16.485: Visual Navigation for Autonomous Vehicles (VNAV) **Fall 2020**

Lecture 7

Lecturer: Luca Carlone *Scribes: Luca Carlone, Markus Ryll*

This lecture introduces the principles of two standard and well-known quadrotor controllers. First, we discuss an almost globally-stable *geometric controller*, which provides a state-of-the-art approach to perform aggressive maneuvers with quadrotors. The second controller is a lightweight cascaded Proportional Derivative (PD) controller, which works well near hovering. Reading the section about the near-hovering controller is optional and such controller is only reviewed as an example of "old-style" quadrotor control design.

In the following we will refer to the variable names depicted in Fig. 7.1.

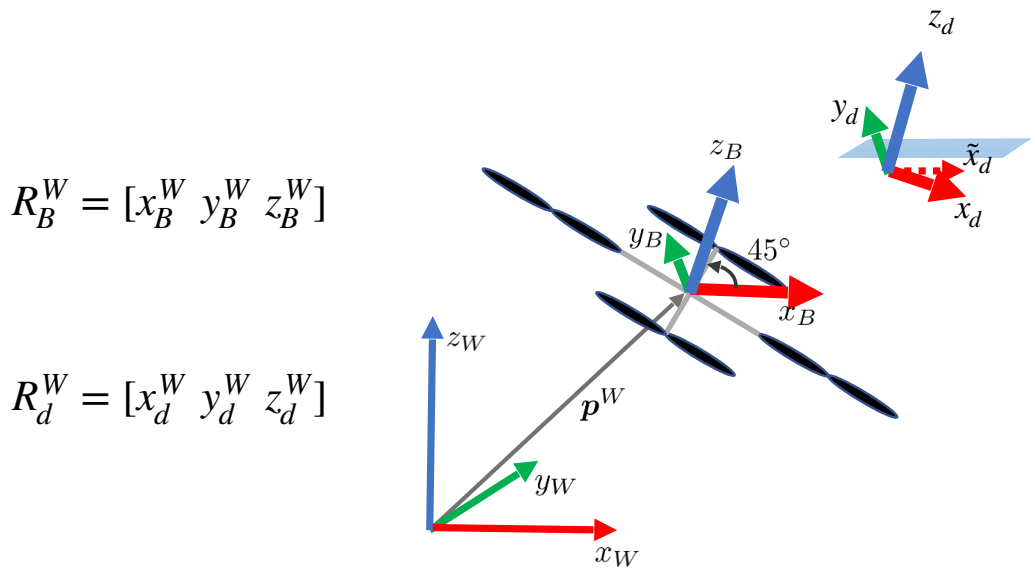

Figure 7.1: Quadrotor - most important variables are labeled. The figure shows the world frame w, the current quadrotor body frame B, and the desired quadrotor frame d (this is where we want our drone to go to). Note that we cannot simultaneously control the position/velocity and roll and pitch, so we only attempt to track a desired yaw direction (indicated by the vector $\tilde{\boldsymbol{x}}_d$), rather than the full attitude $\boldsymbol{R}_d^{\mathrm{w}}$. The rotations on the left are a reminder that the columns of the rotation $\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}}$ correspond to the axis of the body frame (e.g., $\boldsymbol{x}_{\mathrm{B}}$) expressed in the reference frame w (hence $\boldsymbol{x}_{\mathrm{B}}^{\mathrm{w}}$); the same holds for the rotation $\boldsymbol{R}_d^{\mathrm{w}}$.

7.1 Notation and definitions

In the rest of these lecture notes, the subscript $[\cdot]_d$ will indicate that $[\cdot]_d$ is the desired value of $[\cdot]$, e.g. $\boldsymbol{p}_d^{\mathrm{w}}$ refers to the desired position while $\boldsymbol{p}^{\mathrm{w}}$ is the current position (in the world frame).

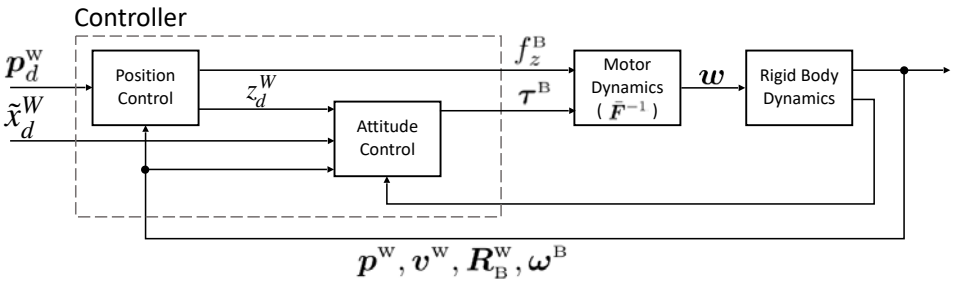

Figure 7.2: The geometric control loops for position and attitude control.

