Chapter 5
Simulation and Experimental Results

5.1 Testbed

In the following we shall present the simulation and hardware setups used for the validation of the proposed methods.

5.1.1 Simulation Setup

All the simulations are carried out using Matlab Simulink and considering an aerial vehicle with mass $m_R = 1$[Kg] and inertia $\mathbf{J}_R = 0.25\mathbf{I}_3$[Kg m^2]. We assume constant link actuator radius and inertia equal to $r_W = 0.2$[m] and $J_W = 0.15$[Kg m^2], respectively. The values of the gains and of the desired trajectories are specified in the following, since they are different for each controller.

5.1.2 Hardware Setup

The unidirectional-thrust aerial vehicle used for the experiments consists of a Quadrotor VTOL (see Fig. 5.1), weighting about 1[Kg]. The hardware of the vehicle is the one of a *Mikrokopter*[1] quadrotor. It is endowed with an IMU, and four brushless motor controllers (BLDC ESC) regulating the propeller speed using an in-house developed closed-loop speed controller [1].

We tested our controller with two different setups:

(a) In the first, the quadrotor is linked to a fixed point on the ground by a rigid structure made by carbon-fiber bars and 3D printed parts (see Fig. 5.1a). The

[1] http://www.mikrokopter.de.

M. Tognon and A. Franchi, *Theory and Applications for Control of Aerial Robots in Physical Interaction Through Tethers*, Springer Tracts in Advanced Robotics 140,
https://doi.org/10.1007/978-3-030-48659-4_5

(a) Quadrotor tethered by a bar-link. (b) Quadrotor tethered by a cable-link.

Fig. 5.1 Test bed used for testing the hierarchical controller for a tethered aerial vehicle. On the left the robot is compressing the bar with a force equal to -12 [N], being in an inclined hovering with $\vartheta_A = -80°$. On the right the robot is pulling the cable with a force equal to 7 [N], being in an inclined hovering with $\vartheta_A = 30°$

system implements the reduced model described in Sect. 4.3. In fact, the bar constraints the vehicle to fly on a 2D vertical plane, but, at the same time, does not constrain the vehicle orientation along the $\mathbf{y}_R$ axes. The structure has been designed such that the quadrotor can freely rotate between the two lateral bars without touching them with the propellers. Furthermore the axis of rotation has been brought as close as possible to the vehicle center of mass. Although this setup constraints the vehicle on a reduced space, actually it allows the aerial vehicle to exert on the link both tension and compression.

(b) In the second, the quadrotor is equipped with a light cable with fixed length, ending with a triple hook that can be anchored to a platform (see Fig. 5.1b). The other end of the link is attached to the vehicle as close as possible to its CoM. Once the cable is made taut, the tethered quadrotor can fly on a sphere but can exert only tension on the link.

In both cases the link has a length of 1[m] while a mass of 0.13[Kg] and less than 0.01[Kg] for the first and second setup, respectively, thus negligible w.r.t. the vehicle one.

The control law, implemented in Matlab–Simulink, runs on a desktop PC sending the commanded propeller velocities at 500 [Hz] through a serial communication. The control loop is then closed based on the measurements of: (i) the position and attitude of the vehicle provided at 1 [kHz] by a UKF that fuses the Motion Capture (Mo-Cap) system measurements at 120 [Hz] with the IMU measurements at 1 [kHz]; (ii) the linear and angular velocities of the vehicle provided by the same UKF filter at 1 [KHz].

Table 5.1 Validating experiments

Test bed	Controller	Section	#	Description	Figure
Bar-link (Fig. 5.1a)	Γ^a_{HC}	5.2.1	1	Quasi static conditions, initial tension to vertical compression	5.2
			2	Dynamic trajectory with vertical compression	5.3
Cable-link (Fig. 5.1b)	Γ^b_{HC}	5.2.2	3	Time varying trajectory on φ	5.4
			4	Time varying trajectory on ϑ_A	5.5
			5	Time varying trajectory on both φ and ϑ_A	5.6

5.2 Hierarchical Controllers: Experimental Validation

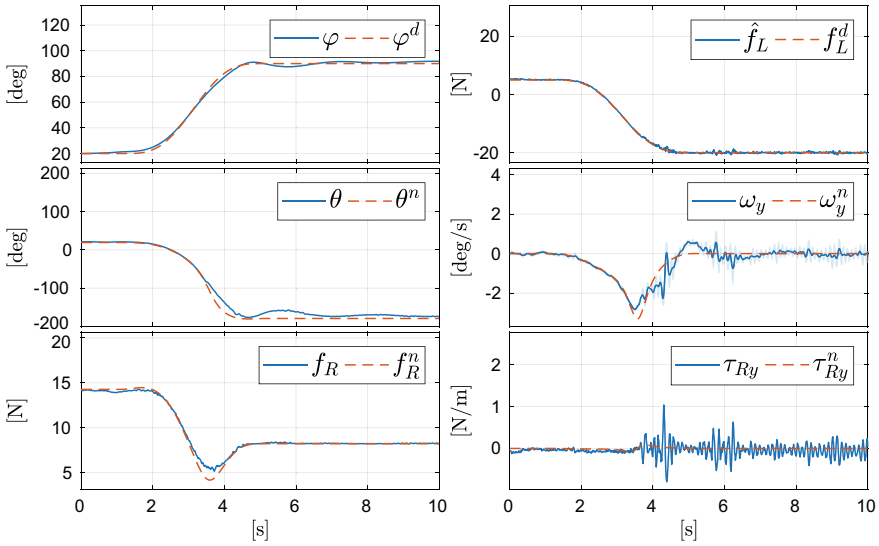
In order to validate and test the performance of the proposed hierarchical controllers we tried to track some dynamical trajectories showing the ability to independently track the entries of $\mathbf{y}^a$ and $\mathbf{y}^b$. The validation has been done by real experiments with the platform described in Sect. 5.1.2, where the other end of the link has been anchored to a very heavy load (much more than the total lifting of the vehicle). Figure 5.1 shows the corresponding test beds.

Table 5.1 gathers the executed experiments specifying the corresponding test bed, controller and giving a short description. The reader can choose to go directly to the section of a specific experiment or to the corresponding plots.

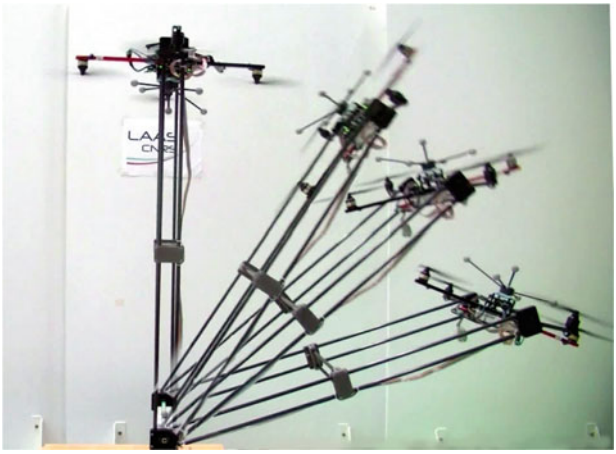
5.2.1 Hierarchical Controller for $\mathbf{y}^a$

For these first tests we use the setup of Fig. 5.1a in order to show the ability of the robot to apply forces to the link, and so to the ground, that go from tension to compression and vice-versa, while changing its position. We shall show such capability in two conditions:

(1) The first is a quasi static condition. The robot is asked to follow a smooth trajectory $\mathbf{y}^{ad}(t) = [\varphi^d(t) \;\; 0 \;\; 0 \;\; {f_L}^d(t)]^\top$ with $t \in [0, T]$ where T is the final time and $\varphi^d(0) = 20$ [°], $\varphi^d(T) = 90°$, ${f_L}^d(0) = 5$ [N], ${f_L}^d(T) = -20N$. Figure 5.2 shows the corresponding results and performed motion. One can notice how the robot is able to keep the bar vertical while pushing it. Since the desired compression is grater then the gravitational one, the robot has to turn and push the bar with an upside down orientation. Even in this unusual configuration for a standard qudrotor, the controlled system remains stable.



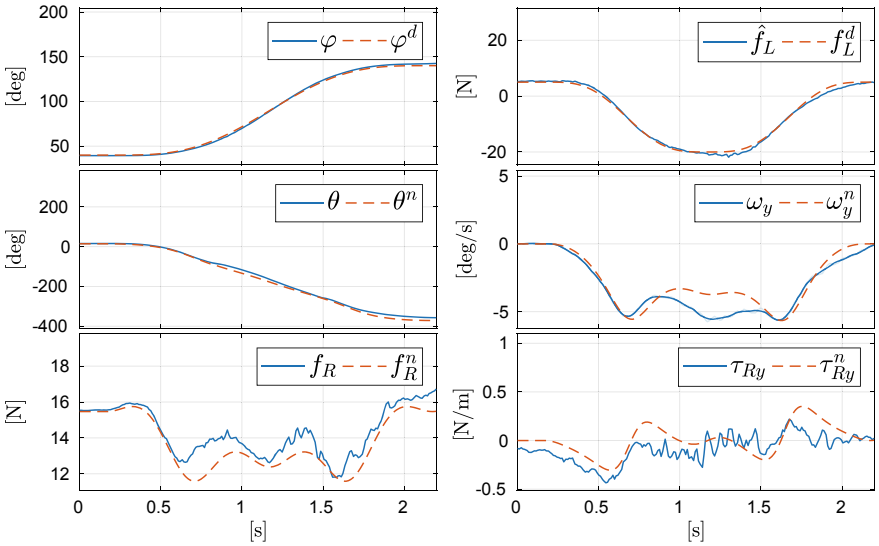
(a) Outputs, state and inputs evolution. Only the variables along the axes of motions are shown.



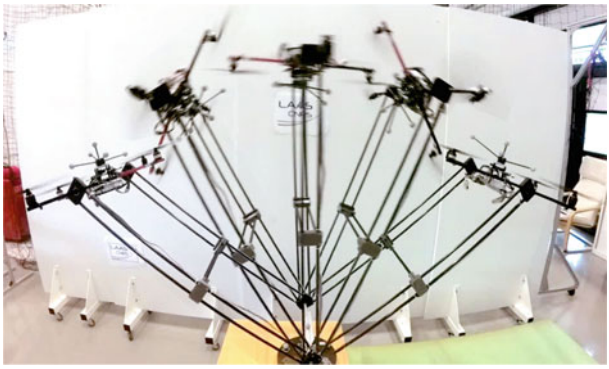
(b) Sequence of snapshots of the experiment.

Fig. 5.2 Results of the the experiment 1 with the hierarchical controller for $\mathbf{y}^a$

(2) The second is a more dynamic trajectory. Like before, $\mathbf{y}^{ad}(t) = [\varphi^d(t)\ 0\ 0\ {f_L}^d(t)]^\top$ is such that $\varphi^d(0) = 40$ [°], $\varphi^d(T/2) = 90°$, $\varphi^d(T/2) = 140°$, and ${f_L}^d(0) = 5$ [N], ${f_L}^d(T/2) = -20N$, ${f_L}^d(T) = 5$ [N]. Figure 5.3 shows the corresponding results and performed motion. In Fig. 5.3b one can notice that to follow the desired trajectory the robot has to flip performing a turn of more than 380 [°] along $\mathbf{y}_R$. We remark that this is not something pre-planned but it is a by-product of the desired internal force trajectory. Although the trajectory is really



(a) Outputs, state and inputs evolution. Only the variables along the axes of motions are shown.



(b) Sequence of snapshots of the experiment.

Fig. 5.3 Results of the the experiment 2 with the hierarchical controller for $\mathbf{y}^a$

acrobatic, the controlled vehicle is able to track the trajectory with sufficient precision.

5.2.2 Hierarchical Controller for $\mathbf{y}^b$

Since our control methods works only for the tethered system (nonzero internal force), a pre-tensioning phase is needed. During this phase, the robot is controlled

with a standard position controller trying to reach a position outside of the feasible sphere. As soon as, at time t_0 the link is taut (detectable using a threshold in the position error) the controller is activated.

In the following we shall show the results of the control action for three different sinusoidal trajectories:

(1) sinusoidal trajectory with time varying frequency on φ while keeping ϑ_A constant,
(2) sinusoidal trajectory with time varying frequency on ϑ_A while keeping φ constant, and
(3) sinusoidal trajectory with fixed frequency on both φ and ϑ_A,

while δ and ψ are kept constantly to zero. We recall that since there is not a link actuator the length of the link cannot be controlled. The first two tests are done firstly to see that the proposed controller can track a desired trajectory on φ or ϑ_A, independently. Secondly we want to show which is the maximum feasible frequency for both dynamics. Notice that with the validation of controller Γ^b_{HC} for the tracking of $\mathbf{y}^b$, we indirectly validate the controller Γ^a_{HC} for the tracking of $\mathbf{y}^a$, as well. Indeed, given the structure of Γ^b_{HC}, it internally uses Γ^a_{HC} (see (4.43)). We recall that the desired trajectory $\mathbf{y}^{bd}(t)$ is transformed into a new desired trajectory $\mathbf{y}^{ad}(t) = \mathbf{g}_a(\mathbf{y}^{bd}(t), \mathbf{X}^2_C)$ that is tracked by Γ^a_{HC}. Checking the tracking errors of both $\mathbf{y}^b$ and $\mathbf{y}^a$, we can evaluate both controllers.

In the *first experiment* we fixed the desired ϑ^d_A at 15[°]. In this way we assure a sufficiently high tension in order to avoid nominal negative tension values during the experiment. The desired sinusoidal trajectory $\varphi^d(t)$ starts with a frequency equal to $\omega_\varphi = \frac{2\pi}{4}$ [rad/s] and it increases linearly until the value of about $\omega_\varphi = \frac{4\pi}{5}$ [rad/s] after which the system becomes unstable. From Fig. 5.4 one can see that the tracking of φ and ϑ_A, thus of φ and f_L, degrades with the increasing of the frequency of the sinusoidal trajectory. We remark that the internal force on the link shown in the plots is an estimation, computed using the model and the knowledge of the state and the input.

