Technische Universität München
Institute of Flight System Dynamics

Design, Implementation and Evaluation of an Incremental Nonlinear Dynamic Inversion Controller for a Nano-Quadrotor

–

Entwurf, Implementierung und Evaluierung eines Inkrementellen Nichtlinearen Dynamischen Inversionsreglers für einen Nano-Quadrotor

Semesterarbeit

Author: Evghenii Volodscoi

Matriculation number: 03663176

Supervisor: Dr. Ewoud Smeur

April 2020

Statutory Declaration

I, Evghenii Volodscoi, declare on oath towards the Institute of Flight System Dynamics of Technische Universitt Mnchen, that I have prepared the present Semester Thesis independently and with the aid of nothing but the resources listed in the bibliography.

This thesis has neither as-is nor similarly been submitted to any other university.

Garching,

Kurzfassung

Deutsche Kurzfassung der Arbeit.

Abstract

English abstract of the thesis.

Table of Contents

List of Figures

List of Tables

Table of Acronyms

Acronym	**Description**
EOM	Equations of Motion
INDI	Incremental Nonlinear Dynamic Inversion
NDI	Nonlinear Dynamic Inversion
NED	North East Down

Table of Symbols

Latin Letters

Symbol	Unit	Description
F	N	Force
g	m/s^2	Gravitational acceleration

Greek Letters

Symbol	Unit	Description
α	rad	Angle of attack
ζ	–	Damping of a linear second order system

Indices

Symbol	Unit	Description
m		Variable related to pitch moment
W		Wind

1 Introduction

1.1 Motivation

1.2 Contribution of the Thesis

1.3 Structure of the Thesis

2 Theoretical Background

The aim of this chapter is to provide the theoretical background which serves as a basis for some of the methods which are presented and applied in the course of this thesis. At the beginning, in section 2.1 general Equations of Motion (EOM) of an aircraft are presented. These are later used to derive the control algorithm and build the Matlab/Simulink model of the Crazyflie quadrotor. Sections 2.2 and 2.3 describe the Nonlinear Dynamic Inversion (NDI) and Incremental Nonlinear Dynamic Inversion (INDI) methods in general. Later in this chapter (subsections 2.3.2 and 2.3.3) these methods are used to derive the inner and the outer loop of the INDI flight controller for the Crazyflie quadrotor.

2.1 Dynamic Equations of Motion of an Aircraft

The dynamic equations which describe the general motion of an aircraft are usually coupled first order implicit nonlinear differential equations. However, in many cases it is sufficient to use their simplified versions (TODO:Source needed!). This section presents simplified general EOM and describes the assumptions which were considered to derive them.

For the derivation of the equations of motion the aircraft system will be assumed to be a rigid body. Such a rigid body system can be described uniquely by 12 states. Note that by taking in account additional effects, such as propulsion system dynamics, multi-body dynamics etc., the number of required state variables will increase.

Additionally to the rigid body assumption, it is also assumed that the earth is flat and non-rotating and the reference point of the sum of all external forces acting on the body corresponds to the center of gravity of the body. Thus, the linear momentum equation is written as

$$\sum(\boldsymbol{F}^G)_B = m \cdot \left[(\dot{\boldsymbol{v}}^G)_B + (\boldsymbol{\omega}^{OB}) \times (\boldsymbol{v}^R)_B\right] \tag{2.1}$$

where $\sum(\boldsymbol{F}^G)_B \in \mathbb{R}^{3\times1}$ is the sum of the external forces acting on the system and applied to the center of gravity G, m the mass of the body, $(\dot{\boldsymbol{v}}^G)_B \in \mathbb{R}^{3\times1}$ the linear acceleration of the point G, $(\boldsymbol{\omega}^{OB}) \in \mathbb{R}^{3\times1}$ the angular velocity of the body-fixed frame (B) with respect to the North East Down (NED) (O) coordinate frame. Subscript B denotes that all variables are specified in the body-fixed coordinate frame.