Also in this lecture we will use the common model with the reference frame rotated by 45° with respect to the quadrotor propeller axes (see Fig. 7.1). This definition is more common as we can align the x-axis with a camera frame (with the camera standing between two quadrotor arms), in a way that is not occluded by the quadrotor arms. This is the same model used in the Lab 3 exercises.

7.2 Geometric Controller

This section reviews the geometric controller presented in [3] that we adapt to our reference frame conventions (Fig. 7.1).

A quadrotor has 4 control inputs (the speed of the 4 rotors) but has 6 degrees of freedom (3 for its translation and 3 for its rotation). Since we have less inputs than controls, we will not be able to control independently all the degrees of freedom. Intuitively, if the quadrotor has a pitch or roll angle, its dynamics will force it to move, hence the quadrotor cannot simultaneously keep a desired position and a desired roll and pitch. Therefore, in this lecture we will only discuss how to control 4 degrees of freedom, and in particular the quadrotor position $\boldsymbol{p}^{\mathrm{W}}$ and the yaw angle ψ. We denote with $\boldsymbol{p}_d^{\mathrm{W}}$ and ψ_d the desired position and the desired yaw. We equivalently define the yaw angle with a vector in the horizontal plane $\tilde{\boldsymbol{x}}_d^{\mathrm{W}} = [\cos(\psi)\ \sin(\psi)\ 0]^{\mathsf{T}}$.

In the following, we first present the control law and then provide some intuition about its effectiveness.

7.2.1 Tracking Errors and Control Design

In Lecture 6 we have seen that the quadrotor dynamics satisfy:

$$\begin{bmatrix} m\ddot{\boldsymbol{p}}^{\mathrm{W}} \\ \boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\mathrm{B}} \end{bmatrix} = \begin{bmatrix} -mg\boldsymbol{e}_3 \\ -\boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\mathrm{B}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix} \boldsymbol{F}\boldsymbol{w} \tag{7.1}$$

Noting that the first two rows of $\boldsymbol{F}$ are always zero in our quadrotor model, we rewrite eq. (6.9) in Lecture 6 as follows:

$$\begin{bmatrix} f_z^{\mathrm{B}} \\ \boldsymbol{\tau}^{\mathrm{B}} \end{bmatrix} = \bar{\boldsymbol{F}}\boldsymbol{w} \tag{7.2}$$

where $\bar{\boldsymbol{F}}$ is now a 4×4 invertible matrix. Since $\bar{\boldsymbol{F}}$ is constant, we can always move back and forth from $\boldsymbol{w}$ to $[f_z^{\mathrm{B}}\ ;\ \boldsymbol{\tau}^{\mathrm{B}}]$ and for this reason in the following we will directly think about $[f_z^{\mathrm{B}}\ ;\ \boldsymbol{\tau}^{\mathrm{B}}]$ as our control inputs and write (7.1) as:

$$\begin{bmatrix} m\ddot{\boldsymbol{p}}^{\mathrm{W}} \\ \boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\mathrm{B}} \end{bmatrix} = \begin{bmatrix} -mg\boldsymbol{e}_3 \\ -\boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\mathrm{B}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{z}_{\mathrm{B}}^{\mathrm{W}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_3 \end{bmatrix} \begin{bmatrix} f_z^{\mathrm{B}} \\ \boldsymbol{\tau}^{\mathrm{B}} \end{bmatrix} \tag{7.3}$$

7.2.2 Tracking Errors

One might think about the geometric controller as a PD (Proportional Derivative) controller, but with error metrics that respect the Lie group structure of the rotations. The errors are defined as follows:

$$\boldsymbol{e}_p = \boldsymbol{p}^{\mathrm{w}} - \boldsymbol{p}_d^{\mathrm{w}} \qquad \text{(position error)} \tag{7.4}$$

$$\boldsymbol{e}_v = \boldsymbol{v}^{\mathrm{w}} - \boldsymbol{v}_d^{\mathrm{w}} \qquad \text{(linear velocity error)} \tag{7.5}$$

$$\boldsymbol{e}_R = \frac{1}{2}\left[(\boldsymbol{R}_d^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}} - (\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_d^{\mathrm{w}}\right]^{\vee} \qquad \text{(rotation error)} \tag{7.6}$$

$$\boldsymbol{e}_\omega = \boldsymbol{\omega}^{\mathrm{B}} - (\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_d^{\mathrm{w}}\boldsymbol{\omega}_d \qquad \text{(angular velocity error)} \tag{7.7}$$

