External Wrench Estimation, Collision Detection, and Reflex Reaction for Flying Robots

Teodor Tomić, Christian Ott, and Sami Haddadin, *Member, IEEE*

Abstract—Flying in unknown environments may lead to unforeseen collisions, which may cause serious damage to the robot and/or its environment. In this context, fast and robust collision detection combined with safe reaction is, therefore, essential and may be achieved using external wrench information. Also, deliberate physical interaction requires a control loop designed for such a purpose and may require knowledge of the contact wrench. In principle, the external wrench may be measured or estimated. Whereas measurement poses large demands on sensor equipment, additional weight, and overall system robustness, in this paper we present a novel model-based method for external wrench estimation in flying robots. The algorithm is based on the onboard inertial measurement unit and the robot's dynamics model only. We design admittance and impedance controllers that use this estimate for sensitive and robust physical interaction. Furthermore, the performance of several collision detection and reaction schemes is investigated in order to ensure collision safety. The identified collision location and associated normal vector located on the robot's convex hull may then be used for sensorless tactile sensing. Finally, a low-level collision reflex layer is provided for flying robots when obstacle avoidance fails, also under wind influence. Our experimental and simulation results show evidence that the methodologies are easily implemented and effective in practice.

Index Terms—Collision detection, flying robots, interaction control, reflex reaction.

I. Introduction

A. Motivation

FLYING robots are being increasingly used in cluttered and unknown environments [1], [2]. Autonomous flight therein requires exteroceptive sensors for mapping and collision avoidance, which may fail or be inaccurate due to several reasons such as environment reflectance properties, limited sensor resolution, or a dynamic environment. In fact, e.g., cameras and laser

T. Tomić was with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany. He is now with Skydio, Redwood City, CA 94061 USA (e-mail: teo@skydio.com).

C. Ott is with Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: christian.ott@dlr.de).

S. Haddadin is with Institut für Regelungstechnik, Leibniz Universät Hannover, 30167 Hannover, Germany (e-mail: sami.haddadin@irt.uni-hannover.de).

scanners might see through windows and do not detect them as obstacles. Thin obstacles, such as power lines, might not be sensed due to low sensor resolution. Under these conditions, unforeseen collisions may, hence, occur despite best efforts. Therefore, navigating in such environments requires safe recovery from collisions and contacts. Collision information may also be used as an additional tactile sensor to implement appropriate high-level reaction strategies. A straightforward approach would be to store collision incidents in a map and systematically using this information for replanning around these collided obstacles. The increased availability and autonomy of vision-based flying robots motivates handling of these cases at the control level to achieve robustness in the most extreme failure cases. However, previous work on collision robustness in flying robots has focused on mechanical robustness [3] and contact/force sensors [4]. Furthermore, the reduction in size of such robots will enable new physical interaction modalities, such as being in direct contact with a person.

Interaction and contact scenarios [5]–[9], such as tool force control and contact inspection, are of growing interest and typically require force sensing. This may be provided by a dedicated force sensor, which unfortunately restricts the measurement to the sensor contact point only. Mounting of such a sensor also requires a robust mechanical frame. A method to accurately estimate the general contact wrench would make it possible to apply the existing methods without the need of additional sensors. Using a contact detection algorithm, the robot then could switch in a reflex manner to an appropriate interaction mode.

B. Contribution

This paper presents a unified framework for external wrench estimation, interaction control, and safe collision reaction for flying robots. Fig. 1 shows an overview of the topics covered in this paper. First, we develop estimators of the external forces and torques acting on a flying robot that exploit the control input, state measurements, and a dynamics model. The so-called momentum-based and acceleration-based approaches are presented first. Furthermore, we propose a hybrid estimator that combines both approaches such that a suitable variant is available that is tailored to the needs of flying systems. It uses proprioceptive sensors only, is independent of the trajectory and the used controller, and makes no assumptions about the environment. Subsequently, we use this estimate for controlled physical interaction with the environment. For this purpose, we design admittance and impedance control, which shape the robot's distur-
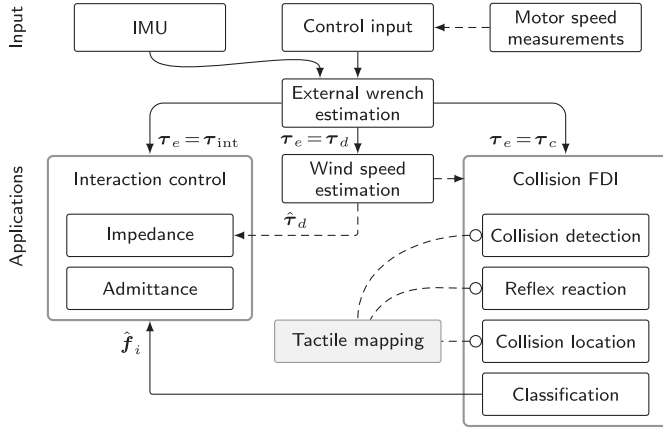
Fig. 1. Overview of the topics covered in this paper. Our external wrench estimator uses the IMU, known control input and a dynamics model. The control input may be obtained from motor feedback and a propulsion model. The external wrench information is then used for interaction control, collision detection, and wind speed estimation. Interaction control assumes that the external wrench $\boldsymbol{\tau}_e$ originates from physical interaction with the environment. Collision FDI assumes that the external wrench is due to the collisions ($\boldsymbol{\tau}_c$) and covers the detection, classification, reflex reaction, and obtaining the collision location (isolation). We discuss how to apply these methods under wind influence—when also the aerodynamic wrench $\boldsymbol{\tau}_d$ is acting on the robot, but refer the reader to references for more details on wind speed estimation itself. Finally, we show how to add a tactile mapping layer to an autonomous flying robot using the described methods.

bance response to external forces. We analyze the stability of the inertia shaping impedance controller together with the dynamics of the external wrench estimator. Finally, we perform the five essential phases of collision fault detection and identification (FDI): detection, identification, isolation, classification, and reaction [10], [11]. We show that our proposed collision reactions based on the identified disturbance wrench improve the robot's behavior upon collisions. The isolated collision location is then used as additional mapping information in an autonomous flight scenario. Together, these algorithms enhance the capabilities of flying robots beyond the current state of the art.

This paper is based on our initial work in [12], with added insights from [13] regarding collision detection under wind influence. In comparison, this work is a more detailed treatise of the subject, adding a stability analysis of the impedance controller with the external wrench estimation dynamics, and with compensation of the aerodynamic wrench. We take inspiration from well-established methods from robot manipulator literature and apply them to underactuated flying robots. Our stability analysis shows the elements involved in the performance of an impedance controller for such a system. Additionally, we extend the analysis to interaction control under wind influence, assuming that the forces may be distinguished. This step is not typically done for robot manipulators, as modeling errors are usually small. However, for aerial robots, the force caused by wind presents a significant disturbance. Typical interaction control schemes would, therefore, fail in this case. Additionally, we performed an evaluation of the external wrench estimator on a force-torque sensor. Finally, we show how to apply all of the presented algorithms in a novel tactile mapping application using an autonomous flying robot with onboard state estimation.

TABLE I
OVERVIEW OF RELATED WORK FROM AERIAL ROBOTICS, ORGANIZED BY TOPIC

Topic	References
External wrench estimation	[5], [15], [16], [17], [18], [19], [20]
Physical interaction control	[6], [7], [8], [9], [15], [16], [17], [19], [21], [22], [20], [23]
Collision handling	[4], [12], [13], [23], [24]

This paper is organized as follows. We discuss related work in Section II. In Section III, we summarize the dynamics model and the control scheme of a quadrotor vehicle. Our approach to external wrench estimation is elaborated in Section IV. Interaction control using this estimate, with a stability proof of the impedance controller, is developed in Section V. We investigate collision detection and reaction strategies in Section VI. In Section VII, we show the effectiveness of the presented methods through experiments. Finally, we conclude in Section VIII.

II. RELATED WORK

External wrench estimation, collision detection, and safe reaction, as well as interaction control, were thoroughly investigated in the context of robotic manipulators; see [11], [14] and references therein. These problems also receive increasing interest from the flying robots community. Our goal is to transfer the original concepts from manipulators and extend them to flying robots. An overview of related work from the flying robots domain can be found in Table I.

A. External Wrench Estimation and Impedance Control

In the context of hybrid pose/wrench control, Bellens et al. [5] mapped offline control inputs to forces and torques generated by the unmanned aerial vehicle (UAV) while being fixed to a base. The estimated wrench is then a function of the control input. It is, however, only valid while the robot is in contact. The work of Ruggireo et al. [15], [16] is related to the wrench estimation and interaction control aspects of this paper. The authors used a momentum-based external wrench estimator with second-order estimation dynamics and implemented an impedance controller based on this estimate. A sketch of the controller's stability proof was given. However, their force estimator requires translational velocity information, which is not readily available on flying systems. Yüksel et al. [17], [18] use a Lyapunov-based design of an nonlinear external wrench observer including also inertia shaping. A port-based Hamiltonian approach is used to design an interconnection and damping assignment passivity-based controller, which allows us to reshape the physical properties of a quadrotor. Only simulation results are presented. McKinnon and Schoellig implemented an unscented-Kalman-filter-based estimator of the external wrench [19]. This allows the explicit incorporation of sensor noise properties into the filter. The results show overshoot of the wrench estimate, which we consider undesired when using the estimate for feedback control.

B. Physical Interaction Control

A dedicated propeller for horizontal force control has been used by Albers *et al.* [21]. However, only a feedforward signal was used to exert a contact force. More recently, force control at a tooltip rigidly attached to a flying robot was investigated in [6]–[8]. The force was assumed to be measured by a force sensor. Impedance control was applied to UAVs for contact inspection by Forte *et al.* [22] and Fumagalli *et al.* [8], [9]. An external force measurement was not required because of reliance on the passivity properties of impedance control. Augugliaro and D'Andrea [20] extend the Kalman filter used for quadcopter state estimation with an external wrench state. They implement an admittance controller for interaction control, using the estimated external force. Mersha *et al.* [23] developed a variable impedance controller for aerial manipulation using a simplified model. A force controller is implemented as the outer loop of the variable impedance controller. It is assumed that the environment can be modeled as a linear spring. The external force is, therefore, estimated using position information and the environment stiffness, making it dependent on the used controller and state estimation method. The effectiveness of this approach is shown in flight experiments in contact scenarios, with ground truth provided by a force-torque sensor.

C. Collision Handling

A control loop supervisor for contact detection was implemented by Naldi *et al.* [24]. If the error of the path-following controller is above a threshold, contact is assumed and the path is interpolated backwards until there is no collision anymore. Here, no wrench information is used, which makes the method trajectory and controller dependent. In [23], contacts are detected through a threshold on the external force, making the implementation equivalent to [24]. Onboard accelerometers and small force sensors attached to elastic springs were used by Briod *et al.* [4] to detect collisions with the environment. The acceleration-based approach detects collisions when the acceleration magnitude is above a predefined threshold. The threshold needs to be chosen larger than nominal accelerations occurring during normal flight. Therefore, the sensitivity of the method depends significantly on the controller and the trajectory. Upon collision, the motors are turned OFF and the robot relies on its robust mechanical design to land safely. It resumes flight opposite of the detected collision direction. This information is used for tactile exploration.

III. DYNAMICS MODELING

The equations of motion of a flying robot about its center of mass can be written as

$$\mathcal{M}\ddot{\boldsymbol{r}} = \mathcal{M}g\boldsymbol{e}_3 + \boldsymbol{R}\boldsymbol{f} + \boldsymbol{R}\boldsymbol{f}_e \tag{1}$$

$$\mathcal{I}\dot{\boldsymbol{\omega}} = \mathcal{M}g\mathcal{S}(\boldsymbol{r}_g)\boldsymbol{R}^T\boldsymbol{e}_3 + \mathcal{S}(\mathcal{I}\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{m} + \boldsymbol{m}_e \tag{2}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\mathcal{S}(\boldsymbol{\omega}) \tag{3}$$

where $\mathcal{M}$ is the robot mass, $\boldsymbol{r} = [x\ y\ z]^T$ is its position in the fixed North-East-Down inertial frame, $\boldsymbol{R} \in SO(3)$ is the
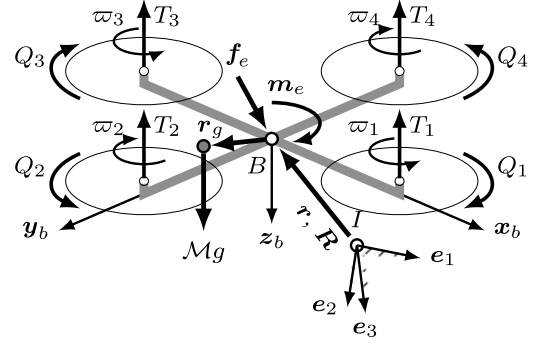

Fig. 2. Free-body diagram of a quadrotor, located at position $\boldsymbol{r}$ and orientation $\boldsymbol{R}$ in the inertial frame I and subject to the external wrench $\boldsymbol{\tau}_e = [\boldsymbol{f}_e\ \boldsymbol{m}_e]^T$. The propellers rotating at angular velocities $\boldsymbol{\Omega} = [\varpi_1\ \varpi_2\ \varpi_3\ \varpi_4]^T$ produce the control wrench $\boldsymbol{\tau} = \boldsymbol{B}\boldsymbol{u} = [\boldsymbol{f}\ \boldsymbol{m}]^T$ through the thrusts T_i and drag torques Q_i.

rotation matrix from body to inertial frame, $\mathcal{S}(\bullet)$ is the skew-symmetric matrix operator such that $\mathcal{S}(\boldsymbol{a})\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$, $\mathcal{I} \in \mathbb{R}^{3\times3}$ is its moment of inertia, $g = 9.81$ m/s^2 is the acceleration of gravity, $\boldsymbol{\omega}$ is the body angular velocity, $\boldsymbol{e}_3$ is the z-axis unit vector, $\boldsymbol{f} = [f_x\ f_y\ f_z]^T$ is the control force in the body frame, $\boldsymbol{f}_e = [f_{e,x}\ f_{e,y}\ f_{e,z}]^T$ is the external force in the body frame, $\boldsymbol{r}_g = [x_g\ y_g\ z_g]^T$ is the position of the center of mass in the body frame, $\boldsymbol{m} = [m_x\ m_y\ m_z]^T$ is the body-frame control torque, and $\boldsymbol{m}_e = [m_{e,x}\ m_{e,y}\ m_{e,z}]^T$ is the external torque in the body frame.

