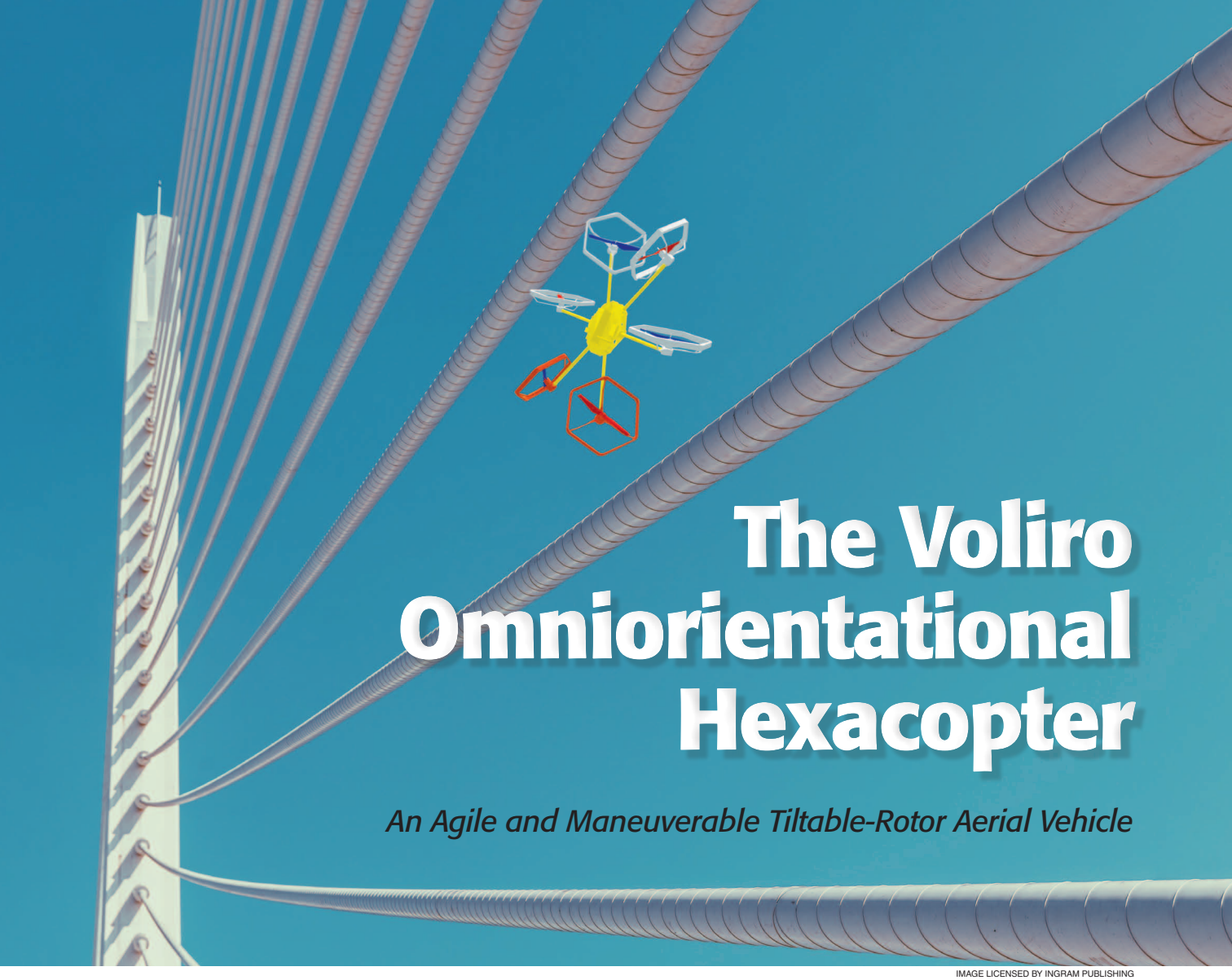

The Voliro Omniorientational Hexacopter

An Agile and Maneuverable Tiltable-Rotor Aerial Vehicle

By Mina Kamel, Sebastian Verling, Omar Elkhatib, Christian Sprecher, Paula Wulkop, Zachary Taylor, Roland Siegwart, and Igor Gilitschenski

Extending the maneuverability of multirotors promises to yield a considerable increase in their scope of applications, such as carrying out more challenging inspection tasks and generating complex, uninterrupted movements of an attached camera. In this article, we address the promise of multirotor maneuverability by presenting Voliro, a novel aerial platform that combines the advantages of existing multirotor systems with the agility of vehicles having omniorientational controllability. In other words, Voliro can fly in any direction while maintaining an arbitrary orientation.

Digital Object Identifier 10.1109/MRA.2018.2866758
Date of publication: 8 October 2018

We propose the use of a hexacopter with tiltable rotors, allowing the system to decouple control of position from control of orientation. The contributions of this article include the overall mechanical design as well as the design of a controller with the corresponding allocation scheme. This work also discusses the design challenges involved when turning the concept of a hexacopter with tiltable rotors into an actual prototype. The agility of the system is demonstrated and evaluated using real-world experiments.

Previous Approaches

As autonomous multirotor unmanned aerial vehicles (UAVs) are increasingly deployed in a broad spectrum of real-world robotics applications [1]–[4], there is a growing demand to broaden the scope of their applicability, effectiveness, and

1070-9932/18©2018IEEE

robustness. Due to this demand, considerable attention is focused on improving a multirotor's agility and enabling aerial manipulation. Advances in these areas may unlock deployment in numerous use cases. For instance, drone-based inspection of bridges [5] may require the aircraft to collect measurements with a sensor that must remain in contact with the bridge structure at all times. This requires the drone to adapt its own orientation to the bridge's geometry while controlling its position independently. Enabling a drone to reach and hold any possible pose also gives rise to many further applications, such as enabling uninterrupted, complex camera motions for film shots [6]. Moreover, physical interaction with the environment while airborne opens the door for various applications, such as contact-based inspection [7], surface cleaning, and sensor deployment [8].

These are only a sampling of the many possible applications that would benefit if the limits imposed by the dynamics of most commercially available multirotor UAVs can be overcome. The demand for and interest in novel multirotor models are reflected in numerous research projects. In [9], a small vertical takeoff and landing platform was proposed. The system begins its flight as a multirotor, before transitioning into a fixed-wing mode for comparatively long-endurance surveillance and inspection flights. The handling of strong winds for platforms of this size was investigated in [10], where a fixed-wing plane was used. This model comes at the cost of lower agility compared to a classical multirotor. The Omnicopter presented in [11] uses intelligent rotor placement in a cube-like structure to generate forces and torques in any direction. It has the downside of counteracting forces that can reduce the system's efficiency and hence its flight time; in addition, certain rotors have only limited usefulness in a given configuration.

In [12] and [13], the Laboratory for Analysis and Architecture of Systems group at the Centre National de la Recherche Scientifique demonstrated the advantages of omniorientational aerial platforms for physical interaction. These systems have fixed tilted rotors and can apply forces and torques in all directions. They have been used for aerial physical interaction and show a high degree of robustness and dexterity. A disadvantage of these systems is the constant presence of internal forces, which significantly reduces efficiency.

Despite these systems' ability to apply forces and torques in all directions, they are limited to a set of possible orientations due to limited forces achievable along the body frame's x and y directions. The use of tilting rotors promises to address some of these challenges.

A concept for and simulation of a quadcopter involving tilting rotors were presented in [14] and experimentally evaluated in [15]. More approaches were evaluated in [16] and [17]. While improving upon maneuverability, these approaches do not offer omniorientational flight capabilities. The work in [18] also involved tilting rotors but considered only two distinct flight modes: one in a horizontal orientation and the other vertical. The design concept presented in [19] is, as far as we are aware, the most similar to that of the proposed Voliro platform. However, this research considered only control of orientation and did not propose any approach for jointly controlling position and orientation. A model-based control approach for a quadrotor with tiltable rotors was presented in [20]. However, knowledge of the system and actuators model is required to achieve high-performance control.