The position and (linear) velocity errors are quite straightforward and indeed resemble what one would use in a standard PD controller; again $\boldsymbol{p}_d^{\mathrm{w}}$ and $\boldsymbol{v}_d^{\mathrm{w}}$ are the desired position and velocity which is assumed to be given[1]. However, the expressions of the rotation and angular velocity errors are more involved. While, we are going to provide some intuitions about these tracking error in the following paragraph, we can think about the rotation and angular velocity errors as carefully-designed vectors where $\boldsymbol{e}_R$ measures the mismatch between the current quadrotor rotation $\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}}$ and the desired rotation $\boldsymbol{R}_d^{\mathrm{w}}$, and where $\boldsymbol{e}_\omega$ measures the mismatch between the current body-frame quadrotor angular velocity $\boldsymbol{\omega}^{\mathrm{B}}$ and the desired body-frame angular velocity $\boldsymbol{\omega}_d$.

Intuition for the rotation error: Let us discuss the intuition behind the rotation error in (7.6). In the Lie group lectures, we mentioned that the angular distance between two rotations $\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}}$ and $\boldsymbol{R}_d^{\mathrm{w}}$ can be computed as:

$$\mathrm{dist}_\theta(\boldsymbol{R}_d^{\mathrm{w}}, \boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}}) = \|\log((\boldsymbol{R}_d^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\vee}\| \tag{7.8}$$

It turns out that the logmap for SO(3) can be simply computed as (see, e.g., https://en.wikipedia.org/ wiki/Axis%E2%80%93angle_representation):

$$\log(\boldsymbol{R}) = \begin{cases} \boldsymbol{0} & \text{if } \boldsymbol{R} = \mathbf{I}_3 \\ \frac{\theta}{2\sin\theta}[\boldsymbol{R} - \boldsymbol{R}^{\mathsf{T}}] & \text{otherwise} \end{cases} \tag{7.9}$$

where θ is the rotation angle.

Therefore, we can develop the log inside the angular distance (7.8) as:

$$\log((\boldsymbol{R}_d^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\vee} = \frac{\theta}{\sin\theta}[(\boldsymbol{R}_d^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}} - (\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_d^{\mathrm{w}}]^{\vee} \tag{7.10}$$

from which it should be clear that the rotation error in (7.6) is simply a vector parametrization of the relative error between the desired rotation $\boldsymbol{R}_d^{\mathrm{w}}$ and the actual rotation $\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}}$ (up to scale).

Intuition for the angular velocity error: The intuition behind the angular velocity error in (7.7) is even simpler. The basic observation is that the desired angular velocity $\boldsymbol{\omega}_d$ and the current angular velocity $\boldsymbol{\omega}^{\mathrm{B}}$ live in two different reference frames. The vector $\boldsymbol{\omega}^{\mathrm{B}}$ lives in the body frame of the quadrotor, while $\boldsymbol{\omega}_d$ lives in the desired body frame of the quadrotor (cf. Fig. 7.1). For these two quantities to be comparable, we should express them in the same reference frame. In (7.7), the term $(\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\mathsf{T}}\boldsymbol{R}_d^{\mathrm{w}}\boldsymbol{\omega}_d$ has the effect of first expressing $\boldsymbol{\omega}_d$ in the world frame "w" (via the multiplication by $\boldsymbol{R}_d^{\mathrm{w}}$) and then transform it to the current body frame "B" (via the multiplication by $(\boldsymbol{R}_{\mathrm{B}}^{\mathrm{w}})^{\mathsf{T}} = \boldsymbol{R}_{\mathrm{w}}^{\mathrm{B}}$).

[1] In Lecture 9-11 we will discuss how to obtain the desired positions and velocities using trajectory optimization

7.2.3 Controller

We can now introduce the geometric controller. For given desired state $\boldsymbol{p}_d^{\mathrm{W}}$, $\tilde{\boldsymbol{x}}_d^{\mathrm{W}}$ (and the corresponding tracking errors $\boldsymbol{e}_p, \boldsymbol{e}_v, \boldsymbol{e}_R, \boldsymbol{e}_\omega$), and some positive constants k_p, k_v, k_R, k_ω, the control inputs $f_z^{\mathrm{B}}, \boldsymbol{\tau}^{\mathrm{B}}$ are chosen as follows:

$$f_z^{\mathrm{B}} = (-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}}) \cdot \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} \boldsymbol{e}_3, \tag{7.11}$$

$$\boldsymbol{\tau}^{\mathrm{B}} = -k_R \boldsymbol{e}_R - k_\omega \boldsymbol{e}_\omega + \boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}} \boldsymbol{\omega}^{\mathrm{B}} - \boldsymbol{\mathcal{J}}([\boldsymbol{\omega}^{\mathrm{B}}]_\times (\boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}})^{\mathsf{T}} \boldsymbol{R}_d^{\mathrm{W}} \boldsymbol{\omega}_d - (\boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}})^{\mathsf{T}} \boldsymbol{R}_d^{\mathrm{W}} \dot{\boldsymbol{\omega}}_d), \tag{7.12}$$

