2017 IEEE International Conference on Robotics and Automation (ICRA)
Singapore, May 29 - June 3, 2017

A global controller for flying wing tailsitter vehicles

Robin Ritz and Raffaello D'Andrea

Abstract— **We present a global controller for tracking nominal trajectories with a flying wing tailsitter vehicle. The control strategy is based on a first-principles model of the vehicle dynamics that captures all relevant aerodynamic effects, and we apply an onboard parameter learning scheme in order to estimate unknown aerodynamic parameters. A cascaded control architecture is used: Based on position and velocity errors an outer control loop computes a desired attitude keeping the vehicle in coordinated flight, while an inner control loop tracks the desired attitude using a lookup table with precomputed optimal attitude trajectories. The proposed algorithms can be implemented on a typical microcontroller and the performance is demonstrated in various experiments.**

I. Introduction

Tailsitter vehicles combine hover capabilities with efficient forward flight: A tailsitter is able to take off and land vertically on its tail with the nose and thrust direction pointing upwards, and for fast forward flight the vehicle tilts to a near-horizontal attitude resulting in a more efficient lift production with conventional wings [1], [2]. Compared to other powered lift aircraft types (such as tiltrotors [3], [4] or tiltwings [5], [6]), the major advantage of a tailsitter is its mechanical simplicity; no mechanism for changing the direction of the propulsive system is required, saving weight and reducing susceptibility to malfunctions. However, this advantage comes at the cost of additional challenges during controller design due to the large flight envelope of a typical mission scenario.

Over the last decade, the research community has developed many successful control strategies for small, unmanned tailsitter vehicles. Typically, the control architecture contains a switching logic that chooses between a hover and a fixed-wing flight controller, depending on the current operating point of the vehicle [7], [8], [9], [10]. Successful transitions between hover and forward flight (and vice-versa) have also been demonstrated using gain scheduling methods [11], [12]. Since transitions are an integral part of the tailsitter flight envelope, the attitude controller must be functional over a large set of attitudes. In order to avoid singularities, typically a quaternion representation for the vehicle's attitude is used, combined with linear feedback on the quaternion error vector in order to obtain the desired angular body rates [13], [14], [15]. However, alternative approaches for representing and controlling attitude have been proposed in order to achieve better tracking performance when large attitude errors occur; see for example the resolved tilt-twist method [16], [17], or methods using heading-independent attitude error representations operating directly in $SO(3)$ [18].

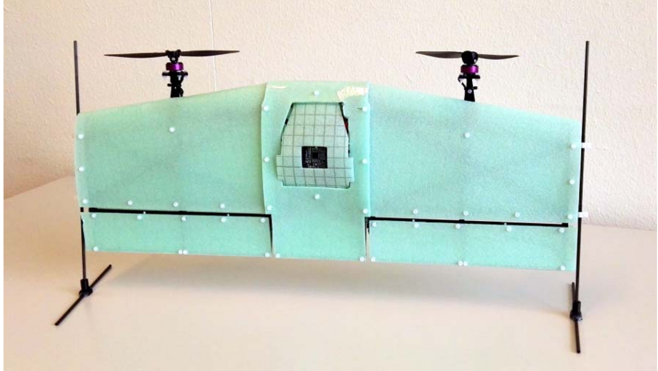

Fig. 1. Picture of the IDSC Tailsitter. The vehicle has been developed at the Institute for Dynamic Systems and Control (IDSC), ETH Zurich, as a testbed for novel control strategies. It is used for evaluating the performance of the controller introduced in this paper.

In this paper, we introduce a global control strategy for a small flying wing tailsitter, which enables the vehicle to track nominal trajectories. Similar to [14] and [15], the controller is based on an adaptive, first-principles model of the vehicle dynamics; parameterized models for the aerodynamic forces and torques are learned using data from onboard sensors. The computational requirements of the proposed algorithms are reasonably low such that they can be implemented on a typical microcontroller. A single control law is used for all flight regimes, such that no manual or automatic switching mechanisms between different controllers are required. The controller computes a desired attitude such that the vehicle is in coordinated flight, and this attitude is tracked with an inner control loop that properly accounts for the different magnitudes of attainable torques around the different rotation axes. The performance of the presented controller is demonstrated and analyzed in various experiments. Fig. 1 shows a picture of the tailsitter vehicle that is used for the experimental validation.

The remainder of the paper is structured as follows: In Section II we present a dynamic model for a small flying wing tailsitter vehicle, and in Section III a method for learning unknown aerodynamic model parameters is described. Section IV introduces a global control strategy, and experimental results are presented in Section V. Finally, we conclude in Section VI.

II. Flying wing tailsitter model

In this section, we derive a model for the dynamics of a small flying wing tailsitter vehicle. In order

This research was supported by the Hans-Eggenberger Stiftung and the SNSF (Swiss National Science Foundation). The authors are with the Institute for Dynamic Systems and Control, ETH Zürich.
{rritz, rdandrea}@ethz.ch

978-1-5090-4633-1/17/$31.00 ©2017 IEEE

to simplify notation, vectors may be expressed as n-tuples $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, with dimension and stacking clear from context. The axes of a coordinate frame $\mathcal{F}$ are denoted as $x_\mathcal{F}$, $y_\mathcal{F}$, and $z_\mathcal{F}$, and if a vector $\boldsymbol{x}$ is expressed in a particular frame $\mathcal{F}$, the vector is written as $\boldsymbol{x}^\mathcal{F}$.

As illustrated in Fig. 2, the vehicle is actuated by two propellers that are mounted in front of the wings, and two flaps located at the wings' trailing edges. We assume that the dynamics of the propeller motors are fast compared to the vehicle dynamics, and that a low-level motor controller accounts for varying airspeed of the vehicle when choosing the nominal angular rate of the propellers. Hence, we assume that we can directly set the propeller forces f_l and f_r without dynamics or delay. Similarly, the flap servos are assumed to be fast such that we can instantaneously set the flap angles δ_l and δ_r. The actuators are subject to the saturations limits

$$\begin{aligned} f_{min} &\leq f_l, f_r \leq f_{max}, \\ \delta_{min} &\leq \delta_l, \delta_r \leq \delta_{max}, \end{aligned} \tag{1}$$

where the thrust limits f_{min} and f_{max} may vary with the vehicle's airspeed. We assume that the propellers can only produce positive thrust, i.e. $f_{min} > 0$.