A free-body diagram of a quadrotor is depicted in Fig. 2. By writing the generalized velocity as $\boldsymbol{v} = [\dot{\boldsymbol{r}}\ \boldsymbol{\omega}]^T$, the equations of motion can be rewritten in Lagrangian form as

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g} = \boldsymbol{J}^T\boldsymbol{\tau} + \boldsymbol{\tau}_e \tag{4}$$

where

$$\boldsymbol{g} = -\begin{bmatrix} \boldsymbol{I}_{3\times3} \\ \mathcal{S}(\boldsymbol{r}_g)\boldsymbol{R}^T \end{bmatrix}\mathcal{M}g\boldsymbol{e}_3, \quad \boldsymbol{J} = \begin{bmatrix} \boldsymbol{R}^T & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix}$$

$$\boldsymbol{M} = \begin{bmatrix} \mathcal{M}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \mathcal{I} \end{bmatrix}, \quad \boldsymbol{C}(\boldsymbol{v}) = \begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & -\mathcal{S}(\mathcal{I}\boldsymbol{\omega}) \end{bmatrix}.$$

Assumptions: The external wrench acting on the robot $\boldsymbol{\tau}_e = [\boldsymbol{f}_e\ \boldsymbol{m}_e]^T$ may be considered as a *lumped* disturbance wrench. The wrench $\boldsymbol{\tau}_e = \boldsymbol{\tau}_m + \boldsymbol{\tau}_d + \boldsymbol{\tau}_i + \boldsymbol{\tau}_c$ consists of the modeling error $\boldsymbol{\tau}_m$, an aerodynamic wrench $\boldsymbol{\tau}_d$, an interaction wrench $\boldsymbol{\tau}_i$, and a collision wrench $\boldsymbol{\tau}_c$. Fig. 1 depicts the specific assumptions made during applications covered in this paper. Without loss of generality, the aerodynamic wrench depends on the relative airspeed $\boldsymbol{v}_r = \boldsymbol{v} - \boldsymbol{v}_w$, with $\boldsymbol{v}_w$ being the wind speed, as $\boldsymbol{\tau}_d(\boldsymbol{v}_r) := \boldsymbol{D}(\boldsymbol{v}_r)\boldsymbol{v}_r$. We refer the reader to [25] and [26] for more detailed models thereof. The aerodynamic and control wrenches are, in general, dependent on the true airspeed $\boldsymbol{v}_r$. In this paper, we neglect the dependence of the propeller thrust on airspeed, as this effect is only noticeable at high airspeeds and, therefore, not relevant to interaction scenarios. For interaction control, we first assume $\boldsymbol{\tau}_e = \boldsymbol{\tau}_i$ and loosen this assumption for analysis. The wrench $\boldsymbol{\tau}_i$ could be measured by a force/torque sensor in the case of operation under wind influence. We treat any modeling errors as external disturbances. Note that these assumptions are not a limitation of the external wrench estimator or

interaction controllers, but are only used for the interpretation of the discussed applications.

Control wrench: The control input $\boldsymbol{\tau}$ is usually obtained by a cascaded control structure. For details on control of flying robots, we refer the reader to [27]–[30], as this is not the focus of the present paper. The applied control wrench $\boldsymbol{\tau}$ is required for external wrench estimation. As it cannot be directly measured during flight, an accurate model is required. The control wrench generated by the propellers in the body frame is $\boldsymbol{\tau} = [\boldsymbol{f}\ \boldsymbol{m}]^T$, which for N propellers is given by

$$\boldsymbol{\tau} = \begin{bmatrix} \sum_{i=1}^{N} T_i \boldsymbol{n}_i \\ \sum_{i=1}^{N} (T_i \boldsymbol{r}_i \times \boldsymbol{n}_i + \delta_i Q_i \boldsymbol{n}_i) \end{bmatrix} = \boldsymbol{B} \boldsymbol{u} \qquad (5)$$

where $\boldsymbol{n}_i = \boldsymbol{R}_{bp,i} \boldsymbol{e}_3$ is the axis of rotation of propeller i located at $\boldsymbol{r}_i$ in the body frame, $\boldsymbol{R}_{pb,i}$ is the rotation matrix from the body to the propeller frame, $\delta_i \in \{-1, 1\}$ is the propeller rotational sense, and $\boldsymbol{B} \in \mathbb{R}^{6 \times N}$ is the control allocation matrix. The desired propeller velocities $\boldsymbol{u} = [\varpi_1^2 \ldots \varpi_N^2]^T$ can be obtained for control purposes by (pseudo)inverting the matrix $\boldsymbol{B}$. The rotor thrust and torque may be obtained by

$$T_i = \rho C_T D^4 \varpi_i^2 \qquad (6)$$

$$Q_i = \rho C_Q D^5 \varpi_i^2 + I_r \dot{\varpi}_i \qquad (7)$$

where C_T and C_Q are the nondimensional rotor thrust and torque coefficients, respectively. Additionally, ρ is the air density, D is the propeller diameter, and I_r is the combined inertia of the rotor and propeller. Alternatively, the generated torque may be directly measured by the motor current, using the motor model [25], [31]

$$Q_i = (K_{q,0} + K_{q,1} i_a) i_a \qquad (8)$$

where i_a is the measured motor current, and $K_{q,0}$ and $K_{q,1}$ are the motor torque parameters. In Section VII, we verify the validity of these models.

IV. EXTERNAL WRENCH ESTIMATION

Knowledge of the system model and control laws can be used to estimate the external wrench acting on the robot [32]. We investigate two schemes—the *momentum-based* method that uses velocity information and the *acceleration-based method*. Finally, we combine the two to obtain a practical choice for a flying robot, using already available sensors.

A. Momentum-Based Estimation

The first method relies on observing the robot's generalized momentum $\boldsymbol{p} = \boldsymbol{M} \boldsymbol{v}$. Rewriting (4) in terms of $\boldsymbol{p}$ gives

$$\dot{\boldsymbol{p}} = \boldsymbol{M} \dot{\boldsymbol{v}} = \boldsymbol{J}^T \boldsymbol{\tau} + \boldsymbol{\tau}_e - \boldsymbol{N} \qquad (9)$$

where $\boldsymbol{N} := \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}$. Following [32], we define a residual vector

$$\boldsymbol{\rho} = \boldsymbol{K}_I \left[\boldsymbol{p} - \int \left(\boldsymbol{J}^T \boldsymbol{\tau} - \boldsymbol{N} + \boldsymbol{\rho} \right) dt - \boldsymbol{p}(0) \right] \qquad (10)$$

TABLE II
MOTIVATION FOR THE HYBRID ESTIMATION SCHEME—THE TRANSLATIONAL VELOCITY WHICH WOULD BE REQUIRED BY THE MOMENTUM OBSERVER WOULD HAVE TO BE ESTIMATED

Sensor	$\dot{r}$	ω	$\ddot{r}$	$\dot{\omega}$
Accelerometer	-	-	●	-
Gyroscope	-	●	-	○
PX4FLOW [33]	△	●	-	○
Skybotix VI [34]	△	●	●	○
IMU–odometry fusion [1], [2]	△	●	●	○

Alternatively, the readily available accelerometer signal can be used for estimation of the external force. The table shows relevant measurements provided by different state-of-the-art sensor suites: translational velocity $\dot{r}$, translational acceleration $\ddot{r}$, angular velocity ω, and angular acceleration $\dot{\omega}$. Measured: ●, obtained numerically: ○, or estimated: △.

with positive-definite diagonal observer matrix $\boldsymbol{K}_I \in \mathbb{R}^{6 \times 6}$. By differentiating (10), we obtain the residual dynamics

$$\dot{\boldsymbol{\rho}} = \boldsymbol{K}_I \boldsymbol{\tau}_e - \boldsymbol{K}_I \boldsymbol{\rho}. \qquad (11)$$

Note that (11) represents a linear exponentially stable system, driven by the true external wrench $\boldsymbol{\tau}_e$. Hence, $\boldsymbol{\rho}$ is the first-order low-pass-filtered reconstruction of $\boldsymbol{\tau}_e$. Therefore, the estimated external wrench is denoted $\hat{\boldsymbol{\tau}}_e := \boldsymbol{\rho}$. Note that this method requires the measurement or estimation of the generalized velocity $\boldsymbol{v}$.

B. Acceleration-Based Estimation

In general, the external wrench could be obtained directly from full acceleration information. By rearranging the terms in (4), we may algebraically calculate the external wrench as

$$\bar{\boldsymbol{\tau}}_e = \boldsymbol{M} \dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g} - \boldsymbol{J}^T \boldsymbol{\tau} \qquad (12)$$

where the acceleration $\dot{\boldsymbol{v}}$ contains significant sensor noise. We, therefore, apply a first-order stable filter to obtain the estimation dynamics

$$\dot{\hat{\boldsymbol{\tau}}}_e = \boldsymbol{K}_I \bar{\boldsymbol{\tau}}_e - \boldsymbol{K}_I \hat{\boldsymbol{\tau}}_e. \qquad (13)$$

By lumping the nonlinear terms in $\boldsymbol{N} := \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}$, we finally obtain

$$\dot{\hat{\boldsymbol{\tau}}}_e = \boldsymbol{K}_I \left(\boldsymbol{M} \dot{\boldsymbol{v}} + \boldsymbol{N} - \boldsymbol{J}^T \boldsymbol{\tau} - \hat{\boldsymbol{\tau}}_e \right). \qquad (14)$$

In contrast to robot manipulators, this method is suitable for observing the translational dynamics of flying robots, as the translational acceleration $\ddot{r}$ and acceleration of gravity $g \boldsymbol{e}_3$ are directly measured by the onboard inertial measurement unit (IMU). The angular acceleration $\dot{\omega}$ may be obtained by numerical differentiation.

C. Practical Considerations for Flying Robots

Table II shows measurement requirements for the two wrench estimation methods. Obtaining a drift-free *translational* velocity requires exteroceptive sensors and a fusion algorithm. This greatly limits the applicability of the momentum-based method for force estimation. Similarly, the *angular* acceleration can only
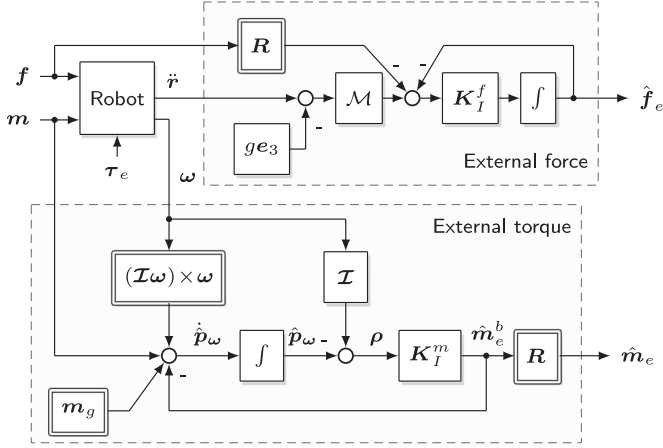
Fig. 3. Structure of the hybrid external wrench estimator near hovering conditions. It uses the robot acceleration $\ddot{r}$ and angular velocity ω, both measured directly by the onboard IMU.
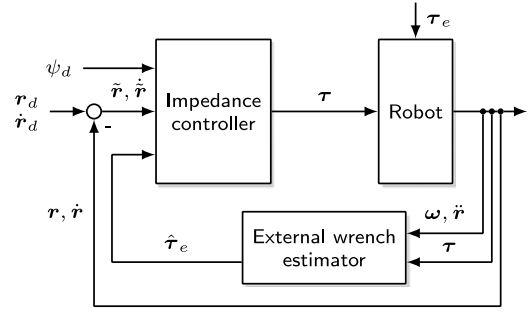

Fig. 4. Structure of an impedance controller (17) if it were possible to apply forces in every direction. This would be the case for a UAV with fully actuated degrees of freedom, where rank($\boldsymbol{B}$) = 6.

be obtained through numerical differentiation. This reduces the quality of the torque estimate by the acceleration-based method.

D. Hybrid Estimation

By considering the directly measurable values, it is intuitively clear that a combination of the two methods solves each one's problems—(14) is used for external force estimation and (10) for external torque estimation. The resulting estimator structure is depicted in Fig. 3. The rotational part of the momentum p is denoted p_ω. The estimator gain K_I is split into its diagonal blocks K_I^f for the force and K_I^m for the torque components. By combining both methods, we can estimate the external wrench using proprioceptive sensors only. The estimated external wrench $\hat{\tau}_e = [\hat{f}_e \ \hat{m}_e]^T$ is obtained by

$$\hat{\tau}_e = \begin{bmatrix} \int_0^t K_{I,f} \left(\mathcal{M}a - f - \hat{f}_e \right) \mathrm{d}t \\ K_{I,m} \left(\mathcal{I}\omega - \int_0^t \left(m + m_g + (\mathcal{I}\omega) \times \omega - \hat{m}_e \right) \mathrm{d}t \right) \end{bmatrix}$$
(15)

where $a = R^T(\ddot{r} - ge_3)$ is the acceleration measured by an accelerometer in the body frame, $m_g = \mathcal{M}g\mathcal{S}(r_g) R^T e_3$ is the torque due to the offset center of gravity, and $\hat{m}_e$ is the estimated external torque, also expressed in the body frame. Note that an accelerometer also measures the acceleration of gravity. The estimator dynamics are linear and decoupled in both methods.

V. INTERACTION CONTROL

Next, the integration of the external wrench estimate $\hat{\tau}_e$ into the robot control feedback loop to enable the system to purposefully interact with its surroundings is considered. Suitable interaction behaviors can be obtained through *impedance control* and *admittance control* [14]. In the impedance control approach, we directly control the interaction properties of the flying system. The resulting controller replaces the existing position control loop of a flying robot. Admittance control, on the other hand, is easily implemented around an existing position control loop

[20]. However, it is more difficult to select the gains to obtain the desired interaction behavior than in the case of a clean impedance controller. In the following, we discuss both interaction control approaches.