where the desired attitude $\boldsymbol{R}_d^{\mathrm{W}}$ is computed as $\boldsymbol{R}_d^{\mathrm{W}} = [\frac{(\boldsymbol{z}_d^{\mathrm{W}} \times \tilde{\boldsymbol{x}}_d^{\mathrm{W}}) \times \boldsymbol{z}_d^{\mathrm{W}}}{\|(\boldsymbol{z}_d^{\mathrm{W}} \times \tilde{\boldsymbol{x}}_d^{\mathrm{W}}) \times \boldsymbol{z}_d^{\mathrm{W}}\|}\ ,\ \frac{\boldsymbol{z}_d^{\mathrm{W}} \times \tilde{\boldsymbol{x}}_d^{\mathrm{W}}}{\|\boldsymbol{z}_d^{\mathrm{W}} \times \tilde{\boldsymbol{x}}_d^{\mathrm{W}}\|}\ ,\ \boldsymbol{z}_d^{\mathrm{W}}] \in \mathrm{SO}(3)$, and:

$$\boldsymbol{z}_d^{\mathrm{W}} = \boldsymbol{R}_d^{\mathrm{W}} \boldsymbol{e}_3 = \frac{-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}}}{\| - k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}}\|}, \tag{7.13}$$

Here, we assume that the denominator of (7.13) is non-zero,

$$\| - k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}}\| \neq 0, \tag{7.14}$$

and that the desired trajectory satisfies

$$\| - mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d\| \ < B \tag{7.15}$$

for a given positive constant B.

The control moment $\boldsymbol{\tau}^{\mathrm{B}}$ given in (7.12) corresponds to a tracking controller on SO(3). For the attitude dynamics of a rigid body described by eq. (7.3), (in both the rotational part) this controller exponentially stabilizes the zero equilibrium of the attitude tracking errors. Similarly, the expression in the parentheses in (7.11) corresponds to a tracking controller for the translational dynamics on $\mathbb{R}^3$. The total thrust f_z^{B} and the desired direction $\boldsymbol{z}_d^{\mathrm{W}}$ of the third body-fixed axis are chosen so that –if there is no attitude tracking error– the control input term $f_z^{\mathrm{B}} \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} \boldsymbol{e}_3$ reduces the position and velocity errors. Therefore, the trajectory tracking error will converge to zero provided that the attitude tracking error is identically zero. Certainly, the attitude tracking error may not be zero at any instant. As the attitude tracking error increases, the direction of the control input term $f_z^{\mathrm{B}} \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} \boldsymbol{e}_3$ of the translational dynamics deviates from the desired direction of $\boldsymbol{R}_d^{\mathrm{W}} \boldsymbol{e}_3$. This may cause instability on the complete dynamics. In (7.11), we carefully design the total thrust f_z^{B} so that its magnitude is reduced when there is a larger attitude tracking error. The expression of f_z^{B} includes the dot product of the desired third body-fixed axis $\boldsymbol{z}_d^{\mathrm{W}} = \boldsymbol{R}_d^{\mathrm{W}} \boldsymbol{e}_3$ and the current third body-fixed axis $\boldsymbol{z}_{\mathrm{B}}^{\mathrm{W}} = \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} \boldsymbol{e}_3$. Therefore, the magnitude of f_z^{B} is reduced when the angle between those two axes becomes larger. A stability proof can be found in [3].

Intuition about the design of the thrust f_z^{B} and the desired rotation $\boldsymbol{R}_d^{\mathrm{W}}$: Let us consider the translational dynamics in eq. (7.3):

$$m\ddot{\boldsymbol{p}}^{\mathrm{W}} = -mg\boldsymbol{e}_3 + \boldsymbol{R}_{\mathrm{B}}^{\mathrm{W}} \boldsymbol{f}^{\mathrm{B}} \tag{7.16}$$

