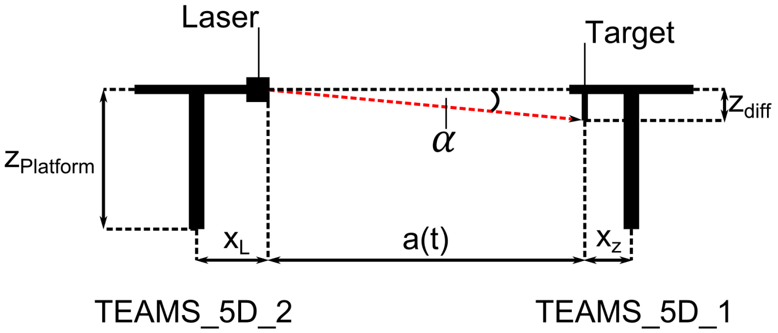

Figure 5: *Relations between the relative position and the inclination of the platforms*

To obtain the angular rate and acceleration for the attitude feedforward controller the first and second derivative of the attitude matrix $\mathbf{R}(t)$ is needed. For that the relative position vector $\mathbf{r}_P(t) = \mathbf{r}_T(t) - \mathbf{r}_C(t)$ can be written as an unitary vector multiplied by a scalar length [2]:

$$\mathbf{r}_P(t) = R(t)\begin{bmatrix} R_x(t) \\ R_y(t) \\ R_z(t) \end{bmatrix}. \tag{20}$$

The first and second derivatives of the scalar term are:

$$\begin{aligned}
R(t) &= \|\mathbf{r}_P(t)\| \\
\dot{R}(t) &= \frac{\mathbf{r}_{Px}(t)\,\dot{\mathbf{r}}_{Px}(t) + \mathbf{r}_{Py}(t)\,\dot{\mathbf{r}}_{Py}(t)}{\|\mathbf{r}_P(t)\|} \\
\ddot{R}(t) &= \frac{\mathbf{r}_{Px}(t)\,\ddot{\mathbf{r}}_{Px}(t) + \mathbf{r}_{Py}(t)\,\ddot{\mathbf{r}}_{Py}(t) + \dot{\mathbf{r}}^2_{Px}(t) + \dot{\mathbf{r}}^2_{Py}(t)}{\|\mathbf{r}_P(t)\|} - \\
&\quad - \frac{(\mathbf{r}_{Px}(t)\,\dot{\mathbf{r}}_{Px}(t) + \mathbf{r}_{Py}(t)\,\dot{\mathbf{r}}_{Py}(t))^2}{\|\mathbf{r}_P(t)\|^3}.
\end{aligned} \tag{21}$$

As the vehicles are moving on a horizontal plane, $\dot{\mathbf{r}}_{Pz}(t) = \ddot{\mathbf{r}}_{Pz}(t) = 0$. The derivatives of the first element of the unitary vector are:

$$\begin{aligned}
R_x(t) &= \frac{\mathbf{r}_{Px}(t)}{R(t)} \\
\dot{R}_x(t) &= \frac{\dot{\mathbf{r}}_{Px}(t) - R_x(t)\,\dot{R}(t)}{R(t)} \\
\ddot{R}_x(t) &= \frac{\ddot{\mathbf{r}}_{Px}(t) - 2\dot{R}_x(t)\,\dot{R}(t) - R_x(t)\,\ddot{R}(t)}{R(t)}.
\end{aligned} \tag{22}$$

The terms for R_y and R_z are similar to R_x. Given the matrices

$$\mathbf{\Omega} = \begin{bmatrix} 0 & \omega_{rz} & -\omega_{ry} \\ -\omega_{rz} & 0 & \omega_{rx} \\ \omega_{ry} & -\omega_{rx} & 0 \end{bmatrix} \qquad \dot{\mathbf{\Omega}} = \begin{bmatrix} 0 & \dot{\omega}_{rz} & -\dot{\omega}_{ry} \\ -\dot{\omega}_{rz} & 0 & \dot{\omega}_{rx} \\ \dot{\omega}_{ry} & -\dot{\omega}_{rx} & 0 \end{bmatrix} \tag{23}$$

the angular rate and angular acceleration can be computed from:

$$\mathbf{\Omega} = \dot{\mathbf{R}}(t)\,\mathbf{R}(t)^T \tag{24}$$

$$\dot{\mathbf{\Omega}} = \ddot{\mathbf{R}}(t)\,\mathbf{R}(t)^T - \left(\dot{\mathbf{R}}(t)\,\mathbf{R}(t)^T\right)^2 . \tag{25}$$