Physical interaction with the environment while airborne opens the door for various applications, such as contact-based inspection, surface cleaning, and sensor deployment.

While the previously discussed works pushed the boundaries of multirotor agility, no tilting rotor system showed the ability to stably achieve any configuration on SE(3). Our goal is to overcome the limitations of common, commercially available multirotor UAVs while preserving their advantages. The main conceptual challenges in achieving this are the mechanical design and control of the system. To be successful, the mechanical design must address three challenges:

1) incorporating the hardware that enables additional maneuverability while keeping the weight low
2) placing the propellers such that they can contribute to the system in any configuration and generate as little counteracting force as possible
3) achieving a safe design, such that a tool can be easily mounted and the drone operated close to a given surface.

These three challenges further translate into hurdles for designing the controller.

In this article, we present Voliro, a hexacopter with tiltable rotors. As the rotor orientation can be fully controlled, this system allows the control of position and orientation to be decoupled. It can be easily manipulated through a proportional-integral-derivative (PID) controller and a simple allocation scheme, which translates the control output into motor configurations. Moreover, the ability of the system to independently orient the thrust generated from each rotor reduces the internal forces and, therefore, increases the efficiency in comparison to the fixed tilted-rotor platforms described previously. A prototype system is demonstrated and controlled in a wide range of configurations using a Vicon system to provide state estimation.

The contributions of this article can be summarized as follows:

- To the best of our knowledge, we present the first hexacopter with tiltable rotors.
- We discuss and justify the UAV's mechanical design and the corresponding control scheme.
- We evaluate the concept in real-world experiments on a prototype platform.

System Description

Mechanical Design

The prototype presented in this article has six rotors arranged evenly in a circle (Figure 1) in a traditional helicopter configuration. The choice of six rotors represents a compromise between simplicity and the ability to generate sufficient thrust to compensate for gravity in any configuration on SE(3). Clearly, the hexacopter configuration has a larger wrench envelope compared to a quadcopter. The rotor arms are constructed of carbon-fiber tubes, and each rotor can rotate around the arm's axis. Only the motor and its housing rotate. This rotation is achieved using brushless dc motors inserted at the end of the arms, as can be seen in Figure 2. The arms themselves are rigidly attached to the platform's main frame, which gives the system increased structural rigidity compared to designs that rotate the entire arm.

The ability of the system to independently orient the thrust generated from each rotor reduces the internal forces and, therefore, increases the efficiency.

The tilting motors are 12 mm in diameter and equipped with three Hall sensors and a high-ratio gearbox for precise position estimation and control. The tilting motors have a maximum rotation speed of 7.85 rad/s and a maximum torque of 0.5 N·m. This allows for fast control of the tilting angles. The decision to use this motor setup over radio control (RC) servo motors was made because such servos are typically limited to 180° of rotation, which would limit the platform's flight envelope. A further advantage of the brushless motors is their narrow profile, which allows them to be hidden inside the aircraft's tubular arms. With this setup, the rotors are capable of rotating 720° around their axes. The limiting factor in this design is the winding of the thrust motor cables around the arms.

Hardware

The main electronic and mechanical components of the system are as follows. For thrust generation, 9-in DJI propellers and KDE2315XF-885 motors were used. This propulsion system is capable of providing a maximum of 13.7 N of thrust per motor. The system requires a high thrust-to-weight ratio so that it can hover at large inclination angles, in cases where most rotors cannot orient their force vector to directly oppose gravity. Faulhaber 1226S012B1 brushless dc motors were used in the rotor-tilting system. Each of these tilting motors requires a motion controller that measures the current position of the motor shaft and applies the required voltage to control its position. To power the motors and the electronic components at different voltages, a six-cell lithium–polymer battery and accompanying power distribution boards were used. Table 1 shows the full list of the electronic components incorporated in this platform.

To house these electronics, a special core was designed using customized parts. The core consists of two 1.5-mm-thick carbon-fiber plates, assembled using aluminum screws. Aluminum spacers keep the plates a fixed distance apart to provide room for the electronics. They also act to clamp the carbon tubes on which the rotor units are mounted. Inside the

Figure 1. The final version of the Voliro platform.

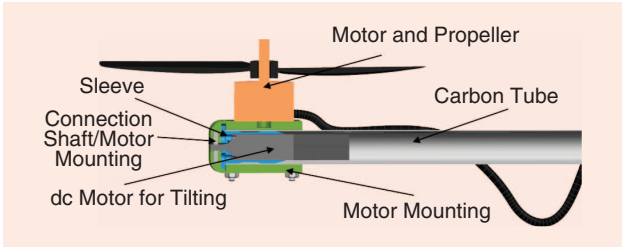

Figure 2. A section view of a rotor unit.

Table 1. The Voliro's components and their specific names and models.

Component	Name
Thrust motors	KDE2315XF-885
ESC	KDEXF-UAS35
Tilt motor	1226S012B1
Motion control board	MCBL 3002
Battery	Swaytronic LiPo 6S
PDB	Piko PDB
BEC 12 V	Micro BEC
BEC 5 V	Reely UBEC
Onboard computer	UP board
Flight controller	Pixhawk
RC receiver	Futaba R3008SB
Wi-Fi antenna	Dlink DWA-1725 GHz

Note: ESC: electronics speed controller; PDB: power distribution board; BEC: battery eliminator circuit.

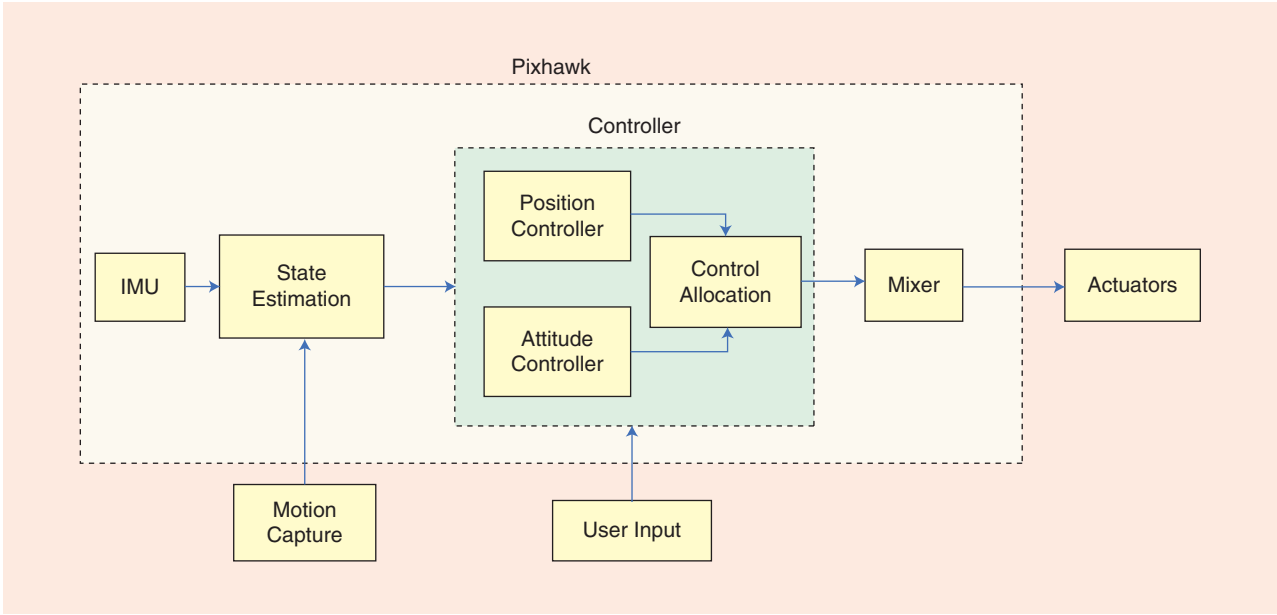

Figure 3. The flight controller structure.

core are a Pixhawk flight controller, a UP board-embedded computer that provides additional computational power, and the motion-control boards for the tilting motors.