Of course, if we compare (7.16) with (7.3) we realize that we can only apply a force $\boldsymbol{f}^{\mathrm{B}}$ along the local $\boldsymbol{z}$ of the quadrotor, but let's put that aside for a moment. In the ideal case where we had the freedom to apply forces in any direction, an ideal force to apply would be:

$$\boldsymbol{f}_{\mathrm{ideal}}^{\mathrm{B}} = \boldsymbol{R}_{\mathrm{w}}^{\mathrm{B}}(-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}}) \tag{7.17}$$

since when we substitute such force into (7.16) we get:

$$m\ddot{\boldsymbol{p}}^{\mathrm{W}} = -mg\boldsymbol{e}_3 - k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}} \iff \tag{7.18}$$

$$m\ddot{\boldsymbol{p}}^{\mathrm{W}} = -k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + m\ddot{\boldsymbol{p}}_d^{\mathrm{W}} \iff \tag{7.19}$$

$$(\text{using } \boldsymbol{e}_v = \dot{\boldsymbol{e}}_p \text{ and } \ddot{\boldsymbol{p}}^{\mathrm{W}} - \ddot{\boldsymbol{p}}_d = \ddot{\boldsymbol{e}}_p) \tag{7.20}$$

$$m\ddot{\boldsymbol{e}}_p = -k_p \boldsymbol{e}_p - k_v \dot{\boldsymbol{e}}_p \tag{7.21}$$

which is now a simple linear system describing the dynamics of the position errors. By suitable tuning of the gains k_p and k_v we can ensure that this system converges to zero position (and velocity) error.

Since the drone dynamics restrict the thrust to be along the local body $\boldsymbol{z}_{\text{B}}^{\text{W}}$ of the quadrotor, the basic idea of the geometric controller is to project the "ideal" thrust force (7.22) along the local body $\boldsymbol{z}_{\text{B}}^{\text{W}}$ to get:

$$f_{\boldsymbol{z}}^{\text{B}} = (-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\text{W}}) \cdot \boldsymbol{z}_{\text{B}}^{\text{W}} = (-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\text{W}}) \cdot \boldsymbol{R}_{\text{B}}^{\text{W}} \boldsymbol{e}_3 \tag{7.22}$$

which matches the thrust design in eq. (7.11).

Note that $f_{\boldsymbol{z}}^{\text{B}}$ matches the "ideal" force $\boldsymbol{f}_{\text{ideal}}^{\text{B}}$ when the local body $\boldsymbol{z}_{\text{B}}^{\text{W}}$ points in the direction of $-k_p \boldsymbol{e}_p - k_v \boldsymbol{e}_v + mg\boldsymbol{e}_3 + m\ddot{\boldsymbol{p}}_d^{\text{W}}$. This is indeed the rational for choosing this vector as the desired local body $\boldsymbol{z}_d^{\text{W}}$ in (7.13). The desired local body $\boldsymbol{x}_d^{\text{W}}$ and $\boldsymbol{y}_d^{\text{W}}$ are built such that the yaw angle matches the desired yaw angle $\tilde{\boldsymbol{x}}_d^{\text{W}}$, see Fig. 7.1, and the axis are orthogonal.

Intuition about the design of the torque $\boldsymbol{\tau}^{\text{B}}$: Similarly to the previous case, we can substitute the torque $\boldsymbol{\tau}^{\text{B}}$ in eq. (7.12) back into the rotational dynamics (7.3):

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\text{B}} = -\boldsymbol{\omega}^{\text{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\text{B}} - k_R \boldsymbol{e}_R - k_\omega \boldsymbol{e}_\omega + \boldsymbol{\omega}^{\text{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\text{B}} + \boldsymbol{\mathcal{J}}(-[\boldsymbol{\omega}^{\text{B}}]_\times (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d + (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \dot{\boldsymbol{\omega}}_d) \iff \tag{7.23}$$

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\text{B}} = -k_R \boldsymbol{e}_R - k_\omega \boldsymbol{e}_\omega + \boldsymbol{\mathcal{J}}(-[\boldsymbol{\omega}^{\text{B}}]_\times (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d + (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \dot{\boldsymbol{\omega}}_d) \tag{7.24}$$

Now we observe that:

$$\dot{\boldsymbol{e}}_\omega = \frac{d}{dt}(\boldsymbol{\omega}^{\text{B}} - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d) = \dot{\boldsymbol{\omega}}^{\text{B}} - (\dot{\boldsymbol{R}}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \dot{\boldsymbol{R}}_d^{\text{W}} \boldsymbol{\omega}_d - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \dot{\boldsymbol{\omega}}_d = \tag{7.25}$$

$$\dot{\boldsymbol{\omega}}^{\text{B}} - (\boldsymbol{R}_{\text{B}}^{\text{W}} [\boldsymbol{\omega}^{\text{B}}]_\times)^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} [\boldsymbol{\omega}_d]_\times \boldsymbol{\omega}_d - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \dot{\boldsymbol{\omega}}_d = \tag{7.26}$$

$$\text{(recalling that } [\boldsymbol{\omega}^{\text{B}}]_\times^\mathsf{T} = -[\boldsymbol{\omega}^{\text{B}}]_\times \text{ and } [\boldsymbol{\omega}_d]_\times \boldsymbol{\omega}_d = \boldsymbol{0}) \tag{7.27}$$

$$\dot{\boldsymbol{\omega}}^{\text{B}} + [\boldsymbol{\omega}^{\text{B}}]_\times (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \boldsymbol{\omega}_d - (\boldsymbol{R}_{\text{B}}^{\text{W}})^\mathsf{T} \boldsymbol{R}_d^{\text{W}} \dot{\boldsymbol{\omega}}_d = \tag{7.28}$$