The rotational motion of a body is described with the angular momentum equation. To derive it, additionally to the assumptions made above it is also considered that the mass and the mass distribution are quasistationary, meaning $\frac{d}{dt}m = 0$ and $\frac{d}{dt}(\boldsymbol{I})_B = 0$ with $(\boldsymbol{I})_B$ being the inertia tensor of the body defined in the body-fixed frame. Thus, the angular momentum is

$$\sum(\boldsymbol{M}^G)_B = (\boldsymbol{I}^G)_B \cdot (\dot{\boldsymbol{\omega}}^{OB}) + (\boldsymbol{\omega}^{OB}) \times \left[(\boldsymbol{I}^G)_B \cdot (\boldsymbol{\omega}^{OB})\right] \tag{2.2}$$

where $\sum(\boldsymbol{M}^G)_B \in \mathbb{R}^{3\times1}$ is the sum of the external moments acting on the system around the center of gravity G.

The remaining two equations, that are necessary to fully describe the motion of a rigid body in space are the attitude differential equation and the position differential equation. The attitude differential equation describes the relationship between angular rates p, q, r and derivatives of the Euler angles $\dot{\Phi}, \dot{\Theta}, \dot{\Psi}$ leading to

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \frac{\sin\Phi}{\cos\Theta} & \frac{\cos\Phi}{\cos\Theta} \end{bmatrix}_B \begin{bmatrix} p \\ q \\ r \end{bmatrix}_B \tag{2.3}$$

where the angular rates are related to the derivatives of the Euler angles through the *strapdown matrix*.

There are different options to represent the position differential equation. Here, for completeness only, it is written as a simple relationship between the change of the position coordinates in the local NED frame and the velocity coordinates of the same frame

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}_O = \begin{bmatrix} \dot{V_N} \\ \dot{V_E} \\ \dot{V_D} \end{bmatrix}_O \tag{2.4}$$

The Equations (2.1)-(2.4) presented above can be used to represent a motion of a general aircraft in a three-dimensional space. Note that terms $\sum(\boldsymbol{F}^G)_B$ and $\sum(\boldsymbol{M}^G)_B$ contain all external forces and moments acting on the rigid body. Considering a general aircraft system, these could be the aerodynamic forces and moments caused by the air flow, propulsion forces and moments, forces caused by the gravitation etc.. The detailed modelling of external forces and moments is presented in chapter 3.2.

2.2 Nonlinear Dynamic Inversion

In this subsection, the Nonlinear Dynamic Inversion (NDI) method is explained. The NDI approach is based on feedback linearization and is also called *Input-Output Linearization*. Often, such type of controllers is involved in tracking control tasks, where objective is to track some desired trajectory [1]. To derive it, consider the following nonlinear system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u} \tag{2.5a}$$

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) \tag{2.5b}$$

where $\boldsymbol{x} \in \mathbb{R}^{n\times1}$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^{m\times1}$ the input vector, $\boldsymbol{y} \in \mathbb{R}^{m\times1}$ the output vector, $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^{n\times1}$ and $\boldsymbol{h}(\boldsymbol{x}) \in \mathbb{R}^{m\times1}$ nonlinear vector fields and $\boldsymbol{G} \in \mathbb{R}^{m\times n}$ an input matrix. Note that the system presented in Equations (2.5) is affine in the input, which is not allways fulfilled. Using a state transformation $\boldsymbol{z} = \boldsymbol{\phi}(\boldsymbol{x})$, the affine system from Equation 2.5a can be transformed into a *normal* (*canonical*) representation.