The presented system has a total weight of 3.2 kg and a flight time of around 8 min in its horizontal orientation. This significantly decreases if the system is required to hover at a large inclination angle.

Flight Controller

The Pixhawk flight controller (Figure 3) used on this platform consists of an inertial measurement unit (IMU), a magnetometer, a barometer, and a Cortex-M4F microprocessor. The Pixhawk runs the PX4 software [21], which handles full control of the multirotor as well as the interfaces with other devices. It also provides a flexible modular framework that allows the integration of new control schemes.

Using the built-in state estimation provided by the PX4 software, the sensor data obtained by the IMU and magnetometer are fused together with the external pose information from a Vicon motion-capture system (or, in the case of outdoor flight, the position information from a global navigation satellite system) to provide an estimate of the system's full pose. This information enters the controller block together with the desired pose trajectory, which is generated by the user on an external computer.

This desired trajectory is then sent to the UP board via Wi-Fi and ultimately to the Pixhawk through a serial connection. The controller block contains both the position and attitude controller, as well as an allocation block that maps the desired forces and moments onto the 12 actuators. This is crucial to the decoupling of position and orientation. After the desired control inputs are calculated, they are fed into the mixer block, which maps these values to the actuator pulse-width modulation signals.

Modeling

Developing a system model for such a vehicle is a necessary step for enabling model-based control synthesis and thorough testing. To fulfill this task, a methodology was used that combines rigid-body dynamics with well-established aerodynamic modeling techniques.

Coordinates and Conventions

Here, we provide an overview of the coordinate systems and notational conventions used subsequently.

Coordinate Frames

Overall, eight coordinate frames were used. The inertial frame $\mathcal{F}_I$ is fixed on the ground, and its z axis points upward. In contrast, the z axis of the body frame $\mathcal{F}_B$ points downward, as seen in Figure 4; its origin is at the center of gravity. Finally, there are the six coordinate frames of the rotor units $\mathcal{F}_{R,i}$,

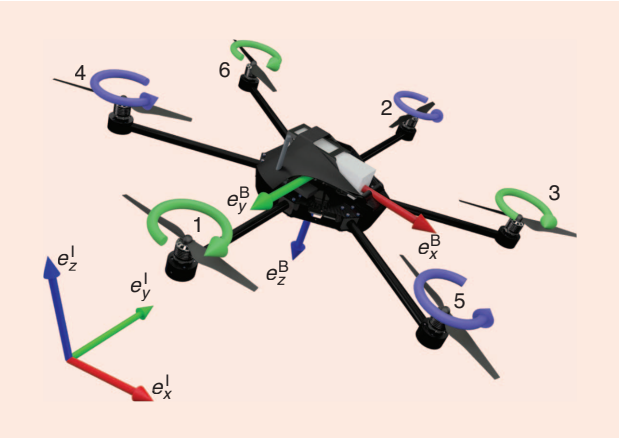

Figure 4. The inertial and body coordinate system.

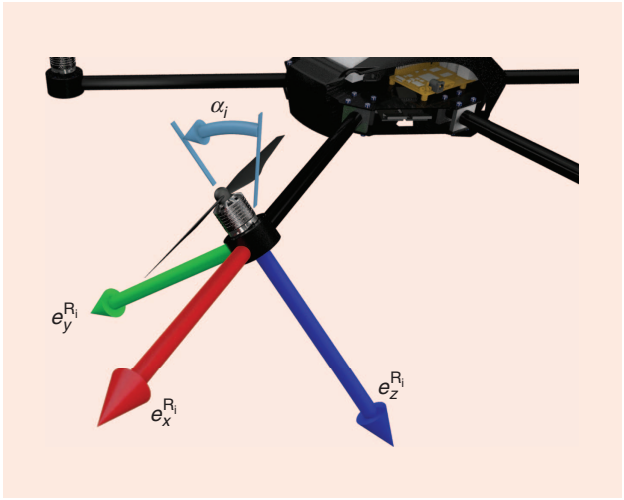

Figure 5. The rotor coordinate system.

shown in Figure 5. Their origins are at the center of the rotor blades, and their x axes are aligned with the axes to the center of mass of the body. These coordinate frames rotate around their x axes with respect to the body frame.

Notation

Throughout this article, we use left-hand subscripts for the coordinate system in which a symbol is referenced. For example, ${}_I\boldsymbol{r}$ indicates a vector $\boldsymbol{r}$ represented in coordinate system $\mathcal{F}_I$. The matrix $\boldsymbol{R}_{AB}$ stands for the rotation matrix that rotates a vector represented in coordinate system $\mathcal{F}_B$ to the coordinate system $\mathcal{F}_A$, i.e., ${}_A\boldsymbol{r} = \boldsymbol{R}_{AB} \cdot {}_B\boldsymbol{r}$. The estimate of a state x will be denoted with a hat on top of the symbol: $\hat{x}$.

Assumptions

For the sake of model simplicity, the following assumptions were made:

- The body structure is rigid and symmetric.
- The rotors are rigid and on the same height as the center of gravity, and they rotate around an axis that passes through the center of gravity.
- The thrust and drag torques are proportional to the square of the rotor speed.
- The linear and angular velocities of the body are small.
- The dynamics of the tilting motors are independent of the rotational speed of the rotors.
- The thrust and drag torques produced by one rotor are independent of the thrust and drag torques of the other rotors, i.e., the resulting air flows do not produce interference.

While most of these assumptions will cause little or no deviation of the flight characteristics of the real-world system from those of the derived system model, the final assumption on this list, which ignores rotor interference, is likely to have a significant impact. However, in our experience and flight tests, we have found that, with a well-tuned controller, this influence can be treated as an unmodeled disturbance.

Rigid Body Model

For the modeling of the body dynamics, the Newton–Euler formalism was used. Its general form is the following:

$$\begin{bmatrix} m\mathcal{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} {}_B\dot{\boldsymbol{v}} \\ {}_B\dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} {}_B\boldsymbol{\omega} \times (m \cdot {}_B\boldsymbol{v}) \\ {}_B\boldsymbol{\omega} \times (I \cdot {}_B\boldsymbol{\omega}) \end{bmatrix} = \begin{bmatrix} {}_B\boldsymbol{F} \\ {}_B\boldsymbol{M} \end{bmatrix}, \tag{1}$$

with $\mathcal{I}_3$ being the three-dimensional identity matrix and $\boldsymbol{J}$ the moment of inertia of the system.

Motor Dynamics

The dynamics of the rotors' dc motors are modeled as a first-order system,

$$\dot{n}_i = \frac{1}{\tau_n}(n_{i,\text{des}} - n_i), \tag{2}$$

with $n_{i,\text{des}}$ being the desired angular velocity and τ_n the motor's time constant. Similarly, the closed-loop tilting velocity $\omega_{\alpha_i} = \dot{\alpha}_i$ dynamics are modeled as a first-order system, with time constant τ_α. Moreover, the tilting velocity is saturated to model the maximum possible tilting velocity as follows:

$$|\omega_{\alpha i}| \leq \omega_{\alpha,\max}, \tag{3}$$

where $\omega_{\alpha,\max}$ is the maximum possible tilting velocity.