An inertial coordinate frame $\mathcal{I}$ is chosen such that the gravitational acceleration is aligned with the negative $z_\mathcal{I}$-axis, and we introduce a body-fixed frame $\mathcal{B}$ with origin at the vehicle's center of mass. The $z_\mathcal{B}$-axis is aligned with the thrust direction, the $x_\mathcal{B}$-axis points along the left wing, and the $y_\mathcal{B}$-axis completes the right-handed coordinate system. The attitude of the vehicle is described by a rotation matrix $\boldsymbol{R}$ that represents a rotation from the inertial frame $\mathcal{I}$ to the body-fixed frame $\mathcal{B}$. The position of the vehicle's center of mass relative to the inertial frame $\mathcal{I}$ is denoted as $\boldsymbol{p} = (p_x, p_y, p_z)$. The translational velocity of the vehicle is denoted as $\boldsymbol{v} = (v_x, v_y, v_z)$, and the rotational velocity is denoted as $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$. The position and attitude kinematics yield

$$\begin{aligned} \dot{\boldsymbol{p}}^\mathcal{I} &= \boldsymbol{v}^\mathcal{I}, \\ \dot{\boldsymbol{R}} &= -[\![\boldsymbol{\omega}^\mathcal{B}]\!]\boldsymbol{R}, \end{aligned} \tag{2}$$

where $[\![\boldsymbol{\omega}^\mathcal{B}]\!]$ denotes the skew-symmetric cross product matrix of the body rate vector [19]. We model the vehicle as a rigid body with mass m and rotational inertia $\boldsymbol{J}$, where $\boldsymbol{J}$ is assumed to be diagonal with entries J_{xx}, J_{yy}, and J_{zz}. The dynamics are given by the Newton-Euler equations:

$$\begin{aligned} m\dot{\boldsymbol{v}}^\mathcal{I} &= \boldsymbol{R}^T \boldsymbol{f}^\mathcal{B}_{tot} + m\boldsymbol{g}^\mathcal{I}, \\ \boldsymbol{J}\dot{\boldsymbol{\omega}}^\mathcal{B} &= \boldsymbol{m}^\mathcal{B}_{tot} - \boldsymbol{\omega}^\mathcal{B} \times \boldsymbol{J}\boldsymbol{\omega}^\mathcal{B}, \end{aligned} \tag{3}$$

where $\boldsymbol{g}$ is the gravitational acceleration, and the vectors $\boldsymbol{f}_{tot}$ and $\boldsymbol{m}_{tot}$ denote the total force and torque, respectively, acting on the vehicle. In the following, models for both the total torque $\boldsymbol{m}_{tot}$ and the total force $\boldsymbol{f}_{tot}$ will be derived.

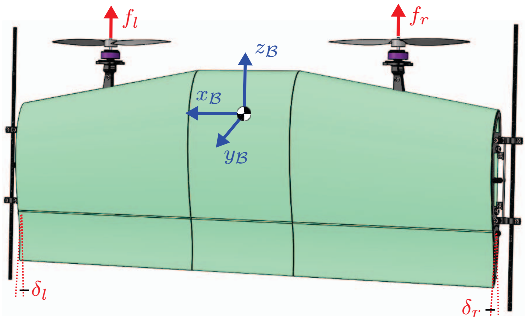

Fig. 2. Illustration of a flying wing tailsitter vehicle. The vehicle is actuated by two propellers that produce thrust forces along the $z_\mathcal{B}$-axis, and two flaps that produce an aerodynamic force and torque by deflecting the airflow behind the wings.

A. Torque model

In Section IV-D, we will present a body rate controller based on the assumption that we can apply a desired torque to the vehicle by choosing corresponding actuator inputs. Therefore, we derive an expression for the total torque that captures all relevant first-order effects of the actuators, as well as first-principles aerodynamic effects of the vehicle body. The total torque is separated into an aerodynamic and a propeller component:

$$\boldsymbol{m}^\mathcal{B}_{tot} = \boldsymbol{m}^\mathcal{B}_{air} + \boldsymbol{m}^\mathcal{B}_{prop}, \tag{4}$$

and in the following we will derive expressions for both terms.

1) Aerodynamic torque: Typically, the body of a flying wing vehicle causes an aerodynamic pitching moment around the wing axis (i.e. the $x_\mathcal{B}$-axis) that depends on the angle of attack α and the reference airspeed v, with

$$\begin{aligned} \alpha &= \operatorname{arctan2}\left(v^\mathcal{B}_y, v^\mathcal{B}_z\right), \\ v &= \sqrt{(v^\mathcal{B}_y)^2 + (v^\mathcal{B}_z)^2}. \end{aligned} \tag{5}$$

Note that we assume that the ambient air is at rest, such that the vehicle's ground velocity is equal to its airspeed. Typically, it is difficult to find an accurate model-based description for the aerodynamic pitching moment. Therefore, we denote it as $m_{pitch}(\alpha, v)$ and treat it as an unknown function during the modeling process. In Section III, we will introduce a method for learning the pitching moment during flight using onboard sensors.

For each wing, the effect of the actuators on the aerodynamic torque is computed separately, assuming no cross coupling between left and right wing. Further, we assume that the entire flap is in the slipstream of the propeller, and that if the vehicle is flying forward (i.e. for positive $v^\mathcal{B}_z$) the slipstream velocity is uniform and can be approximated using Momentum Theory [20]. For backward flight (i.e. for negative $v^\mathcal{B}_z$) Momentum Theory is not valid. In this case, we neglect the relative airflow and assume that backward velocities are small and negligible. Thus, for each side s, $s \in \{l, r\}$, the reference airspeed over the flap along the $z_\mathcal{B}$-axis is modeled as

$$v^\mathcal{B}_{s,z} = \sqrt{\frac{2f_s}{\rho d} + \max\left(0, v^\mathcal{B}_z\right)^2}, \tag{6}$$

where ρ denotes the density of air and d is the propeller disk area. The total reference airspeed over the flap yields

$$v_s = \sqrt{(v_y^{\mathcal{B}})^2 + (v_{s,z}^{\mathcal{B}})^2}. \tag{7}$$

We assume that the flap deflects the airflow behind the wing by a small angle proportional to the flap angle δ_s. Hence the aerodynamic torque can be written as

$$\boldsymbol{m}_{air}^{\mathcal{B}} = \left(m_{air,x}^{\mathcal{B}}, m_{air,y}^{\mathcal{B}}, m_{air,z}^{\mathcal{B}}\right), \tag{8}$$

with

$$\begin{aligned} m_{air,x}^{\mathcal{B}} &= (b_x + c_x\delta_l)\, v_l^2 + (b_x + c_x\delta_r)\, v_r^2 + m_{pitch}\,(\alpha, v)\,, \\ m_{air,y}^{\mathcal{B}} &= b_y v_l^2 - b_y v_r^2, \\ m_{air,z}^{\mathcal{B}} &= (b_z + c_z\delta_l)\, v_l^2 - (b_z + c_z\delta_r)\, v_r^2, \end{aligned} \tag{9}$$

where the constants b_x, b_y, b_z, c_x, and c_z are wing and flap characteristics.