A. Impedance Control and Inertia Shaping

For interaction control, a robot should ideally represent an impedance, as was argued in the seminal work of Hogan [35]. Define the target closed-loop dynamics to be

$$M_v \dot{\tilde{v}} + D_v \tilde{v} + K_v \tilde{x} = \tau_e$$
(16)

where $M_v \in \mathbb{R}^{6\times6}$ is the desired positive-definite apparent inertia matrix, $D_v \in \mathbb{R}^{6\times6}$ is the desired positive-definite diagonal damping matrix, and $K_v = \mathrm{blockdiag}\{K_{v,t}, \ c\,I_{3\times3}\}$ is the desired positive-definite diagonal stiffness matrix. Here, $K_v \in \mathbb{R}^{3\times3}$ is the translational stiffness, and c is the rotational stiffness about all three axes. The state and velocity tracking errors are $\tilde{x} = [\tilde{r} \ \tilde{\eta}\tilde{\epsilon}]^T$ and $\tilde{v} = v - v_d$, respectively. Here, $\tilde{q} := [\tilde{\eta} \ \tilde{\epsilon}]^T$ is the quaternion representation of the geometric orientation error, with $\tilde{\eta}$ being the scalar, and $\tilde{\epsilon}$ the vector part. By inserting (16) into (4), the required control input becomes

$$J^T \tau = (MM_v^{-1} - I)\tau_e + M\dot{v}_d$$
$$- MM_v^{-1}(D_v\tilde{v} + K_v\tilde{x}) + N.$$
(17)

Since a measurement of τ_e is not available, we use its estimate $\hat{\tau}_e$ in (17). The resulting control structure for a fully actuated flying robot (rank($\boldsymbol{B}$) = 6) is shown in Fig. 4. The impedance control structure for a nonholonomic flying robot (e.g., quadcopter) is shown in Fig. 5. The resulting system response along the inertial x-direction is shown in Fig. 6. The impedance controller has a simpler overall structure, and it is easier to select its gains than for the admittance controller.

Remark 1 (Closed-loop dynamics): By applying $\hat{\tau}_e$ in (17), the closed-loop dynamics of controller (17) for system (4) is

$$M_v \dot{\tilde{v}} + D_v \tilde{v} + K_v \tilde{x} = \tau_e + (M_v M^{-1} - I)\tilde{\tau}_e$$
(18)

where $\tilde{\tau}_e := \tau_e - \hat{\tau}_e$ is the external wrench estimation error. (18) shows that controller (17) achieves the desired closed-loop dynamics up to the scaled wrench estimation error. The steady state ($\dot{v} = 0$, $v = 0$, $\tilde{\tau}_e = 0$, $\dot{\tilde{\tau}}_e = 0$) equilibrium point
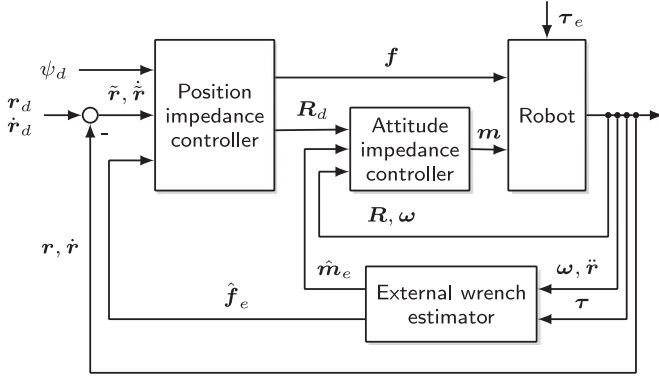
Fig. 5.   Control structure of an impedance controller for an underactuated UAV, like a typical quadrotor with rank $(\boldsymbol{B}) < 6$. Position impedance is only achievable in the thrust direction. We must, therefore, use the attitude controller to obtain the desired impedance in Cartesian translational directions. The controller calculates a virtual control force in the inertial frame $\boldsymbol{f}_c = T\boldsymbol{R}_d\boldsymbol{e}_3$. This force is then decomposed into the desired thrust T and attitude $\boldsymbol{R}_d$ using state-of-the-art methods.
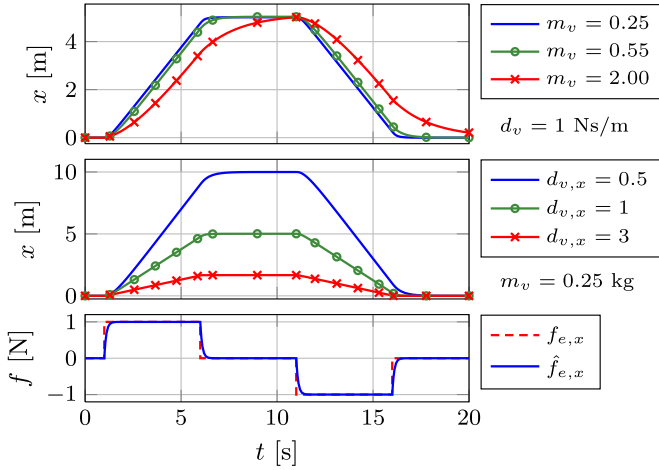

Fig. 6.   Simulated response of the position impedance controller to an external force in the inertial x-direction, for varying virtual inertia and damping parameters. The system is in damping mode—the virtual stiffness is $\boldsymbol{K}_v = \boldsymbol{0}$. The real system mass is $m = 0.55$ kg.

is obviously $\tilde{\boldsymbol{x}} = \boldsymbol{K}_v^{-1}\boldsymbol{\tau}_e$. We may rewrite system (18) and (11) using $\boldsymbol{z} := [\tilde{\boldsymbol{x}} \; \tilde{\boldsymbol{v}} \; \tilde{\boldsymbol{\tau}}_e]^T$, in matrix form

$$\dot{\boldsymbol{z}} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{g}(t, \boldsymbol{z}) \qquad (19)$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{T} & \boldsymbol{0} \\ -\boldsymbol{M}_v^{-1}\boldsymbol{K}_v & -\boldsymbol{M}_v^{-1}\boldsymbol{D}_v & \boldsymbol{M}^{-1} - \boldsymbol{M}_v^{-1} \\ \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{K}_I \end{bmatrix} \qquad (20)$$

$$\boldsymbol{g}(t) = [\boldsymbol{0} \quad \boldsymbol{M}_v^{-1}\boldsymbol{\tau}_e \quad \dot{\boldsymbol{\tau}}_e]^T. \qquad (21)$$

Here, $\boldsymbol{T}$ is the kinematic transformation matrix with $\dot{\boldsymbol{x}} = \boldsymbol{T}\boldsymbol{v}$ and $\det(\boldsymbol{T}) = 1$. The nominal (unperturbed) system $\dot{\boldsymbol{z}} = \boldsymbol{A}\boldsymbol{z}$ will be asymptotically stable if $\mathrm{Re}(\boldsymbol{A}) < 0$. Note that in that case the coupling term $\left(\boldsymbol{M}^{-1} - \boldsymbol{M}_v^{-1}\right)$ in (20) does not affect stability due to zero submatrices in $\boldsymbol{A}$. Next, we show passivity and stability of the impedance controller.

We note that with most of the flying systems (17) cannot be obtained in the general case. Flying systems such as quadrotors are underactuated and, therefore, cannot generate horizontal

forces in the body frame. The target impedance dynamics (16) can only be obtained in the direction of the rotor thrust, which is typically the body z-axis. Therefore, to obtain a Cartesian impedance, the robot must turn its body z-axis in the direction of the desired force. This is achieved by using a cascaded control structure, as depicted in Fig. 5, which will be the subject of our analysis. By introducing $\mu := \mathcal{M}\mathcal{M}_v^{-1}$ and $\boldsymbol{\Gamma} := \mathcal{I}\mathcal{I}_v^{-1}$, the position and attitude impedance controllers can be written as

$$\boldsymbol{R}_d\boldsymbol{f} = \mathcal{M}\ddot{\boldsymbol{r}}_d + (\mu - 1)\,\hat{\boldsymbol{f}}_e - \mu\big(\boldsymbol{D}_1\dot{\tilde{\boldsymbol{r}}} + \boldsymbol{K}_1\tilde{\boldsymbol{r}}\big) - \boldsymbol{g} \qquad (22)$$

$$\boldsymbol{m} = \mathcal{I}\dot{\boldsymbol{\omega}}_d + (\boldsymbol{\Gamma} - \boldsymbol{I})\hat{\boldsymbol{m}}_e - \boldsymbol{\Gamma}(\boldsymbol{D}_2\boldsymbol{\omega} + c_v\,\tilde{\boldsymbol{e}}) - \boldsymbol{n} \qquad (23)$$

where $\boldsymbol{g} = \mathcal{M}g\boldsymbol{e}_3$, $\tilde{\boldsymbol{e}} = \tilde{\eta}\tilde{\boldsymbol{\epsilon}}$, $\boldsymbol{n} = \mathcal{S}(\mathcal{I}\boldsymbol{\omega})\,\boldsymbol{\omega} - \boldsymbol{m}_g$. The desired attitude $\boldsymbol{R}_d$ and thrust T may be obtained from the relation $\boldsymbol{f}_c = T\boldsymbol{R}_d\boldsymbol{e}_3$ (see, e.g., [30]). Here, $\tilde{\boldsymbol{q}}$ is the quaternion representation of the geometric orientation error $\tilde{\boldsymbol{R}} = \boldsymbol{R}_d\boldsymbol{R}^T$. Note that this formulation of $\tilde{\boldsymbol{e}}$ will result in two asymptotically stable equilibria at $\tilde{\eta} = \pm 1$ and one unstable equilibrium at $\tilde{\eta} = 0$. For very large attitude errors (around 180°), and because of the unstable equilibrium, this feedback might not take the shortest path, but go through $\tilde{\eta} = 0$. Therefore, in tracking scenarios, the attitude trajectory should be initialized close to $\|\tilde{\boldsymbol{\epsilon}}\| = 0$ such that this does not occur. The overall structure of the described system is depicted in Fig. 7. In order to incorporate the attitude error dynamics into the error dynamics of the translation subsystem, we define the *control force error*.

Definition 1 (Control force error): Suppose that the desired thrust T can be obtained instantaneously, i.e., that the thrust dynamics can be neglected w.r.t. the attitude dynamics. This assumption is valid due to the motor dynamics and speed control being faster than the attitude control loop. The desired control force $\boldsymbol{f}_c := T\boldsymbol{R}_d\boldsymbol{e}_3$ is then achieved perfectly, whereas the actual control force depends on the actual attitude as $\boldsymbol{f} = T\boldsymbol{R}\boldsymbol{e}_3$. The control force error $\tilde{\boldsymbol{f}} := \boldsymbol{f} - \boldsymbol{f}_c$ is

$$\tilde{\boldsymbol{f}} = T(\tilde{\boldsymbol{R}} - \boldsymbol{I})\boldsymbol{R}_d\boldsymbol{e}_3$$
$$= 2T\big((\tilde{\eta}\boldsymbol{I} + \mathcal{S}(\tilde{\boldsymbol{\epsilon}}))\mathcal{S}(\tilde{\boldsymbol{\epsilon}})\big)\boldsymbol{R}_d\boldsymbol{e}_3 \qquad (24)$$

from which we obtain the norm estimates $\|\tilde{\boldsymbol{f}}\| \le 2T\|\tilde{\boldsymbol{\epsilon}}\|$ and $\|\dot{\tilde{\boldsymbol{f}}}\| \le 2\dot{T}\|\tilde{\boldsymbol{\epsilon}}\| + T\|\tilde{\boldsymbol{\omega}}\|$. By limiting the thrust to the range $0 < T < T_{\max}$, we obtain the upper bound $\|\tilde{\boldsymbol{f}}\| < 2T_{\max}\|\tilde{\boldsymbol{\epsilon}}\|$.

The error dynamics of the impedance controllers is

$$\mathcal{M}_v\ddot{\tilde{\boldsymbol{r}}} + \boldsymbol{D}_1\dot{\tilde{\boldsymbol{r}}} + \boldsymbol{K}_1\tilde{\boldsymbol{r}} = \boldsymbol{f}_e + (\bar{\mu} - 1)\tilde{\boldsymbol{f}}_e + \bar{\mu}\tilde{\boldsymbol{f}} \qquad (25)$$

$$\mathcal{I}_v\dot{\tilde{\boldsymbol{\omega}}} + \boldsymbol{D}_2\tilde{\boldsymbol{\omega}} + c_v\tilde{\eta}\tilde{\boldsymbol{\epsilon}} = \boldsymbol{m}_e + (\bar{\boldsymbol{\Gamma}} - \boldsymbol{I})\tilde{\boldsymbol{m}}_e \qquad (26)$$

where $\bar{\mu} = \mu^{-1} = \mathcal{M}_v\mathcal{M}^{-1}$, $\bar{\boldsymbol{\Gamma}} = \mathcal{I}_v\mathcal{I}^{-1}$, $\tilde{\boldsymbol{f}}_e := \boldsymbol{f}_e - \hat{\boldsymbol{f}}_e$ and $\tilde{\boldsymbol{m}}_e := \boldsymbol{m}_e - \hat{\boldsymbol{m}}_e$ are the force and torque estimation errors, respectively, and $\tilde{\boldsymbol{f}}$ is the control force error resulting from the attitude tracking error.

Proposition 1: The attitude impedance controlled subsystem (26) is passive w.r.t. the pair $(\tilde{\boldsymbol{\omega}}, \boldsymbol{m}_e + (\bar{\boldsymbol{\Gamma}} - \boldsymbol{I})\tilde{\boldsymbol{m}}_e)$.