The core idea behind the input-output linearization method is to find a direct relationship between the desired system output and the control input. After the relationship is found it is inverted to generate the control law. To derive this relationship the output $\boldsymbol{y}$ is differentiated until the input $\boldsymbol{u}$ appears

$$\begin{aligned}\dot{\boldsymbol{y}} = \frac{\partial \boldsymbol{y}}{\partial t} = \frac{\partial \boldsymbol{h}(\boldsymbol{x})}{\partial t}\frac{\partial \boldsymbol{x}}{\partial t} = \frac{\partial \boldsymbol{h}(\boldsymbol{x})}{\partial t}\dot{\boldsymbol{x}} = \nabla \boldsymbol{h}(\boldsymbol{x})[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}] \\ = \nabla \boldsymbol{h}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}) + \nabla \boldsymbol{h}(\boldsymbol{x})\boldsymbol{G}(\boldsymbol{x})\boldsymbol{u} = L_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) + L_{\boldsymbol{G}}\boldsymbol{h}(\boldsymbol{x})\boldsymbol{u}\end{aligned} \tag{2.6}$$

In Equation (2.6) $L_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})$ is called Lie derivative of $\boldsymbol{h}(\boldsymbol{x})$ with respect to $\boldsymbol{f}(\boldsymbol{x})$. The Lie derivative is defined as $L_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = \nabla \boldsymbol{h}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x})$ with ∇ being the Nabla operator. Thus, it represents a directional derivative of $\boldsymbol{h}(\boldsymbol{x})$ along the direction of the vector field $\boldsymbol{f}(\boldsymbol{x})$. If the term $L_{\boldsymbol{G}}\boldsymbol{h}(\boldsymbol{x})$ is nonzero, the relationship between input and output is

$$\dot{\boldsymbol{y}} = L_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) + L_{\boldsymbol{G}}\boldsymbol{h}(\boldsymbol{x})\boldsymbol{u} \tag{2.7}$$

Now Equation (2.7) can be used to formulate the control law by solving it for $\boldsymbol{u}$ and substituting $\dot{\boldsymbol{y}}$ with $\boldsymbol{\nu}$

$$\boldsymbol{u} = L_{\boldsymbol{G}}\boldsymbol{h}(\boldsymbol{x})^{-1}(\boldsymbol{\nu} - L_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})) \tag{2.8}$$

The variable $\boldsymbol{\nu}$ is called an *equivalent input* and represents the desired output of the system.

In the example provided above the input-output relationsip was found after the first differentiation of the output $\boldsymbol{y}$. But if after the first differentiation the term $L_{\boldsymbol{G}}\boldsymbol{h}(\boldsymbol{x})$ is zero, output $\boldsymbol{y}$ has to be differentiated until the Lie derivative with respect to $\boldsymbol{G}$ is nonzero. The i-th derivative of the output is then

$$\frac{\partial^i \boldsymbol{y}}{\partial t} = \frac{\partial^i \boldsymbol{h}(\boldsymbol{x})}{\partial t} = L^i_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) + L_{\boldsymbol{G}}L^{i-1}_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})\boldsymbol{u} \tag{2.9}$$

with i being the *relative degree* of the system. Using Equation (2.9) to formulate the control law leads to the following expression for the control input $\boldsymbol{u}$

$$\boldsymbol{u} = L_{\boldsymbol{G}}L^{i-1}_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})^{-1}(\boldsymbol{\nu} - L^i_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})) \tag{2.10}$$

Thus, Equation (2.10) applied to Equation (2.9) yields the simple linear relation

$$\boldsymbol{y}^i = \boldsymbol{\nu} \tag{2.11}$$

The NDI method was widely adopted for civil and military aircrafts and has numerous of extensions [2]. Nevertheless, it has some drawbacks. The major one is that the control law derived using the NDI approach is dependent on the full system dynamics model [3]. The equations of motion of an aircraft usually have a complicated nonlinear character. Thus, describing complex physical phenomena often leads to the inconsistency between the real aircraft and its mathematical representation used in the model. As some of those inconsistencies are inevitable there is a need to build a control law which performance is less dependent on the uncertainties of the model. A method called Incremental Nonlinear Dynamic Inversion (INDI) can be used to achieve this, it is discussed in the next section.