Aerodynamics

The aerodynamics modeling transfers the actual angular velocities n and tilting angles α into the forces $\boldsymbol{F}$ and moments $\boldsymbol{M}$ acting on the center of gravity of the body. Given the stated assumptions of low body velocities, we can neglect the aerodynamic drag of the main body and assume that the thrust and reaction torques are proportional to the square of the rotational velocity of the rotors. The force and torque produced by one rotor are given by

$$F = \mu n^2, \tag{4}$$

$$\tau = \kappa n^2, \tag{5}$$

with μ being the lift force coefficient and κ the drag torque coefficient. These forces and moments are produced in the z direction of the rotor's coordinate frames. The forces have a negative sign because the z axis points downward.

The forces and moments are then rotated into the body frame according to

$${}_{R_i}\boldsymbol{F}_i = -\mu n_i^2 \cdot {}_{R_i}\boldsymbol{e}_z, \tag{6}$$

$${}_{R_i}\boldsymbol{\tau}_i = -c_i \kappa n_i^2 \cdot {}_{R_i}\boldsymbol{e}_z, \tag{7}$$

with $c_i = 1$ for $i \in \{1, 3, 6\}$ and -1 otherwise. Furthermore,

$${}_B\boldsymbol{F} = \sum_i {}_B\boldsymbol{F}_i = \sum_i \boldsymbol{R}_{BR_i} \cdot {}_{Ri}\boldsymbol{F}_i, \tag{8}$$

$$\begin{aligned} {}_B\boldsymbol{M} &= \sum_i {}_B\boldsymbol{\tau}_i + \sum_i {}_{R_i}\boldsymbol{M}_{\text{ind},i}, \\ &= \sum_i \boldsymbol{R}_{BR_i} \cdot {}_{R_i}\boldsymbol{\tau}_i + {}_B\boldsymbol{r}_i \times {}_B\boldsymbol{F}_i, \end{aligned} \tag{9}$$

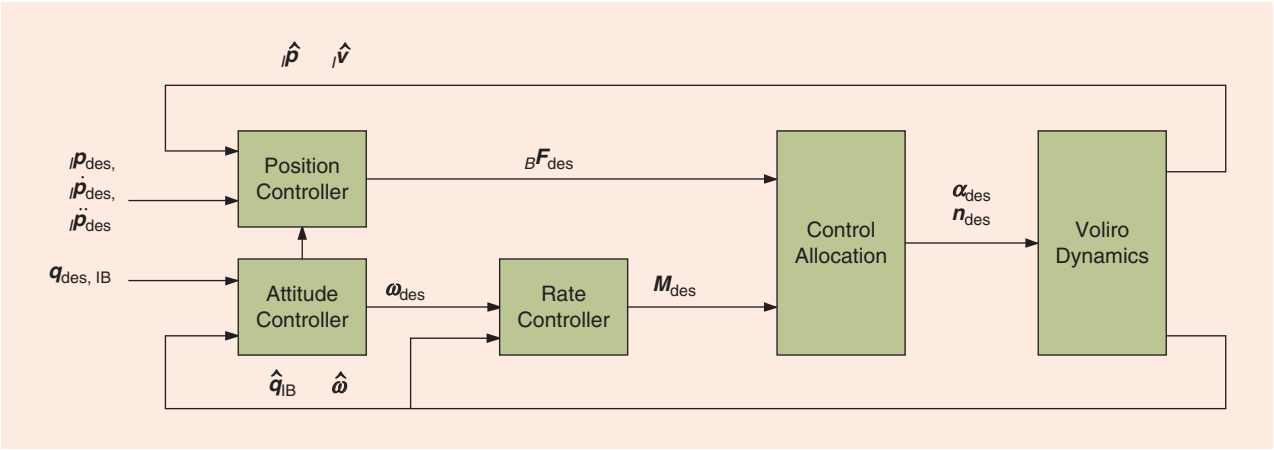

Figure 6. The control structure of the Voliro platform. The controller is running on board the Pixhawk flight controller.

where $\boldsymbol{R}_{BR_i}$ is the rotation matrix of the ith rotor frame to the body frame and ${}_B\boldsymbol{r}_i$ is the distance from the center of gravity to the ith rotor. The relation between the forces, moments, tilting angles, and angular velocities can then be described by

$$\begin{pmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{pmatrix} = \boldsymbol{A}(\boldsymbol{\alpha})\boldsymbol{N}, \tag{10}$$

where $N = (n_1^2, n_2^2, \ldots, n_6^2)$ and $\boldsymbol{A}(\boldsymbol{\alpha}) \in \mathbb{R}^{6\times 6}$, as described in (11), shown in the box at the bottom of the page, with $s(\cdot)$ representing the sine and $c(\cdot)$ the cosine. $\boldsymbol{A}(\boldsymbol{\alpha})$ will henceforth be referred to as the allocation matrix. For a conventional quad- or hexacopter, this matrix is static. However, for this system it is a function of the tilting angles α, which is an advantage because the total force vector can be directed in any orientation through the selection of appropriate tilting angles.

In addition, the matrix is size 6×6, as opposed to the 4×6 size used by a conventional hexacopter. This means that the system can produce a desired force in all three axes, instead of just in the z direction. Because the matrix has full rank in most configurations (it is singular in some special cases, such as when all αs are zero and in other, more complex cases), all forces and moments can be produced independently, meaning that translation and rotation of the platform can be controlled independently. The allocation matrix is singular when all tilting angles αi are set to zero, which means that the forces along the x and y axes cannot be generated. This is not a problem, as the proposed control approach explained in the "Control Allocation" section generates a tilting angles command αi and props the angular velocities ni simultaneously.

Control

Control Structure

In this section, we present a baseline controller for the Voliro platform. Given the omnidirectional nature of the platform, it is possible to decouple the position and attitude dynamics, as the actuation forces and torques are independent. Therefore, we consider two separate controllers for position and attitude reference tracking. A cascade control structure is employed to control the attitude, as shown in Figure 6.

The desired forces and torques are achieved by changing the angular velocity and the orientation of each propeller. Therefore, the control allocation problem becomes challenging; as for the 6 degrees-of-freedom (DoF) output, there are 12 control inputs. We here introduce the position and attitude controllers, followed by a novel solution to this control allocation problem.

Position Control

The position control is based on a PID controller with a feedforward term that generates a desired force vector in the vehicle body frame $\mathcal{F}_B$. The position error is defined as the difference between the desired and estimated position ${}_I\boldsymbol{p}_{\text{err}} = {}_I\boldsymbol{p}_{\text{des}} - {}_I\hat{\boldsymbol{p}}$. The position control law is given by

$${}_B\boldsymbol{F}_{\text{des}} = \boldsymbol{R}_{IB}^T\left(k_{p,pI}\boldsymbol{p}_{\text{err}} + k_{p,pI}\dot{\boldsymbol{p}}_{\text{err}} + k_{i,p}\int \boldsymbol{p}_{\text{err}}dt + m\boldsymbol{g} + {}_I\ddot{\boldsymbol{p}}_{\text{des}}\right), \tag{12}$$

where $\boldsymbol{R}_{IB}$ is the rotation matrix between the body frame $\mathcal{F}_B$, and inertial frame $\mathcal{F}_I$, $k_{p,p}, k_{d,p}, k_{i,p}$ are the proportional,