Proof: Take the storage function

$$V_1 = \tfrac{1}{2}\tilde{\boldsymbol{\omega}}^T\mathcal{I}_v\tilde{\boldsymbol{\omega}} + \tfrac{1}{2}c_v\,\tilde{\boldsymbol{\epsilon}}^T\tilde{\boldsymbol{\epsilon}}.$$
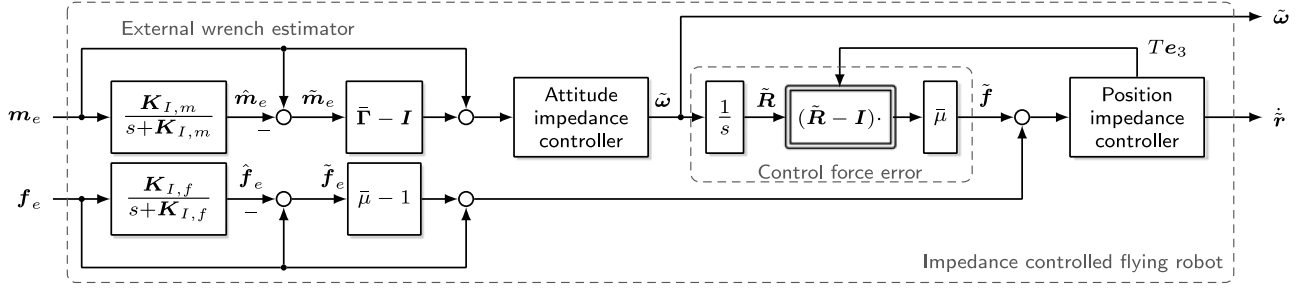
Fig. 7. Blocks considered in the passivity proof of the cascaded impedance controlled flying robot, considering the external wrench estimator and attitude error dynamics. In order for the complete system to be passive, the interconnected subsystems must be passive or lossless. The external wrench estimator has first-order error dynamics and is, therefore, passive. We show passivity of the attitude impedance controller w.r.t. the pair $(\tilde{\boldsymbol{\omega}}, \boldsymbol{m}_e + (\bar{\boldsymbol{\Gamma}} - \boldsymbol{I})\tilde{\boldsymbol{m}}_e)$. Furthermore, the control force error $\tilde{\boldsymbol{f}}$ is obtained by a lossless transformation. Finally, we show passivity of the position impedance controller w.r.t. the pair $(\dot{\tilde{\boldsymbol{r}}}, \boldsymbol{f}_e + (\bar{\mu} - 1)\tilde{\boldsymbol{f}}_e + \bar{\mu}\tilde{\boldsymbol{f}})$.

The derivative of V_1 is

$$\dot{V}_1 = \tilde{\boldsymbol{\omega}}^T \left(\boldsymbol{m}_e + (\bar{\boldsymbol{\Gamma}} - \boldsymbol{I})\tilde{\boldsymbol{m}}_e - \boldsymbol{D}_2 \tilde{\boldsymbol{\omega}} - c_v \tilde{\eta}\tilde{\boldsymbol{\epsilon}} \right) + c_v \tilde{\eta}\tilde{\boldsymbol{\omega}}^T \tilde{\boldsymbol{\epsilon}}$$

$$\dot{V}_1 = -\tilde{\boldsymbol{\omega}}^T \boldsymbol{D}_2 \tilde{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}^T \left(\boldsymbol{m}_e + (\bar{\boldsymbol{\Gamma}} - \boldsymbol{I})\tilde{\boldsymbol{m}}_e \right)$$

which shows passivity of the subsystem. Here, the estimator error $\tilde{\boldsymbol{m}}_e$ is an additional scaled input. ∎

Proposition 2: The position impedance controlled subsystem (25) is passive w.r.t. the pair $(\dot{\tilde{\boldsymbol{r}}}, \boldsymbol{f}_e + (\bar{\mu} - 1)\tilde{\boldsymbol{f}}_e + \bar{\mu}\tilde{\boldsymbol{f}})$.

Proof: Take the storage function

$$V_2 = \tfrac{1}{2}\dot{\tilde{\boldsymbol{r}}}^T \mathcal{M}_v \dot{\tilde{\boldsymbol{r}}} + \tfrac{1}{2}\tilde{\boldsymbol{r}}^T \boldsymbol{K}_1 \tilde{\boldsymbol{r}}.$$

The derivative of V_2 is

$$\dot{V}_2 = -\dot{\tilde{\boldsymbol{r}}}^T \boldsymbol{D}_1 \dot{\tilde{\boldsymbol{r}}} + \dot{\tilde{\boldsymbol{r}}}^T \left(\boldsymbol{f}_e + (\bar{\mu} - 1)\tilde{\boldsymbol{f}}_e + \bar{\mu}\tilde{\boldsymbol{f}} \right)$$

which shows passivity of the subsystem. ∎

Note that in the case without inertia shaping, the interaction ports of the attitude controller correspond to the physical power ports. For the position controller, this only holds for $\tilde{\boldsymbol{f}} = 0$. Note that for a full passivity proof for a passive environment w.r.t $(\boldsymbol{f}_e, \boldsymbol{m}_e)$ and $(\dot{\tilde{\boldsymbol{r}}}, \tilde{\boldsymbol{\omega}})$, further steps would be necessary.

However, in the following, we show local input-to-state stability (ISS) [36] of the complete impedance controlled system, taking into account the external wrench estimation dynamics. In the analysis, we assume that the external wrench is bounded and differentiable, but do not impose any specific form on the bounds. In this way, we allow the analysis to be applied to more specific assumptions on the external wrench, such as, e.g., [29]. As an example of bounds on the external force, one may use $\|\boldsymbol{f}_e\| \le c_{f1} + c_{f2}\|\dot{\boldsymbol{r}} - \boldsymbol{v}_w\|$ (see [29]). First, we recall the definition of ISS and Theorem 4.19 from [36].

Definition 2: The system $\dot{x} = f(t, x, u)$ is said to be input-to-state stable if there exist a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ class function γ such that for any initial state $x(t_0)$ and any bounded input $u(t)$, the solution $x(t)$ exists for all $t \ge t_0$ and satisfies

$$\|x(t)\| \le \beta(\|x(t_0)\|, t - t_0) + \gamma\left(\sup_{t_0 \le \tau \le t} \|u(\tau)\| \right).$$

Theorem 1: Let $V : [0, \infty) \times \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function such that

$$\alpha_1(\|\boldsymbol{x}\|) \le V(t, x) \le \alpha_2(\|\boldsymbol{x}\|)$$

$$\frac{\partial V}{\partial t} + \left(\frac{\partial V}{\partial \boldsymbol{x}} \right)^T f(t, \boldsymbol{x}, \boldsymbol{u}) \le -W_3(\boldsymbol{x}), \quad \forall \|\boldsymbol{x}\| \ge \rho(\|\boldsymbol{u}\|) > 0$$

$\forall (t, \boldsymbol{x}, \boldsymbol{u}) \in [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m$, where α_1, α_2 are class $\mathcal{K}_\infty$ functions, ρ is a class $\mathcal{K}$ function, and $W_3(\boldsymbol{x})$ is a continuous positive-definite function on $\mathbb{R}^n$. Then, the system

$$\dot{\boldsymbol{x}} = f(t, \boldsymbol{x}, \boldsymbol{u})$$

is input-to-state stable with $\gamma = \alpha_1^{-1} \circ \alpha_2 \circ \rho$. ◇

Proposition 3: The impedance controlled flying robot, as depicted in Fig. 7, is locally input-to-state stable for inputs $\boldsymbol{u}^T = [\boldsymbol{f}_e^T \ \dot{\boldsymbol{f}}_e^T \ \boldsymbol{m}_e^T \ \dot{\boldsymbol{m}}_e]$ and states $\boldsymbol{x}^T = [\tilde{\boldsymbol{r}}^T \ \dot{\tilde{\boldsymbol{r}}}^T \ \tilde{\boldsymbol{f}}_e^T \ \tilde{\boldsymbol{\epsilon}}^T \ \tilde{\boldsymbol{\omega}}^T \ \tilde{\boldsymbol{m}}_e^T]$ for all eig($\boldsymbol{P}$) $\succ 0$, eig($\boldsymbol{N}$) $\succ 0$, and $\boldsymbol{Q} \succ 0$, with $\boldsymbol{P}$ defined in (28), $\boldsymbol{N}$ defined in (31), and $\boldsymbol{Q}$ defined in (32).

Proof: Take the candidate function for the complete system

$$V = \tfrac{1}{2}\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} \tag{27}$$

where

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{K}_1 & c_1 \boldsymbol{I} & \cdot & \cdot & \cdot & \cdot \\ c_1 \boldsymbol{I} & \mathcal{M}_v \boldsymbol{I} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \boldsymbol{K}_{I,f}^{-1} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & c_v \boldsymbol{I} & c_2 \boldsymbol{I} & \cdot \\ \cdot & \cdot & \cdot & c_2 \boldsymbol{I} & \mathcal{I}_v & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \boldsymbol{K}_{I,m}^{-1} \end{bmatrix} \tag{28}$$

with $\cdot \equiv \boldsymbol{0}_{3 \times 3}$ used for readability, and $c_1 > 0$, $c_2 > 0$ chosen such that eig($\boldsymbol{P}$) $\succ 0$ (see, e.g., [27]). Then, V is bounded by

$$\tfrac{1}{2}\lambda_{\min}\{\boldsymbol{P}\} \|\boldsymbol{x}\|_2^2 \le V \le \tfrac{1}{2}\lambda_{\max}\{\boldsymbol{P}\} \|\boldsymbol{x}\|_2^2 \tag{29}$$

$$\alpha_1(\|\boldsymbol{x}\|) = \tfrac{1}{2}\lambda_{\min}\{\boldsymbol{P}\} \|\boldsymbol{x}\|_2^2$$

$$\alpha_2(\|\boldsymbol{x}\|) = \tfrac{1}{2}\lambda_{\max}\{\boldsymbol{P}\} \|\boldsymbol{x}\|_2^2$$

and after algebraic manipulation its derivative can be found as

$$\dot{V} < -\boldsymbol{y}^T \boldsymbol{Q} \boldsymbol{y} + \boldsymbol{y}^T \boldsymbol{N} \boldsymbol{v} \tag{30}$$

where

$$y^T = \left[\|\tilde{r}\| \ \|\dot{\tilde{r}}\| \ \|\tilde{f}_e\| \ \|\tilde{\epsilon}\| \ \|\tilde{\omega}\| \ \|\tilde{m}_e\| \right]$$

$$v^T = \left[\|f_e\| \ \|\dot{f}_e\| \ \|m_e\| \ \|\dot{m}_e\| \right]$$

contain the state as $y(\|x\|)$ and input bounds as $v(\|u\|)$, and their coupling matrix $N \in \mathbb{R}^{6 \times 4}$ is

$$N = \begin{bmatrix} c_{\mathcal{M}} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & \lambda_{\min}^{-1}\{K_{I,f}\} & 0 & 0 \\ 0 & 0 & c_{\mathcal{I}} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \lambda_{\min}^{-1}\{K_{I,m}\} \end{bmatrix} \quad (31)$$

where $c_{\mathcal{M}} = c_1 \mathcal{M}_v^{-1}$ and $c_{\mathcal{I}} = c_2 \lambda_{\min}^{-1}\{\mathcal{I}_v\}$ for $v > 0$. The matrix $Q \in \mathbb{R}^{6 \times 6}$ can be found in (32), shown at the bottom of this page.

Take $\theta : 0 < \theta < 1$ and rewrite $\dot{V}$ as

$$\dot{V} < -\theta y^T Q y + y^T N v - (1 - \theta) y^T Q y. \quad (33)$$

The quadratic term in $\dot{V}$ is negative definite iff $y^T Q y > 0$, i.e., Q must be positive definite. Then, $\dot{V}$ will be negative definite at least in the region where the quadratic term is larger than the linear term, i.e., where $\theta y^T Q y > y^T N v$. From

$$\theta \lambda_{\min}\{Q\} \|y\|^2 > \lambda_{\max}\{N\} \|y\| \|v\| \quad (34)$$

we obtain

$$\dot{V} < -(1 - \theta) y^T Q y, \quad \forall \|y\| > \rho(\|u\|) = \frac{\|v\|}{\theta} \frac{\lambda_{\max}\{N\}}{\lambda_{\min}\{Q\}} \quad (35)$$

for all $\lambda_{\max}\{N\} > 0$. It follows that the complete system is locally input-to-state stable with

$$\gamma = \alpha_1^{-1}(\alpha_2(\rho(\|u\|))) = \frac{\|v\|}{\theta} \frac{\lambda_{\max}\{N\}}{\lambda_{\min}\{Q\}} \sqrt{\frac{\lambda_{\max}\{P\}}{\lambda_{\min}\{P\}}} \quad (36)$$

where $\rho(\|u\|)$ defines the region of local ISS. ∎

Remark 2: Notice that the matrix Q can be subdivided into the position control subsystem with force estimation dynamics in $Q_p \in \mathbb{R}^{3 \times 3}$, the attitude control subsystem with torque estimation dynamics in $Q_a \in \mathbb{R}^{3 \times 3}$, and the coupling matrix $Q_{pa} \in \mathbb{R}^{3 \times 1}$ as

$$Q = \begin{bmatrix} Q_p & Q_{pa} \\ Q_{pa}^T & Q_a \end{bmatrix}, \ Q_p = \begin{bmatrix} Q_p^0 & Q_{p1}^0 \\ Q_{p1}^{0,T} & 1 \end{bmatrix}, \ Q_a = \begin{bmatrix} Q_a^0 & Q_{a1}^0 \\ Q_{a1}^{0,T} & 1 \end{bmatrix} \quad (37)$$

and Q_p and Q_a can be further subdivided into the respective error dynamics Q_p^0 and Q_a^0, estimator dynamics, and coupling matrices Q_{p1}^0 and Q_{a1}^0. Therefore, showing positive definiteness of Q can be shown by showing positive definiteness—and therefore local ISS—of the respective subsystems. Due to space constraints and complicated analytical expressions, we omit the particular conditions for the positive definiteness of Q and P. This can be also seen as a linear matrix inequality problem [37] and solved numerically.

Remark 3: The following qualitative thoughts can be given from analyzing (36). First, the bounds will obviously grow with the external inputs $\|v\|$, as well as a smaller θ. Increasing the maximum eigenvalue of Q will decrease the ISS bounds. This can be achieved by increasing the minimum damping, stiffness, and virtual inertia. Analyzing the eigenvalues of N, one finds that $c_{\mathcal{M}}$ is the only nonzero eigenvalue. Therefore, increasing the virtual mass will reduce the ISS bounds in this term. Finally, decreasing the eigenvalues of P (the bounds of the Lyapunov function) and improving the conditioning of P will reduce the bounds further. This can be interpreted as increasing the observer gains, and reducing the virtual inertia and stiffness, and them being of the same order of magnitude as the inverse of the observer gains. This is clearly a compromise with tuning $\lambda_{\min}\{Q\}$. Finally, a numerical example applied to one of the experiments is given in the Appendix.