2.3 Incremental Nonlinear Dynamic Inversion

The INDI is an incremental form of the NDI for which the lack of the accurate system dynamics model does not critically affect the performance of the control algorithm [4]. At first the general form of the INDI is presented in 2.3.1, then the inner and outer control loops for the Crazyflie quadrotor are derived in 2.3.2 and 2.3.3.

2.3.1 General INDI

The incremental form of the system can be obtained by taking a Taylor series expansion of the Equation 2.5a

$$\begin{aligned}\dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}_0) + \boldsymbol{G}(\boldsymbol{x}_0)\boldsymbol{u}_0 \\ &+ \frac{\partial}{\partial \boldsymbol{x}}[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}]\bigg|_{\boldsymbol{x}=\boldsymbol{x}_0,\boldsymbol{u}=\boldsymbol{u}_0}(\boldsymbol{x}-\boldsymbol{x}_0) \\ &+ \frac{\partial}{\partial \boldsymbol{u}}[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}]\bigg|_{\boldsymbol{x}=\boldsymbol{x}_0,\boldsymbol{u}=\boldsymbol{u}_0}(\boldsymbol{u}-\boldsymbol{u}_0)\end{aligned} \tag{2.12}$$

The first term on the right side of the Equation 2.12 is $\dot{\boldsymbol{x}}_{\boldsymbol{0}}$. Also, evaluating the differentiation of the third term leads to

$$\begin{aligned}\dot{\boldsymbol{x}} &= \dot{\boldsymbol{x}}_0 \\ &+ \frac{\partial}{\partial \boldsymbol{x}}[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\boldsymbol{u}]\bigg|_{\boldsymbol{x}=\boldsymbol{x}_0,\boldsymbol{u}=\boldsymbol{u}_0}(\boldsymbol{x}-\boldsymbol{x}_0) \\ &+ \boldsymbol{G}(\boldsymbol{x}_0)(\boldsymbol{u}-\boldsymbol{u}_0)\end{aligned} \tag{2.13}$$

The second term of the Equation 2.13 contains partial derivative with respect to the state vector. Considering very small time increments of the controller loop and applying the *principle of time scale separation* the second term vanishes. This is a valid assumption if the dynamics of the actuators is fast compared with the dynamics of the system [4]. Thus, the Equation 2.13 is further simplified to

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_0 + \boldsymbol{G}(\boldsymbol{x}_0)(\boldsymbol{u}-\boldsymbol{u}_0) \tag{2.14}$$

By solving Equation 2.14 for $\boldsymbol{u}$ and substituting $\dot{\boldsymbol{y}}$ with $\boldsymbol{\nu}$, the INDI control law is obtained

$$\boldsymbol{u} = \boldsymbol{u}_0 + \boldsymbol{G}(\boldsymbol{x}_0)^{-1}(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_0) \tag{2.15}$$

where $\dot{\boldsymbol{x}}_0$ is a measurable value from the previous step, $\boldsymbol{u}_{\boldsymbol{0}}$ the control input from the previous step, $\boldsymbol{\nu}$ the reference value and $\boldsymbol{G}(\boldsymbol{x}_0)$ the control effectiveness matrix. With $\boldsymbol{\Delta u} = \boldsymbol{u} - \boldsymbol{u}_0$ the control law from Equation 2.15 represents an incremental version of the Equation 2.8. Instead of computing the complete control input command $\boldsymbol{u}$, this control law results in computing the increment of the control input $\boldsymbol{\Delta u}$ and adding it to the previous value $\boldsymbol{u}_0$. As it is less dependent on the model of the system dynamics, it is able to increase the robustness of the system [3].