Noting that this expression matches terms present in (7.24), eq. (7.24) can be written as:

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{e}}_\omega = -k_R \boldsymbol{e}_R - k_\omega \boldsymbol{e}_\omega \tag{7.29}$$

which again is a simple linear system describing the rotation and angular velocity error dynamics. By suitable tuning of the gains k_R and k_ω we can ensure that this system converges to zero rotation and angular velocity error.

7.3 Near-Hovering Controller

This section is optional and is given as an example of how quadrotor controllers were designed before the geometric controller was proposed. The presentation is mostly based on [4].

The aerial robot is controlled by nested feedback loops as shown in Fig. 7.3. The inner attitude control loop usually utilizes an onboard IMU with accelerometers and gyroscopes to control the roll ϕ, pitch θ, and yaw ψ and runs typically at a high frequency (200 Hz – 2 kHz), while the outer position control loop uses estimates of position and velocity of the center of mass to control the trajectory in three dimensions. The controllers are derived by linearizing the equations of motion at an operating point that corresponds to the nominal hover state, $\boldsymbol{p}^{\text{W}} = \boldsymbol{p}_0^{\text{W}}$, $\theta = \phi = 0$, $\psi = \psi_0$, $\dot{\boldsymbol{p}}^{\text{W}} = \boldsymbol{0}$, and $\dot{\theta} = \dot{\phi} = \dot{\psi} = 0$, where the roll and pitch angles are small ($\cos(\phi) \approx 1$, $\cos(\theta) \approx 1$, $\sin(\phi) \approx \phi$, and $\sin(\theta) \approx \theta$).

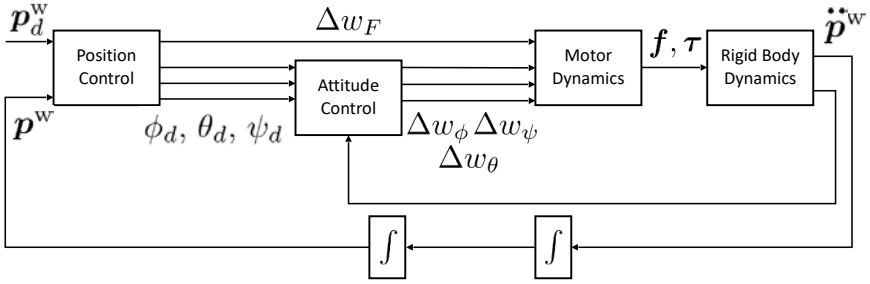

Figure 7.3: The nested control loops for position and attitude control.

At this hover state, the nominal thrusts from the propellers must satisfy

$$\boldsymbol{T}_{i,0} = -\frac{m\boldsymbol{g}^{\mathrm{W}}}{4}, \tag{7.30}$$

meaning that the thrust of each propeller is in charge of compensating for 1/4 of the gravity force $m\boldsymbol{g}^{\mathrm{W}}$.

In order to produce this torque, at hovel, the motor speeds must be set to:

$$\omega_{i,0} = \omega_h = \sqrt{\frac{mg}{4c_f}}, \tag{7.31}$$

where $g = \|\boldsymbol{g}^{\mathrm{W}}\|$ ($\approx 9.81 m/s^2$).

7.3.1 Attitude Control

We will now discuss the attitude controller to track trajectories in $SO(3)$ that are close to the nominal hover state where the roll ϕ and pitch θ angles are small. Using the rotational part of the Newton-Euler equation (see eq. (6.8) in the previous lecture), with angular velocities of the rotors $\boldsymbol{w} = [w_1|w_1|\ w_2|w_2|\ w_3|w_3|\ w_4|w_4|]^T$, yields to

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\mathrm{B}} = -\boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\mathrm{B}} + \boldsymbol{\tau}^{\mathrm{B}} \quad \Longleftrightarrow \tag{7.32}$$