Remark 4: In the foregoing analysis, we implicitly assumed that the external wrench is caused by interaction forces only. However, aerial robots are typically subject to wind influence. In that case, we would ideally compensate for the wind influence and apply an impedance only to the interaction wrench. Suppose now that we are able to distinguish between the external wrench components $\tau_e = \tau_d + \tau_i$, consisting of the aerodynamic drag wrench τ_d and interaction wrench τ_i. It is then possible to compensate the aerodynamic wrench and apply an impedance on the interaction wrench τ_i. To analyze the stability of such a controller, we focus on the position controller for simplicity. The *compensated impedance controller* is then

$$R_d f = \mathcal{M} \ddot{r}_d - \hat{f}_d + (\mu - 1) \hat{f}_i - \mu (D_1 \dot{\tilde{r}} + K_1 \tilde{r}) - g \quad (38)$$

where $\hat{f}_d$ is the estimated aerodynamic force and $\hat{f}_i$ is the estimated interaction force. We do not assume any properties on $\hat{f}_d$ and $\hat{f}_i$ other than $\hat{f}_e = \hat{f}_d + \hat{f}_i$. The previous stability analysis may be applied to this controller by expanding the state $\tilde{f}_e$ into $\tilde{f}_i$ and $\tilde{f}_d$, and setting the appropriate coupling matrix Q_{p1}^0. The stability bounds will in this case depend on the

$$Q = \begin{bmatrix} c_{\mathcal{M}} \lambda_{\min}\{K_1\} & \frac{1}{2} c_{\mathcal{M}} \lambda_{\min}\{D_1\} & -\frac{1}{2} c_{\mathcal{M}} (\bar{\mu} - 1) & c_{\mathcal{M}} \bar{\mu} T_{\max} & 0 & 0 \\ \frac{1}{2} c_{\mathcal{M}} \lambda_{\min}\{D_1\} & \lambda_{\min}\{D_1\} - c_1 & -\frac{1}{2}(\bar{\mu} - 1) & \bar{\mu} T_{\max} & 0 & 0 \\ -\frac{1}{2} c_{\mathcal{M}} (\bar{\mu} - 1) & -\frac{1}{2}(\bar{\mu} - 1) & 1 & 0 & 0 & 0 \\ c_{\mathcal{M}} \bar{\mu} T_{\max} & \bar{\mu} T_{\max} & 0 & c_v c_{\mathcal{I}} & \frac{1}{2} c_{\mathcal{I}} \lambda_{\min}\{D_2\} & -\frac{1}{2} c_{\mathcal{I}} (\lambda_{\min}\{\bar{\Gamma}\} - 1) \\ 0 & 0 & 0 & \frac{1}{2} c_{\mathcal{I}} \lambda_{\min}\{D_2\} & \lambda_{\min}\{D_2\} - \frac{1}{2} c_2 & \frac{1}{2}(\lambda_{\min}\{\bar{\Gamma}\} - 1) \\ 0 & 0 & 0 & -\frac{1}{2} c_{\mathcal{I}} (\lambda_{\min}\{\bar{\Gamma}\} - 1) & \frac{1}{2}(\lambda_{\min}\{\bar{\Gamma}\} - 1) & 1 \end{bmatrix} \quad (32)$$
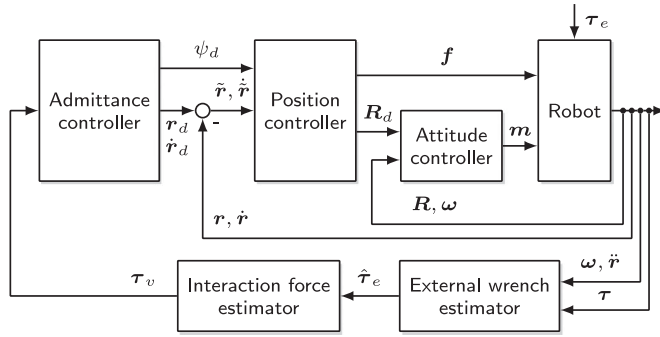
Fig. 8. Structure of the admittance controller (39). The controller generates a position and yaw reference, which is then tracked by a position controller. The external wrench causes a velocity.
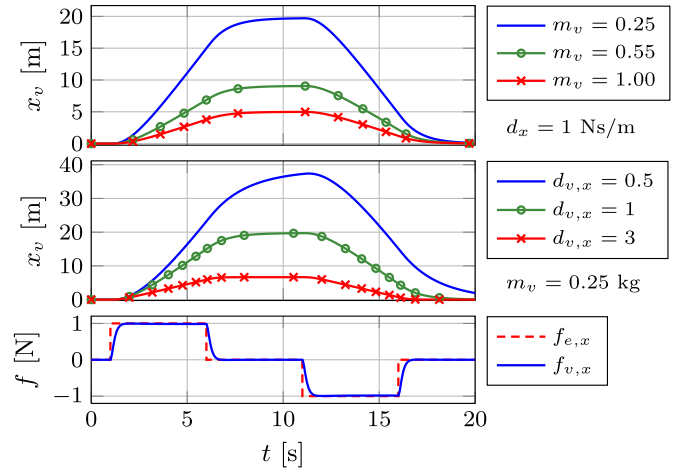

Fig. 9. Admittance controller response. The parameters determine the steady-state velocity of the virtual system. The system is in damping mode—the virtual stiffness is $K_v = 0$, and the real system mass is $m = 0.55$ kg.

quality of discrimination between the external force components. Furthermore, the estimation error $\tilde{f}_i$ must be bounded and will have the same effect as the attitude control error $\tilde{f}$. Note that in the case of $f_i = 0$ and $\mu = 1$, we obtain a disturbance-observer-based (DOB) controller [30]. Interestingly, the DOB controller is equivalent to setting $(\mathcal{M}\mathcal{M}_v^{-1} - 1)f_e = -f_e$ in the impedance controller, i.e., setting the virtual mass to $\mathcal{M}_v = \infty$.

B. Admittance Control

Essentially, admittance control generates velocity commands as a function of the contact wrench. The resulting trajectory may then be tracked using a position and attitude controller. In effect, this adds an additional cascade around the position control loop. The reference velocity and position are typically generated via the virtual dynamical system

$$M_a \ddot{\tilde{x}}_v + D_a \dot{\tilde{x}}_v + K_a \tilde{x}_v = \tau_v \tag{39}$$

where $x_v = [r_d \ \psi_d]$ is the resulting position trajectory, $M_a = \text{blockdiag}\{\mathcal{M}_a I_{3\times3} \ \mathcal{I}_{a,z}\}$ with $m_a > 0$ and $\mathcal{I}_{a,z} > 0$ is the virtual admittance inertia matrix, $D_a \in \mathbb{R}^{4\times4}$ is the positive-definite diagonal virtual damping gain matrix, $K_a \in \mathbb{R}^{4\times4}$ is the positive diagonal virtual spring gain matrix, and $\tau_v = [\hat{f}_e \ \hat{m}_{e,z}]^T$ is the estimated external wrench, with $\hat{m}_{e,z}$ being the torque about the z-axis. The torque information is used to implement an admittance on the yaw angle. The desired roll and pitch angles are commanded via the position controller. Similarly to the compensated impedance controller, a compensated admittance controller may be achieved by using a DOB position tracking controller and applying the interaction wrench in (39) as $\tau_v = [\hat{f}_i \ \hat{m}_{e,z}]^T$.

In the case $K_a = 0_{4\times4}$, a human operator can move the robot freely in space. The steady-state velocity will then be $\dot{x}_{v,ss} = D_a^{-1} M_a^{-1} \tau_v$, which, in turn, could be used to design the desired behavior of the admittance system. The admittance control structure is shown in Fig. 8, and the response of (39) in Fig. 9. The admittance controller adds another cascade into the control loop. This makes it simple to implement the method on a system where a position controller is already available. Note that now there is a fourth-order system between the input τ_v

and the system states, as opposed to the impedance controller, which imposes a second-order system.

C. Discussion and Practical Considerations

Figs. 6 and 9 show responses of impedance and admittance controllers under the same external force, using various parameters. It can be seen that the behavior of the impedance controller is intuitively more consistent, e.g., a different virtual mass does not change the steady-state position. It is thus easier to select the appropriate gains than for an admittance controller. An impedance controller replaces the position control loop of a flying robot. In contrast, the admittance controller can be easily implemented on the existing systems around an existing position control loop.

VI. COLLISION DETECTION AND REACTION

During flight in unknown environments, collisions with the environment might occur due to unfavorable sensing conditions. In these cases, it is necessary to have a low-level collision detection and reflex strategy to minimize harm to the robot and its environment. For this purpose, the presented external wrench estimate can be exploited for the four essential phases of collision fault detection, identification, isolation, and recovery [10], [11]. *Collision detection* provides binary information whether a collision with the environment has occurred. *Collision classification* provides information about the collision type. The location of the collision is obtained through *collision isolation*. Finally, appropriate *collision recovery* mitigates danger.

A. Detection

Discrimination between aerodynamic and *collision* forces may be achieved by considering the respective signals' frequency characteristics. Following [13], collisions may be detected by applying a high-pass filter $H(f)$ on the external force [11], [13], [32] to obtain the *collision detection signal* that is
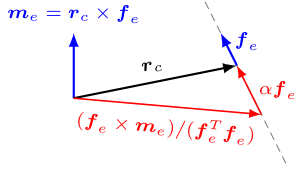
Fig. 10. We obtain the collision position $\boldsymbol{r}_c$ from the external force $\boldsymbol{f}_e$ and torque $\boldsymbol{m}_e$. The cross product $\boldsymbol{m}_e = \boldsymbol{r}_c \times \boldsymbol{f}_e$ cannot be inverted directly. This is because all forces lying on the dashed ray (42) produce the same torque. We, therefore, find the free parameter α by imposing a constraint on the solution. In our case, we limit $\boldsymbol{r}_c$ to lie on the robot's convex hull.

robust under wind influence

$$\text{CD} = \begin{cases} 1, & \text{if } \exists i : H(|\hat{f}_{e,i}|) > f_{c,i} \\ 0, & \text{otherwise} \end{cases} \quad (40)$$

where $f_{c,i}$ is the experimentally determined collision detection threshold. This signal will be robust even under wind influence, since the collision force is easy to distinguish from the aerodynamic forces based on the frequency [13].

When no wind is present, we may use a constant threshold on the estimated external force as a *collision detection signal* as discussed in [4]

$$\text{CD}_0 = \begin{cases} 1, & \text{if } \|\hat{\boldsymbol{f}}_e\| \geq f_k \\ 0, & \text{otherwise.} \end{cases} \quad (41)$$

However, in this way, both slow contacts and impacts will be detected as collisions, but the detection sensitivity will then highly depend on modeling errors and measurement noise. The signal will also be highly susceptible to wind influence. A more robust method of detecting *contacts* is to require the external wrench to be larger than a threshold f_k for a certain amount of time T_k. Ideally, the signal parameterization will provide clear class boundaries. Obviously, it is possible to create more classes. However, this goes beyond the scope of this paper.

B. Isolation—Obtaining the Collision Position

For a free-flying robot, we may safely assume that contact forces act on its hull. Upon collision, the force acts toward the robot, i.e., it is *pushing* the robot. Under a single point of contact assumption, we can reliably obtain the collision location $\boldsymbol{r}_c$ from the estimated contact wrench during the collision phase. The body-frame contact force $\boldsymbol{f}_e$ generates the torque $\boldsymbol{m}_e = \boldsymbol{r}_c \times \boldsymbol{f}_e$. This equation cannot be solved in closed form for $\boldsymbol{r}_c$. As depicted in Fig. 10, the contact location lies on the ray

$$\boldsymbol{r}_c = \left(\hat{\boldsymbol{f}}_e \times \hat{\boldsymbol{m}}_e\right) / \left(\hat{\boldsymbol{f}}_e^T \hat{\boldsymbol{f}}_e\right) + \alpha \hat{\boldsymbol{f}}_e \quad (42)$$

where the free parameter α must be found. Here, we use the high-pass-filtered wrench signal. In this way, we clearly separate collision from wind forces, thereby correctly estimating the collision position under wind influence [13]. One can find α easily, e.g., by intersecting (42) with the robot's convex hull. We assume that the contact force points into the robot; hence, $\alpha < 0$. We may now estimate a collision plane in the inertial frame
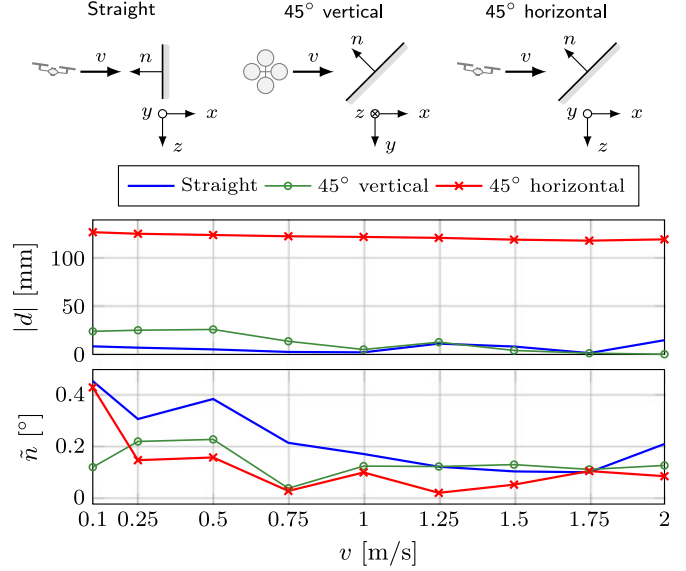

Fig. 11. Simulation of collision location determination for a quadrotor in different collision configurations. The upper plot shows the absolute distance of the estimated plane position to the real plane $d = \boldsymbol{n}^T(\boldsymbol{r}_p - \boldsymbol{o})$. The lower plot shows the angular error of the estimated normal $\tilde{n} = \text{acos } \boldsymbol{n}^T \boldsymbol{n}_p$. The mean of multiple collisions is shown for each velocity. The standard deviation in both cases is too small to be visible (2 mm and $10^{-3\circ}$, respectively).