$$\boldsymbol{A}(\boldsymbol{\alpha}) = \begin{bmatrix} -\mu\,\mathrm{s}(\alpha_1) & \mu\,\mathrm{s}(\alpha_2) & \mu\,\frac{1}{2}\mathrm{s}(\alpha_3) & -\mu\,\frac{1}{2}\mathrm{s}(\alpha_4) & -\mu\,\frac{1}{2}\mathrm{s}(\alpha_5) & \mu\,\frac{1}{2}\mathrm{s}(\alpha_6) \\ 0 & 0 & \mu\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_3) & -\mu\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_4) & \mu\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_5) & -\mu\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_6) \\ -\mu\,\mathrm{c}(\alpha_1) & -\mu\,\mathrm{c}(\alpha_2) & -\mu\,\mathrm{c}(\alpha_3) & -\mu\,\mathrm{c}(\alpha_4) & -\mu\,\mathrm{c}(\alpha_5) & -\mu\,\mathrm{c}(\alpha_6) \\ -\mu l\,\mathrm{c}(\alpha_1) - \kappa\,\mathrm{s}(\alpha_1) & \mu l\,\mathrm{c}(\alpha_2) - \kappa\,\mathrm{s}(\alpha_2) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_3) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_3) & -\mu l\,\frac{1}{2}\mathrm{c}(\alpha_4) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_4) & -\mu l\,\frac{1}{2}\mathrm{c}(\alpha_5) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_5) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_6) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_6) \\ 0 & 0 & \mu l\,\frac{\sqrt{3}}{2}\mathrm{c}(\alpha_3) + \kappa\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_3) & -\mu l\,\frac{\sqrt{3}}{2}\mathrm{c}(\alpha_4) + \kappa\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_4) & \mu l\,\frac{\sqrt{3}}{2}\mathrm{c}(\alpha_5) - \kappa\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_5) & -\mu l\,\frac{\sqrt{3}}{2}\mathrm{c}(\alpha_6) - \kappa\,\frac{\sqrt{3}}{2}\mathrm{s}(\alpha_6) \\ \mu l\,\mathrm{c}(\alpha_1) - \kappa\,\mathrm{s}(\alpha_1) & \mu l\,\mathrm{c}(\alpha_2) + \kappa\,\mathrm{s}(\alpha_2) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_3) - \kappa\,\frac{1}{2}\mathrm{s}(\alpha_3) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_4) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_4) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_5) + \kappa\,\frac{1}{2}\mathrm{s}(\alpha_5) & \mu l\,\frac{1}{2}\mathrm{c}(\alpha_6) - \kappa\,\frac{1}{2}\mathrm{s}(\alpha_6) \end{bmatrix} \tag{11}$$

derivative, and integral gains respectively; m is the vehicle mass; and $\boldsymbol{g}$ is the gravity vector.

Attitude Control

The attitude controller consists of two cascade controllers: an outer loop for the attitude control that generates desired body rates and a rate controller that creates desired moments. The attitude controller is based on quaternion error, which is given by

$$\boldsymbol{q}_{\text{err}} = \boldsymbol{q}_{\text{des},IB} \otimes \hat{\boldsymbol{q}}_{IB}^{*} = \begin{pmatrix} q_{w,\text{err}} \\ \boldsymbol{q}_{v,\text{err}} \end{pmatrix}. \tag{13}$$

The desired body rate $\boldsymbol{\omega}_{\text{des}}$ is generated from the vector part of the quaternion error $\boldsymbol{q}_{v,\text{err}}$ as follows:

$$\boldsymbol{\omega}_{\text{des}} = k_q \,\text{sign}\,(q_{w,\text{err}})\, \boldsymbol{q}_{v,\text{err}}, \tag{14}$$

where k_q is a tuning parameter. Note that the sign of the real part of the quaternion error is used to avoid the unwinding phenomenon.

The desired moments $\boldsymbol{M}_{\text{des}}$ are computed as

$$\boldsymbol{M}_{\text{des}} = k_r\,(\boldsymbol{\omega}_{\text{des}} - \hat{\boldsymbol{\omega}}) - \boldsymbol{r}_{\text{off}} \times {}_B\boldsymbol{F}_{\text{des}} + \hat{\boldsymbol{\omega}} \times \boldsymbol{J}\hat{\boldsymbol{\omega}}, \tag{15}$$

where k_r is the rate controller gain, $\hat{\boldsymbol{\omega}}$ is the estimated angular velocity, $\boldsymbol{r}_{\text{off}}$ is the center of mass offset, and $\boldsymbol{J}$ is the vehicle's inertia matrix.

Control Allocation

The control allocation problem deals with finding rotor speeds $n_1, \ldots, n_6$ and rotor angular positions $\boldsymbol{\alpha}$ to satisfy (10). This control allocation problem has been studied in detail in [22] for marine vessels, while a general survey about control allocation can be found in [23]. Solving this equation is challenging for two reasons. First, any solution is not unique because of the overactuated nature of the platform. Second, the rotor angular positions appear in a nonlinear fashion in the allocation matrix (11).

One approach to solve this problem would be to perform a nonlinear least-squares optimization. However, the computation time required and the available computation resources on board the platform inhibit such a solution. Another approach is to use a nonlinear model- predictive attitude controller [24] that generates near-optimal rotor-speed and tilting-angle commands. This method is appealing because it can easily account for tilting-angle dynamics, but it can be computationally expensive. Note that this allocation needs to be solved at a high rate to allow the system to perform in an agile manner.

Here, we propose a means to transform the nonlinear allocation problem into a linear problem through variable transformation. We show that this solution also satisfies the constraint of a positive rotor speed. To this end, we decompose the force generated by each rotor, as shown in (4), along the vertical and lateral axes of the rotor. We define the vertical and lateral forces of the ith rotor as

$$F_{v,i} = \mu n_i^2 \cos\alpha_i\,, \tag{16}$$

$$F_{l,i} = \mu n_i^2 \sin\alpha_i\,. \tag{17}$$

We also define the vector of dimension 12×1 of all of the vertical and lateral forces as

$$\boldsymbol{F}_{\text{dec}} = (F_{v,1}, F_{l,1}, \ldots, F_{v,6}, F_{l,6})^{\top}. \tag{18}$$

By rearranging (10), we obtain

$$\begin{pmatrix} \boldsymbol{F} \\ \boldsymbol{M} \end{pmatrix} = \boldsymbol{A}_{\text{static}} \boldsymbol{F}_{\text{dec}}, \tag{19}$$

where $\boldsymbol{A}_{\text{static}}$ is a static allocation matrix of dimensions 6×12 that does not depend on the rotor orientation $\boldsymbol{\alpha}$. At this point, we can easily invert (19) to calculate $\boldsymbol{F}_{\text{dec}}$ from a desired wrench $(\boldsymbol{F}^{\top}, \boldsymbol{M}^{\top})^{\top}$. We use the Moore–Penrose pseudoinverse of $\boldsymbol{A}_{\text{static}}$ to calculate $\boldsymbol{F}_{\text{dec}}$ as follows:

$$\boldsymbol{F}_{\text{dec}} = \boldsymbol{A}_{\text{static}}^{\dagger} \begin{pmatrix} \boldsymbol{F}_{\text{des}} \\ \boldsymbol{M}_{\text{des}} \end{pmatrix}. \tag{20}$$

Because the system is underdetermined, the Moore–Penrose pseudoinverse is the minimum norm solution of (19). It is straightforward to show that the norm of $\boldsymbol{F}_{\text{dec}}$ is proportional to the sum of the fourth power of the rotor speed, as

$$\begin{aligned} \|\boldsymbol{F}_{\text{dec}}\|^2 &= \mu^2 \sum_{i=1}^{6} (n_i^2 \cos\alpha_i)^2 + (n_i^2 \sin\alpha_i)^2 \\ &= \mu^2 \sum_{i=1}^{6} n_i^4. \end{aligned} \tag{21}$$

This minimization leads to more consistent and equally distributed angular velocities as well as to a reduction of power consumption, all of which are desirable.