2.3.2 Inner INDI loop for Quadrotor

In this subsection, using theory from 2.3.1 the inner loop INDI controller is derived to control angular acceleration of the Crazyflie quadrotor. The derivation of the inner loop, as well as the outer loop controllers is based on the INDI controller architecture introduced by Smeur [5], [6]. Equation 2.2 from section 2.1 serves as a basis for this derivation. The desired variable to be controlled by the inner loop INDI is the angular acceleration of the quadrotor in the body-fixed coordinate frame. As in the case of a real quadrotor control problem the value of the thrust can be seen as an output of the dynamic system, it makes sense to incorporate thrust as a control variable into the control law as well. Thus, the angular momentum Equation 2.2 is augmented with the total thrust T of all four rotors [6]. As only the body-fixed frame is used in the following control law derivation, the subscripts B are not used in this subsection. Also all external forces and moments apply to the center of gravity of the quadrotor, thus the superscripts G are also omitted. Solving Equation 2.2 for the angular acceleration results in

$$\begin{bmatrix} \dot{\boldsymbol{\omega}} \\ T \end{bmatrix} = \underbrace{\begin{bmatrix} -\boldsymbol{I}^{-1}(\boldsymbol{\omega} \times \boldsymbol{I}\boldsymbol{\omega}) \\ 0 \end{bmatrix}}_{\boldsymbol{F}(\boldsymbol{\omega})} + \underbrace{\begin{bmatrix} \boldsymbol{I}^{-1}(\boldsymbol{M}_G + \boldsymbol{M}_A + \boldsymbol{M}_P) \\ T \end{bmatrix}}_{\boldsymbol{G}(\boldsymbol{\omega}, \boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}})} \tag{2.16}$$

where $\boldsymbol{\omega}$ and $\dot{\boldsymbol{\omega}}$ are angular velocity and acceleration of the body-fixed frame (B) with respect to the NED (O) coordinate frame, $\boldsymbol{M}_G$ the gravitational moment, $\boldsymbol{M}_A$ the aerodynamic moment and $\boldsymbol{M}_P$ the propulsion moment. The vector $\boldsymbol{\Omega}$ contains angular velocities of all four rotors and serves as an input variable of the system. It is assumed that the gravitational force is applied to the center of gravity of the quadrotor and does not cause any moment around it. Due to the absence of the aerodynamic moment this term is also omitted and can be seen as a disturbance [5]. The remaining propulsion moment is written as $\boldsymbol{M}_P = \boldsymbol{M}_C - \boldsymbol{M}_{gyro}$ where $\boldsymbol{M}_C$ is the control moment generated by the rotors and $\boldsymbol{M}_{gyro}$ the moment containing the gyroscopic effect of the rotors. This two moments can be explicitly written as

$$\boldsymbol{M}_C = \begin{bmatrix} -bk_F & bk_F & bk_F & -bk_F \\ lk_F & lk_F & -lk_F & -lk_F \\ k_M & -k_M & k_M & -k_M \end{bmatrix} \boldsymbol{\Omega} \tag{2.17}$$

$$\boldsymbol{M}_{gyro} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ I_{rzz} & -I_{rzz} & I_{rzz} & -I_{rzz} \end{bmatrix} \dot{\boldsymbol{\Omega}} + \begin{bmatrix} \boldsymbol{\omega}_y & 0 & 0 \\ 0 & \boldsymbol{\omega}_x & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_{rzz} & -I_{rzz} & I_{rzz} & -I_{rzz} \\ -I_{rzz} & I_{rzz} & -I_{rzz} & I_{rzz} \\ 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{\omega} \tag{2.18}$$

where I_{rzz} is the element of the inertia matrix I_r of the rotor, l and b lever arms as denoted in the Figure (TODO:Figure), k_F and k_M force and moment constants of the rotors.

As already explained in the previous section to derive the incremental control law the Taylor series expansion is perfromed on the Equation 2.16