$\boldsymbol{n}_p^T(\boldsymbol{r}_p - \boldsymbol{o}) = 0$ with the associated normal $\boldsymbol{n}_p = \hat{\boldsymbol{f}}_e/\|\hat{\boldsymbol{f}}_e\|$ and position $\boldsymbol{r}_p = \boldsymbol{r} + \boldsymbol{R}\boldsymbol{r}_c$.

To show the effectiveness of this method, a quadrotor is collided with a plane for different collision speeds in simulation. The contact with the plane occurs at a single point on the robot's convex hull. The robot geometry is approximated using only eight rectangular surfaces, which underlines the robustness of the method. The resulting plane estimation errors are depicted in Fig. 11. The estimated surface normal is very accurate for the entire velocity range and all collision configurations. This is due to the normal being estimated from the force direction, which has a high signal-to-noise ratio on collision. The position determination accuracy depends highly on the used geometric model. Even for the quite approximate geometric model, the accuracy is better than 1 cm for *straight* collisions. The maximum standard deviation of *all* collision cases is 2 mm, and it is consistent across collision velocities. This shows strong robustness and repeatability. The error of the *45° horizontal* case is caused by the geometric approximation. The approximated robot's bottom surface is already 5 cm below the actual contact point, which causes a systematic error in the position estimation.

C. Reaction

We investigate several basic collision reflex reaction strategies. A *reflex* can be described as an "involuntary" action that does not involve active planning. The x-axis trajectory of a quadrotor colliding with a stiff surface (polycarbonate) is depicted in Fig. 12. The Hunt–Crossley surface contact model and parameters are taken from [38]. The collision occurs at a velocity of 1 m/s, and the surface is located at $x = 1$ m. The quadrotor
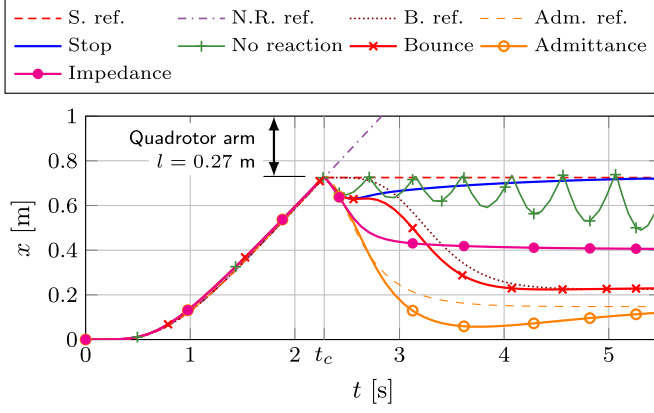
Fig. 12. Simulated robot trajectories for different collision reaction strategies. Without reaction, the robot might become unstable. A stop reaction halts the robot at the collision position. A bounce reaction with $\delta_c = 0.5$ m sets the equilibrium position in the opposite direction of the collision normal. The interaction controllers ($K_{r,x} = 1$) show very fast reaction. The system mass is $m = 0.55$ kg and $m_v = 0.25$ kg. Damping during admittance control is $d_{v,x} = 3$ N·s/m, and during impedance control $d_{v,x} = 0.25$ N·s/m.

arm is 27 cm in length and acts as an offset. Depending on the desired robot behavior, we investigate the *no reaction*, *trajectory stop*, *equilibrium bounce*, and *collision force amplification* reactions.

No reaction: The collision is not considered; hence, the quadrotor becomes unstable and may crash.

Trajectory stop: Upon collision, the trajectory is stopped at the current reference position. This halts the robot approximately at the obstacle position.

Equilibrium bounce: Upon detecting a collision, we set the equilibrium position to $\boldsymbol{r}_d = \boldsymbol{r}_k + \delta_c \hat{\boldsymbol{f}}_e^0$, where $\boldsymbol{r}_k$ is the robot position at the time of collision, and $\delta_c > 0$ is the bounce distance. This will drive the path backwards along the collision normal. The robot will then stabilize at a remote and safe distance from the obstacle.

Collision force amplification: Immediately upon collision, we instantaneously switch the controller to impedance or admittance mode. Only damping is performed, i.e., $\boldsymbol{K}_v = \boldsymbol{0}$. The input to the control system becomes the amplified collision force $\boldsymbol{\tau}_v = \boldsymbol{K}_r \hat{\boldsymbol{\tau}}_e$. For admittance control, we then have

$$\boldsymbol{M}_v \ddot{\boldsymbol{x}}_v + \boldsymbol{D}_v \dot{\boldsymbol{x}}_v = \boldsymbol{K}_r \hat{\boldsymbol{\tau}}_e \qquad (43)$$

and for impedance control

$$\boldsymbol{J}^T \boldsymbol{\tau} = (\boldsymbol{M}\boldsymbol{M}_v^{-1} - \boldsymbol{I})\boldsymbol{K}_r \hat{\boldsymbol{\tau}}_e - \boldsymbol{M}\boldsymbol{M}_v^{-1}\boldsymbol{D}_v \tilde{\boldsymbol{v}} + \boldsymbol{N}. \qquad (44)$$

This essentially produces an energy dissipation reflex. The reaction is in both cases significantly faster than trajectory-based approaches. Note that the amplified wrench $\boldsymbol{\tau}_v$ acting on the robot should be saturated to prevent instability.

D. Use Case: Takeoff and Landing Detection

When landed, the steady-state external force acting on the robot is of course its weight (see Fig. 20). The estimated force along the inertial z-direction can, therefore, be monitored to detect takeoff and landing, e.g., in the form of a threshold. We

TABLE III
SYSTEM AND CONTROL PARAMETERS USED IN THE EXPERIMENTS
WITH THE ASCTEC HUMMINGBIRD

Parameter	Value	Unit
$\mathcal{M}$	0.55	kg
$\boldsymbol{K}_1$	diag{2.25, 2.25, 5.06}	s^{-2}
$\boldsymbol{D}_1$	diag{3, 3, 4.5}	s^{-1}
$\mathcal{I}$	diag{3, 3, 4}	10^{-3} kg·m^2
c_v	256	-
$\boldsymbol{D}_2$	diag{32, 32, 32}	rad $^{-1}$·s^{-1}

can use the signals CD_0 and CD_1 along with state information to robustly determine landing events. Furthermore, the consistency of a landing could be verified from the external wrench. If the wrench acting on the robot becomes statically inconsistent, recovery behavior may be employed.

VII. EXPERIMENTAL RESULTS

In this section, we present the effectiveness of the presented methods experimentally. First, we describe the experimental setup. Then, we show the performance of the presented external force estimation method. This is followed by an impedance control experiment, where the robot is in the damping mode and can thus be freely moved in the horizontal plane. The most important result is the collision reaction and location experiment, which showcases the main contribution of this paper. Finally, we outline how collision reflexes and tactile mapping may be included in the autonomy stack of a flying robot.

A. Setup

We performed experiments on three different flying systems. The propulsion model and the external wrench estimator were verified on a custom-built hexacopter with coaxial propeller pairs, in a configuration as described in [25]. Collision detection, reaction, and interaction control experiments were performed on an AscTec Hummingbird quadrotor, as described in [30], using parameters from Table III. Finally, the tactile mapping experiment was performed on an autonomous AscTec Pelican quadrotor, as described in [2].

B. External Wrench Estimation

In order to validate the external wrench estimation, as well as the propulsion model, we fixed a coaxial hexacopter to an ATI 85 Mini force-torque sensor, as depicted in Fig. 13. The pose of the hexacopter and force-torque sensor are obtained by a motion capture system at 250 Hz, while the onboard attitude controller runs at 500 Hz. During the test, we logged the pose, IMU, motor speed, and current as measured by the speed controllers, the commanded control input, and the force and torque. From this, we identified the system parameters, as listed in Table IV. Note the different motor constants between upper and lower propellers. This model best describes the measurements and is most likely due to unmodeled interaction effects in the coaxial pairs. Interestingly, upper propellers generate less thrust (C_t) and require more power (C_q) in comparison to the lower propellers.
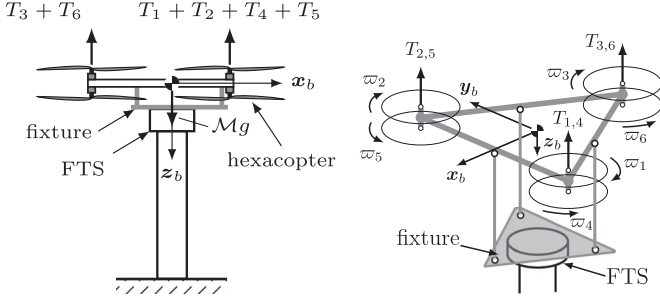
Fig. 13. Setup of the force-torque-sensor experiment. We fixed a hexacopter with coaxial propeller pairs to a force-torque sensor. The lower and upper propellers have different thrust and torque coefficients due to interaction effects.

TABLE IV
SYSTEM PARAMETERS IDENTIFIED IN THE FORCE-TORQUE-SENSOR EXPERIMENT

Parameter		Value
Mass	$\mathcal{M}$	2.445 kg
Propeller diameter	D	0.254 m
Thrust coefficient	C_t	$[5.1137, 7.8176] \cdot 10^{-2}$
Torque coefficient	C_q	$[7.5183, 4.7597] \cdot 10^{-3}$ -
Motor torque coef.	$K_{q,0}$	$[2.9404, 1.4545] \cdot 10^{-2}$ N·m/A
Motor torque coef.	$K_{q,1}$	$[-1.4099, -3.3360] \cdot 10^{-3}$ N·m/A²
Rotor inertia	I_r	$2.1748 \cdot 10^{-4}$ kg m²
Inertia	$\mathcal{I}$	diag{3.06, 3.16, 4.50} $\cdot 10^{-2}$ kg·m²
Center of gravity	r_g	$[4.39, -1.18, 6.42]^T \cdot 10^{-4}$ m

Coaxial propeller pair coefficients are written as [upper, lower]. The inertia and center of gravity position were identified using data from an identification flight and identified propulsion parameters.

The inertia and center of gravity were identified from flight data, and using the identified propulsion model.

Fig. 14 shows the external wrench estimation on the hexacopter while rigidly fixed to the force-torque sensor. As expected, the identified wrench is the propulsion wrench because of the fixture. In this case, using the measured motor speeds to obtain the control wrench shows a negligible effect. Hence, using only the commanded motor speeds to obtain the control input is sufficient for most of the applications. Using motor feedback is beneficial in the case of actuator failure.

Fig. 15 shows the external yaw torque estimation using three different propulsion models. The most widely used model uses only the motor speed and is shown as $\hat{m}_{e,z}^{\varpi}$. Here, accuracy is lost during fast transitions due to the rotor acceleration torque. Adding also rotor acceleration ($\hat{m}_{e,z}^{\varpi,\dot{\varpi}}$) fixes this issue. Finally, the motor torque may be obtained from the measured motor current, shown as $\hat{m}_{e,z}^{i_a}$. In this case, we do not need a propeller model, and the accuracy is similar to the model using rotor acceleration. Note that in the case of actuator failure (e.g., losing a propeller), using the motor current will provide a better estimate, as the motor torque model is not affected by the propeller.

C. Impedance Control

An impedance controller for the position subsystem is implemented (see Fig. 5). The rotational inertia is unchanged, i.e.,
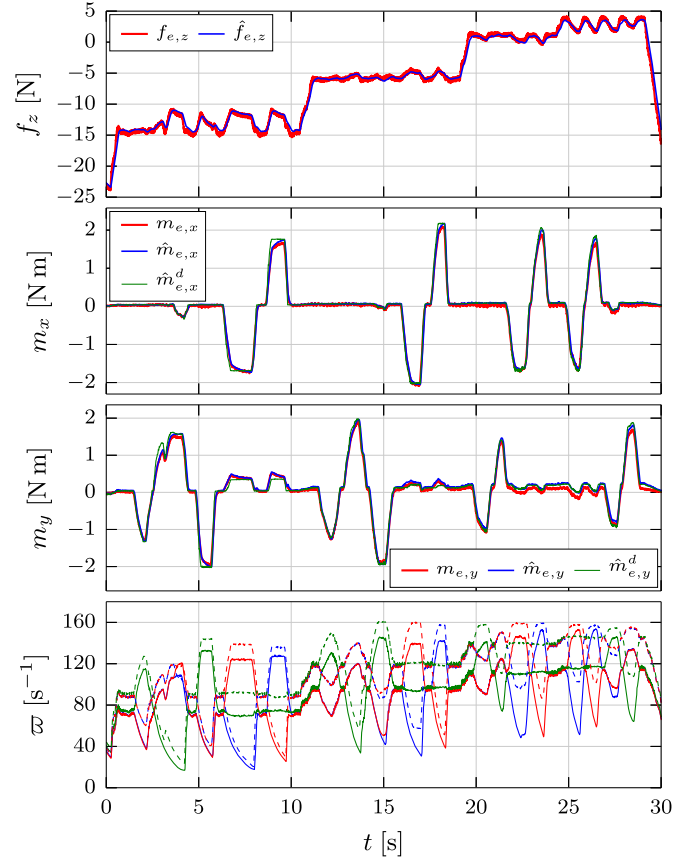

Fig. 14. Validation of the external wrench estimator on a force-torque sensor using motor feedback on a coaxial hexacopter as depicted in Fig. 13, using $K_{i,f} = 10$. For the torques, we also show the estimate using commanded torques in green ($m_{e,x}^d$ and $m_{e,y}^d$). The control input τ is obtained using the measured motor speeds, shown in the bottom plot, and the propulsion model as presented in Section III. Note that there is a small error on fast changes due to slow motor dynamics when braking. The measured motor speeds shown in solid (—, —, —) represent the upper propellers, and in dashed (- -, - -, - -) the lower propellers. An improvement of this motor speed tracking is commonly achieved by active braking. This experiment also validates the propulsion wrench model, as the robot is fixed to the force-torque sensor; hence, $\tau_e = -\tau$.
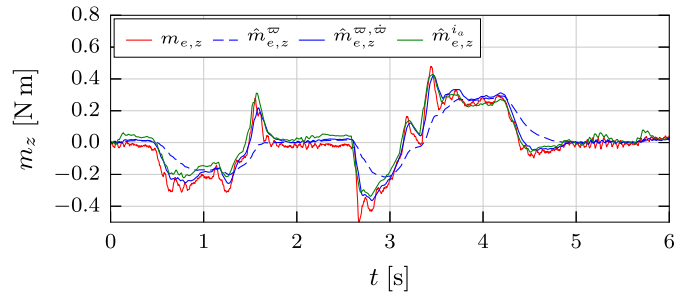