The *second experiment* is the dual, indeed we propose a sinusoidal desired trajectory with varying frequency on ϑ_A while keeping a desired constant $\varphi^d = 45$[°]. For what concerns the frequency of the sinusoidal desired trajectory $\vartheta^d_A(t)$, it starts from a value of $\omega_{\vartheta_A} = \frac{2\pi}{6}$ [rad/s] and increase up to a value of about $\omega_{\vartheta_A} = \frac{8\pi}{9}$ [rad/s]. After that, as it is possible to see from the plots in Fig. 5.5, the tracking error becomes very high. However, the system remains always stable.

Finally, for the *third experiment*, we gave as reference a sinusoidal trajectory on both φ and ϑ_A. The two signals have different frequency and phases, in particular $\omega_\varphi = \frac{2\pi}{4}$ [rad/s] and $\omega_{\vartheta_A} = \frac{2\pi}{6}$ [rad/s], respectively. The results can be seen in Fig. 5.6. As one can see, the trajectories are both tracked with a sufficiently small error. This analysis finally shows that the proposed controller is able to independently track sufficiently slow time varying desired trajectories of $\mathbf{y}^b$ with small tracking errors. On the other hand, as expected, the controller shows increasing tracking errors when asked to follow more dynamic trajectories, revealing its limitations. Anyway, a time varying reference governor (see [2] and references therein) could be applied

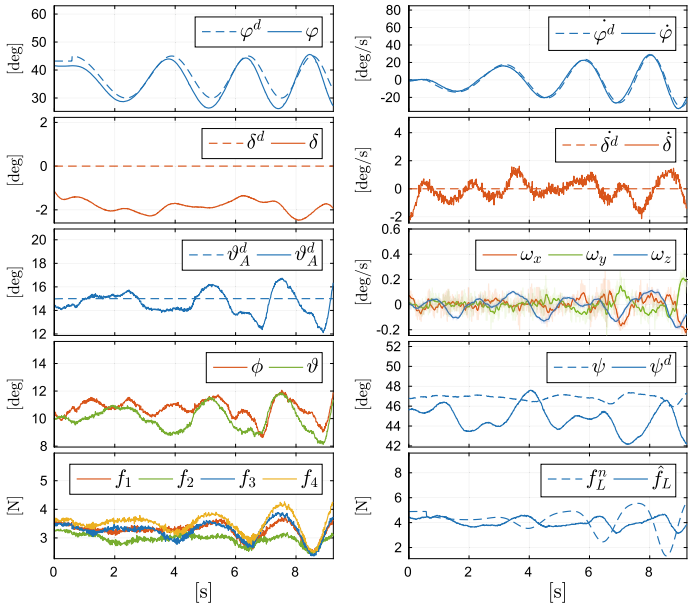


Fig. 5.4 Experimental results: tracking of a sinusoidal input on elevation with varying period with fixed attitude. ${f_L}^n$ is the nominal internal force computed by the flatness from ϑ_A^d. $(f_1, \ldots, f_4)$ are the forces produced by each thruster

to improve tracking performance. We did not report the results of the tracking of δ because they are analogous to the ones related to φ. We also encourage the reader to watch the first part of [3] where some static inclined hovering for a tethered aerial vehicle are shown.

5.3 DFL-Controller for $\mathbf{y}^a$ with Observer

In this section we consider the generic system described in Sect. 4.3 with a cable-like link (only positive internal forces are allowed.). Like in a real patrolling task, the platform follows a certain trajectory in the 3D space mimicking, e.g., a ground robot following a road. We require the aerial vehicle at time t_0 to takeoff from the moving platform, at time t_{circ} to circle above the platform at a certain altitude, and at time t_{land} to land on the moving platform. The yaw angle of the aerial vehicle has to follow the one of the platform. Notice that takeoff and landing are performed while the platform is moving, making these standard maneuvers non trivial.

We firstly test the controller Γ^a_{DFL} for the tracking of $\mathbf{y}^a$, together with the observer designed for the generic system (see Table 4.6). We set $\mathbf{k}_i$ and $\mathbf{k}_j$ such that the error dynamics $\boldsymbol{\xi}_i$ and $\boldsymbol{\xi}_j$ have poles in $(-1, -2, -3, -4)$ and $(-1, -2)$ respectively.

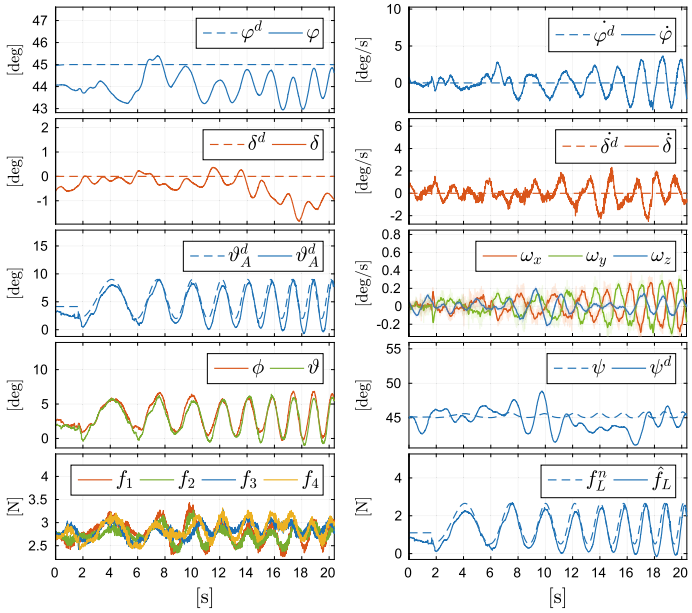


Fig. 5.5 Tracking of a desired sinusoidal trajectory of ϑ_A with varying frequency and fixed φ

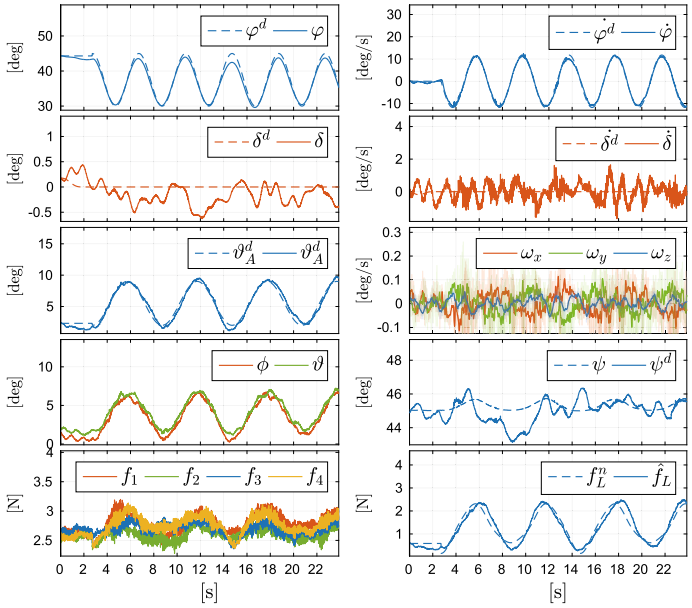
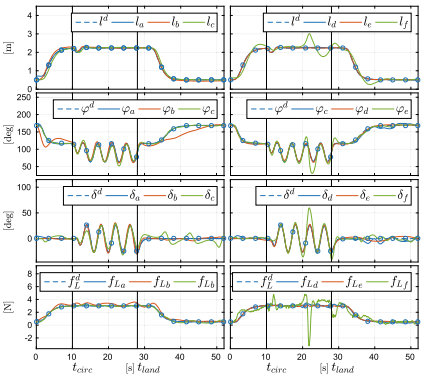
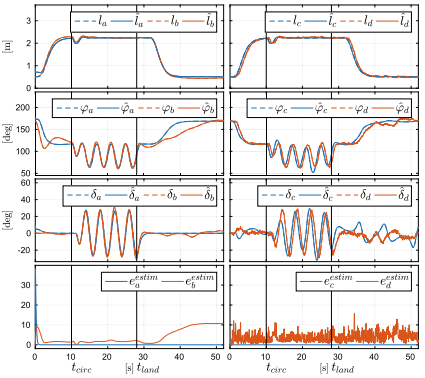


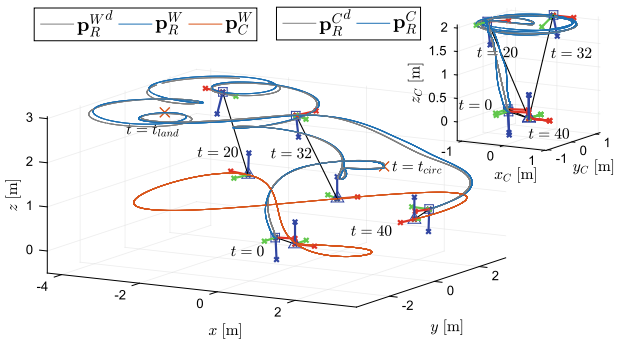
Fig. 5.6 Experimental results: tracking of a desired sinusoidal trajectory on both φ and ϑ_A with fixed period



(a) Controller performance: the tracking of the output of interest for each case is plotted. The subscript of the variables indicates the corresponding case. The blue dots highlight the desired trajectory.



(b) Observer performance. The *cases* e) is not plotted for space limitation. However it can be found in Sec. 5.3.5.



(c) Trajectories visualization for *Case d)*. Left: trajectories in the world frame. Right: aerial vehicle trajectory in the moving platform frame. Red line: platform trajectory. Blue line: aerial robot trajectory. Gray line: desired aerial robot trajectory.

Fig. 5.7 Simulation results: plausible task trajectory. The performance for each non-ideal case are compared. © 2020 IEEE. Reprinted, with permission, from [4]

For the observer we choose $\epsilon = 0.1$ and (α_1, α_2) such that $s^2 + \alpha_1 s + \alpha_2$ has roots $(-3, -4)$. Those values guarantee the stability and ensure a sufficiently fast exponential tracking. During the takeoff the desired tension must go from a small initial tension of 0.5[N] to a steady-state value of 3[N], that is kept for the whole of the circling phase, and then has to go back to the initial value during the landing.

To fully validate our method for real applications we test the convergence and the robustness for different *non ideal* cases commented in the following. Figure 5.7 gathers the main results.

(a) With an *initial position and estimation errors*, after the convergence of the observer (less than one second) the outputs follow the desired trajectories with high fidelity. An animation of this simulation is available at [5].
(b) With a *parametric variation of* 5% we notice a small constant error in the estimation of the state, but we obtain a monotonically decreasing tracking error thanks to the addition of an integral term in the outer loop (4.56), e.g., $v_1 = y_1^{d(4)} + \mathbf{k}_1^\top \boldsymbol{\xi}_1 + k_{I1} \int_0^\top \xi_1(\tau) d\tau$.
(c) For a moving platform a standard sensory set (e.g., optical flow, IMU and magnetometer) usually is sufficient to measure its trajectory variables up to $\ddot{\mathbf{p}}_C^C$ and ω_C. In this case, in which we have a *partial knowledge of the platform motion*, we can consider as zero the higher derivatives. We observe that the estimation and tracking errors are very small and remain always bounded under a reasonable threshold.
(d) In the presence of *Gaussian noise in the measurements* with typical variance values, we notice that the state estimate becomes slightly noisy but the error remains bounded within small values. The non zero estimation error implies a non zero but bounded tracking error as well.
(e) Since in practice one cannot assume the link attached exactly to O_R, we tested the method for a *vertical offset* of 5[cm] with respect to $\mathcal{F}_R$. In this case the tracking error does not go to zero but remains bounded below a small threshold.
(f) We also compared the dynamic feedback linearizing controller with the hierarchical one. We noticed that to obtain good tracking performance, the hierarchical controller requires very high gains that cause instability in the presence of the same noisy measurements of case (d). Therefore we lowered the gains until we obtained a stable behavior. However these gains are not enough to obtain good tracking performance anymore. Moreover notice that the cable becomes even slack ($f_{L_f} < 0$). Further discussions about the hierarchical control will follow in the next section.

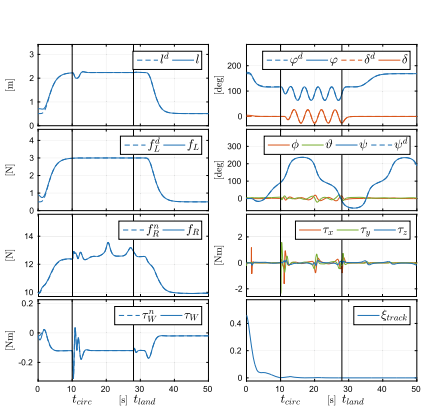
In the following section we provide additional plots and discussions for the previous non ideal cases. We also consider other additional non idealities such as non diagonal inertia matrix, saturation of the inputs and non ideal motors.

For each case we show the control performances plotting the tracking of each output of interest, the global tracking error ξ_{track} computed as the sum of each errors, and the inputs. Concerning f_R and τ_W we also show the nominal input coming from the flatness, ${f_R}^n$ and τ_W^n, that should be applied to obtain the desired output tracking in the nominal case. We also show the observer performances comparing the estimated state and the actual one. The estimation error $e^{estimation}$ is simply calculated as the sum of the estimation error for each entry of the state. Finally we display the trajectories of the aerial vehicle and of the moving platform in the world frame and with respect to $\mathcal{F}_C$. In the 3D plots the position of the moving platform and of the aerial vehicle in some particular instants are represented with a triangle and a square respectively.

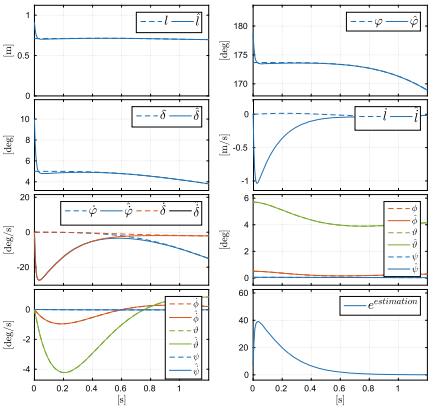
5.3.1 Initial Errors

In this section we want to show the closed loop stability of the system in dynamic condition even with some initialization error. The system starts with an error on l of 0.1[m], on φ and δ of 2[°] and on f_L of 0.5[N]. Similarly the initialization of the observer is done with an error of 0.2[m] on $\hat{l}$, of 5[°] on $\hat{\varphi}$ and $\hat{\delta}$, while their velocity are initialized to zero.