$$\begin{aligned}\begin{bmatrix}\dot{\boldsymbol{\omega}}\\ T\end{bmatrix} &= \boldsymbol{F}(\boldsymbol{\omega}_0) + \boldsymbol{G}(\boldsymbol{\omega}_0, \boldsymbol{\Omega}_0, \dot{\boldsymbol{\Omega}}_0)\\ &+ \frac{\partial}{\partial \boldsymbol{\omega}}\big[\boldsymbol{F}(\boldsymbol{\omega}) + \boldsymbol{G}(\boldsymbol{\omega}_0, \boldsymbol{\Omega}_0, \dot{\boldsymbol{\Omega}}_0)\big]\Big|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0}(\boldsymbol{\omega} - \boldsymbol{\omega}_0)\\ &+ \frac{\partial}{\partial \boldsymbol{\Omega}}\big[\boldsymbol{G}(\boldsymbol{\omega}_0, \boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}_0)\big]\Big|_{\boldsymbol{\Omega}=\boldsymbol{\Omega}_0}(\boldsymbol{\Omega} - \boldsymbol{\Omega}_0)\\ &+ \frac{\partial}{\partial \dot{\boldsymbol{\Omega}}}\big[\boldsymbol{G}(\boldsymbol{\omega}_0, \boldsymbol{\Omega}_0, \dot{\boldsymbol{\Omega}})\big]\Big|_{\dot{\boldsymbol{\Omega}}=\dot{\boldsymbol{\Omega}}_0}(\dot{\boldsymbol{\Omega}} - \dot{\boldsymbol{\Omega}}_0)\end{aligned} \tag{2.19}$$

By applying differentiation and rewriting some of the terms, following equation is obtained

$$\begin{bmatrix}\dot{\boldsymbol{\omega}}\\ T\end{bmatrix} = \begin{bmatrix}\dot{\boldsymbol{\omega}}_0\\ T_0\end{bmatrix} + \boldsymbol{G}_1(\boldsymbol{\Omega} - \boldsymbol{\Omega}_0) + T_S \boldsymbol{G}_2(\dot{\boldsymbol{\Omega}} - \dot{\boldsymbol{\Omega}}_0) \tag{2.20}$$

The first term of the Equation 2.20 is the angular acceleration based on the current angular rates ω_0 and inputs $\boldsymbol{\Omega}_0$ and can be denoted as $\dot{\omega}_0$. T_0 is the current thrust value. The last term on the right side of the Equation 2.20 is scaled with the sample time T_s which is introduced only to simplify further mathematical transformations. The expressions of the control moment $\boldsymbol{M}_C$ and the gyroscopic moment of the rotors $\boldsymbol{M}_{gyro}$ have been summarized to control effectiveness matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$

$$\boldsymbol{G}_1 = \begin{bmatrix} \boldsymbol{I}^{-1}\begin{bmatrix}-bk_F & bk_F & bk_F & -bk_F\\ lk_F & lk_F & -lk_F & -lk_F\\ k_M & -k_M & k_M & -k_M\end{bmatrix} - \begin{bmatrix}\boldsymbol{\omega}_y & 0 & 0\\ 0 & \boldsymbol{\omega}_x & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}I_{rzz} & -I_{rzz} & I_{rzz} & -I_{rzz}\\ -I_{rzz} & I_{rzz} & -I_{rzz} & I_{rzz}\\ 0 & 0 & 0 & 0\end{bmatrix}\\ k_F \cdot \mathbf{1}_{1\times 4}\end{bmatrix} \tag{2.21}$$

$$\boldsymbol{G}_2 = \begin{bmatrix} -\boldsymbol{I}^{-1}T_S^{-1}\begin{bmatrix}0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ I_{rzz} & -I_{rzz} & I_{rzz} & -I_{rzz}\end{bmatrix}\\ \mathbf{0}_{1\times 4}\end{bmatrix} \tag{2.22}$$

with the terms $\mathbf{1}_{1\times 4}$ and $\mathbf{0}_{1\times 4}$ being 1×4 vectors of ones and zeros, respectively.