Now, obtaining the actual rotor speed n_i and rotor orientation αi for each rotor is straightforward and can be accomplished by solving the system of equations in (16). We have

$$n_i^2 = \frac{1}{\mu}\sqrt{F_{v,i}^2 + F_{l,i}^2}, \tag{22}$$

$$\alpha_i = a\tan 2\,(F_{l,i}, F_{v,i})\,. \tag{23}$$

With this approach, the calculation of motor speeds and tilting angles can be performed by the simple multiplication of a matrix by a vector. This allows it to be calculated at a rate of several hundred hertz on the small microcontroller found in the Pixhawk. It must be noted, however, that allocating the actuator commands using the pseudoinverse may result in inadmissible rotor speeds that are higher than physically possible. This may occur even if the desired forces and moments lie in the set of attainable forces and moments. One approach to overcome this is the exploitation of the null space of the pseudoinverse. Exploration of this remains for further analysis.

There is one major disadvantage of this allocation method, i.e., that the tilting angles and the angular velocities are

nonlinearly coupled. Therefore, it is difficult to constrain them separately. This means that this allocation does not account for the slower dynamics of the tilting angles. An example of a situation in which this becomes critical is hovering at a roll angle of 90°. In this configuration, one of the three rotor axes is completely vertical. Therefore, the two thrust motors on this axis can produce a force only in the horizontal direction. Hence, they are not used for counteracting gravity but for disturbance rejection and horizontal position stabilization.

These two motors are required to constantly rotate 180° back and forth at a high rate to provide the desired counteracting moments and forces. The dynamics of the tilting motors are, however, much slower than those of the thrust motors and, as a result, cannot keep up with the desired allocation commands. This leads to forces and moments being produced in directions that have not been commanded. To overcome this, the allocation was modified to exclude these two motors in this particular configuration. In this configuration, the system flies with the four remaining motors, which still allow it to control all 6 DoF. This is further explained in [25].

Evaluation

Simulation

The controller was tested in simulation, which used Gazebo [26] as a physics environment and to model the tilting motors. To simulate the modeled sensors and the thrust motors, the RotorS plugin [27] was used. The sensor data were sent via the MAVLink protocol (pixhawk.org/dev/mavlink) to the software-in-the-loop version of the PX4 software. The model of the system was then loaded into this environment (Figure 7). Similar to the actual system, the commands were sent via the robot operating system to the controller.

Experimental Results

In this section, we present our experimental results. The position and orientation of the vehicle were determined with an external motion-capture system, sent to the flight controller at 10 Hz, and fused with the onboard IMU. To showcase the vehicle's capabilities, two different maneuvers were demonstrated. Both maneuvers are not feasible with a standard multirotor. Moreover, we show experimental results in which the Voliro platform was able to interact with and move along a wall.

Free-Flight Experiments

The first experiments show the rotation of the vehicle around the body's y axis from its initial horizontal orientation to upside-down and back (Figure 8). During this maneuver, the other desired rotations and positions were set to zero. The results of this experiment are shown in Figure 9. This maneuver was conducted slowly to display the vehicle's capability to stabilize in all of the rotations around the y axis.

The second experiment displays a horizontal translation in the x and y directions. The desired rotation around the x axis was set constant to 50°, while the other rotations were set to zero. The results are shown in Figure 10.

Wall Interaction Experiments

To enable physical interaction with the wall, a three-sphere module that can roll passively was mounted on the Voliro platform, as shown in Figure 11. The module was passively compliant to improve interaction stability and reduce oscillation during contact. The phases to achieve physical interaction with the wall can be summarized as follows:

1) transition from horizontal flight to vertical flight, which is achieved at pitch 90°, as shown in Figure 12(a)
2) approaching the wall while maintaining a pitch angle of 90° until contact is established with the compliant three-sphere module; this phase is shown in Figure 12(b)
3) driving on the wall is achieved by generating a force vector using a simple proportional controller; tracking a circle on the wall is shown in Figure 12(c).

More details about wall interaction control and experiments are available in [28].

The experiments demonstrated the vehicle's omnidirectionality. It was able to obtain all orientations along one rotation axis and perform translations at an inclined orientation. However, the controller was not able to counteract all of the system's translational and rotational dynamics,

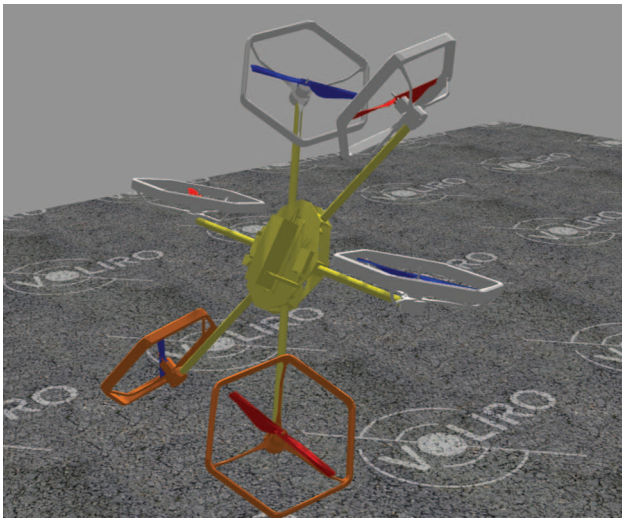

Figure 7. The model of the system in the simulation environment.

Figure 8. Snapshots of the system's rotation from horizontal to upside-down flight.

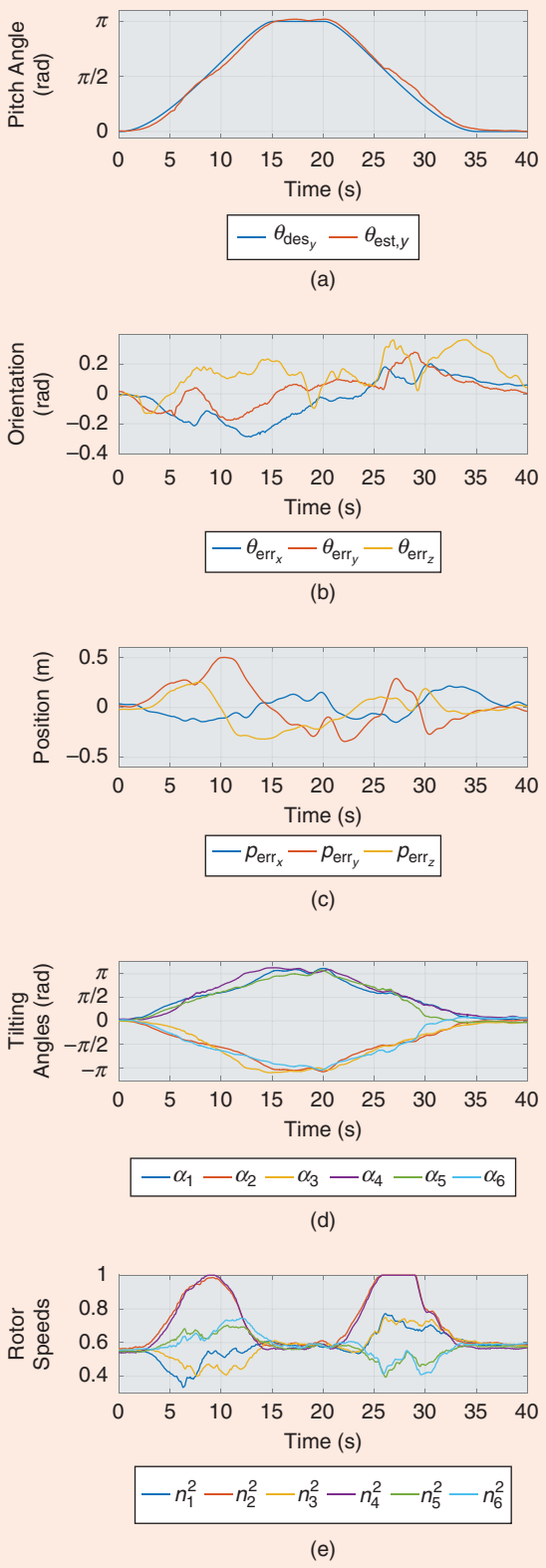

Figure 9. The (a) pitch angle, (b) orientation, (c) position, (d) tilting angles, and (e) rotor speeds during the Voliro's transition to upside-down flight. The x axis of all of the plots displays the time in seconds. The orientation is given in zxy-Euler angles $\bar{\theta}$. The squared rotor speeds are scaled, with 1 representing the maximal thrust produced by a rotor.