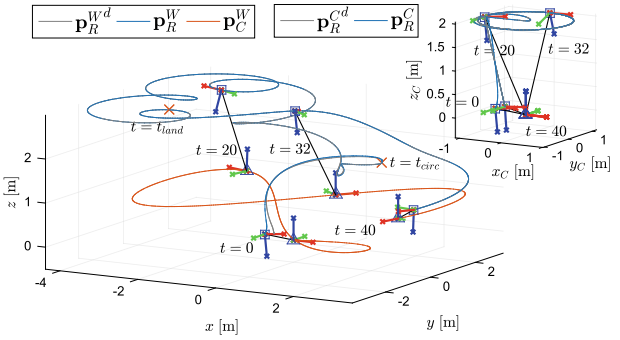
In Fig. 5.8 one can see that after the convergence of the observer, that takes less than one second, the controller exponentially steers the outputs along the desired trajectories, while the moving platform is following its own dynamic trajectory.



(a) Controller performances.

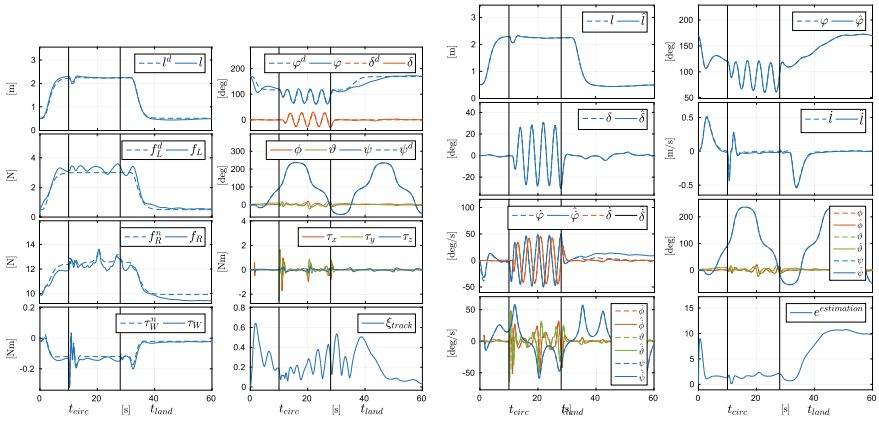


(b) Observer performances. We show here only the first second because after the estimated state follows the actual one with high fidelity for all the remaining simulation.



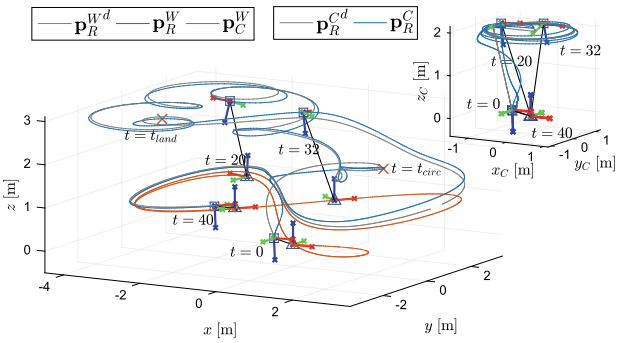
(c) Trajectories visualization.

Fig. 5.8 Simulation: initial errors



(a) Controller performances.

(b) Observer performances.



(c) Trajectories visualization.

Fig. 5.9 Simulation: parametric variations

5.3.2 Parametric Variations

The purpose of the next sections is to investigate the robustness of the proposed method. In particular in this one, we consider some parameter variation between the real model and controller/observer. Indeed in a real scenario we can not know exactly each parameter of the system, thus the controller and observer would be based on some parameter value different from the real one.

Figure 5.9 displays the results of the simulation with a parametric variation of the 5% for each entry, i.e., m_R, $\mathbf{J}_R$, J_W and r_W. In order to partially compensate the effects of the uncertainties we added in the controller an integral term with gain $k_I = 3$.

We can notice that due to the uncertainty of the model we have some nonzero errors in the tracking and in the estimation of the state. Nevertheless the error system

remains stable and thanks to the integrator terms, during the landing maneuver we obtained a decreasing tracking error that allows a correct landing of the aerial vehicle.

We performed additional extensive simulations in which we observed that the system remains stable up to a parametric variation of the 20%, after this value the system results unstable. However notice that in reality those parameters are very well measurable with small errors, certainly lower than the 20%.

5.3.3 Limited Knowledge of $\mathbf{p}_C^W(t)$

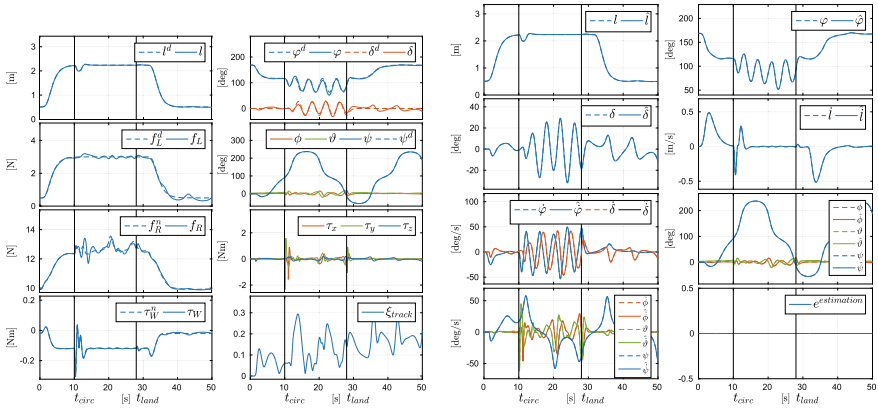
In Sect. 4.6.1 we saw that the knowledge of $\mathbf{X}_C^4$ is needed in order to compute the control action. In other words, to obtain a perfect tracking one has to know the derivative of $\mathbf{p}_C^C(t)$ up to the fourth order and of $\boldsymbol{\omega}_C(t)$ up to the third order. Although those variables have to be known to obtain zero tracking error, actually, without a posteriori knowledge of the trajectory or the model and control inputs of the system, it is difficult to measure the higher-order derivatives. Nevertheless, in this section we want to show that even with only a partial measurement of $\mathbf{x}_C^4$ the system stays stable and the tracking error remains bounded.

Indeed, for a real moving platform, a standard onboard sensorial configuration, such as optical flow, IMU and magnetometer, is sufficient to obtain $\boldsymbol{\omega}_C(t)$ and $\mathbf{p}_C^C(t)$ up to its second derivative.

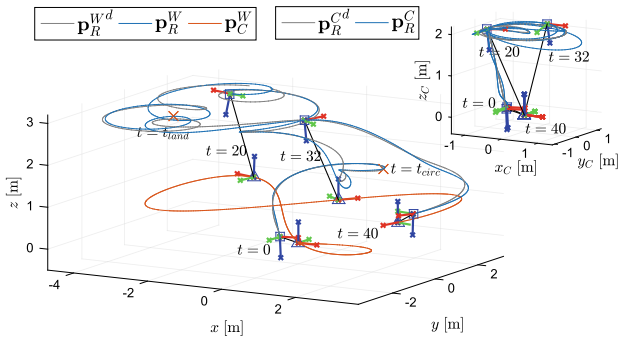
In Fig. 5.10b we can observe that the estimation error is almost constantly zero even if $\dot{\boldsymbol{\omega}}_C$ is assumed zero. While in Fig. 5.10a one can notice that the outputs oscillates around the desired value and the tracking error does not go to zero but remains bounded under a reasonable threshold. Nevertheless, with a more "aggressive" platform trajectory the negative effects would be more significants. In Fig. 5.11 the entries of $\mathbf{x}_C^i$ for $i = 1, 2, 3, 4$ are plotted. The last five entries are assumed zero by the observer and the controller.

5.3.4 Noise on the Measurements

In this section we investigate the robustness of the proposed method with the presence of noise in the measurements. Table 5.2 gathers the variance magnitude set for each measurement. For the encoder and the gyroscope we set some reasonable value found in the literature [6]. On the other hand, instead of adding noise on $\mathbf{w}_5$ and $\mathbf{w}_7$ we preferred inserting the noise directly in the measure of the rotational matrix $\mathbf{R}_R$, i.e., in $\mathbf{W}_R$. This is done because the direct measure of $\mathbf{R}_R$ using the accelerometer and the magnetometer is normally filtered with the gyroscope [7], in order to obtain a less noisy estimation of both $\mathbf{R}_R$ and $\boldsymbol{\omega}_R$. The noise added directly to $\mathbf{R}_R$ is comparable to the one we would obtain after the filtering.



(a) Controller performances. (b) Observer performances.



(c) Trajectories visualization.

Fig. 5.10 Simulation: limited measurements of the moving platform trajectory

From Fig. 5.12 we can observe that the estimated state shows some noise but the corresponding error remains limited. Due to the noisy component on the estimated state the outputs presents some oscillation as well, especially on the stress that seams to be the more sensitive output to the noise. Nevertheless the tracking error remains small and always bounded.

5.3.5 *Tethered Offset*

Exact attachment of the link to the center of mass of the aerial vehicle is practically unfeasible. Therefore there will always be a non zero offset, although small, between the tether attachment and the center of gravity. This offset makes the translational and rotational dynamics of the aerial robot coupled and can potentially lead to the instability of the controlled system. In this section we want to show the robustness of

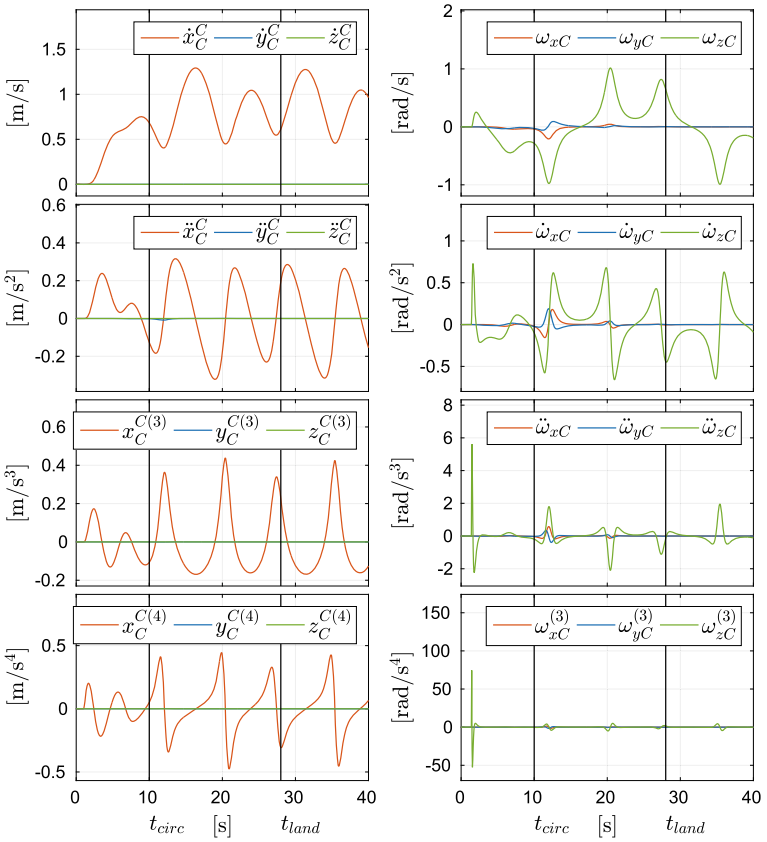
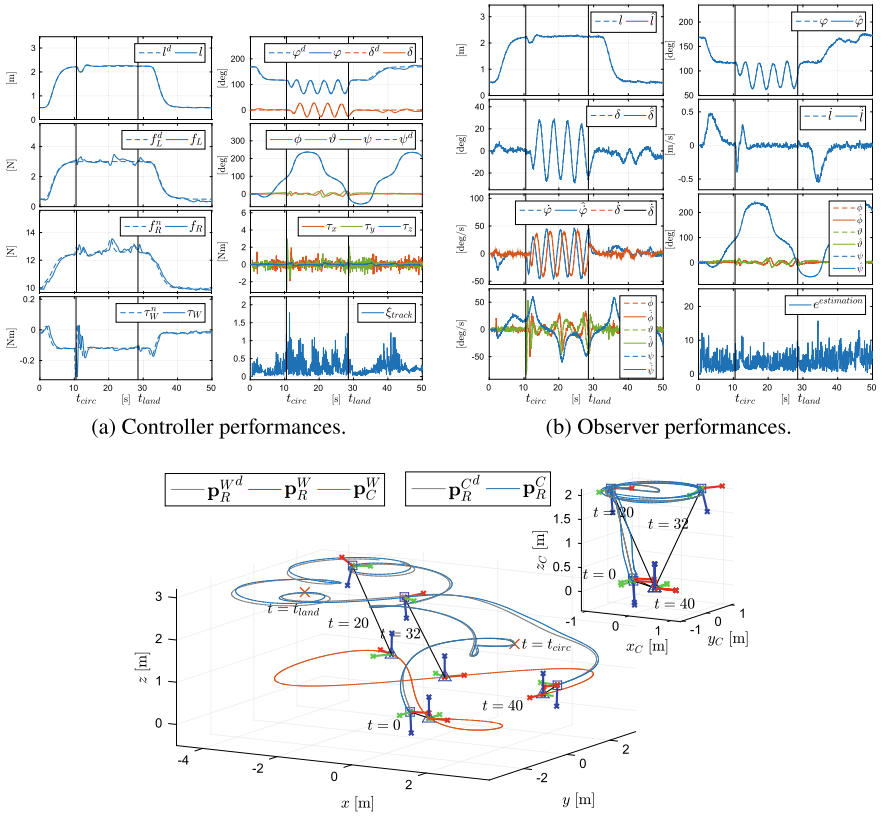


Fig. 5.11 Plot of $\mathbf{x}_C^i$, for $i = 1, 2, 3, 4$. In Simulation (c) all the variables in the last five plots are considered zero by the controller and the observer

Table 5.2 List of sensors

#	Type	Measurement	Noise variance
w_2	Abs. encoder	$\vartheta_W \approx l$	0.008[rad]
w_3	Abs. encoder	φ	0.008[rad]
w_4	Abs. encoder	δ	0.008[rad]
$\mathbf{w}_5$	Accelerometer	$\mathbf{R}_R(\ddot{\mathbf{p}}_R^W + g\mathbf{e}_3)$	–
$\mathbf{w}_6$	Gyroscope	$\boldsymbol{\omega}_R$	0.01[rad/s]
$\mathbf{w}_7$	Magnetometer	$\mathbf{R}_R\mathbf{h}^W$	–
$\mathbf{W}_R$	Complementary filter	$\mathbf{R}_R$	0.001



(a) Controller performances.