Using Euler's equation (9) one can also get the torques needed to keep the satellite on the reference attitude trajectory.

5. RESULTS

The results of the real tests of the controllers on the TEAMS_5D vehicles are compared with the simulation of a formation flight. Both satellites have the same relative orbit in relation to the reference point. But the starting points are on opposite positions on that orbit.

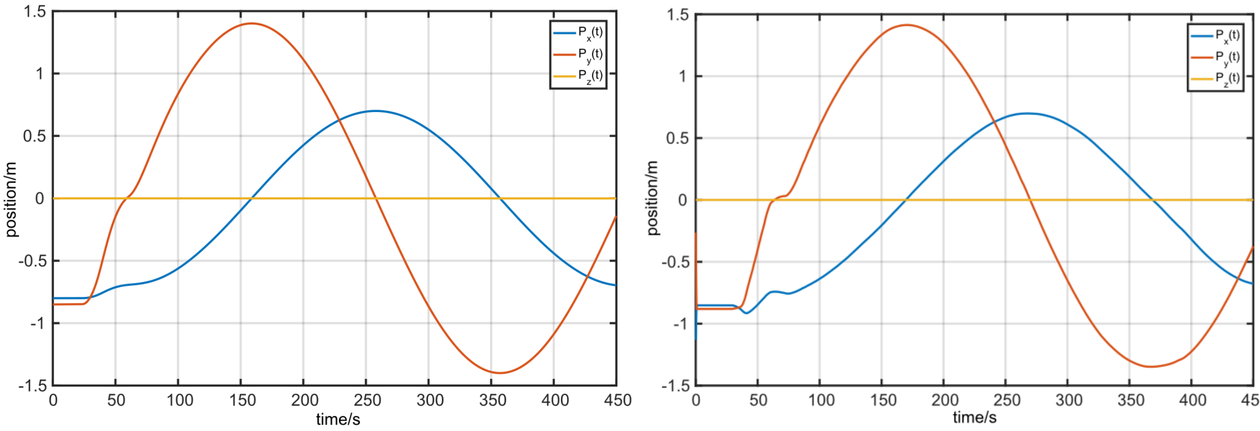

Figure 6: *Simulation results of the relative position for one satellite (left); Real results of the relative position for one satellite (right)*

Figure 6 shows the results for the position controller. During the first 70 s to 75 s, the satellite is in a starting mode. In that mode, the attitude platform is balanced (0 s to 40 s) and the satellite drives to the starting position (40 s to 75 s). After the starting position was reached, both trajectories start and the satellites begin the formation flight. It can be seen, that the starting mode in the real experiment needs a bit more time (see figure 6 right). Both orbits (simulation and real orbit) take the same shapes during the flight. The plots in figure 7 show results of the experiment in TEAMS.

In the left plot in figure 7 the displacement of the laser beam on target can be seen. There are some deviation with a maximum of about 0.04 m. One reason for this error can be seen in the right plot of figure 7 where the position of the target during the test is shown. On top of the nominal sinusoidal behaviour of the x and y position (which is forming the elliptical relative orbit) some oscillations are added due to disturbances. The source of these oscillations is a rotating lower position platform. The air pads of each vehicle cause sometimes friction due to dusk particles and small imperfections of the table. This leads to a rotation of the lower platform. If the lower platform rotates and the center of