2) Propeller torque: The propellers are mounted with an offset of l along the positive and negative $x_{\mathcal{B}}$-axis, respectively, and the thrust direction is parallel to the $z_{\mathcal{B}}$-axis. Consequently, the torque created by the propellers is given by

$$\boldsymbol{m}_{prop}^{\mathcal{B}} = (0, l\,(f_r - f_l)\,, \kappa\,(f_r - f_l))\,, \tag{10}$$

where κ denotes the torque-to-thrust ratio of the propellers. Note that the left propeller rotates counter-clockwise, and the right propeller rotates clockwise.

B. Force model

Similar to the total torque, the total force acting on the vehicle is separated into an aerodynamic and a propeller component:

$$\boldsymbol{f}_{tot}^{\mathcal{B}} = \boldsymbol{f}_{air}^{\mathcal{B}} + \boldsymbol{f}_{prop}^{\mathcal{B}}. \tag{11}$$

1) Aerodynamic force: As we will introduce in Section IV-B, the expression for the total force is used to compute a desired attitude and a desired total thrust value in a high-level control block. Therefore, in order to keep the problem tractable, we derive a simplified model for the aerodynamic force that does not capture differences in actuator inputs between the left and the right wing. In particular, we apply a first-principles aerodynamic model based on the following two assumptions: 1) the lateral aerodynamic force acting on the vehicle is negligible[1], and 2) the aerodynamic force is a function of the angle of attack α, the reference airspeed v, and the average propeller force $f_a = (f_l + f_r)/2$. The aerodynamic force can thus be written as

$$\boldsymbol{f}_{air}^{\mathcal{B}} = (0, f_{lift}(\alpha, v, f_a), -f_{drag}(\alpha, v, f_a))\,, \tag{12}$$

where $f_{lift}(\alpha, v, f_a)$ and $f_{drag}(\alpha, v, f_a)$ denote the aerodynamic lift and drag force, respectively. Typically, it is difficult to derive an accurate model-based description for the aerodynamic forces, and therefore in Section III we will introduce a method for learning these forces during flight using onboard sensors.

2) Propeller force: The force due to propeller thrust is

$$\boldsymbol{f}_{prop}^{\mathcal{B}} = (0, 0, 2f_a)\,. \tag{13}$$

III. Learning aerodynamic force and torque

Accurate maps for the pitching moment $m_{pitch}(\alpha, v)$ and the aerodynamic forces $f_{lift}(\alpha, v, f_a)$ and $f_{drag}(\alpha, v, f_a)$ might be obtained, for example, by wind tunnel measurements. However, if a wind tunnel is not available or if the aerodynamic surfaces are changed regularly, this approach is impractical. Another method for modeling the aerodynamics is to simulate the airflow around the vehicle numerically using tools from computational fluid dynamics. For a vehicle that has its wings and flaps in the slipstream of the propellers, however, the aerodynamics are complex which imposes high requirements on the spatial and temporal resolution of the simulation and on the accuracy of the three-dimensional model of the aerodynamic surfaces. Therefore, in this paper we use a different approach for approximating the aerodynamics of the vehicle: We estimate the aerodynamic properties using onboard sensor data. This approach provides the advantage that we can run the estimation onboard during flight and are thus adaptive to changes in the aerodynamic properties. A drawback of this approach is that the model of the aerodynamic forces and torques might be inaccurate if we enter a new flight regime.

A. Parameterized aerodynamic functions

We apply a parameter learning scheme, due to its low requirements on computation and memory compared to other learning methods [21]. The structure of the parameterized functions is based on heuristics and first-principles aerodynamic effects, and might depend on the particular vehicle design. Herein, we use the following parameterized functions for the pitching moment and aerodynamic lift and drag force:

$$\begin{aligned} m_{pitch}\,(\alpha, v) &= k_{p1}\sin(\alpha) v^2, \\ f_{lift}\,(\alpha, v, f_a) &= \left(k_{l1}\sin(\alpha)\cos(\alpha)^2 + k_{l2}\sin(\alpha)^3\right) v^2 \\ &\quad + k_{l3} f_a, \\ f_{drag}\,(\alpha, v, f_a) &= \left(k_{d1}\sin(\alpha)^2\cos(\alpha) + k_{d2}\cos(\alpha)\right) v^2 \\ &\quad + k_{d3} f_a, \end{aligned} \tag{14}$$

where $k_{(\cdot)}$ are the parameters that are learned. Ideally, these parameters are constant over the domain of the aerodynamic functions.

B. Parameter learning

For each learning update, we first compute the aerodynamic torque and force that is currently acting on the vehicle, and then update the function parameters $k_{(\cdot)}$ based on these values.

[1] Here, the *lateral direction* refers to the direction along the $x_{\mathcal{B}}$-axis.

By combining and rearranging (3), (4), and (9) we obtain an expression for the pitching moment acting on the vehicle:

$$\begin{aligned} m_{pitch} &= J_{xx}\dot{\omega}_x^{\mathcal{B}} + (J_{zz} - J_{yy})\omega_y^{\mathcal{B}}\omega_z^{\mathcal{B}} \\ &\quad - (b_x + c_x\delta_l)\, v_l^2 - (b_x + c_x\delta_r)\, v_r^2. \end{aligned} \tag{15}$$

Similarly, by combining and rearranging (3), (11), (12), and (13) we obtain an expression for the lift and drag force:

$$\begin{aligned} f_{lift} &= a_{p,y}^{\mathcal{B}} m, \\ f_{drag} &= f_l + f_r - a_{p,z}^{\mathcal{B}} m, \end{aligned} \tag{16}$$

where $\boldsymbol{a}_p$ denotes the proper acceleration[2]. An inertial measurement unit provides measurements for the proper acceleration $\boldsymbol{a}_p^{\mathcal{B}}$ and for the body rates $\boldsymbol{\omega}^{\mathcal{B}}$. Through numerical differentiation we obtain the angular acceleration $\dot{\boldsymbol{\omega}}^{\mathcal{B}}$, and we assume that an estimate of the current velocity $\boldsymbol{v}^{\mathcal{B}}$ is available. Hence, we can evaluate (15) and (16) in order to obtain the aerodynamic torque and forces currently acting on the vehicle. These values are then used to estimate the parameters $k_{(\cdot)}$ with an extended Kalman filter, as described for example in [21].

IV. Control Strategy

In this section, we introduce a global control strategy for tracking a desired trajectory. The desired trajectory is given by position and velocity trajectory $\boldsymbol{p}_{des}$ and $\boldsymbol{v}_{des}$, respectively, and feed-forward acceleration $\boldsymbol{a}_{ff}$. If the desired trajectory contains sequences where the nominal aerodynamic forces vanish (such as hover), we might also specify a desired yaw angle trajectory ψ_{des}. (When the vehicle is hovering, the yaw angle corresponds to the vehicle's rotation around the $z_{\mathcal{I}}$-axis.) We assume that an onboard state estimator delivers a full state estimate of the vehicle, i.e. estimated position $\boldsymbol{p}_{est}$, estimated velocity $\boldsymbol{v}_{est}$, estimated attitude $\boldsymbol{R}_{est}$, and estimated body rates $\boldsymbol{\omega}_{est}$.