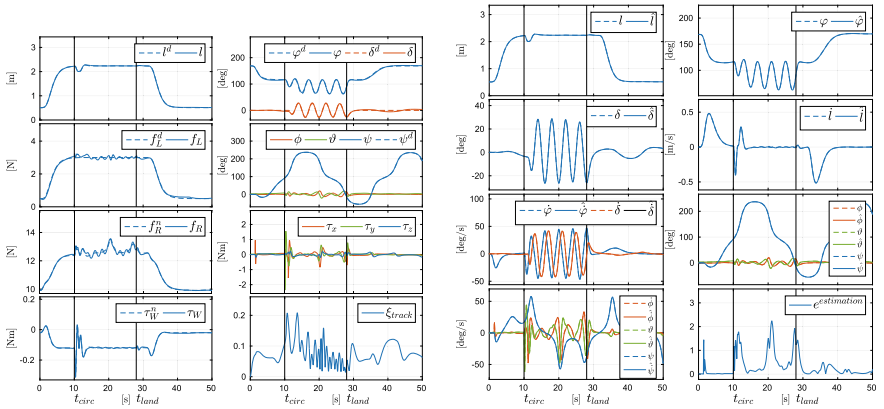
(b) Observer performances.

(c) Trajectories visualization.

Fig. 5.12 Simulation: noisy measurements

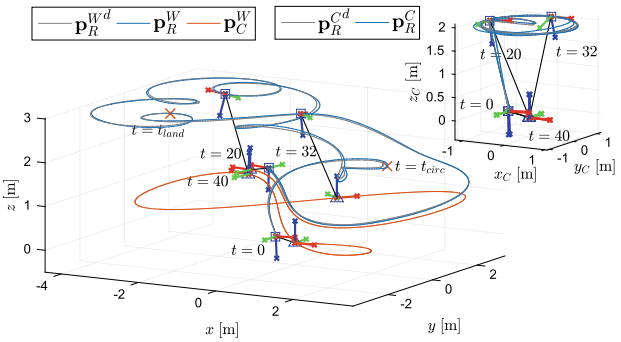
the proposed method when the distance between the attaching point and the center of gravity of the aerial vehicle is non zero (Fig. 5.13). In particular in this simulation the link is attached 5 [cm] vertically below O_R with respect to $\mathcal{F}_R$. As expected, the tracking error does not go to zero but however remains bounded, showing good tracking performances. Notice that the error is higher during the circling phase since this part of the global trajectory is very dynamical and the unmodeled effects due to the offset are larger. However we remark that a good mechanical design could make the tracking error almost negligible.

We tested the method with even larger offsets and we saw that the system remains stable up to a vertical offset of 30[cm], that is an exaggerated value for the system considered in the simulation (small-size quadrotor like vehicle). In fact, note that a larger quadrotor means a larger inertia which actually reduces the negative effects of the offset. In additional simulations, which are not reported here for the sake of brevity, we also tested the robustness of the method with a more general offset



(a) Controller performances.

(b) Observer performances.



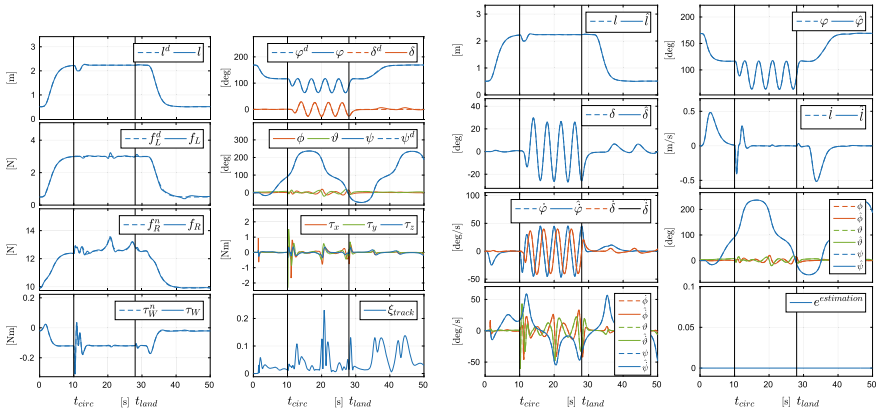
(c) Trajectories visualization.

Fig. 5.13 Simulation: non zero offset between tether attachment and center of gravity of the aerial vehicle

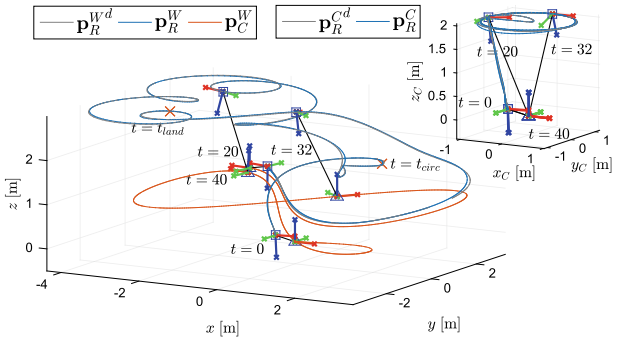
(not only vertical) noticing that, within some reasonable bounds, the system remains stable and with acceptable tracking performances.

5.3.6 Nondiagonal Inertia Matrix

In the derivation of the model and of the controller as well, we assumed a diagonal inertia matrix. In this section we check the robustness of the method if the aerial vehicle has a non diagonal inertia matrix. In particular, in Fig. 5.14, we show the results for a test in which the real inertia matrix is



(a) Controller performances. (b) Observer performances.



(c) Trajectories visualization.

Fig. 5.14 Simulation: non-diagonal inertia matrix $\mathbf{J}_R$

$$\mathbf{J}_R = \begin{bmatrix} 0.25 & 0.05 & 0.05 \\ 0.05 & 0.25 & 0.05 \\ 0.05 & 0.05 & 0.25 \end{bmatrix},$$

while the controller still assumes a diagonal inertial matrix.

One can observe that the tracking error is not exactly zero but is kept limited within a small bound. For the observer this does not constitute a non ideality, in fact the estimation error is constantly zero.

With further simulations we observed that the system remains stable up to a value of 0.15 in the off diagonal terms (60% of the values on the main diagonal). With larger values the system becomes unstable.

5.3.7 Input Saturation

For how we planned the desired trajectory, the nominal input needed to track the desired outputs is always within the limits of the considered system. Indeed, exploiting the flatness, we are able to a priori check if the inputs exceed the minimum and maximum values. Nevertheless, in this section we want to show that the system is still stable if the inputs are hardly saturated for some instants. Thus we set some very restrictive limits on the input, i.e., $f_R \leq \bar{f}_R$ and $\underline{\tau} \leq \tau_i \leq \bar{\tau}$, where $i = x, y, z$, $\bar{f}_R = 13[\text{N}]$, $\underline{\tau} = -1[\text{Nm}]$ and $\bar{\tau} = 1[\text{Nm}]$. In order to let the saturation show up during execution we did not re-plan the desired trajectory.

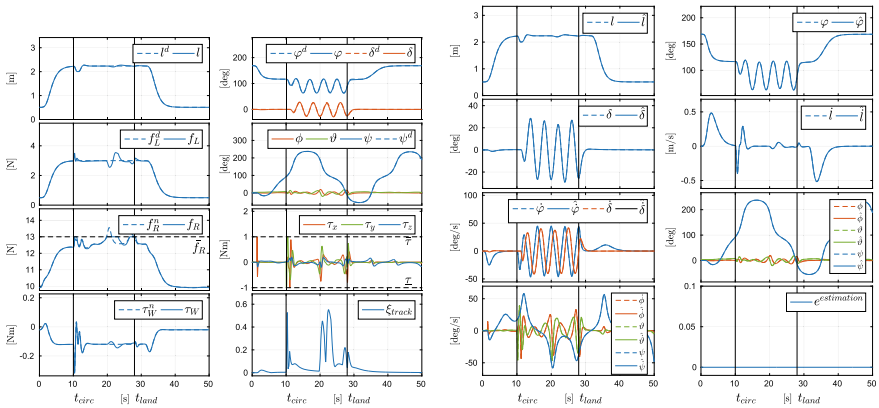
In Fig. 5.15a it can be seen that the inputs are saturated for some time instants during the execution of the task. When the inputs are saturated the tracking error increases, but, as soon as the inputs come back within the limits, the error exponentially decreases to zero.

We stress again the fact that the saturation of the inputs can be avoided exploiting the flatness. Using the flatness one can check if the desired trajectory requires inputs that are too large. In the worst case one can re-plan the trajectory such that the input limits are respected.

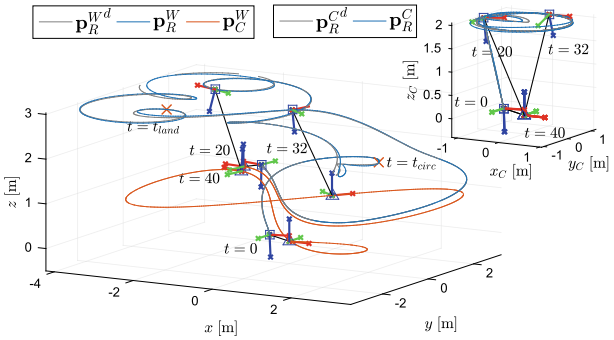
5.3.8 Motor Time Constant

With this simulation we want to further enlarge the set of non ideal models considered for the testing of the proposed control method. Considering an aerial vehicle actuated by rotating propellers, in this simulation we add the dynamical model of the motors described with a first order system characterized by a time constant of $\tau_M = 0.1[\text{s}]$. In practice the propeller dynamics inserts a frequency dependent phase shift between the commanded control input and the actuated one, whose amplitude depends on the time constant. In other words, the models acts as a low pass filter on the commanded input, cutting its high frequency components. Those effects could dramatically decrease the performances or even make the system unstable. However, from Fig. 5.16, one can notice that our method is robust to the unmodeled effects of the propellers dynamics. Indeed, in some instant, where the trajectory is more dynamical and requires fast varying inputs, the tracking error increases but it is always bounded and at steady state converges to zero.

We remark that, if needed, one can increase the smoothness of the control inputs considering an higher order in the dynamic feedback control. Indeed adding more integrators on the control channels one can increase the degree of smoothness of the control input thus guarantying that it is always below the cutting frequency proper of the system, and in particular of the propellers. Another possible strategy is to exploit the flatness to plan a trajectory that fulfills the system limitations.



(a) Controller performances. (b) Observer performances.



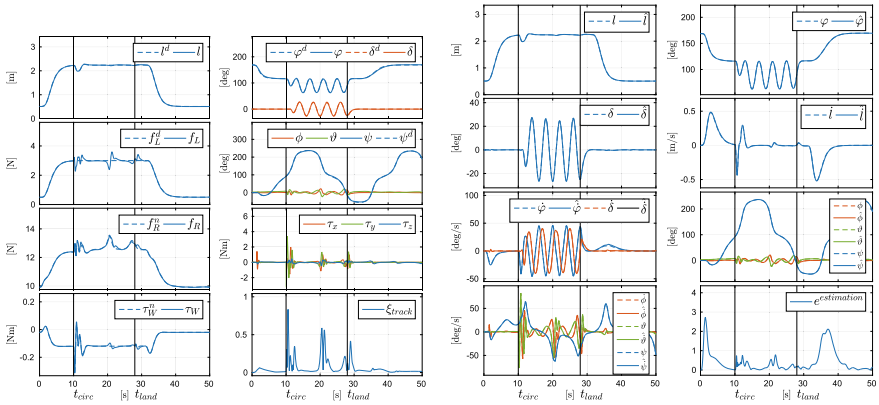
(c) Trajectories visualization.

Fig. 5.15 Simulation: saturation of the input

5.3.9 Hierarchical Control Versus DFL Control for $\mathbf{y}^a$

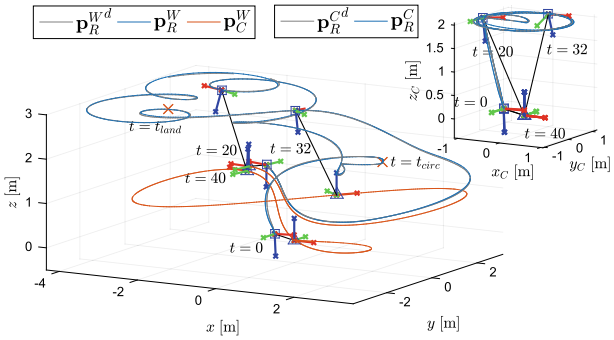
As we noted in Sect. 5.2, the hierarchical controller guarantees sufficiently good tracking performance in quasi static conditions. However the performance gets worse when the desired velocities and accelerations increase. In this section we shall compare the hierarchical controller tracking performance with respect to the one provided by the dynamic feedback linearizing controller. In particular, we simulate the system with Γ^a_{HC} together with the observer Γ^a_{HC}, tracking the same desired trajectory $\mathbf{y}^{ad}(t)$ used in Sect. 5.3, with an initial tracking error and under noisy estimated state. We shall then compare the results with the ones obtained with Γ^a_{DFL} tracking the same trajectory, and in particular under the non ideal cases (a) and (c) of Sect. 5.3 (see Fig. 5.7).