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\mathrm{B}} = -\boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\mathrm{B}} + \boldsymbol{\tau}^{\mathrm{B}}_{\mathrm{drag}} + \boldsymbol{\tau}^{\mathrm{B}}_{\mathrm{thrust}} \quad \Longleftrightarrow \tag{7.33}$$

$$\boldsymbol{\mathcal{J}}\dot{\boldsymbol{\omega}}^{\mathrm{B}} = -\boldsymbol{\omega}^{\mathrm{B}} \times \boldsymbol{\mathcal{J}}\boldsymbol{\omega}^{\mathrm{B}} + \begin{bmatrix} \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_1^{\mathrm{B}}|| & \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_2^{\mathrm{B}}|| & -\sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_3^{\mathrm{B}}|| & -\sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_4^{\mathrm{B}}|| \\ -\sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_1^{\mathrm{B}}|| & \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_2^{\mathrm{B}}|| & \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_3^{\mathrm{B}}|| & -\sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}_4^{\mathrm{B}}|| \\ -c_d & c_d & -c_d & c_d \end{bmatrix} \begin{bmatrix} w_1|w_1| \\ w_2|w_2| \\ w_3|w_3| \\ w_4|w_4| \end{bmatrix} \tag{7.34}$$

where we assumed that the arms are mounted at 45° as in Fig. 7.1.

Let us call the rotational velocity components $\boldsymbol{\omega}^{\mathrm{B}} = [p,\ q,\ r]$. Assuming that $||\boldsymbol{\rho}_1^{\mathrm{B}}|| = ||\boldsymbol{\rho}_2^{\mathrm{B}}|| = ||\boldsymbol{\rho}_3^{\mathrm{B}}|| = ||\boldsymbol{\rho}_4^{\mathrm{B}}|| = ||\boldsymbol{\rho}^{\mathrm{B}}||$ and that $\boldsymbol{\mathcal{J}}$ is a diagonal matrix with $\boldsymbol{\mathcal{J}} = \mathrm{diag}[\mathcal{J}_{xx}\ \mathcal{J}_{yy}\ \mathcal{J}_{zz}]$, we can reformulate (7.34) as

$$\mathcal{J}_{xx}\dot{\mathrm{p}} = \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}^{\mathrm{B}}||(w_1|w_1| + w_2|w_2| - w_3|w_3| - w_4|w_4|) - qr(\mathcal{J}_{zz} - \mathcal{J}_{yy}) \tag{7.35}$$

$$\mathcal{J}_{yy}\dot{\mathrm{q}} = \sqrt{\frac{1}{2}}c_f||\boldsymbol{\rho}^{\mathrm{B}}||(-w_1|w_1| + w_2|w_2| + w_3|w_3| - w_4|w_4|) - pr(\mathcal{J}_{xx} - \mathcal{J}_{zz}) \tag{7.36}$$

$$\mathcal{J}_{zz}\dot{\mathrm{r}} = c_d(-w_1|w_1| + w_2|w_2| - w_3|w_3| + w_4|w_4|) - pq(\mathcal{J}_{yy} - \mathcal{J}_{xx}). \tag{7.37}$$

The last term in eq. (7.37) is small, since $\mathcal{J}_{xx} \approx \mathcal{J}_{yy}$ because of the symmetry. Additionally, we assume the component of the angular velocity in the $\boldsymbol{z}_{\mathrm{B}}$ direction, r, is small so the rightmost terms in (7.35) and (7.36) which are products involving r are also small compared to the other terms.

Before using the expressions above, we reparametrize the vector of desired rotor speeds in terms of the nominal motor speed ω_h and the deviations $\Delta\omega_F, \Delta\omega_\phi, \Delta\omega_\theta, \Delta\omega_\psi$:

$$\begin{bmatrix} w_1^{\mathrm{d}} \\ w_2^{\mathrm{d}} \\ w_3^{\mathrm{d}} \\ w_4^{\mathrm{d}} \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \omega_h + \Delta\omega_F \\ \Delta\omega_\phi \\ \Delta\omega_\theta \\ \Delta\omega_\psi \end{bmatrix} \iff \begin{bmatrix} \omega_h + \Delta\omega_F \\ \Delta\omega_\phi \\ \Delta\omega_\theta \\ \Delta\omega_\psi \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} w_1^{\mathrm{d}} \\ w_2^{\mathrm{d}} \\ w_3^{\mathrm{d}} \\ w_4^{\mathrm{d}} \end{bmatrix} \tag{7.38}$$

where the nominal rotor speed required to hover in steady state is ω_h (see eq. (7.31)), and the deviations from this nominal vector are $\Delta\omega_F, \Delta\omega_\phi, \Delta\omega_\theta$, and $\Delta\omega_\psi$. The term $\Delta\omega_F$ results in a net force along the $\boldsymbol{z}_{\mathrm{B}}$ axis, while $\Delta\omega_\phi, \Delta\omega_\theta$, and $\Delta\omega_\psi$ produce moments causing roll, pitch, and yaw, respectively.