We use a cascaded control architecture: Based on position and velocity errors an outer control loop computes a desired attitude keeping the vehicle in coordinated flight, while an inner control loop tracks the desired attitude using a lookup table with precomputed optimal attitude trajectories. A flowchart of the controller is shown in Fig. 3, and the individual control blocks are introduced in the following.

A. Position control

As proposed in [22], a desired acceleration $\boldsymbol{a}_{des}$ is computed based on position and velocity errors. The control loop is shaped such that for each coordinate the system behaves like a linear second-order system with some desired time constant τ_p and damping ratio ζ_p. The desired acceleration is thus given by

$$\boldsymbol{a}_{des}^{\mathcal{I}} = \boldsymbol{a}_{ff}^{\mathcal{I}} + \frac{1}{\tau_p^2}\left(\boldsymbol{p}_{des}^{\mathcal{I}} - \boldsymbol{p}_{est}^{\mathcal{I}}\right) + \frac{2\zeta_p}{\tau_p}\left(\boldsymbol{v}_{des}^{\mathcal{I}} - \boldsymbol{v}_{est}^{\mathcal{I}}\right). \tag{17}$$

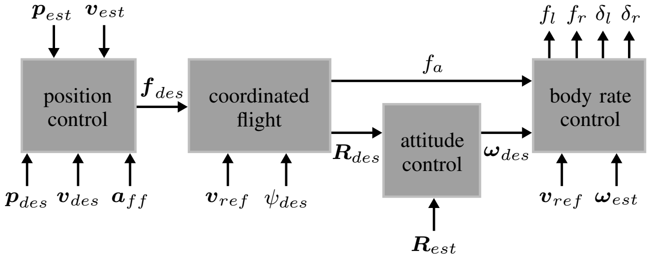
$\boldsymbol{p}_{est}$ $\boldsymbol{v}_{est}$
position control
$\boldsymbol{f}_{des}$
coordinated flight
f_a
$\boldsymbol{R}_{des}$
attitude control
$\boldsymbol{\omega}_{des}$
body rate control
f_l f_r δ_l δ_r
$\boldsymbol{p}_{des}$ $\boldsymbol{v}_{des}$ $\boldsymbol{a}_{ff}$
$\boldsymbol{v}_{ref}$ ψ_{des}
$\boldsymbol{R}_{est}$
$\boldsymbol{v}_{ref}$ $\boldsymbol{\omega}_{est}$

Fig. 3. Flowchart illustrating the control architecture. The controller is based on a cascaded control scheme, and the individual control blocks are described in Section IV.

Subsequently, the desired acceleration is converted to a desired force vector

$$\boldsymbol{f}_{des}^{\mathcal{I}} = \left(\boldsymbol{a}_{des}^{\mathcal{I}} - \boldsymbol{g}^{\mathcal{I}}\right) m, \tag{18}$$

which is then fed to the next controller block.

B. Coordinated flight

In the following, the average propeller thrust f_a and the desired attitude $\boldsymbol{R}_{des}$ are computed such that two conditions are satisfied: 1) the vehicle is in coordinated flight[3], and 2) the resulting total force $\boldsymbol{f}_{tot}$ acting on the vehicle matches the desired force $\boldsymbol{f}_{des}$ demanded by the position controller.

The desired attitude can be written as three subsequent rotations:

$$\boldsymbol{R}_{des} = \boldsymbol{R}_c \boldsymbol{R}_b \boldsymbol{R}_a, \tag{19}$$

where the the individual rotations are introduced in the following. For a more intuitive notation, we will describe the rotations using the axis-angle representation, i.e. a rotation is described by an angle σ and a unit vector $\boldsymbol{n}$ denoting the rotation axis [19]. A vector $\boldsymbol{x}$ expressed in the intermediate coordinate frame defined by $\boldsymbol{R}_a$ is denoted as $\boldsymbol{x}^{(a)}$. Similarly, vectors expressed in the coordinate frames defined by $\boldsymbol{R}_b\boldsymbol{R}_a$ and $\boldsymbol{R}_c\boldsymbol{R}_b\boldsymbol{R}_a$ are denoted as $\boldsymbol{x}^{(ab)}$ and $\boldsymbol{x}^{(abc)}$, respectively.

The first rotation $\boldsymbol{R}_a$ is a rotation that, starting from the inertial frame $\mathcal{I}$, aligns the vehicle's forward axis (i.e. the $z_{\mathcal{B}}$-axis) with the current direction of flight:

$$\begin{aligned} \sigma_a &= \arccos\left(\frac{v_{ref,z}^{\mathcal{I}}}{||\boldsymbol{v}_{ref}||}\right), \\ \boldsymbol{n}_a &= \left(\frac{-v_{ref,y}^{\mathcal{I}}}{||(v_{ref,x}^{\mathcal{I}}, v_{ref,y}^{\mathcal{I}}, 0)||}, \frac{v_{ref,x}^{\mathcal{I}}}{||(v_{ref,x}^{\mathcal{I}}, v_{ref,y}^{\mathcal{I}}, 0)||}, 0\right), \end{aligned} \tag{20}$$

where $||\cdot||$ denotes the Euclidean norm, and the reference velocity $\boldsymbol{v}_{ref}$ can be chosen to be either the desired velocity $\boldsymbol{v}_{des}$ or the estimated velocity $\boldsymbol{v}_{est}$. (For the experiments presented in this paper we use $\boldsymbol{v}_{ref} = \boldsymbol{v}_{des}$.) If the vertical velocity components $v_{ref,x}^{\mathcal{I}}$ and $v_{ref,y}^{\mathcal{I}}$ are zero, then either the vehicle's forward axis is already aligned with the direction of flight ($v_{ref,z}^{\mathcal{I}} \neq 0$) or the vehicle is hovering

[2] In this context, *proper acceleration* refers to the acceleration relative to free fall.

[3] Here, *coordinated flight* refers to a flight state with no sideslip and no lateral acceleration.

($v^{\mathcal{I}}_{ref,z} = 0$). In both cases, the rotation $\boldsymbol{R}_a$ can be chosen to be identity:

$$\boldsymbol{R}_a = \boldsymbol{I}_3. \tag{21}$$

In practice, we define a threshold v_{th} and choose $\boldsymbol{R}_a$ according to (21) if $||(v^{\mathcal{I}}_{ref,x}, v^{\mathcal{I}}_{ref,y}, 0)|| < v_{th}$.