In Fig. 5.17 the results of the hierarchical controller in *ideal conditions* are reported. The initial tracking and estimation errors are the ones of case (a) in Sect. 5.3. After a tuning phase we were able to get a good performance and a small bounded



(a) Controller performances.

(b) Observer performances.



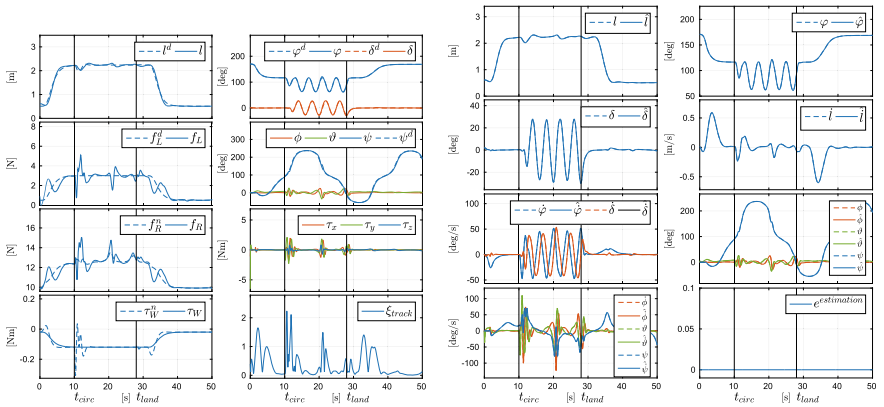
(c) Trajectories visualization.

Fig. 5.16 Simulation: system with motors dynamics

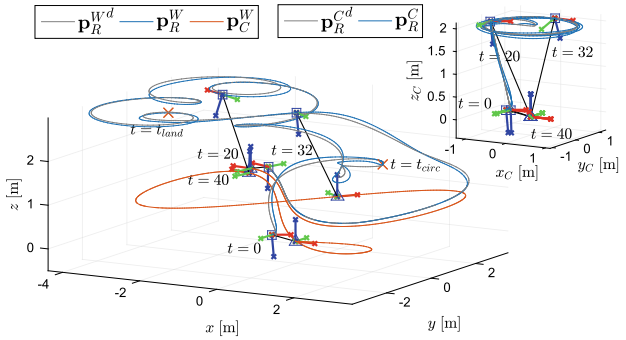
tracking error, even if the error does not converge exactly to zero. On the other hand, in the same conditions the controller based on dynamic feedback linearization is able to steer the output along the desired trajectory with zero error (see Fig. 5.7 or Sect. 5.3.1 for more details). However, to obtain good tracking performance with the hierarchical controller we had to set *very high* gains that make the system more reactive and thus able to follow the desired trajectory. Nevertheless this requirement has two main drawbacks.

The first drawback is that, due to the large control gains, the control effort increases thus possibly requiring an input that is out of the physical limits of the actuators. Indeed with this configuration we reach a maximum thrust and a maximum torque of about 15[N] and 2.5[Nm] respectively. This values are higher than the nominal inputs required to track the desired trajectory.

The second extremely serious issue arises in the presence of noise in the measurements and so in the estimated state. Indeed, the higher the gains, the larger the noise in the commands and the closer the controlled system is to instability. In fact,



(a) Controller performance. (b) Observer performance.

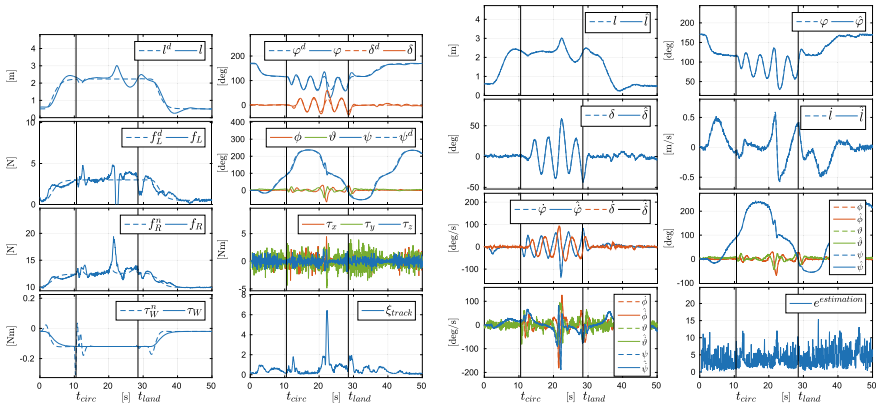


(c) Trajectories visualization.

Fig. 5.17 Simulation results: hierarchical control in ideal conditions with initial tracking error

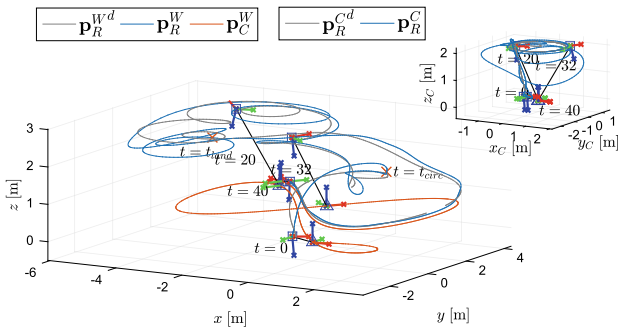
simulating the system with the same measurement noise described in Sect. 5.3 (see Sect. 5.3.4 for more details) the closed loop system becomes unstable. In order to get a stable behavior we had to significantly lower the gains, an action that, however, clearly degrades the tracking performance. As we can see in Fig. 5.18 the performance with noise is much worse than the one obtained using the dynamic feedback linearizing controller in the same noisy condition.

Therefore, the hierarchical approach presents a strictly penalizing trade-off between applicability with noise and tracking performance. One cannot obtain both. Attainment of both objectives is instead possible with the DFL controller Γ^a_{DFL}. Nevertheless, we experimentally proven that, in standard conditions, with not too dynamic desired trajectories, the hierarchical controller can still guarantees good tracking performance with a minimum implementation effort. This controller allowed us to perform the landing and takeoff maneuvers on surfaced inclined up to 60°, in a very robust and reliable way. On the other hand, although the dynamic feedback linearization control provide much better performance, even in non ideal conditions,



(a) Controller performance.

(b) Observer performance.



(c) Trajectories visualization.

Fig. 5.18 Simulation results: hierarchical control in the noisy case. To preserve stability lower gains have to be used with noise, therefore the performance is significantly degraded. The hierarchical controller presents a strictly penalizing trade off between tracking performance and robustness to noise

it comes with an higher computational cost, that makes it also more difficult to be implemented on a real robot.

5.3.10 *DFL-Controller for $\mathbf{y}^c$ in Case of Passive Link Actuator*

Recalling the discussion in Sect. 4.4.3.1, in order to obtain a steady state internal force $f_L^\star = 5\,[\mathrm{N}]$, we set the torque winch $\tau_W = -1\,[N]$. To obtain a sufficiently fast exponentially tracking, we set the controller gains $\mathbf{k}_i$ and $\mathbf{k}_4$ such that the error dynamics ξ_i and ξ_4 have poles in $(-0.5, -1, -1.5, -2)$ and $(-0.5, -1)$, respectively,

for $i = 1, 2, 3$. Since the observer has been already tested in the previous section, here the control loop is closed with a direct measure of the state.

As before, we design the platform motion and the aerial vehicle desired trajectory in order to simulate a patrol-like task of a delimited area. The platform simply follows a certain trajectory shown in Fig. 5.19. The aerial vehicle, after the takeoff maneuver, at time t_{circ} has to loiter above the platform. Then, starting from time t_{land}, the aerial vehicle has to land on the platform.

To validate the control method and to test its robustness we performed several simulations in different non ideal conditions:

(a) We initialized the system with an *initial tracking error* of 10 [°] for the elevation, of 5 [°] for the azimuth and of 0.5 [m] for the link length. Looking at Fig. 5.19a we can notice that after a transient, the controller steers the output of interest along the desired trajectory. Notice that the internal force along the link remains always positive and close to the desired steady state value $f_L^\star$. Furthermore it is exactly equal to $f_L^\star$ whenever $\ddot{l}$ is zero.

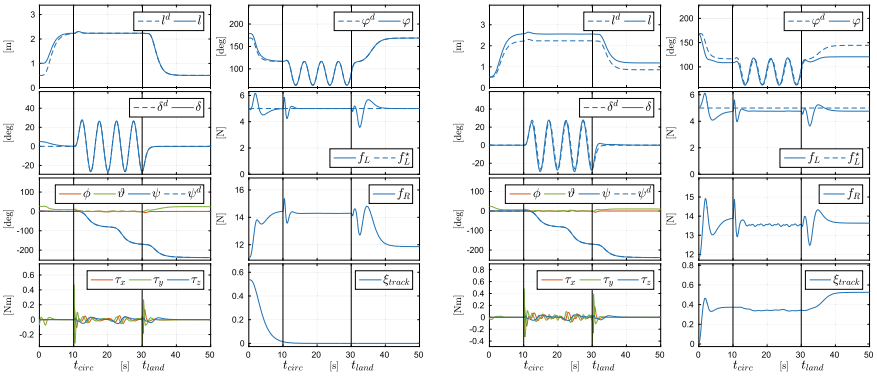
(b) We tested the robustness of the control method with a *variation of the* 5% *on all the model's parameters* (see Fig. 5.19b). Due to the mismatch between real and nominal model, the feedback linearization is not exact and the error does not go to zero. However it remains always bounded showing nicely degrading and sufficiently good tracking performance. Moreover, in order to eliminate the constant error at steady state we have seen that a simple integral term in the linear control loop is sufficient. With further simulations we noticed that the system remains stable showing acceptable tracking errors up to a parametric variation of 50%, proving the robustness of the proposed method. Above the system becomes unstable.

(c) Although the control loop is not closed with the observer, we tested the robustness of the proposed method injecting *Gaussian noise on the measured state* used to close the control loop. The power of the noise has the same value of the one noticed in Sect. 5.3 out of the observer based on noisy sensors. From Fig. 5.19c one can see that the error does not converge to zero but remains always bounded showing good and practically viable tracking performance.

(d) In this simulation we considered the thrust and the torque of the aerial vehicle generated by *non ideal motors* modeled as a first order system characterized by a time constant of 0.2 [s]. The results displayed in Fig. 5.19d show a very small tracking error, validating the robustness of the control method to this additional non ideality.

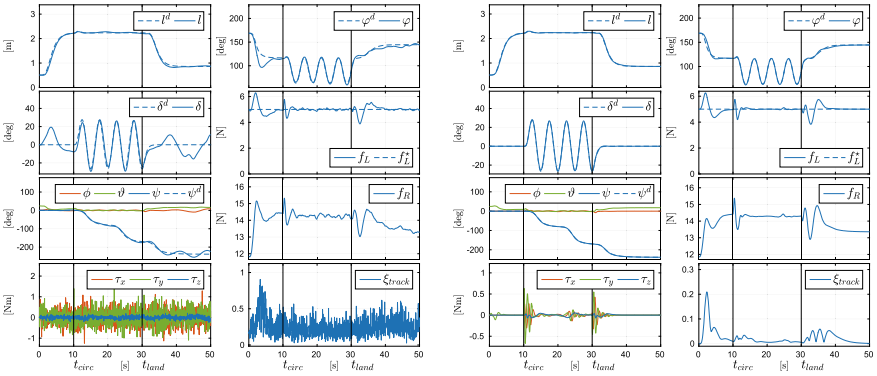
5.4 Observer Based DFL-Controllers for Reduced Model

In this section we validate the observer based on IMU only, together with the 2D version of controllers Γ^a_{DFL} and Γ^b_{DFL} (see [8] for the corresponding details). In the first subsection we show the capability of Γ^a_{DFL} of independently controlling φ and



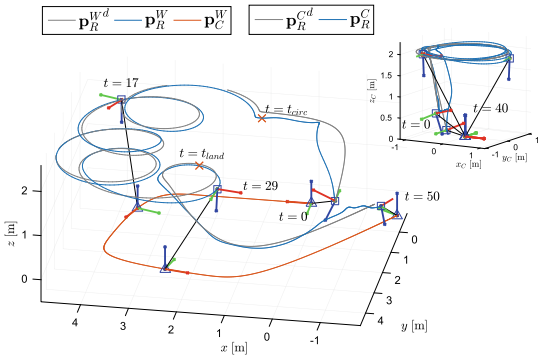
(a) Initial tracking error.

(b) Parameters uncertainties.



(c) Additional noise on the state.

(d) Non ideal motors.



(e) Trajectories visualization for the case c).

Fig. 5.19 Simulation results: plausible task trajectory for the case of a passive link actuator. The performance for each non-ideal case are shown

f_L, even when the desired internal force trajectory goes from tension to compression, and vice versa. In the second subsection we instead provide a thorough analysis of the robustness of the 2D version of Γ^a_{DFL} and Γ^b_{DFL} together with the observe against non ideal conditions.

5.4.1 Controlling f_L for Both Tension and Compression

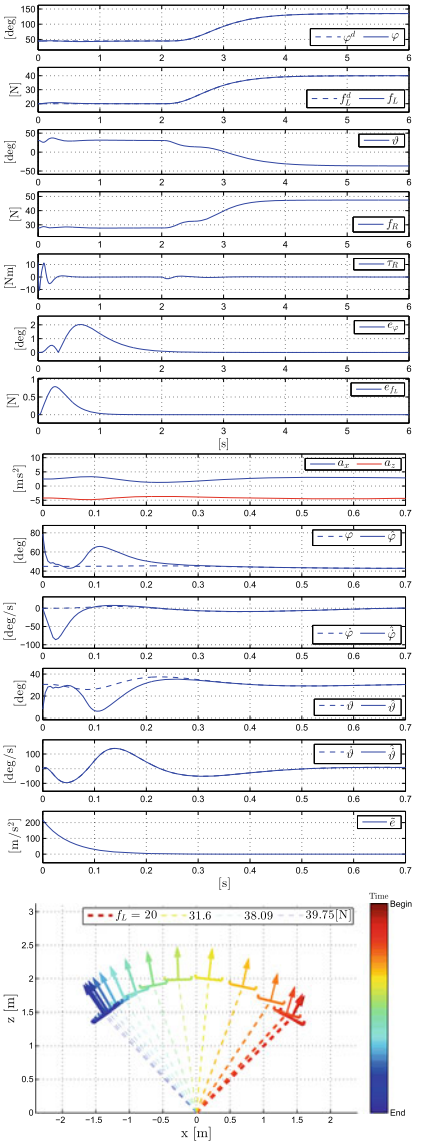
Figures 5.20a, b and 5.21a, b show the behavior of the system following smooth trajectories from an initial to a final output configuration. The plots of the tracking errors show that the proposed controller is able, after a short transient, to perfectly follow the time varying smooth trajectories of class C^3 and C^1 for the elevation and link internal force, respectively. An animation of the presented simulations is available at [9]. Notice that in Fig. 5.21b, the pick of torque at around time 2.6[s], arises due to the crossing of the control singularity, i.e., zero thrust. However, since that singularity is crossed only for one instant, the system remains stable.