Fig. 15. Estimation of the external yaw torque, using the measured motor speed only ($m_{e,z}^{\varpi}$), using the motor speed and rotor acceleration ($m_{e,z}^{\varpi,\dot{\varpi}}$), and using motor current ($m_{e,z}^{i_a}$). The estimator gain is $K_{i,m} = 36$ to make the signals more discernible. Using the rotor acceleration improves accuracy during fast changes of the desired torque, whereas the motor current provides direct information about the torque. Although susceptible to noise, the motor current method is less sensitive to, e.g., propeller failures, as it does not depend on the rotor inertia.
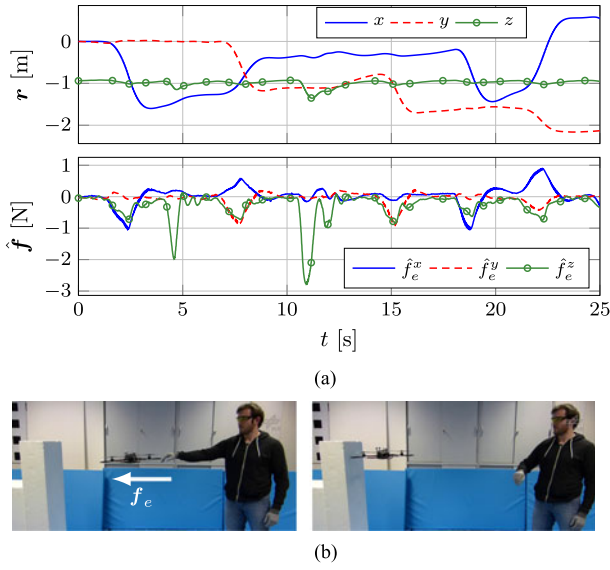
(a)

(b)

Fig. 16. Trajectory and estimated external force in the impedance controller experiment with $m_v = 0.25$ kg, $\boldsymbol{D}_v = \text{diag}\{1, 1, 1\}$ N/s, $\boldsymbol{K}_v = \text{diag}\{0, 0, 2\}$ N/m [see (a)]. The quadrotor can be freely moved along the inertial x- and y-axes [see (b)]. (a) Position response and estimated external force of the impedance controller during interaction without collisions. (b) Video stills of an interaction experiment. The human pushes the robot, which causes a collision. The secondary collision can be recovered from safely because the interaction controller amplifies the collision force; as a result, the robot bounces away from the obstacle.

$\mathcal{I}_v = \mathcal{I}$. During the experiment, the controller was in the damping mode in the inertial (x, y) plane, and a human applied a force to the robot. Fig. 16 depicts the position response and estimated force during interaction. Due to the spring term, the altitude is controlled to 1 m, while the external force causes a proportional altitude error as expected. The robot may be freely moved in the horizontal axes by applying a force, as expected. The energy injected by a collision is dissipated by the impedance controller's damping term. A numerical analysis of the local ISS conditions for this experiment is given in the Appendix.

D. Inertia Shaping

Fig. 17 shows the altitude response of the quadrotor with different virtual masses to the same external load. We clearly see that inertia shaping changes the effect that the external force has on the robot. At $t = 0$ s [see Fig. 17(b)], a 100-g weight (1 N) is attached to the robot in hover, so that a constant external force is acting on it. At $t = 2$ s [see Fig. 17(c)], we can see that a lower inertia results in a faster response to the load, as expected. At $z \approx 0.2$ m, the robot hits the ground, which is covered by a soft surface. This causes slight bouncing and oscillations in the external force.

E. Collision Reaction and Location

A polystyrene block was placed on the robot's flight path and used it as an obstacle. Fig. 18 depicts the position response of different collision reaction schemes along the inertial x-direction. The obstacle position did not change between the experiments. Video stills of an experiment are shown in Fig. 19. We used the
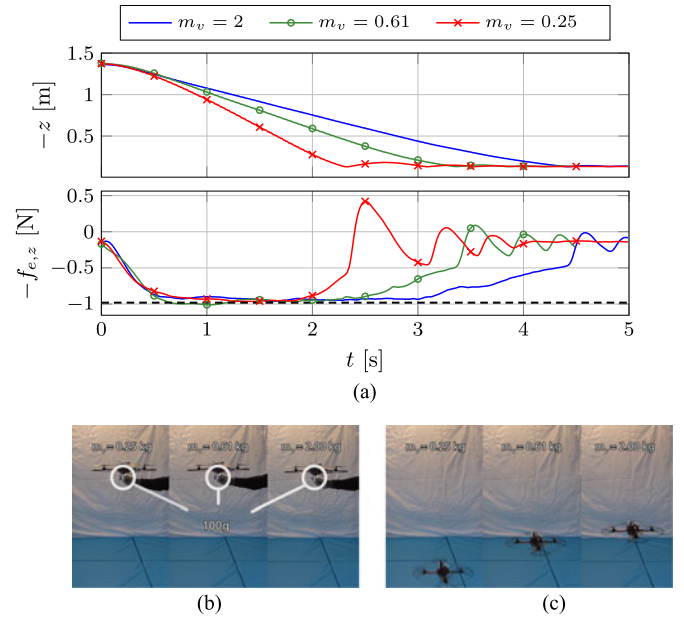

(a)

(b)   (c)

Fig. 17. Altitude response and estimated external force for different virtual masses, showing the effect of inertia shaping. At $z \approx 0.2$ m, the robot hits the ground, which is covered by a soft surface. This causes slight bouncing and oscillations in the external force. (a) Altitude and external force for inertia shaping. The dashed line shows the ground truth weight of 100 g (0.981 N). (b) $t = 0$ s. (c) $t = 2$ s.
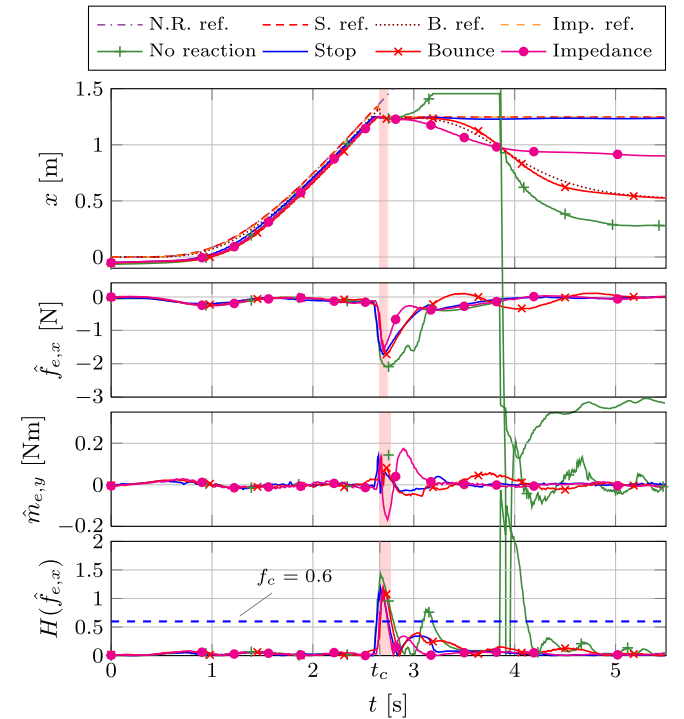

Fig. 18. Collision reaction experiment with a polystyrene block. From top to bottom: position, estimated external force, estimated external pitch torque, and collision detection signal. The force estimate is filtered with $K_l^f = 3$ rad/s, and the torque with $K_l^m = 12$ rad/s. The collision detection threshold is set to $f_c = 0.6$. Collisions are detected at $\approx t_c$. The red shaded area depicts the contact phase. The quadrotor crashed when the collision was not accounted for. All reaction schemes stabilize the robot upon collision. Impedance mode with collision force augmentation provides the fastest response. Here, we used $m_v = 0.25$ kg, $d_v = 0.5$ N·s/m, and $K_{r,x} = 1$.
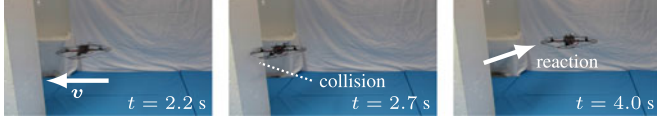
Fig. 19. Video stills of the equilibrium bounce collision reaction experiment depicted in Fig. 18. From left to right: shortly before impact ($t = 2$ s), at impact ($t = 2.7$ s), and stable state after the impact ($t = 4$ s). Upon collision, the quadrotor is brought to a safe and stable state.

TABLE V
ESTIMATED OBSTACLE LOCATION AND ASSOCIATED SURFACE NORMAL IN THE COLLISION REACTION EXPERIMENT

Reaction	x	y	z	n_x	n_y	n_z
No reaction	1.129	−0.339	−0.841	−0.964	0.051	−0.262
Stop	1.140	−0.335	−0.864	−0.983	−0.006	−0.180
Bounce	1.139	−0.242	−0.851	−0.982	−0.072	−0.175
Impedance	1.145	−0.304	−0.834	−0.989	0.034	−0.145

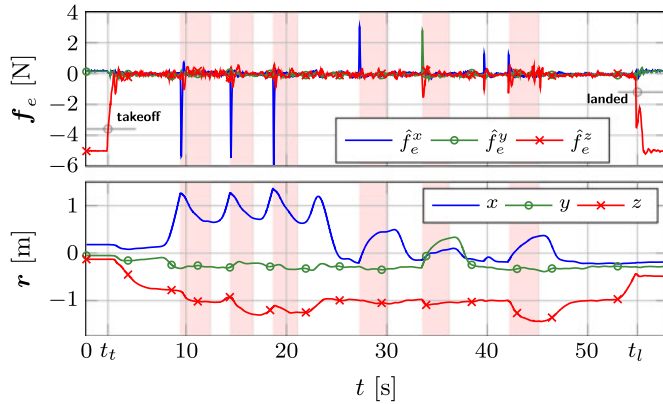The position is expressed in meters, in the inertial frame.

Fig. 20. Flight with multiple collisions. *Equilibrium bounce* reactions are indicated by shaded areas. Normal flight is resumed after stabilization. After $t = 25$ s, the quadrotor hovers and is hit by a human. Landed phases are clearly visible from $\hat{f}_{e,z}$. Takeoff is detected at $t = t_t$, and landing at $t = t_l$. External force estimation in flight was performed with $K_I^f = 50$.

(a)

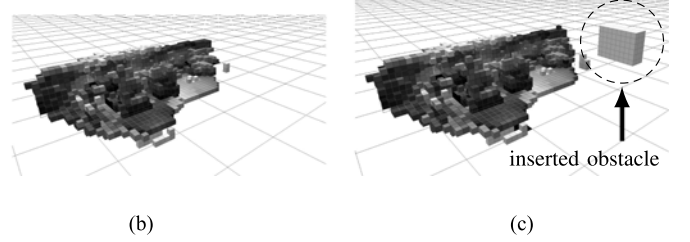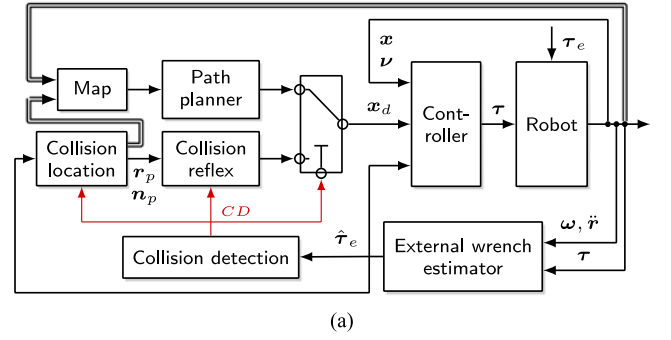(b)   (c)

Fig. 21. Tactile mapping application. The collision reaction is a low-level reflex. Upon detecting a collision, we add an object at the estimated contact position and orientation to the onboard octomap and replan to avoid further collisions. This assumes collisions with a static environment. (a) Implementation of the collision reflex in an autonomous control structure. The collision reflex is independent of the path planner, and aids environment mapping parallel to exteroceptive sensors. (b) Octomap before collision. (c) Octomap after collision—the inserted collision object is visible.
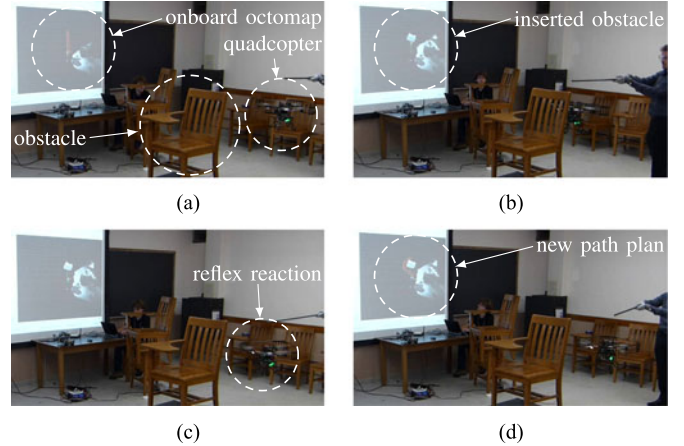

(a)   (b)

(c)   (d)

Fig. 22. Video stills from a tactile mapping experiment. The quadcopter [2] flies a predefined set of waypoints while not looking in the direction of travel. At $t = 0$ s, a straight-line path through the unknown obstacle is planned and executed. At $t = 4$ s, the quadcopter collides with the chair, adds a collision obstacle into the octomap and starts executing the collision reflex. At $t = 6$ s, the collision reflex is finished (timeout), and at $t = 7$ s, a new plan around the newly added obstacle is planned. (a) $t = 0$ s. (b) $t = 4$ s. (c) $t = 6$ s. (d) $t = 7$ s.

high-pass-filtered external force as a collision detection signal. Even for a highly filtered (delayed) external force, the presented schemes are still effective. Without a reaction scheme, the robot crashed. The trajectory-based schemes (stop, bounce) successfully stabilize the quadrotor after a collision. Switching to impedance damping mode provides the fastest and smoothest response to the collision. The estimated obstacle locations are shown in Table V. The obstacle plane estimation is precise and consistent across experiments.