The second rotation $\boldsymbol{R}_b$ rolls the vehicle around its current $z_{\mathcal{B}}$-axis (i.e. the $z_{\mathcal{B}}$-axis after rotating by $\boldsymbol{R}_a$) such that the desired total force has no lateral component and is positive along the new $y_{\mathcal{B}}$-axis. The rotation $\boldsymbol{R}_b$ yields

$$\begin{aligned}\sigma_b &= \arccos\left(\frac{f^{(a)}_{des,y}}{||(f^{(a)}_{des,x}, f^{(a)}_{des,y}, 0)||}\right),\\ \boldsymbol{n}_b &= \left(0, 0, -\operatorname{sign}\left(f^{(a)}_{des,x}\right)\right).\end{aligned} \tag{22}$$

If the vehicle is hovering or if the desired force is parallel to the vehicle's direction of flight, then both $f^{(a)}_{des,x}$ and $f^{(a)}_{des,y}$ are zero and the angle σ_b is undefined. In this case, we choose $\boldsymbol{R}_b$ as

$$\begin{aligned}\sigma_b &= \psi_{des},\\ \boldsymbol{n}_b &= (0, 0, 1),\end{aligned} \tag{23}$$

where ψ_{des} is the desired yaw angle that is part of the desired trajectory. Note that if the vehicle is hovering, then ψ_{des} corresponds to the vehicle's rotation around the $z_{\mathcal{I}}$-axis. Again, in practice we define a threshold f_{th} and choose $\boldsymbol{R}_b$ according to (23) if $||(f^{(a)}_{des,x}, f^{(a)}_{des,y}, 0)|| < f_{th}$.

Finally, the third rotation $\boldsymbol{R}_c$ pitches the vehicle around its current $x_{\mathcal{B}}$-axis (i.e. the $x_{\mathcal{B}}$-axis after rotating by $\boldsymbol{R}_a$ and $\boldsymbol{R}_b$) such that the resulting total force $\boldsymbol{f}_{tot}$ matches the desired force $\boldsymbol{f}_{des}$. Consequently, the rotation axis is chosen as

$$\boldsymbol{n}_c = (-1, 0, 0), \tag{24}$$

and the rotation angle σ_c corresponds to the desired angle of attack α (if the vehicle is not hovering). Our objective is to choose the angle σ_c and the desired average propeller thrust f_a such that the following condition is satisfied:

$$\boldsymbol{f}^{\mathcal{B}}_{tot} \overset{!}{=} \boldsymbol{f}^{(abc)}_{des}, \tag{25}$$

where the total force $\boldsymbol{f}^{\mathcal{B}}_{tot}$ is given by (11). Noting that the x-component of (25) is trivially satisfied, the condition can be written as

$$\begin{aligned}f_{lift}(\sigma_c, v_{ref}, f_a) &\overset{!}{=} \cos(\sigma_c) f^{(ab)}_{des,y} + \sin(\sigma_c) f^{(ab)}_{des,z},\\ 2f_a - f_{drag}(\sigma_c, v_{ref}, f_a) &\overset{!}{=} \cos(\sigma_c) f^{(ab)}_{des,z} - \sin(\sigma_c) f^{(ab)}_{des,y}.\end{aligned} \tag{26}$$

Since the reference airspeed v_{ref} is invariant under a rotation around the $x_{\mathcal{B}}$-axis, we can compute it with

$$v_{ref} = \sqrt{(v^{(ab)}_{ref,y})^2 + (v^{(ab)}_{ref,z})^2}. \tag{27}$$

Hence, (26) is a system of two equations for the two unknowns σ_c and f_a. For reasonably complex aerodynamic maps $f_{lift}(\alpha, v, f_a)$ and $f_{drag}(\alpha, v, f_a)$ no closed-from solution to the above system of equations exists, moreover there might not exist a solution at all. Therefore, we numerically minimize a quadratic residual function

$$r(\sigma_c, f_a) = \Delta f_y(\sigma_c, f_a)^2 + \Delta f_z(\sigma_c, f_a)^2, \tag{28}$$

with

$$\begin{aligned}\Delta f_y(\sigma_c, f_a) &= f_{lift}(\sigma_c, v_{ref}, f_a)\\ &\quad - \cos(\sigma_c) f^{(ab)}_{des,y} - \sin(\sigma_c) f^{(ab)}_{des,z},\\ \Delta f_z(\sigma_c, f_a) &= 2f_a - f_{drag}(\sigma_c, v_{ref}, f_a)\\ &\quad - \cos(\sigma_c) f^{(ab)}_{des,z} + \sin(\sigma_c) f^{(ab)}_{des,y}.\end{aligned} \tag{29}$$

The pitch angle σ_c is constrained to the interval $[-\pi, \pi]$, and we constrain the average propeller force f_a to be within the saturation limits $[f_{min}, f_{max}]$. Thus, we get the following optimization problem:

$$\begin{aligned}\text{minimize} \quad & r(\sigma_c, f_a)\\ \text{subject to} \quad & \sigma_c \in [-\pi, \pi],\\ & f_a \in [f_{min}, f_{max}].\end{aligned} \tag{30}$$

For every controller update, we apply 50 iterations of the Nelder-Mead method [23], in order to find a local optimum of the above optimization problem. This delivers the desired rotation angle σ_c and the desired average propeller force f_a, and thus completes the coordinated flight controller block.

C. Attitude control

We apply an attitude controller based on Optimal Control, enabling the vehicle to recover from large attitude errors. The attitude control law is obtained by solving an optimal control problem, with the objective of correcting large attitude errors by turning primarily around the vehicle's strongly actuated $y_{\mathcal{B}}$-axis[4]. Solutions for a set of initial attitude errors are precomputed and stored in a lookup table. For each controller update, the desired body rates are obtained by reading the optimal body rates from this table:

$$\boldsymbol{\omega}^{\mathcal{B}}_{des} = \boldsymbol{\omega}_{opt}(\boldsymbol{R}_{err}), \tag{31}$$

where $\boldsymbol{\omega}_{opt}(\cdot)$ denotes the lookup table containing the optimal body rates, and $\boldsymbol{R}_{err}$ is the attitude error given by

$$\boldsymbol{R}_{err} = \boldsymbol{R}_{est}(\boldsymbol{R}_{des})^{-1}. \tag{32}$$

For a better intuition of the attitude control law, we note that for small attitude errors the map with the optimal body rates can be written as

$$\boldsymbol{\omega}^{\mathcal{B}}_{des} \approx -\frac{1}{\tau_\alpha} \boldsymbol{n}_{err} \sigma_{err}, \tag{33}$$

where τ_α is a desired first-order time constant, and $\boldsymbol{n}_{err}$ and σ_{err} describe the axis-angle representation of the attitude error $\boldsymbol{R}_{err}$. For a more detailed introduction of the attitude controller we refer to [24].

[4] Typically, compared to the other axes, the control authority around the $y_{\mathcal{B}}$-axis is high. This is because the control torque around this axis is directly produced by the propellers' differential thrust.