To prepare the linearized dynamic Equation 2.20 for discrete implementation on a computing system, the discrete approximation (z domain) of the derivative is used: $\dot{\boldsymbol{\Omega}} = (\boldsymbol{\Omega} - \boldsymbol{\Omega} z^{-1})T_S^{-1}$. Furthermore, angular acceleration $\dot{\omega}_0$ which is obtained from the differentiated gyroscope measurements is usually noisy. It has been shown, that using a second order filtering can help to reduce measurement nose. At the same time such a filter introduces time delay, which has to be considered in the derivation, as it is important to have a unique delay for all variables which are used in the Taylor expansion [5]. Thus, the same second order filter is applied to all variables with a subscript 0. By applying filtering (subscript f) and finite differences method to the Equation 2.20, its discrete version results in

$$\begin{bmatrix} \dot{\boldsymbol{\omega}} \\ T \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{\omega}}_f \\ T_f \end{bmatrix} + (\boldsymbol{G}_1 + \boldsymbol{G}_2)(\boldsymbol{\Omega} - \boldsymbol{\Omega}_f) - \boldsymbol{G}_2 z^{-1}(\boldsymbol{\Omega} - \boldsymbol{\Omega}_f) \tag{2.23}$$

By solving Equation 2.23 for $\dot{\boldsymbol{\Omega}}$ and substituting $\dot{\boldsymbol{\omega}}$ with the equivalent input $\boldsymbol{\nu}_{ang}$, the control law of the inner loop is obtained

$$\dot{\boldsymbol{\Omega}}_c = \dot{\boldsymbol{\Omega}} = \dot{\boldsymbol{\Omega}}_f + (\boldsymbol{G}_1 + \boldsymbol{G}_2)^{-1}\left(\begin{bmatrix} \boldsymbol{\nu}_{ang} - \dot{\boldsymbol{\omega}}_f \\ \tilde{T} \end{bmatrix} + \boldsymbol{G}_2 z^{-1}(\boldsymbol{\Omega} - \boldsymbol{\Omega}_f) \right) \tag{2.24}$$

where $\dot{\boldsymbol{\Omega}}$ is a vector of commanded rotational rates for every rotor and $\tilde{T} = T - T_f$ being the thrust increment which is provided by the outer loop INDI. The block diagram of the inner loop INDI controller is presented in Figure (TODO:Figure). In the figure the derived controller which controls the angular acceleration is augmented with two simple controllers which control the angular velocity $\boldsymbol{\omega}$ and the attitude $\boldsymbol{\eta}$ of the quadrotor. Each controller consists of a single gain. Thus, the reference values of the INDI part of the inner loop are provided by these two controllers.

2.3.3 Outer INDI loop for Quadrotor

In this subsection the derivation of the outer loop INDI controller is introduced. The outer loop INDI controls translational acceleration of the quadrotor. Linear momentum Equation 2.1 serves as a basis for this derivation. For the simplicity of the further transformations the derivation is performed in the NED frame (subscript O). Thus, the Equation 2.1 becomes

$$\sum (\boldsymbol{F}^G)_O = m \cdot (\dot{\boldsymbol{v}}^G)_O \tag{2.25}$$

As it is assumed that all forces apply to the center of gravity of the quadrotor, the superscript G is omitted in the future. Thus, solving Equation 2.25 for $\dot{\boldsymbol{v}}$ and assuming that the sum of all forces acting on the quadrotor consists only of gravitational, propulsive and aerodynamic forces, following equation is obtained

$$\dot{\boldsymbol{v}} = m^{-1}((\boldsymbol{F}_G)_O + (\boldsymbol{F}_P)_O + (\boldsymbol{F}_A)_O) \tag{2.26}$$

The aerodynamic force $(\boldsymbol{F}_A)_O$ is modelled as an unknown function of velocity $\boldsymbol{v}$ and wind vector $\boldsymbol{\chi}$. For the gravitational force $(\boldsymbol{F}_G)_O$ the simplest one-dimensional gravity model is assumed.

$$(\boldsymbol{F}_A)_O = \boldsymbol{f}(\boldsymbol{v}, \boldsymbol{\chi}) \tag{2.27}$$

$$(\boldsymbol{F}_G)_O = m \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}_O \tag{2.28}$$