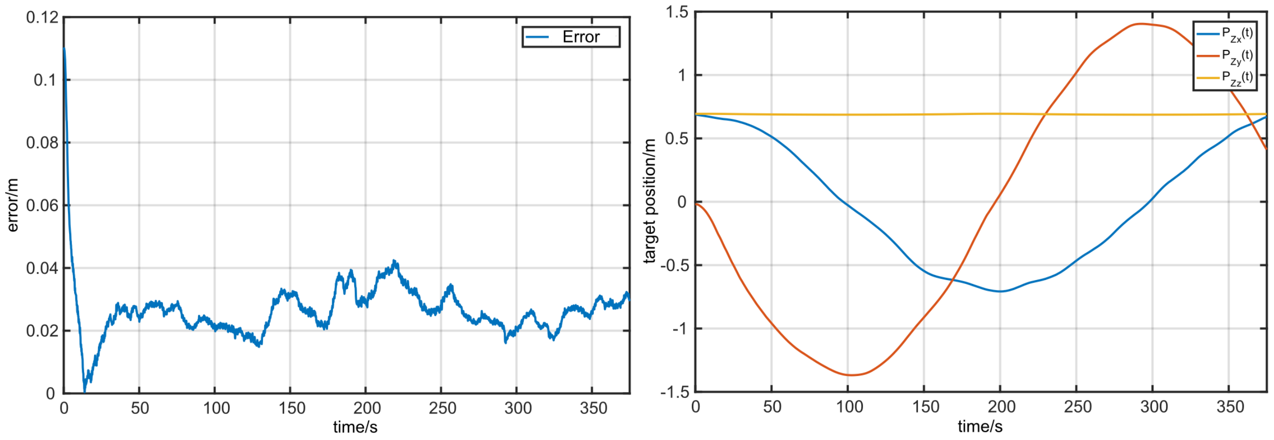

Figure 7: *Pointing error during a real experiment (left); Target position during a real experiment (right)*

mass of the total vehicle is not perfectly aligned with the center of the air bearing it can cause small rotations of the air bearing center around the nominal position. A more precise balancing of the lower platform would minimize this effect.

6. CONCLUSION

This paper gives an overview of the steps taken towards the objective of a coordinated orbit and attitude control of a satellite formation on an air bearing table. The different steps for the solution were explained and at the end a running system can be shown. During the work some ideas to improve the whole system occurred. In future work, the Clohessy-Wiltshire equations can be enhanced in such a way that the restriction for the circular reference orbit can be neglected. Furthermore the integral term in the position controller should be considered in the LQR design.

REFERENCES

[1] Riccardo Benvenuto, Michèle Lavagna, Markus Schlotterer, and Stephan Theil. Experimental setup and tests' results for uncooperative objects capture and manoeuvring with robotic arm. In *7th European Conference on Space Debris*, April 2017.

[2] Henrique Daitx, Markus Schlotterer, James Whidborne, and Marco Sagliano. Development of a combined attitude and position controller for a satellite simulator. In *67th International Astronautical Congress (IAC)*. International Astronautical Federation, September 2016.

[3] J. Mattingley and S. Boyd. Cvxgen: A code generator for embedded convex optimization. *Optimization and Engineering*, 13:1–27, 2012.

[4] Markus Schlotterer and Sergej Novoschilov. On-ground path planning experiments for multiple satellites. In *23rd International Symposium on Space Flight Dynamics 2012*, Oktober 2012.

[5] Markus Schlotterer and Stephan Theil. Testbed for on-orbit servicing and formation flying dynamics emulation. In *AIAA Guidance, Navigation, and Control Conference 2010*, August 2010. Paper-Nr: AIAA 2010-8108.

[6] M. J. Sidi. *Spacecraft Dynamics and Control*. Cambridge University Press, 2006.

[7] S. Vromen, F.J. de Bruijn, and Erwin Mooij. Guidance for autonomous rendezvous and docking with envisat using hardware-in-the-loop simulations. In *IWSCFF 2015*, Juni 2015.

[8] Sebastian Wehrmann. Koordinierte Bahn- und Lageregelung einer Satellitenformation. Master's thesis, Universität Bremen, Februar 2017.