D. Body rate control

The body rate controller is designed such that the elements of the body rate error vector follow three decoupled linear first-order systems with desired time constant τ_ω. The desired torque is obtained by rearranging the angular dynamics (3), which yields

$$\boldsymbol{m}^{\mathcal{B}}_{des} = \frac{1}{\tau_\omega}\boldsymbol{J}\left(\boldsymbol{\omega}^{\mathcal{B}}_{des} - \boldsymbol{\omega}^{\mathcal{B}}_{est}\right) + \boldsymbol{\omega}^{\mathcal{B}}_{est} \times \boldsymbol{J}\boldsymbol{\omega}^{\mathcal{B}}_{est}. \tag{34}$$

The actuator inputs are computed such that the desired torque is acting on the vehicle, and such that we achieve the desired average propeller thrust. Hence, the following conditions must be satisfied:

$$\begin{aligned} \boldsymbol{m}^{\mathcal{B}}_{tot} &\overset{!}{=} \boldsymbol{m}^{\mathcal{B}}_{des}, \\ f_l + f_r &\overset{!}{=} 2f_a, \end{aligned} \tag{35}$$

where $\boldsymbol{m}^{\mathcal{B}}_{tot}$ is given by (4). Solving for the actuator inputs yields

$$\begin{aligned} f_l &= f_a - \frac{m^{\mathcal{B}}_{des,y}\rho d}{2\rho dl - 4b_y}, \\ f_r &= f_a + \frac{m^{\mathcal{B}}_{des,y}\rho d}{2\rho dl - 4b_y}, \\ \delta_l &= \frac{b_z v_r^2 - b_z v_l^2 + f_l\kappa - f_r\kappa + m^{\mathcal{B}}_{des,z}}{2c_z v_l^2} \\ &\quad + \frac{b_x v_l^2 + b_x v_r^2 - m^{\mathcal{B}}_{des,x} + m_{pitch}\left(\alpha_{ref}, v_{ref}\right)}{-2c_x v_l^2}, \\ \delta_r &= \frac{b_z v_r^2 - b_z v_l^2 + f_l\kappa - f_r\kappa + m^{\mathcal{B}}_{des,z}}{-2c_z v_r^2} \\ &\quad + \frac{b_x v_l^2 + b_x v_r^2 - m^{\mathcal{B}}_{des,x} + m_{pitch}\left(\alpha_{ref}, v_{ref}\right)}{-2c_x v_r^2}, \end{aligned} \tag{36}$$

where α_{ref} is computed by inserting $\boldsymbol{v}_{ref}$ into (5). If the resulting actuator inputs are not within their attainable range, we adapt the desired torque and propeller forces with the prioritization method described in [24]. Finally, these actuator inputs are then applied to the system and this completes the derivation of the control algorithm.

V. Results

In this section, we present results demonstrating the performance of the control strategy introduced herein. A video of the vehicle performing various maneuvers is available at `https://youtu.be/wfmf-eJ89T4`. The vehicle has been showcased in front of a live audience during Raffaello D'Andrea's talk at the TED conference 2016, available at `https://www.ted.com/talks/raffaello_d_andrea_meet_the_dazzling_flying_machines_of_the_future`.

A. The IDSC Tailsitter

The IDSC Tailsitter has been designed at the Institute for Dynamic Systems and Control (IDSC), ETH Zurich, as a testbed for novel control algorithms for tailsitter vehicles. The airframe of the vehicle is based on a Clark Y profile and designed such that for regular flight the pitching moment vanishes for zero flap angle, and such that the aerodynamic neutral point is close to the vehicle's center of gravity. The objective of this design is to allow agile maneuvers for all flight regimes and to avoid flap angle saturation problems caused by large trim angles. Fig. 1 shows a picture of the vehicle.

The vehicle is equipped with custom onboard electronics: The control algorithms run on a $168\,\mathrm{MHz}$ Cortex M4F microcontroller with an update rate of $500\,\mathrm{Hz}$. The onboard state estimation algorithms and technology are provided by the ETH spin-off company Verity Studios[5]. The vehicle is able to communicate with a ground station using a $2.4\,\mathrm{GHz}$ radio connection. Communication can be established for recording telemetry data or for sending high-level commands to the vehicle, such as 'take-off' or 'land'. However, the vehicle is able to fly autonomously and the trajectories being tracked are stored onboard.

For the plots shown in the following, the data is sampled at a rate of $25\,\mathrm{Hz}$ and for the acceleration measurements a low-pass filter is applied in order to remove high-frequency noise and generate more readable figures.

B. Parameter values

Table I shows the control and physical parameter values that are used for the experiments. The propeller force boundaries, aerodynamic coefficients, and torque-to-thrust-ratio are measured on a load cell, and we assume that the values do not change significantly over the vehicle's flight envelope. The components of the rotational inertia around the different axes are obtained with a CAD model of the vehicle.

[5] http://www.veritystudios.com/

TABLE I
Parameter values

param	value	description
f_{min}	$0.1\,\mathrm{N}$	minimum propeller force
f_{max}	$1.2\,\mathrm{N}$	maximum propeller force
δ_{min}	$-0.87\,\mathrm{rad}$	minimum flap angle
δ_{max}	$0.79\,\mathrm{rad}$	maximum flap angle
m	$150\,\mathrm{g}$	mass of the vehicle
J_{xx}	$4.62 \times 10^{-4}\,\mathrm{kg\,m^2}$	rotational inertia around $x_{\mathcal{B}}$-axis
J_{yy}	$2.32 \times 10^{-3}\,\mathrm{kg\,m^2}$	rotational inertia around $y_{\mathcal{B}}$-axis
J_{zz}	$1.87 \times 10^{-3}\,\mathrm{kg\,m^2}$	rotational inertia around $z_{\mathcal{B}}$-axis
ρ	$1.2\,\mathrm{kg/m^3}$	air density
d	$133\,\mathrm{cm^2}$	propeller disk area
b_x	$7.46 \times 10^{-6}\,\mathrm{Nm/(m/s)^2}$	aerodynamic coefficient
b_y	$4.12 \times 10^{-6}\,\mathrm{Nm/(m/s)^2}$	aerodynamic coefficient
b_z	$3.19 \times 10^{-6}\,\mathrm{Nm/(m/s)^2}$	aerodynamic coefficient
c_x	$2.18 \times 10^{-4}\,\mathrm{Nm/(m/s)^2/rad}$	aerodynamic coefficient
c_z	$3.18 \times 10^{-4}\,\mathrm{Nm/(m/s)^2/rad}$	aerodynamic coefficient
l	$14\,\mathrm{cm}$	propeller offset along $x_{\mathcal{B}}$-axis
κ	$8.72 \times 10^{-3}\,\mathrm{N\,m/N}$	torque-to-thrust ratio
τ_p	$0.6\,\mathrm{s}$	position control time constant
ζ_p	1	position control damping ratio
τ_α	$0.25\,\mathrm{s}$	attitude control time constant
τ_ω	$0.05\,\mathrm{s}$	body rate control time constant

C. *Figure eight maneuvers*

Using the control strategy described herein the vehicle can track arbitrary feasible trajectories. As an example, we track a figure eight trajectory at a speed of about $6\,\mathrm{m/s}$ at a constant height. The horizontal position trajectory consists of two half-circle segments with a radius of $2.1\,\mathrm{m}$ and two splines connecting the two segments. The order of the splines is chosen such that the trajectory is continuous in snap. Fig. 4 shows a top and side view of the desired and estimated position during the figure eight maneuver, and in Fig. 5 the attitude trajectory of the vehicle is illustrated. Further, in Fig. 6 the angle of attack and aerodynamic force trajectories are shown.