We also tested controller Γ^a_{DFL} to track a desired trajectory of f_L that goes from an initial tension to a final compression, while following a desired elevation motion as well. In Fig. 5.21a it is interesting to notice that to pass from tension to compression the vehicle turns upside-down keeping the thrust always positive. On the other side, in the simulation of Fig. 5.21b the transition from tension to compression is obtained with the thrust that passes through zero and inverts its sign in order to obtain the same final compressing force of the simulation of Fig. 5.21a. This happens because in the second case the desired trajectory requires zero thrust at a certain moment. Then, since it is not possible to instantaneously turn the vehicle, the controller inverts the sign of the thrust in order to provide compression. In the case of vehicles able to provide also negative thrust this is not a problem. While, in the case of robots providing only positive thrust a planning phase is needed in order to generate feasible trajectories.

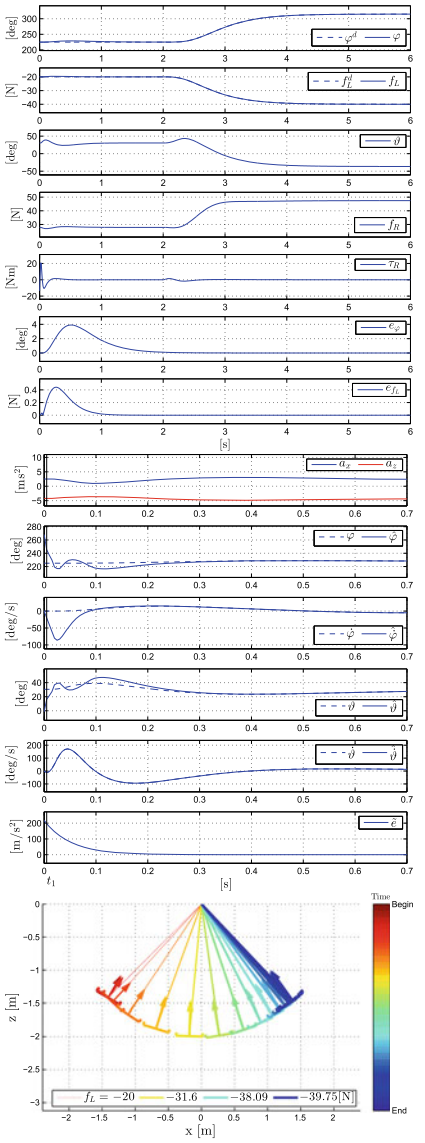
5.4.2 Robustness Investigation Against Non Ideal Conditions

In this section we present a comprehensive analysis of the robustness of the designed dynamic feedback linearizing controllers together with the observer based on IMU only, against non-ideal conditions. This shows both their strengths and possible limits when applied on a real system. In order to test the observer based only on the IMU measurements, the following analysis is carried out for the reduced model presented in Sect. 4.3, consisting of an aerial vehicle constrained on a 2D vertical plane tethered to a fixed point on the ground by a link with a constant length $l = 2$[m].

For this system we tested the reduced version of Γ^a_{DFL} and Γ^b_{DFL} together with the observer based only on the IMU measurement. We recall that the details of such reduced version of the presented DFL controllers can be found in [8].

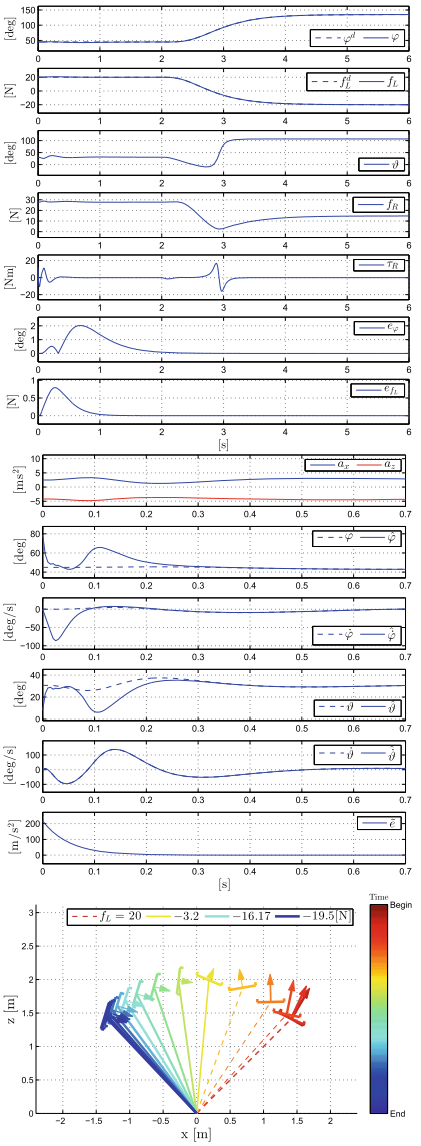


(a) The desired φ trajectories goes from the initial value $\varphi_0^d = \pi/4$ [rad] to the final $\varphi_f^d = 3\pi/4$ [rad]. The desired f_L trajectory goes from the initial tension of ${f_L}^d{}_0 = 20$ [N] to the final tension of ${f_L}^d{}_f = 40$ [N].

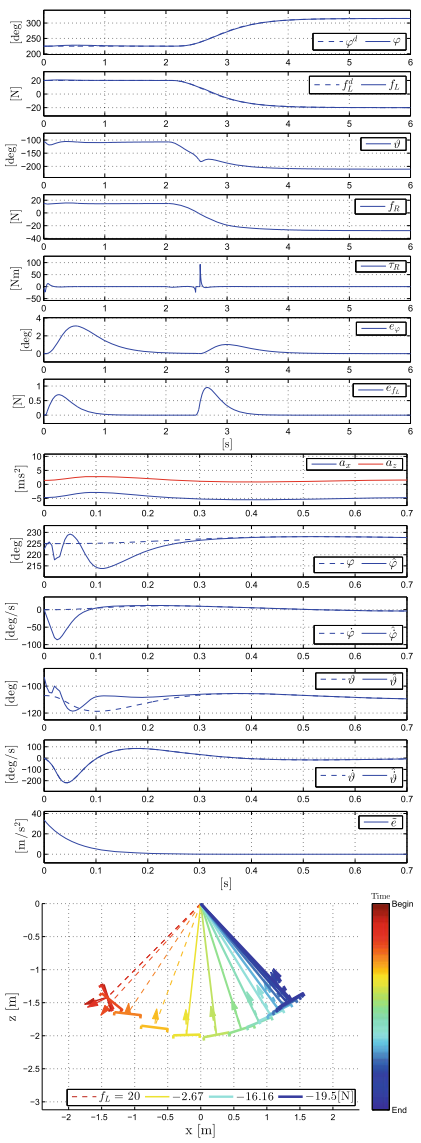


(b) The desired φ trajectories goes from the initial value $\varphi_0^d = 5\pi/4$ [rad] to the final $\varphi_f^d = 7\pi/4$ [rad]. The desired f_L trajectory goes from the initial compression of ${f_L}^d{}_0 = -20$ [N] to the final compression of ${f_L}^d{}_f = -40$ [N].

Fig. 5.20 Simulation results: controlling φ and the tension (on the left) or the compression (on the right)

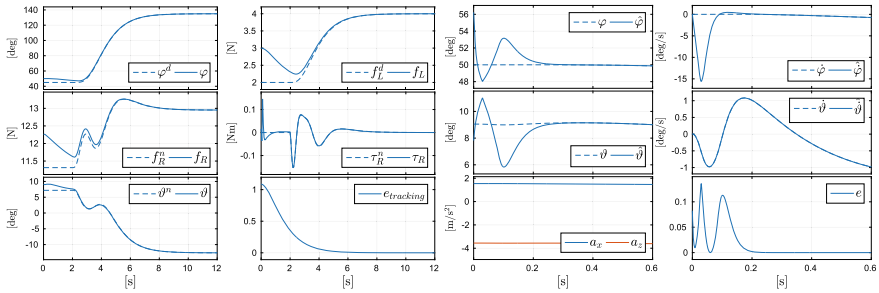


(a) The desired φ trajectories goes from the initial value $\varphi_0^d = \pi/4$ [rad] to the final $\varphi_f^d = 3\pi/4$ [rad]. The desired f_L trajectory goes from the initial compression of ${f_L}^d{}_0 = 20$ [N] to the final compression of ${f_L}^d{}_f = -20$ [N].

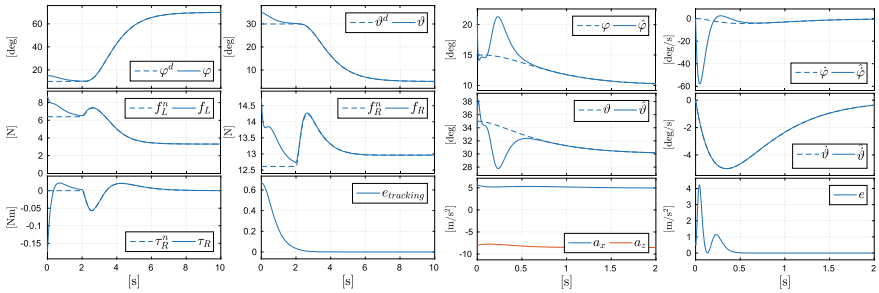


(b) The desired φ trajectories goes from the initial value $\varphi_0^d = 5\pi/4$ [rad] to the final $\varphi_f^d = 7\pi/4$ [rad]. The desired f_L trajectory goes from the initial tension of ${f_L}^d{}_0 = 20$ [N] to the final tension of ${f_L}^d{}_f = -20$ [N].

Fig. 5.21 Simulation results: controlling φ and f_L going from tension to compression. In figure (**b**)



(a) Results of Γ^a_{DFL} and of the observer, on the left and on the right, respectively.



(b) Results of Γ^b_{DFL} and of the observer, on the left and on the right, respectively.

Fig. 5.22 Simulation results: nonzero initial tracking error

Concerning the controller Γ^a_{DFL} we set the gains of the linear outer control loop, $\mathbf{k}^a_1$ and $\mathbf{k}^a_2$, such that the error dynamics of φ and f_L has poles in $(-1, -1.5, -2, -2.5)$ and $(-1, -1.5)$ respectively. While for the controller Γ^b_{DFL} we set the gains $\mathbf{k}^b_1$ and $\mathbf{k}^b_2$, such that the error dynamics of φ and ϑ_A has poles in $(-0.5, -1, -1.5)$ and $(-0.5, -1)$ respectively. For the gains of the observer we set $\epsilon = 0.1$ and $(\alpha_1, \alpha_2, \alpha_3)$ such that the root of $s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3$ are $(-6, -4.5, -3)$. Those values guarantee the stability of the closed loop system and a sufficiently rapid convergence of the observer and controller.

For the controller Γ^a_{DFL}, the desired trajectory is a smooth step, continuous up to the fourth order for φ and up to the second order for f_L, from the initial values $\varphi^d_0 = 45°$, ${f_L}^d_0 = 3$[N], to the final values $\varphi^d_f = 135°$, ${f_L}^d_f = 5$[N], respectively. Smooth step-like trajectories (see Fig. 5.22), as it will be clear later, have the benefit of clearly showing the performance of the controllers under three important conditions: the initial transient, the tracking of a fast time-varying signal, and the steady state. For the controller Γ^b_{DFL}, the desired trajectory is a smooth step, continuous up to the third order for φ and up to the second order for θ, from the initial values $\varphi^d_0 = 10°$, ${\vartheta_A}^d_0 = 30°$ to the final values ${\vartheta_A}^d_f = 50°$, ${\vartheta_A}^d_f = 5°$, respectively. Notice that, since the system is constrained to the 2D vertical plane with $\mathbf{y}_R = \mathbf{y}_W$, we have that $\vartheta_A = \vartheta$, where we recall that ϑ is the pitch of the vehicle.

To obtain a complete validation, in the following we show a concise summary of the results about the stability and robustness of the proposed method under different non-ideal conditions, such as: (a) nonzero initial tracking and estimation errors, (b) parametric variations, (c) generic CoM position and non-negligible link mass, (d) noisy sensor measurements, and (e) non-ideal motors.

5.4.2.1 Validation for Nonzero Initial Tracking/Estimation Errors

In order to show the asymptotic convergence performance of both the controller and the observer we initialize the control system with nonzero initial tracking and estimation errors. One can see in Fig. 5.22 that, after the convergence of the observer, which takes less than one second, the controller Γ^a_{DFL} is able to steer the outputs along the desired trajectories with zero error. A similar behavior is obtained for the controller Γ^b_{DFL}. We then performed many other similar simulations with different initial errors and we observed always the same asymptotically convergent behavior, as expected from the almost-global nature of the proposed observer and control laws.

5.4.2.2 Parametric Variations

We notice that in principle one could try to design an adaptive control law that is able to compensate for parametric uncertainties. However, this is clearly a tough objective, because the system is nonlinear and the available measurements are only the (nonlinear) accelerometer and the gyroscope readings. Therefore this goal is left as future work. Instead, we concentrate in this section on assessing the ranges of parameter variations that causes a degradation of the performance that remains within an acceptable bound. By doing so, we shall see in fact that the proposed control scheme possesses a remarkable robustness even without the presence of an adaptive design.