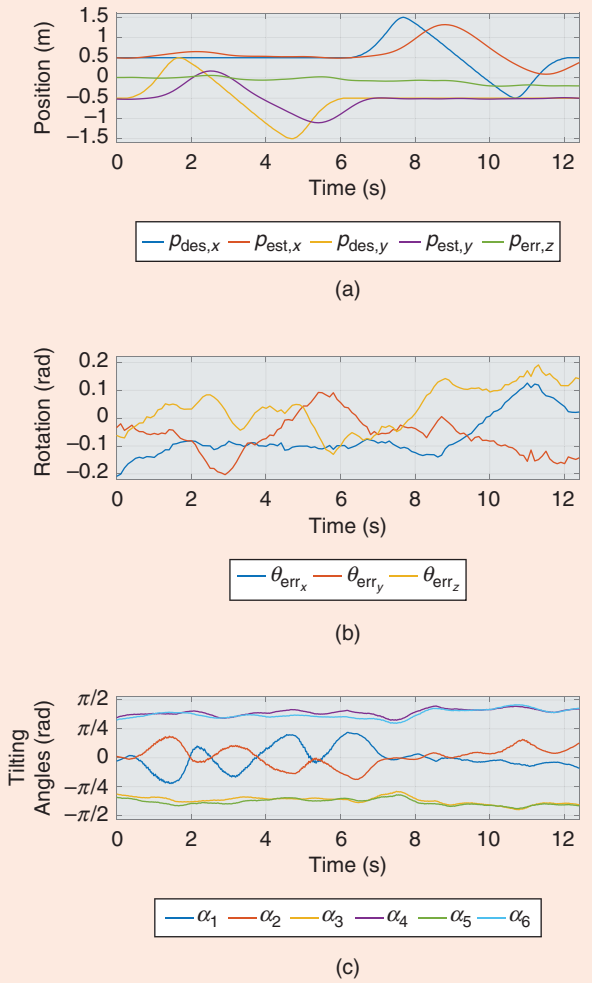

Figure 10. The (a) position, (b) rotation, and (c) tilting angles of the Voliro UAV during the horizontal plus maneuver. The orientation is given in zxy-Euler angles $\bar{\theta}$. The desired rotation angle around the x axis is 50°.

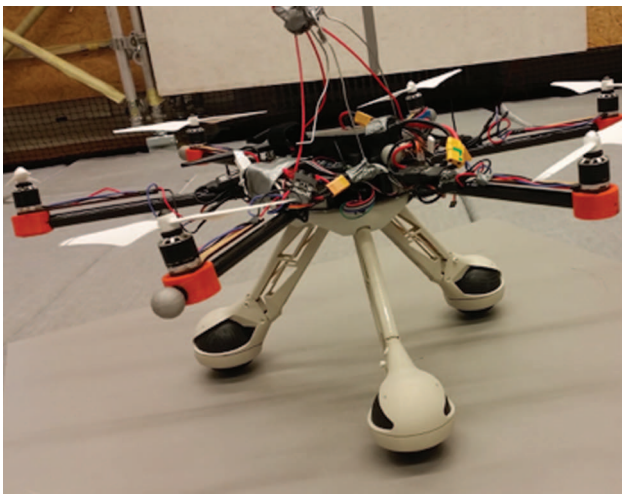

Figure 11. The three-sphere module mounted on the Voliro prototype.

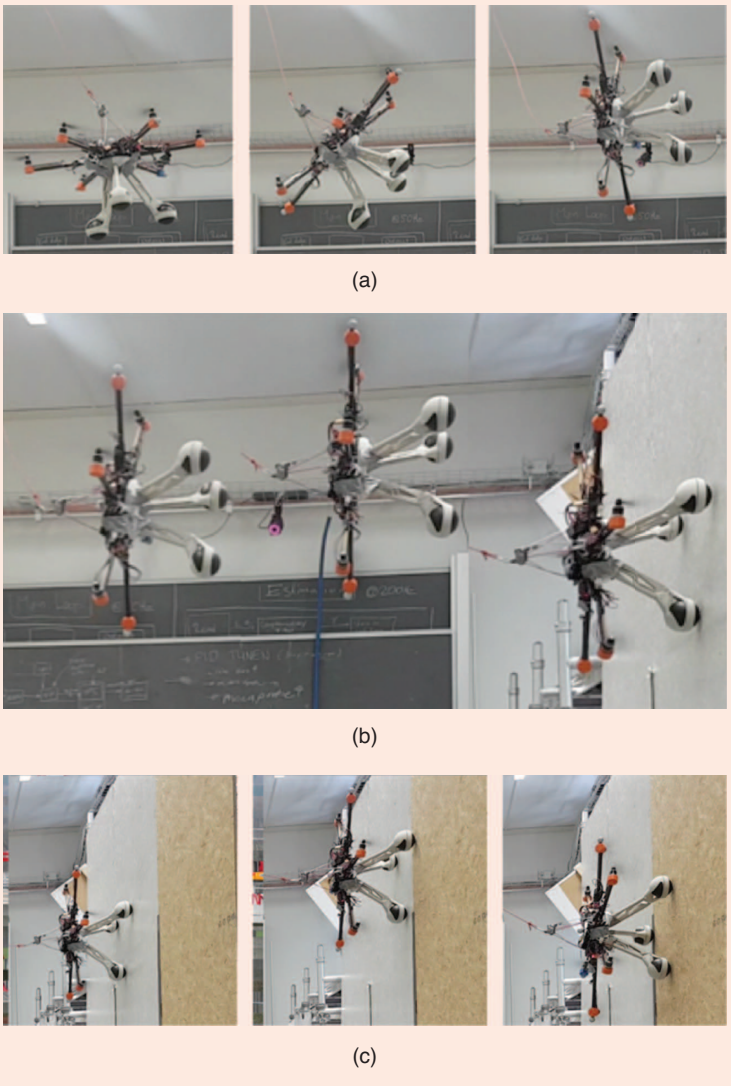

Figure 12. The Voliro UAV during its wall interaction. (a) A vertical orientation is achieved to align the three-sphere module with the wall. (b) The wall is approached while maintaining the aforementioned orientation. (c) The platform is able to establish stable contact with the wall and move in any direction while maintaining force against the wall.

such that the position and the orientation were still slightly coupled. This can be partially explained by the slower dynamics of the tilting motors when compared to the thrust motors. The experiments showed that the roll and pitch angles were tracked better than the position. This occurred because the rotation dynamics are controlled mainly by changing the rotational speeds of the rotors and using their fast dynamics. Deviations in position and yaw are generally corrected by the tilting of the rotors and are more sensitive to their slow dynamics.

While flying at the pitch angle of 90°, the allocation demanded inadmissible rotor speeds, leading to two thrust motors becoming saturated. This is further discussed in the "Control Allocation" section. Nevertheless, the system was still able to track the desired position and orientation.