D. *Transitions maneuvers*

Transitions between hover and fast forward flight are an integral part of typical tailsitter missions. Therefore, it is crucial that the controller can track trajectories than contain such transitions. Fig. 7 shows the trajectory of a constant-height transition from hover to forward flight.

VI. Conclusion

We have introduced a global control strategy for a flying wing tailsitter vehicle, and its performance has been analyzed in experiments where the vehicle autonomously tracks desired trajectories. The proposed controller shows good tracking performance and is able to perform transitions between hover and fast forward flight. However, the experiments also unveil some systematic position tracking errors, and a bias between desired and estimated pitch angle in forward flight is observed. The reason for this might be, among other things, unmodeled aerodynamic effects or an artifact caused by the state estimation algorithms.

Future work includes experiments with a ground truth sensor, such that the performance can be quantified in a more rigorous way. Further, with experiments in a larger flying space the controller's performance could be analyzed for higher velocities, which might uncover additional aerodynamic effects.

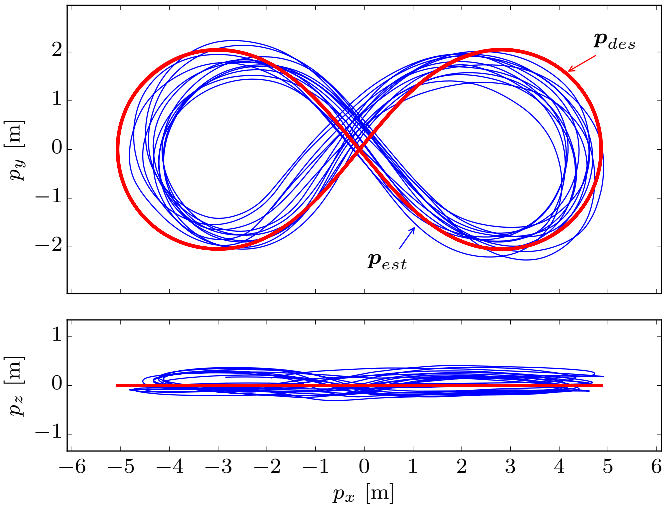

Fig. 4. Desired and estimated position of the vehicle while tracking a horizontal figure eight trajectory. The plot on the top shows a top view of the maneuver, and the plot on the bottom shows a side view. The position estimation accuracy depends on the orientation of the vehicle, which might be the reason for the repeatable tracking offsets in the different segments of the trajectory. Another reason could be aerodynamic effects not captured by the first-principles model of the vehicle.

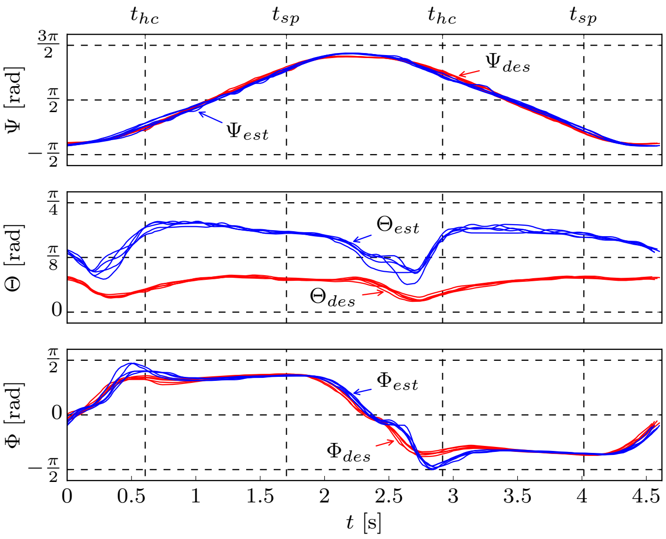

Fig. 5. Attitude of the vehicle while tracking a figure eight trajectory. The plots show data from five executions of the maneuver, each line illustrates the trajectory of one execution. For an intuitive illustration, the desired and estimated attitudes are converted to Tait-Bryan angles being the yaw Ψ, pitch Θ, and roll Φ angle that are typically used to describe the attitude of fixed-wing airplanes [19]. The time when the vehicle enters a half-circle or a spline segment is denoted with t_{hc} or t_{sp}, respectively. The plot on the top shows the yaw trajectories, indicating that the yaw angle tracking error is reasonably small. In the middle plot, showing the pitch trajectories, we can observe that there is a systematic error between desired and estimated pitch angle. The reason for this might be, for example, inaccurate models for the aerodynamic forces that are used to compute the desired angle of attack, or an offset of the vehicle's center of gravity. The plot on the bottom shows the roll trajectories, and we can see that a high roll angle is required during the half-circle segments in order to make the turn. The plot also indicates that the roll angle tracking exhibits some overshooting at the beginning of the turns. Note that the scale of the vertical axes is not equal.

References

[1] J. P. Campbell, "Research on VTOL and STOL aircraft in the United States," in *Advances in Aeronautical Sciences: Proceedings of the first International Congress in the Aeronautical Sciences*, vol. 2. Pergamon Press, 1959.

[2] R. J. Woods, "Convertiplanes and other VTOL aircraft," SAE Technical Paper, Tech. Rep., 1957.

[3] K. G. Wernicke, "Tilt prop-rotor composite research aircraft," U.S. Army Aviation Materiel Laboratories, Tech. Rep., 1968.

[4] R. L. Lichten, "Some aspects of convertible aircraft design," *Journal of the Aeronautical Sciences*, vol. 16, no. 10, pp. 611–622, 1949.

[5] J. Stuart, "Tilt wing propelloplane potentialities," *Journal of the American Helicopter Society*, vol. 4, no. 1, pp. 29–33, 1959.

[6] L. P. Tosti, "Flight investigation of the stability and control characteristics of a 1/4-scale model of a tilt-wing vertical-take-off-and-landing aircraft," NASA, Tech. Rep., 1959.

[7] R. H. Stone, P. Anderson, C. Hutchison, A. Tsai, P. Gibbens, and K. Wong, "Flight testing of the T-wing tail-sitter unmanned air vehicle," *Journal of Aircraft*, vol. 45, no. 2, pp. 673–685, 2008.