Considering l_0, m_{R0} and J_{R0} the real parameters value and l, m_R and J_R the nominal ones, we set $l = (1 + \Delta l)l_0$, $m_R = (1 + \Delta m_R)m_{R0}$ and $J_R = (1 + \Delta J_R)J_{R0}$, where Δm_R, Δl and ΔJ_R denote the corresponding parametric variations.

For obtaining a comprehensive analysis we tested the behavior for several different parametric variation combinations. The results are plotted in Figs. 5.23 and 5.24, where we show the mean tracking error, $\bar{e}_{track}$, and the corresponding standard deviation $\sigma_{\bar{e}_{track}}$, defined as

$$e_{track}(t) = \frac{\left\| y_1^d(t) - y_1(t) \right\|}{y_1^d(t)} + \frac{\left\| y_2^d(t) - y_2(t) \right\|}{y_2^d(t)}$$

$$\bar{e}_{track} = \frac{1}{t_f - t_0} \int_{t_0}^{t_f} e_{track}(t) dt$$

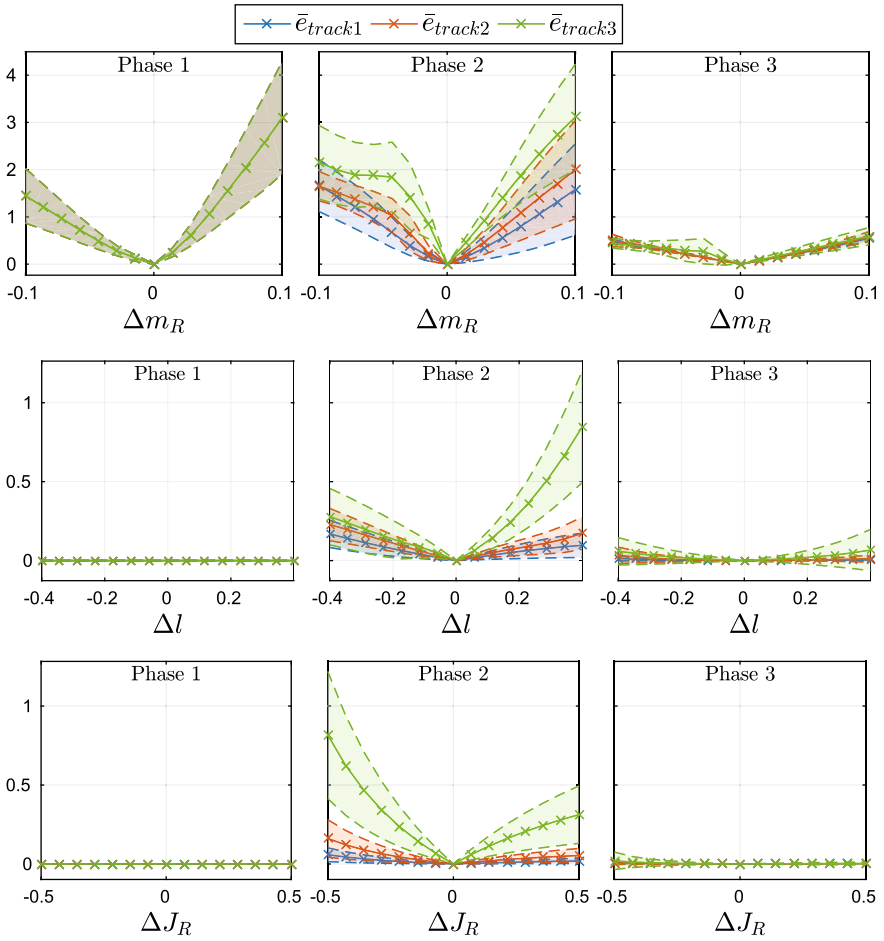


Fig. 5.23 Simulation results: parametric variation—Controller Γ^a_{DFL} (elevation and link force). The subscript 1, 2, and 3 correspond to the three different trajectory times. Outside of the displayed range of parametric variation the performance is unacceptable or the closed loop system results to be even unstable. © 2020 IEEE. Reprinted, with permission, from [8]

$$\sigma_{\bar{e}_{track}} = \sqrt{\frac{1}{t_f - t_0} \int_{t_0}^{t_f} (e_{track}(t) - \bar{e}_{track})^2 dt},$$

where t_0 and t_f are the initial and final time, respectively. Notice that for the reduced model $y_1 = \varphi$ and $y_2 = f_L$ or $y_2 = \vartheta_A$ for Γ^a_{DFL} and Γ^b_{DFL} respectively. In the plots the solid line corresponds at the mean tracking error while the dashed lines correspond at the mean tracking error plus and minus its standard deviation.

The effect of an unknown parameter could also change with respect to the trajectory and in particular with respect to the velocity and acceleration at which the

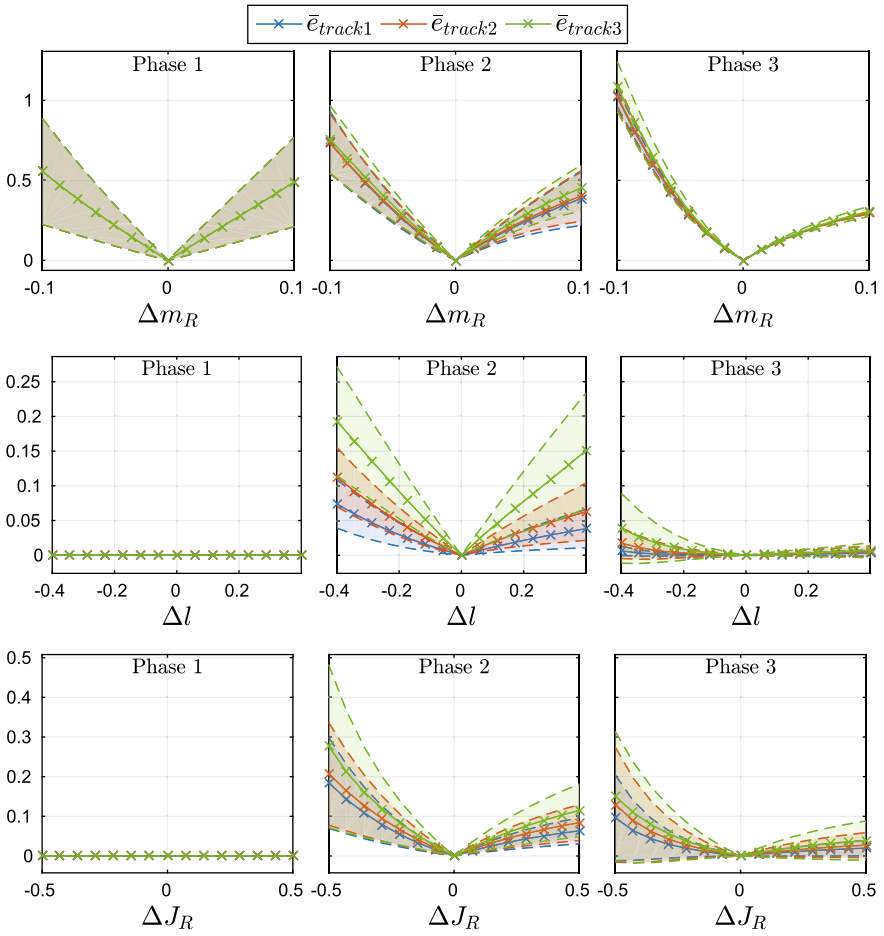


Fig. 5.24 Simulation results: parametric variation—Controller Γ^{b}_{DFL}(elevation and attitude). © 2020 IEEE. Reprinted, with permission, from [8]

path is followed. Consequently we plotted the mean tracking error, $\bar{e}_{track1}$, $\bar{e}_{track2}$ and $\bar{e}_{track3}$, for the same type of desired path (smooth step) but executed with three different durations (increasing velocity): (1) 7[s], (2) 5[s] and (3) 3[s] respectively. We also analyze the error behavior dividing the trajectory into three phases: in the *Phase 1* (transient) the desired trajectory is constant and the analysis is more focused on the convergence of the observer; *Phase 2* constitutes the dynamic part where the desired trajectory quickly goes from the initial value to the final one; the *Phase 3*, the last, corresponds to the steady state condition where the desired trajectory is again constant. We show the tracking error for each of the three phases to better understand if a parameter variation affects more the transient, the dynamic phase, or the static one.

From Figs. 5.23 and 5.24 one can notice that, as expected, the performance gets worse increasing the parametric variation. Furthermore, the same variation has more effect if the trajectory is more "aggressive" and it is followed with higher speed ($\bar{e}_{track3}$). This is due to the fact that with higher speed and acceleration the inertial and Coriolis/centripetal terms become larger, and thus also the error in the feedback linearization increases, which in turn implies a worst tracking.

Comparing the performance between the two controllers, we notice that controller Γ^b_{DFL} results to be more robust than controller Γ^a_{DFL} in term of mean tracking error. This is due to the fact that for the controller Γ^b_{DFL}, the dynamics of one of the controlled outputs, namely ϑ_A, is not influenced by the parameters such as mass and length of the link. This means that any variation on these parameters does not generates a worse tracking of ϑ^d_A, which results in a lower tracking error.

One can also notice that the mean tracking error is not in general symmetric with respect to the sign of the corresponding parametric variation. For example for the controller Γ^b_{DFL} it is better to overestimate the mass, and the length of the link rather than underestimating them, while for controller Γ^a_{DFL} it results to be the opposite, even if these consideration are more relevant for the dynamic phase. Indeed, during the steady state phase the behavior is almost symmetrical.

Another fact that appears clear from the plots is that the variation that most influences the performance is the one on the mass of the aerial vehicle. Fortunately, in practice this parameter can be easily measured with high precision.

5.4.2.3 Generic CoM Position and Non-negligible Link Mass

The controllers developed in this paper assume that the system can be represented with the model given in Sect. 4.3, where the CoM of the aerial vehicle coincides with the attachment point of the link to the vehicle and the link has a negligible mass. Figure 5.25 represents instead a more general model, for which the assumptions done in Sect. 4.3 are not fulfilled. Taking into account the definitions made in Sect. 4.3 we then define a *body frame*, $\mathcal{F}_l$, attached to the link, with axes $\{\mathbf{x}_l, \mathbf{y}_l, \mathbf{z}_l\}$ and origin O_l coinciding with the center of mass (CoM) of the link. The position of O_l, defined in $\mathcal{F}_W$, is denoted with $\mathbf{p}_l = [x_l \;\; y_l \;\; z_l]^\top$. As for $\mathcal{F}_R$ we have that $\mathbf{y}_l \equiv \mathbf{y}_B \equiv \mathbf{y}_W$ and $y_l \equiv$

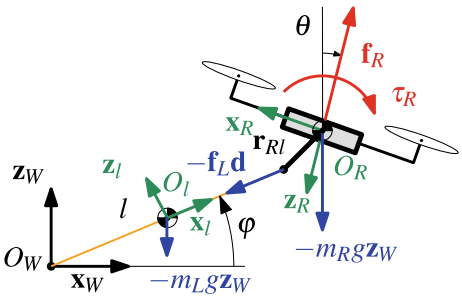


Fig. 5.25 Representation of the more general system and its variables, still constrained in the 2D vertical plane. © 2020 IEEE. Reprinted, with permission, from [8]

0. For the validation we model the link as a rigid body of mass $m_L \in \mathbb{R}_{>0}$ and inertia $J_L \in \mathbb{R}_{>0}$. Considering the inertia of the link as the inertia of an infinitesimally thin rigid tie with uniform distributed mass, we have also that $J_L = m_L l^2/12$. Assuming links with high stiffness, the deformations and the elongations results negligible with respect to the length of the cable itself, in the range of forces of our concern. Therefore the link length is fixed. The link is connected at one end to a fixed point coinciding with O_W and at the other end to a point rigidly attached to the aerial vehicle whose constant position in $\mathcal{F}_R$ is denoted with $\mathbf{r}_{Rl} = [r_x \;\; 0 \;\; r_z]^\top$. If $\|\mathbf{r}_{Rl}\| = 0$ then the link is directly attached to the CoM of the aerial vehicle.

The mechanical model of the more general robotic system can be then derived writing the dynamics as the one in (4.18) plus a disturbance due to the non idealities:

$$\mathbf{M}'(\mathbf{q}')\ddot{\mathbf{q}}' + \mathbf{g}'(\mathbf{q}') + \boldsymbol{\delta}(\mathbf{q}', \dot{\mathbf{q}}', \ddot{\mathbf{q}}', \mathbf{u}') = \mathbf{Q}'(\mathbf{q}')\mathbf{u}',$$

where

$$\begin{aligned}
\boldsymbol{\delta}(\mathbf{q}', \dot{\mathbf{q}}', \ddot{\mathbf{q}}', \mathbf{u}') &= \bar{\mathbf{M}}(\mathbf{q}')\ddot{\mathbf{q}}' + \bar{\mathbf{c}}(\mathbf{q}', \dot{\mathbf{q}}') + \bar{\mathbf{g}}(\mathbf{q}') - \bar{\mathbf{Q}}(\mathbf{q}')\mathbf{u}', \\
\bar{\mathbf{M}} &= \begin{bmatrix} \bar{J}_\varphi & J_{\varphi\theta} \\ J_{\theta\varphi} & \bar{J}_\theta \end{bmatrix}, \quad \bar{\mathbf{c}} = \begin{bmatrix} \bar{c}\dot{\theta}^2 \\ \bar{c}\dot{\varphi}^2 \end{bmatrix}, \\
\bar{\mathbf{g}} &= \begin{bmatrix} \frac{m_L}{2} l g \mathbf{d}^\perp \cdot \mathbf{e}_3 \\ -m_R l g \bar{\mathbf{R}}_B^W \mathbf{r}_{Rl} \cdot \mathbf{e}_3 \end{bmatrix}, \quad \bar{\mathbf{Q}} = \begin{bmatrix} 0 & 0 \\ -\mathbf{r}_{Rl} \cdot \mathbf{e}_3 & 0 \end{bmatrix},
\end{aligned}$$

where $\bar{\mathbf{R}}_B^W = \partial \mathbf{R}_B^W / \partial\theta$, $\bar{J}_\varphi = m_L l^2/3$, $\bar{J}_\theta = m_R \|\mathbf{r}_{Rl}\|^2$, $J_{\varphi\theta} = J_{\theta\varphi} = -m_R l \bar{\mathbf{R}}_B^W \mathbf{r}_{Rl} \cdot \mathbf{d}^\perp$, $\bar{c} = m_R l \bar{\mathbf{R}}_B^W \mathbf{r}_{Rl} \cdot \mathbf{d}$.