Discussion and Conclusions

Throughout the development of the Voliro system, a range of issues were encountered on which we are basing our development of future platforms. While building the tilt motors into the arm tips has a range of advantages, it also has the following practical limitations:

1) Significant backlash occurs as the parts wear due to the difficulty of maintaining a tight fitting.
2) Accessing and replacing the internal motors is challenging.
3) The limited airflow can cause overheating issues.
4) The overall system is difficult to repair, as the forces from a crash can often be transmitted down the arms to the internal components.
5) The thrust-to-weight ratio of the system is marginal when hovering in its least efficient configurations, limiting its ability to carry payloads.

In current follow-up designs, these issues are being addressed by the design of arms that employ dual-thrust motors and are driven by servos in the base through a simple clutch. On the software side, as previously mentioned, some problems can be caused by slow rotor-tilting dynamics when the system hovers at 90°.

Overall, however, we demonstrated how the basic idea of a hexacopter with tiltable rotors can be used to achieve omniorientational maneuverability. At the same time, we avoid wasting energy on the generation of counteracting forces, which is an issue that inhibits the use of fixed-orientation rotors in omnidirectional designs. We presented the mechanical design of the platform and a compact tiltable rotor. The employed control allocation technique has the advantage of limiting the computational effort required to compute tilting angles and rotor speeds.

In various experiments, we demonstrated a transition from horizontal to upside-down flight and physical interaction with a wall. While the increased maneuverability gives rise to a broader scope of applications, some of them may require a powering tether to overcome the limitations imposed by battery life. In conclusion, the tiltable-rotor multirotor showed great capabilities and can push the boundaries of what is currently achievable by standard multirotors.

Acknowledgments

We would like to acknowledge Philipp Andermatt, Cliff Li, Alexis Müller, Kamil Ritz, and Kevin Schneider, who were

part of the Voliro focus project team and contributed in numerous ways to the development and deployment of the Voliro prototypes.

References

[1] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon path planning for 3D exploration and surface inspection," *Auton. Robots*, pp. 1–16. 2016.

[2] A. Gawel, M. Kamel, T. Novkovic, J. Widauer, D. Schindler, B. P. von Altishofen, R. Siegwart, and J. Nieto, "Aerial picking and delivery of magnetic objects with MAVs," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2017, pp. 5746–5752.

[3] K. Steich, M. Kamel, P. Beardsleys, M. K. Obrist, R. Siegwart, and T. Lachat, "Tree cavity inspection using aerial robots," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2016.

[4] A. Girard, A. Howell, and J. Hedrick, "Border patrol and surveillance missions using multiple unmanned air vehicles," in *Proc. 43rd IEEE Conf. Decision and Control*, 2004.

[5] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, "Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots," *Auton. Robots*, vol. 40, no. 6, pp. 1059–1078, 2016.

[6] T. Nägeli, L. Meier, A. Domahidi, J. Alonso-Mora, and O. Hilliges, "Real-time planning for automated multi-view drone cinematography," *ACM Trans. Graph. (TOG)*, vol. 36, no. 4, 2017. doi: 10.1145/3072959.3073712.

[7] M. Fumagalli, R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli, and L. Marconi, "Modeling and control of a flying robot for contact inspection," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2012, pp. 3532–3537.

[8] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-UAV cooperation and control for load transportation and deployment," in *Selected Papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*. Berlin: Springer-Verlag, 2009, pp. 417–449.

[9] S. Verling, B. Weibel, M. Boosfeld, K. Alexis, M. Burri, and R. Siegwart, "Full attitude control of a VTOL tailsitter UAV," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2016.

[10] L. Furieri, T. Stastny, L. Marconi, R. Siegwart, and I. Gilitschenski, "Gone with the wind: Nonlinear guidance for small fixed-wing aircraft in arbitrarily strong windfields," in *Proc. American Control Conf.*, 2017.

[11] D. Brescianini and R. D'Andrea, "Design, modeling and control of an omni-directional aerial vehicle," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2016.

[12] N. Staub, D. Bicego, Q. Sablé, V. Arellano, S. Mishra, and A. Franchi, "Towards a flying assistant paradigm: The OTHex," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2018.

[13] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, and A. Franchi, "6d physical interaction with a fully actuated aerial robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2017, pp. 5190–5195.

[14] M. Ryll, H. H. Bulthoff, and P. R. Giordano, "Modeling and control of a quadrotor UAV with tilting propellers," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2012.

[15] M. Ryll, H. H. Bülthoff, and P. R. Giordano, "First flight tests for a quadrotor UAV with tilting propellers," in *Proc. IEEE Int. Conf. Robotics and Automation* , 2013, pp. 295–302.

[16] M. Ryll, D. Bicego, and A. Franchi, "Modeling and control of FAST-Hex: A fully-actuated by synchronized-tilting hexarotor," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, 2016.

[17] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, and A. Franchi, "6D physical interaction with a fully actuated aerial robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2017.

[18] A. Oosedo, S. Abiko, S. Narasaki, A. Kuno, A. Konno, and M. Uchiyama, "Flight control systems of a quad tilt rotor unmanned aerial vehicle for a large attitude change," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2015.

[19] E. Kaufman, K. Caldwell, D. Lee, and T. Lee, "Design and development of a free-floating hexrotor UAV for 6-DOF maneuvers," in *Proc. Aerospace Conf.*, 2014.

[20] G. Scholz and G. F. Trommer, "Model based control of a quadrotor with tiltable rotors," *Gyroscopy Navigation*, vol. 7, no. 1, pp. 72–81, 2016.

[21] L. Meier, D. Honegger, and M. Pollefeys, "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2015.

[22] O. Sørdalen, "Optimal thrust allocation for marine vessels," *Control Eng. Practice*, vol. 5, no. 9, pp. 1223–1231, 1997.

[23] T. A. Johansen and T. I. Fossen, "Control allocation: A Survey," *Automatica*, vol. 49, no. 5, pp. 1087–1103, 2013.

[24] M. Kamel, K. Alexis, M. Achtelik, and R. Siegwart, "Fast nonlinear model predictive control for multicopter attitude tracking on so (3)," in *Proc. IEEE Conf. Control Applications*, 2015, pp. 1160–1166.

[25] O. Elkhatib, "Control allocation of a tilting rotor hexacopter," B.S. thesis, Swiss Federal Institute of Technology, ETH Zürich, 2017.

[26] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2004, vol. 3, pp. 2149–2154.

[27] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Cham, Switzerland: Springer-Verlag, 2016, pp. 595–625.

[28] P. Wulkop, "Development of a controller enabling a tilting rotor hexacopter to move omnidirectionally on a wall," B.S. thesis, Swiss Federal Institute of Technology, ETH Zürich, 2017.

Mina Kamel, Autonomous Systems Lab, ETH Zürich. E-mail: mina.kamel@mavt.ethz.ch.

Sebastian Verling, Autonomous Systems Lab, ETH Zürich. E-mail: sebastian.verling@mavt.ethz.ch.

Omar Elkhatib, Autonomous Systems Lab, ETH Zürich. E-mail: eomar@student.ethz.ch.

Christian Sprecher, Autonomous Systems Lab, ETH Zürich. E-mail: cspreche@student.ethz.ch.

Paula Wulkop, Autonomous Systems Lab, ETH Zürich. E-mail: pwulkop@student.ethz.ch.

Zachary Taylor, Autonomous Systems Lab, ETH Zürich. E-mail: ztaylor@ethz.ch.

Roland Siegwart, Autonomous Systems Lab, ETH Zürich. E-mail: rsiegwart@ethz.ch.

Igor Gilitschenski, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge. E-mail: igilitschenski@mit.edu.