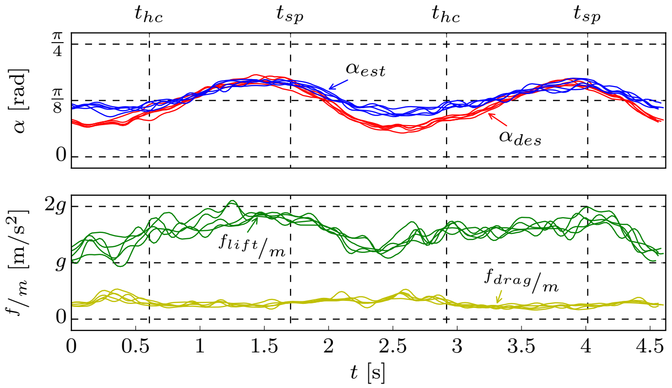

Fig. 6. Angle of attack and aerodynamic forces that are acting on the vehicle during a figure eight maneuver. The plots show data from five executions of the maneuver, each line illustrates the trajectory of one execution. The time when the vehicle enters a half-circle or a spline segment is denoted with t_{hc} or t_{sp}, respectively. The plot on the top shows the desired and estimated angle of attack. The plot on the bottom shows the mass-normalized aerodynamic forces acting on the vehicle, measured by the onboard accelerometer (where propeller forces are subtracted). In the label of the vertical axis, $g = 9.81\,\mathrm{m/s^2}$ denotes the gravitational acceleration.

[8] Y. Jung, S. Cho, and D. H. Shim, "A comprehensive flight control design and experiment of a tail-sitter UAV," in *AIAA Guidance, Navigation, and Control Conference (GNC)*, 2013.

[9] M. Hochstenbach, C. Notteboom, B. Theys, and J. De Schutter, "Design and control of an unmanned aerial vehicle for autonomous parcel delivery with transition from vertical take-off to forward flight – vertikul, a quadcopter tailsitter," *International Journal of Micro Air Vehicles*, vol. 7, no. 4, pp. 395–405, 2015.

[10] X. Wang, Z. Chen, and Z. Yuan, "Modeling and control of an agile tail-sitter aircraft," *Journal of the Franklin Institute*, vol. 352, no. 12, pp. 5437–5472, 2015.

[11] K. Kita, A. Konno, and M. Uchiyama, "Transition between level flight and hovering of a tail-sitter vertical takeoff and landing aerial robot," *Advanced Robotics*, vol. 24, no. 5-6, pp. 763–781, 2010.

[12] D. Kubo and S. Suzuki, "Transitional flight control of tail-sitter vertical takeoff and landing mini unmanned aerial vehicle," in *AIAA Infotech Aerospace Conference and Exhibit*, 2007.

[13] E. N. Johnson, A. Wu, J. C. Neidhoefer, S. K. Kannan, and M. A. Turbe, "Flight-test results of autonomous airplane transitions between steady-level and hovering flight," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 2, 2008.

[14] N. B. Knoebel and T. W. McLain, "Adaptive quaternion control of a miniature tailsitter UAV," in *American Control Conference (ACC)*. IEEE, 2008, pp. 2340–2345.

[15] V. Martinez, O. Garcia, A. Sanchez, V. Parra, and A. Escobar, "Adaptive backstepping control for a convertible UAV," in *Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*. IEEE, 2015, pp. 298–307.

[16] T. Matsumoto, K. Kita, R. Suzuki, A. Oosedo, K. Go, Y. Hoshino, A. Konno, and M. Uchiyama, "A hovering control strategy for a tail-sitter VTOL UAV that increases stability against large disturbance," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 54–59.

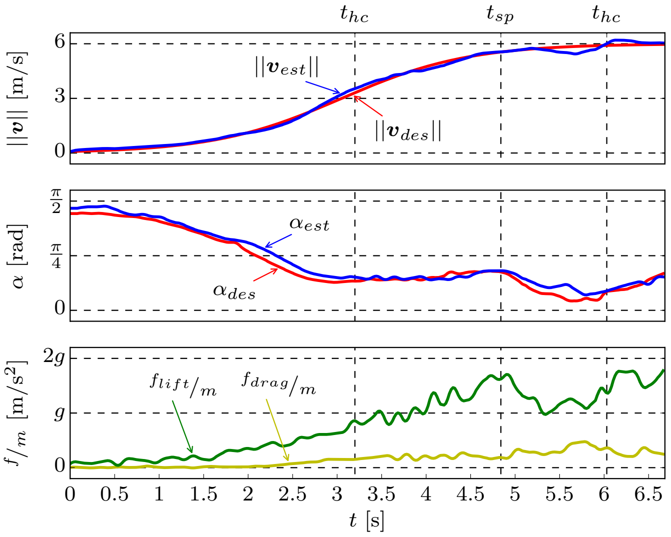

Fig. 7. Velocity, angle of attack, and aerodynamic force trajectories of a constant-height transition from hover to forward flight. Due to space restrictions, the vehicle does not enter a linear forward motion after the transition, but instead starts tracking a figure eight trajectory. The time when the vehicle enters a half-circle or a spline segment is denoted with t_{hc} or t_{sp}, respectively. The plot on the top shows desired and estimated total velocity increasing from $0\,\mathrm{m/s}$ to $6\,\mathrm{m/s}$. The plot in the middle shows the desired and estimated angle of attack during the transition. The plot on the bottom shows the mass-normalized aerodynamic forces acting on the vehicle, measured by the onboard accelerometer (where propeller forces are subtracted). In the label of the vertical axis, $g = 9.81\,\mathrm{m/s^2}$ denotes the gravitational acceleration. The mass-normalized lift force becomes larger than the gravitational acceleration, because the vehicle performs a turn in order to track the figure eight trajectory.

[17] J. M. Beach, M. E. Argyle, T. W. McLain, R. W. Beard, and S. Morris, "Tailsitter attitude control using resolved tilt-twist," in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2014, pp. 768–779.

[18] S. Verling, B. Weibel, M. Boosfeld, K. Alexis, M. Burri, and R. Siegwart, "Full attitude control of a VTOL tailsitter UAV," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 3006–3012.

[19] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," Stanford University, Tech. Rep., 2006.

[20] W. Johnson, *Helicopter theory*. Courier Corporation, 2012.

[21] G. Chowdhary and R. Jategaonkar, "Aerodynamic parameter estimation from flight data applying extended and unscented kalman filter," *Aerospace science and technology*, vol. 14, no. 2, pp. 106–117, 2010.

[22] S. Lupashin, M. Hehn, M. W. Mueller, A. P. Schoellig, M. Sherback, and R. D'Andrea, "A platform for aerial robotics research and demonstration: The Flying Machine Arena," *Mechatronics*, vol. 24, no. 1, pp. 41–54, 2014.

[23] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[24] R. Ritz and R. D'Andrea, "A global strategy for tailsitter hover control," in *International Symposium on Robotics Research (ISRR)*, 2015.