For a plausible case in which the link consists of a cable of mass $m_L = 0.01 m_R$ and inertia (during taut condition) $J_L = m_L l^2/12$, and it is attached to the robot in the position $r_{BL} = [0.03 \;\; 0 \;\; 0.03]^\top$[m] with respect to $\mathcal{F}_R$, we noticed that the controlled system is stable but the error does not converge exactly to zero. Indeed, due to the nonzero $\|\mathbf{r}_{Rl}\|$, the force along the link generates an extra torque on the aerial vehicle that is not compensated and so a constant steady state error appears.

In order to understand how each parameter of the more general model affects the tracking performance, as before, we show in Fig. 5.26 the mean tracking error and its standard deviation for different parameter values and in the three phases described before. In particular the mass of the link is taken as $m_L = \Delta m_L m_R$.

We noticed that the negative effects due to a nonzero offset $\mathbf{r}_{BL}$ reduce or increase if the rotational inertia is increased or reduced, respectively. Indeed, looking at the rotational dynamics in the case of non zero offset:

$$\ddot{\vartheta} = \tau_R / J_R - f_L \mathbf{d} \cdot \mathbf{r}_{BL} / J_R, \tag{5.1}$$

one can notice that the effect of the link force on the angular acceleration decreases if the inertia increases. Intuitively, a bigger inertia would mean a bigger mass or a bigger dimension of the vehicle. In the second case, the bigger the vehicle the more

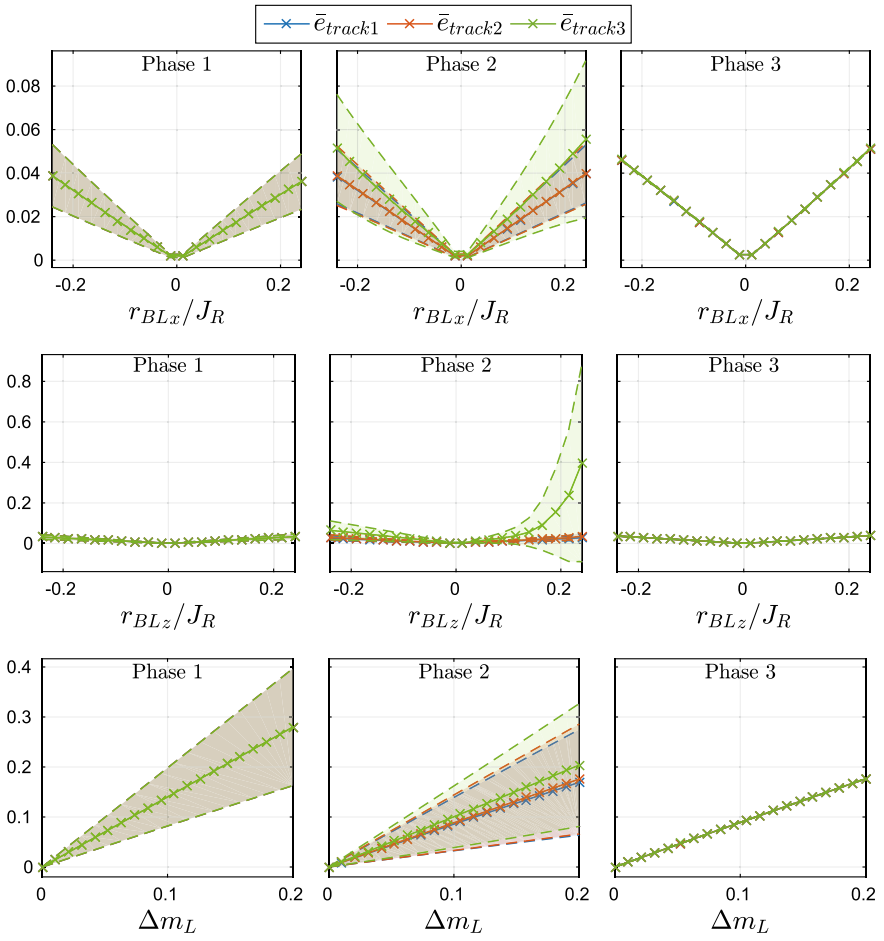


Fig. 5.26 Simulation results: controller Γ^a_{DFL}. Mean tracking error when changing the parameters of the general model. © 2020 IEEE. Reprinted, with permission, from [8]

the effect of the offset becomes negligible. For this reason in Fig. 5.26 we plot the mean tracking error with respect to $r_{BL_\star}/J_R$, thus normalizing this effect.

We did the same test for the controller Γ^b_{DFL}, which resulted to be much more sensitive to link mass and to the offset than the controller Γ^a_{DFL}. This is due to the fact that one of the output, namely the attitude of the aerial vehicle ϑ_A, is directly influenced by the offset as it is shown in (5.1) (we recall the in the 2D case $\vartheta_A = \vartheta$). Furthermore, even with a small offset the tracking error is such that the actual trajectory passes through the singularity of the controller Γ^b_{DFL} (see Sect. 4.6.2) causing an unstable behavior.

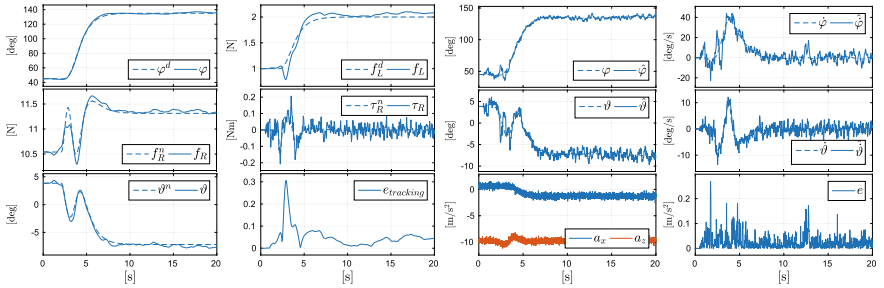
On the other hand, the controller Γ^a_{DFL} turned to be much more robust to these sort of structural model variations. From Fig. 5.26 we can see that the parameters

that mostly affect an increase of the error are the entries of $\mathbf{r}_{BL}$, i.e., r_{BLx} and r_{BLz}. One can notice that it is more advisable to attach the link such that one is sure that $r_{BLz} \leq 0$, especially if agile motions are required. The effect of the displacement along $\boldsymbol{x}_R$ is instead almost symmetric. The small asymmetry is due to the particular trajectory passing from the first to the second quadrant. The mean tracking error increases instead almost linearly with respect to the mass of the link. Nevertheless, even with m_L equal to the 20% of m_R the closed loop system remains still perfectly stable.

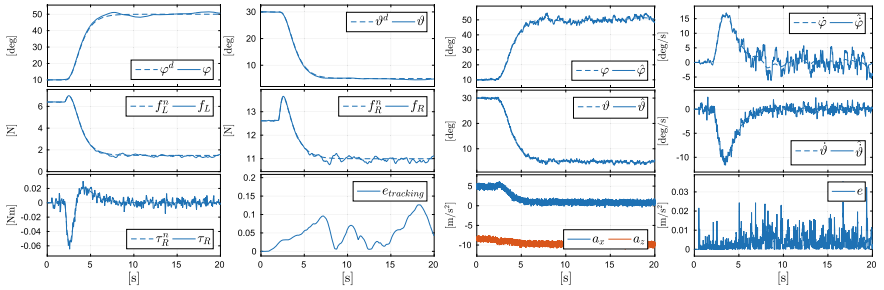
5.4.2.4 Noisy Measurements

In this section we investigate the robustness of the proposed method in presence of noisy measurements, which always exist in reality. We consider both the accelerometer and the gyroscope measures being affected by a white Gaussian noise of variance $0.1[\mathrm{m/s^2}]$ and $0.01[\mathrm{rad/s}]$ respectively.

From Fig. 5.27 we can observe that the estimated state shows some noise but the corresponding error remains always bounded. Due to the noisy component on the estimated state the control action presents some oscillations that imply a non exact

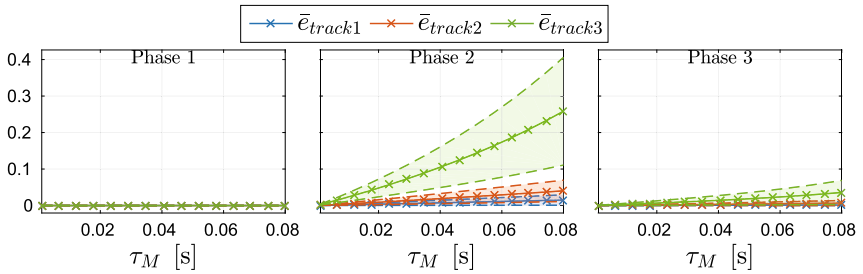


(a) Results of Γ^a_{DFL} and of the observer, on the left and on the right, respectively.

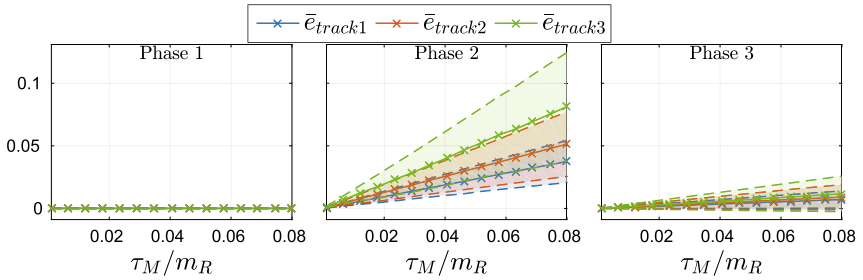


(b) Results of Γ^b_{DFL} and of the observer, on the left and on the right, respectively.

Fig. 5.27 Simulation results: noisy measurements. © 2020 IEEE. Reprinted, with permission, from [8]



(a) Controller Γ^a_{DFL}.



(b) Controller Γ^b_{DFL}.

Fig. 5.28 Simulation results: non ideal motors. Mean tracking error when changing the motor time constant

tracking of the desired trajectory. Nevertheless the tracking error remains small and always bounded. Notice that to achieve these results we had to reduce the gains of both the controller and observer. Indeed, high gain values increase the convergence speed but also amplify the sensitivity to noisy measurements. In general the two controllers does not show particularly different behaviors in face of the presence of noise.

5.4.2.5 Non-ideal Motors

In a real scenario, one motor cannot instantaneously change the spinning velocity of the propeller, and in turn the thrust produced. Indeed, this discontinuous variation of the speed would require the application of an infinite torque by the motor, that is clearly not possible. Instead the dynamics of the motor is characterized by a certain time constant, $\tau_M \in \mathbb{R}$, that quantifies the time needed to change the motor speed. In this section we analyze this additional non ideality testing the proposed method with different non ideal motors characterized by an increasing time constant. In Fig. 5.28 we show the relative mean and variance of the tracking error for the different time constant values τ_M. The plots clearly shows that increasing the time constant the tracking error increases as well, especially during the dynamic part of the desired

trajectory (Phase 2). Indeed, for motors with higher time constant, the error between commanded and actuated thrust on each propeller increases causing a bigger tracking error. However, the system remains stable up to a time constant of 0.08[s], which is completely acceptable in real systems.

This analysis is important for the scalability of the system. Indeed, bigger vehicles with higher mass imply the need of an higher lift that can be in general generated by bigger propellers. This in turn requires the use of bigger motors that are characterized by a larger time constant. Finally, as shown in Fig 5.28, the larger mass of the system, and so the larger time constant of the motors, reduces the tracking performance of the system for dynamic trajectories. Therefore, when we increase the dimension and the mass of the vehicle, in order to still obtain good tracking performance, it is necessary to reduce the agility of the desired maneuver reducing the demanded accelerations.

References

1. Franchi, A., Mallet, A.: Adaptive closed-loop speed control of BLDC motors with applications to multi-rotor aerial vehicles. In: 2017 IEEE International Conference on Robotics and Automation, pp. 5203–5208. Singapore (2017)
2. Kolmanovsky, I., Garone, E., Cairano, S.D.: Reference and command governors: a tutorial on their theory and automotive applications. In: 2014 American Control Conference, pp. 226–241. IEEE (2014)
3. Video 1. Takeoff and landing on slopes via inclined hovering with a tethered aerial robot (2016). https://www.youtube.com/watch?v=01UYN289YXk
4. Tognon, M., Dash, S.S., Franchi, A.: Observer-based control of position and tension for an aerial robot tethered to a moving platform. IEEE Robot. Autom. Lett. **1**(2), 732–737 (2016)
5. Video 2. Observer-based control of position and tension for an aerial robot tethered to a moving platform (2016). https://www.youtube.com/watch?v=VGnTBFCPbWg
6. Sandino, L., Santamaria, D., Bejar, M., Viguria, A., Kondak, K., Ollero, A.: Tether-guided landing of unmanned helicopters without GPS sensors. In: 2014 IEEE International Conference on Robotics and Automation, pp. 3096–3101. Hong Kong, China (2014)
7. Mahony, R., Hamel, T., Pflimlin, J.-M.: Nonlinear complementary filters on the special orthogonal group. IEEE Trans. Autom. Control **53**(5), 1203–1218 (2008)
8. Tognon, M., Franchi, A.: Dynamics, control, and estimation for aerial robots tethered by cables or bars. IEEE Trans. Robot. **33**(4), 834–845 (2017)
9. Video 3. Nonlinear observer-based tracking control of link stress and elevation for a tethered aerial robot using inertial-only measurements (2015). https://www.youtube.com/watch?v=6Y3NDnhLWr0