Finally, Fig. 20 shows a complete flight with multiple *bounce* collision reactions. Individual events can clearly be seen from the external forces. The first three reactions were due to the quadrotor hitting a static obstacle. Subsequent reactions are caused by a human hitting the robot during hovering. Clearly, the method's sensitivity is evident, as it reacts also to small peak forces. Takeoff and landing events can be clearly recognized from the magnitude of the external force.

F. Tactile Mapping

Following a collision reflex, information about the identified collision location may be built into an obstacle map. Such tactile mapping complements mapping by exteroceptive sensation. For carrying out the experiment, the hybrid external force estimator and the equilibrium bounce collision reaction strategy were

implemented on an Asctec Pelican quadrotor (see Figs. 21 and 22). The quadrotor [2] navigates using IMU–stereo fusion only and builds an onboard *octomap*, which is used for subsequent path planning. Our onboard cameras purposefully look 90° rotated from the flight direction, so that the polystyrene-block obstacle cannot be seen. Upon collision, a $60 \times 60 \times 30$ cm obstacle is added to the octomap at the position and orientation of the collision, estimated via (42). We subsequently apply an A* path planner, which uses the map information obtained from the collision, to replan around the collided obstacle.

VIII. CONCLUSION

In this paper, we have solved the problem of collision detection and safe reaction for flying robots without the need of additional sensors. This is enabled by a novel hybrid method for obtaining external wrench information. It uses proprioceptive measurements from the onboard IMU, a dynamic model, and the control input only. Our method is independent of any particular controller and the desired trajectory. Furthermore, it does not require translational velocity measurements. Collision detection, isolation, classification, and reaction make different use of the complete external wrench information. This additionally allows for mapping the environment upon collisions. The collision information was exploited in a fully autonomous flight scenario. Furthermore, we developed a set of interaction control schemes based on well-established admittance and impedance concepts. We have presented a thorough stability analysis of the impedance controller that uses the external wrench estimate in closed-loop form.

To sum up, we have investigated interaction controllers and collisions with the environment of flying robots under a unified framework. We show that sensitive interaction control coupled with fast collision reflexes significantly enrich the robot's behavior and are a significant step toward enabling flying robots to operate safely and robustly in cluttered environments.

APPENDIX
ISS NUMERICAL EXAMPLE

The following is a numerical example of the local ISS conditions for the impedance control experiment shown in Fig. 16, parameters from Table III , and $c_1 = 0.1$, $c_2 = 0.01$ taken numerically such that $\boldsymbol{P}$ and $\boldsymbol{Q}$ are positive definite. Because the x- and y-position stiffness is zero, these position coordinates are obviously unbounded. We, therefore, omit them from the state for this analysis. Note that the analysis includes the x- and y-translational velocities nonetheless. With $\boldsymbol{K}_{i,m} = 30\boldsymbol{I}$, we obtain the numerical values

$$\boldsymbol{P} = \begin{bmatrix} 2 & 0.1 & \cdot & \cdot & \cdot & \cdot \\ 0.1 & 0.25 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0.1\,\boldsymbol{I} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & c_v\,\boldsymbol{I} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \boldsymbol{\mathcal{I}} & 0.01\,\boldsymbol{I} \\ \cdot & \cdot & \cdot & \cdot & 0.01\,\boldsymbol{I} & 30\,\boldsymbol{I} \end{bmatrix} \quad (45)$$

$$\boldsymbol{Q} = \begin{bmatrix} 0.80 & 0.20 & 0.12 & 1.97 & 0 & 0 \\ 0.20 & 0.90 & 0.30 & 4.92 & 0 & 0 \\ 0.12 & 0.30 & 1.00 & 0 & 0 & 0 \\ 1.97 & 4.92 & 0 & 853.33 & 53.33 & 0 \\ 0 & 0 & 0 & 53.33 & 32 & 0 \\ 0 & 0 & 0 & -0 & -0 & 1 \end{bmatrix}. \quad (46)$$

Their eigenvalues (with multiplicity in braces) are

$$\text{eig}(\boldsymbol{P}) = \left(256^{(3)}, 30^{(3)}, 2.006, 0.244, 0.1^{(3)}, 0.004, 0.003^{(2)}\right)$$

$$\text{eig}(\boldsymbol{Q}) = \left(856.81, 28.55, 1.32, 1, 0.75, 0.59\right)$$

and are obviously positive, confirming the local ISS conditions from Proposition 3.

REFERENCES

[1] K. Schmid, T. Tomić, F. Ruess, H. Hirschmüller, and M. Suppa, "Stereo vision based indoor/outdoor navigation for flying robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 3955–3962.
[2] K. Schmid, P. Lutz, T. Tomić, E. Mair, and H. Hirschmüller, "Autonomous vision-based micro air vehicle for indoor and outdoor navigation," *J. Field Robot.*, vol. 31, no. 4, pp. 537–570, 2014. [Online]. Available: http://dx.doi.org/10.1002/rob.21506
[3] A. Klaptocz, A. Briod, L. Daler, J.-C. Zufferey, and D. Floreano, "Euler spring collision protection for flying robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1886–1892.
[4] A. Briod *et al.*, "Contact-based navigation for an autonomous flying robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 3987–3992.
[5] S. Bellens, J. De Schutter, and H. Bruyninckx, "A hybrid pose/wrench control framework for quadrotor helicopters," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 2269–2274.
[6] H.-N. Nguyen and D. Lee, "Hybrid force/motion control and internal dynamics of quadrotors for tool operation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Tokyo, Japan, Nov. 2013, pp. 3458–3464.
[7] S. Jung, "A position-based force control approach to a quad-rotor system," in *Proc. 9th Int. Conf. Ubiquitous Robots Ambient Intell.*, 2012, pp. 373–377.
[8] 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. Intell. Robots Syst.*, 2012, pp. 3532–3537.
[9] M. Fumagalli and R. Carloni, "A modified impedance control for physical interaction of UAVs," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1979–1984.
[10] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical Human-Robot Interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 3356–3363.
[11] S. Haddadin, *Towards Safe Robots—Approaching Asimov's 1st Law* (ser. Springer Tracts in Advanced Robotics), vol. 90. Berlin, Germany: Springer-Verlag, 2014.
[12] T. Tomić and S. Haddadin, "A unified framework for external wrench estimation, interaction control and collision reflexes for flying robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 4197–4204.
[13] T. Tomić and S. Haddadin, "Simultaneous estimation of aerodynamic and contact forces in flying robots: Applications to metric wind estimation and collision detection," in *Proc. IEEE Int. Conf. Robot. Autom.*, Seattle, WA, USA, May 2015, pp. 5290–5296.
[14] C. Ott, R. Mukherjee, and Y. Nakamura, "A hybrid system framework for unified impedance and admittance control," *J. Intell. Robot. Syst.*, vol. 78, pp. 359–375, Jun. 2015.

[15] F. Ruggiero, J. Cacace, H. Sadeghian, and V. Lippiello, "Impedance control of VToL UAVs with a Momentum-based external generalized forces estimator," in *Proc. IEEE Int. Conf. Robot. Autom.*, Hong Kong, Jun. 2014, pp. 2093–2099.

[16] F. Ruggiero, J. Cacace, H. Sadeghian, and V. Lippiello, "Passivity-based control of VToL UAVs with a momentum-based estimator of external wrench and unmodeled dynamics," *Robot. Auton. Syst.*, vol. 72, pp. 139–151, 2015.

[17] B. Yüksel, C. Secchi, H. H. Bülthoff, and A. Franchi, "Reshaping the physical properties of a quadrotor through IDA-PBC and its application to aerial physical interaction," in *Proc. IEEE Int. Conf. Robot. Autom.*, Hong Kong, Jun. 2014, pp. 6258–6265.

[18] B. Yüksel, C. Secchi, H. H. Buelthoff, and A. Franchi, "A nonlinear force observer for quadrotors and application to physical interactive tasks," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, Besançon, France, Jul. 8–11, 2014, pp. 433–440.

[19] C. D. McKinnon and A. P. Schoellig, "Unscented external force and torque estimation for quadrotors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2016, pp. 5651–5657.

[20] F. Augugliaro and R. D'Andrea, "Admittance control for physical human-quadrocopter interaction," in *Proc. Eur. Control Conf.*, Zürich, Switzerland, Jul. 2013, pp. 1805–1810.

[21] A. Albers, S. Trautmann, T. Howard, T. A. Nguyen, M. Frietsch, and C. Sauter, "Semi-autonomous flying robot for physical interaction with environment," in *Proc. IEEE Conf. Robot. Autom. Mechatronics*, 2010, pp. 441–446.

[22] F. Forte, R. Naldi, A. Macchelli, and L. Marconi, "Impedance control of an aerial manipulator," in *Proc. Amer. Control Conf.*, 2012, pp. 3839–3844.

[23] A. Y. Mersha, S. Stramigioli, and R. Carloni, "Variable impedance control for aerial interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 3435–3440.

[24] R. Naldi, A. Torre, and L. Marconi, "Robust blind navigation for a miniature ducted-fan aerial robot," in *Proc. Amer. Control Conf.*, 2013, pp. 988–993.

[25] T. Tomić, K. Schmid, P. Lutz, A. Mathers, and S. Haddadin, "The flying anemometer: Unified estimation of wind velocity from aerodynamic power and wrenches," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Daejeon, South Korea, Oct. 2016, pp. 1637–1644.

[26] S. Omari, M.-D. Hua, G. Ducard, and T. Hamel, "Hardware and software architecture for nonlinear control of multirotor helicopters," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 6, pp. 1724–1736, Dec. 2013.

[27] T. Lee, M. Leoky, and N. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. 49th IEEE Conf. Decision Control*, Dec. 2010, pp. 5420–5425.

[28] G. M. Hoffmann, S. L. Wasl, and C. J. Tomlin, "Quadrotor helicopter trajectory tracking control," in *Proc. AIAA Guid., Navigat., Control Conf.*, 2008, Art. no. AIAA 2008-7410.

[29] M.-D. Hua, T. Hamel, P. Morin, and C. Samson, "Introduction to feedback control of underactuated VTOL vehicles: A review of basic control design ideas and principles," *IEEE Control Syst.*, vol. 33, no. 1, pp. 61–75, Feb. 2013.

[30] T. Tomić, "Evaluation of acceleration-based disturbance observation for multicopter control," in *Proc. Eur. Control Conf.*, Jun. 2014, pp. 2937–2944.

[31] M. Bangura, H. Lim, H. J. Kim, and R. Mahony, "Aerodynamic power control for multirotor aerial vehicles," in *Proc. IEEE Int. Conf. Robot. Autom.*, Hong Kong, Jun. 2014, pp. 529–536.

[32] A. De Luca, A. Albu-Schaffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2006, pp. 1623–1630.

[33] D. Honegger, L. Meier, P. Tanskanen, and M. Pollefeys, "An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 1736–1741.

[34] J. Nikolic *et al.*, "A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, Hong Kong, May 31–Jun. 7, 2014, pp. 431–437.

[35] N. Hogan, "Impedance control - An approach to manipulation. I—Theory. II—Implementation. III—Applications," *ASME Trans., J. Dyn. Syst. Meas. Control B*, vol. 107, pp. 1–24, Mar. 1985.

[36] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[37] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA, USA: SIAM, 1994.

[38] N. Diolaiti, C. Melchiorri, and S. Stramigioli, "Contact impedance estimation for robotic systems," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 925–935, Oct. 2005.

Teodor Tomić received the master's degree in mechanical engineering from University of Zagreb, Zagreb, Croatia, in 2010.

From 2008 to 2010, he was a GNC System Engineer on modeling and control of an omnidirectional airship with HiperSfera, Zagreb. From 2011 to 2017, he was a Research Scientist with Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Weßling, Germany, where he worked on control for flying robots. Since 2017, he has been a Planning and Controls Software Engineer with Skydio, Redwood City, CA, USA, working on fully autonomous flying robots. His main research interests include nonlinear control, trajectory generation, interaction control under wind influence, software architectures, and full autonomy for flying robots.

Christian Ott received the Dipl.-Ing. degree in mechatronics from Johannes Kepler University, Linz, Austria, in 2001 and the Dr.-Ing. degree in control engineering from Saarland University, Saarbrücken, Germany, in 2005.

From 2001 to 2007, he was with Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Weßling, Germany. From May 2007 to June 2009, he was a Project Assistant Professor with the Department of Mechano-Informatics, University of Tokyo, Tokyo, Japan. Since then, he has been with DLR as a Team Leader of the Helmholtz Young Investigators Group for "Dynamic Control of Legged Humanoid Robots." In January 2014, he became the Head of the Department of Analysis and Control of Advanced Robotic Systems, DLR. His research interests include nonlinear robot control, flexible joint robots, impedance control, and control of humanoid robots.

Sami Haddadin (M'10) received degrees in electrical engineering, computer science, and technology management from Technische Universität München, München, Germany, and Ludwig-Maximilians Universität München, München, and the Ph.D. degree from RWTH Aachen University, Aachen, Germany.

He is a Full Professor and Director of the Institute of Automatic Control, Leibniz Universität Hannover, Germany. He organized/edited several international robotics conferences and journals and authored/coauthored more than 120 scientific articles. His research topics include physical human–robot interaction, nonlinear robot control, real-time motion planning, real-time task and reflex planning, robot learning, optimal control, human motor control, variable impedance actuation, and safety in robotics.

Dr. Haddadin received numerous awards at the top international robotics conferences and journals, including the 2015 IEEE/RAS Early Career Award, the 2015 RSS Early Career Spotlight, and the 2015 Alfried Krupp Award. He was selected in 2015 and 2016 as 2015 Capital Young Elite Leader under 40 in Germany for the domain "Politics, State & Society."