Now we linearize (7.35), (7.36) and (7.37) about the hovering operating point and write the desired angular accelerations in terms of the new control inputs

$$\dot{\mathrm{p}}_d = \frac{4\sqrt{\frac{1}{2}c_f}||\boldsymbol{\rho}^{\mathrm{B}}||\omega_h}{\mathcal{J}_{xx}}\Delta\omega_\phi, \tag{7.39}$$

$$\dot{\mathrm{q}}_d = \frac{4\sqrt{\frac{1}{2}}||\boldsymbol{\rho}^{\mathrm{B}}||\omega_h}{\mathcal{J}_{yy}}\Delta\omega_\theta, \tag{7.40}$$

$$\dot{\mathrm{r}}_d = \frac{4c_d\omega_h}{\mathcal{J}_{zz}}\Delta\omega_\psi. \tag{7.41}$$

As near the nominal hover state $\dot{\phi} \approx p, \dot{\theta} \approx q$, and $\dot{\psi} \approx r$, we use proportional derivative control laws that take the form

$$\begin{aligned} \Delta\omega_\phi &= k_{p,\phi}(\phi_d - \phi) + k_{d,\phi}(p_d - p) \\ \Delta\omega_\theta &= k_{p,\theta}(\theta_d - \theta) + k_{d,\theta}(q_d - q) \\ \Delta\omega_\psi &= k_{p,\psi}(\psi_d - \psi) + k_{d,\psi}(r_d - r)\ . \end{aligned} \tag{7.42}$$

where ϕ_d, θ_d, ψ_d are the desired roll, pitch, and yaw angles. Substituting (7.42) into (7.38) yields the desired rotor speeds.

7.3.2 Position Control

The 3D position controller is used to follow three-dimensional trajectories with modest accelerations so the near-hover assumptions hold. Let's define again, $\boldsymbol{p}_d^{\mathrm{w}}$ as the desired position and $\boldsymbol{p}^{\mathrm{w}}$ as the current position (both in the world frame). Let the unit vector tangent to the trajectory at point $\boldsymbol{p}_d^{\mathrm{w}}$ be $\hat{\boldsymbol{t}} = \frac{\dot{\boldsymbol{p}}_d^{\mathrm{w}}}{||\dot{\boldsymbol{p}}_d^{\mathrm{w}}||}$ and the desired velocity vector be $\dot{\boldsymbol{p}}_d^{\mathrm{w}}$. We define the position and velocity errors as

$$\boldsymbol{e}_p = ((\boldsymbol{p}_d^{\mathrm{w}} - \boldsymbol{p}^{\mathrm{w}}) \cdot \hat{\boldsymbol{n}})\hat{\boldsymbol{n}} + ((\boldsymbol{p}_T^W + \boldsymbol{p}) \cdot \hat{\boldsymbol{b}})\hat{\boldsymbol{b}} \tag{7.43}$$

and

$$\boldsymbol{e}_v = \dot{\boldsymbol{p}}_d^{\mathrm{w}} - \dot{\boldsymbol{p}}^{\mathrm{w}}. \tag{7.44}$$

where $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{b}}$ are unit vectors orthogonal to $\hat{\boldsymbol{t}}$. Note that the position error $\boldsymbol{e}_p$ ignores the error in the tangent direction $\hat{\boldsymbol{t}}$ and only considers the position error in the normal directions $\hat{\boldsymbol{n}}$ and $\hat{\boldsymbol{b}}$.

We calculate the commanded acceleration, $\ddot{\boldsymbol{p}}$, from PD feedback of the position and velocity errors:

$$\ddot{\boldsymbol{p}} = k_p \boldsymbol{e}_p + k_d \boldsymbol{e}_v + \ddot{\boldsymbol{p}}_d^{\mathrm{w}} \tag{7.45}$$

This and very similar control approaches have been presented in [1, 2, 4], which also show how to convert this desired acceleration into the desired roll, pitch, and yaw angles ϕ_d, θ_d, ψ_d.

References

[1] Samir Bouabdallah. Design and control of quadrotors with application to autonomous flying. Technical report, Epfl, 2007.

[2] Daniel Gurdan, Jan Stumpf, Michael Achtelik, Klaus-Michael Doth, Gerd Hirzinger, and Daniela Rus. Energy-efficient autonomous four-rotor flying robot controlled at 1 khz. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 361–366. IEEE, 2007.

[3] Taeyoung Lee, Melvin Leoky, and N Harris McClamroch. Geometric tracking control of a quadrotor uav on se (3). In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5420–5425. IEEE, 2010.

[4] Nathan Michael, Daniel Mellinger, Quentin Lindsey, and Vijay Kumar. The grasp multiple micro-uav testbed. *IEEE Robot. Automat. Mag.*, 17(3):56–65, 2010.

16.485 Visual Navigation for Autonomous Vehicles (VNAV)
Fall 2020