The propulsive force $(\boldsymbol{F}_P)_O$ (produced by the rotors) has only one nonzero component if defined in the body-fixed frame. To transform it to the NED frame the rotation matrix $\boldsymbol{M}_{OB}$ is used. This results in

$$(\boldsymbol{F}_P)_O = \boldsymbol{M}_{OB}\begin{bmatrix}0\\0\\-T\end{bmatrix}_B = \begin{bmatrix}* & * & c\Psi s\Theta c\Phi + s\Psi s\Phi\\ * & * & s\Psi s\Theta c\Phi + c\Psi s\Phi\\ * & * & c\Theta c\Phi\end{bmatrix}\begin{bmatrix}0\\0\\-T\end{bmatrix}_B = \begin{bmatrix}(c\Psi s\Theta c\Phi + s\Psi s\Phi)(-T)\\(s\Psi s\Theta c\Phi + c\Psi s\Phi)(-T)\\(c\Theta c\Phi)(-T)\end{bmatrix}_O \tag{2.29}$$

To obtain the incremental form of the linear momentum equation, the same strategy as in the case of the inner loop derivation is applied: the Equation 2.26 is linearized using Taylor series expansion

$$\begin{aligned}\dot{\boldsymbol{v}} &= m^{-1}[\boldsymbol{F}_G + \boldsymbol{F}_P(\Phi_0,\Theta_0,\Psi_0,T_0) + \boldsymbol{F}_A(\boldsymbol{v}_0,\boldsymbol{\chi}_0)]\\
&+ \frac{\partial}{\partial \boldsymbol{v}}\big[\boldsymbol{F}_A(\boldsymbol{v},\boldsymbol{\chi}_0)\big]\Big|_{\boldsymbol{v}=\boldsymbol{v}_0}(\boldsymbol{v}-\boldsymbol{v}_0)\\
&+ \frac{\partial}{\partial \boldsymbol{\chi}}\big[\boldsymbol{F}_A(\boldsymbol{v}_0,\boldsymbol{\chi})\big]\Big|_{\boldsymbol{\chi}=\boldsymbol{\chi}_0}(\boldsymbol{\chi}-\boldsymbol{\chi}_0)\\
&+ \frac{\partial}{\partial \Phi}\big[\boldsymbol{F}_P(\Phi,\Theta_0,\Psi_0,T_0)\big]\Big|_{\Phi=\Phi_0}(\Phi-\Phi_0)\\
&+ \frac{\partial}{\partial \Theta}\big[\boldsymbol{F}_P(\Phi_0,\Theta,\Psi_0,T_0)\big]\Big|_{\Theta=\Theta_0}(\Theta-\Theta_0)\\
&+ \frac{\partial}{\partial \Psi}\big[\boldsymbol{F}_P(\Phi_0,\Theta_0,\Psi,T_0)\big]\Big|_{\Psi=\Psi_0}(\Psi-\Psi_0)\\
&+ \frac{\partial}{\partial T}\big[\boldsymbol{F}_P(\Phi_0,\Theta_0,\Psi_0,T)\big]\Big|_{T=T_0}(T-T_0)\end{aligned} \tag{2.30}$$

The first term on the right side of the Equation 2.30 is the acceleration at previous time step $\dot{\boldsymbol{v}}_0$. It can be obtained from the onboard accelerometer and then tranformed to the NED frame. The next two terms on the right are assumed to be zero since it is very complicated to develop an accurate model of the aerodynamic force as the nature of the wind influence can vary and remains unknown. It is also assumed that the change of the yaw angle Ψ is small and can be neglected as well. By applying considered assumptions Equation 2.30 is written as

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{v}}_0 + m^{-1}\boldsymbol{G}(\Phi_0,\Theta_0,\Psi_0,T_0)(\boldsymbol{u}-\boldsymbol{u}_0) \tag{2.31}$$

where

$$\boldsymbol{G}(\Phi_0,\Theta_0,\Psi_0,T_0) = \underbrace{\begin{bmatrix}(-c\Psi_0 s\Theta_0 s\Phi_0 + s\Psi_0 c\Phi_0)T_0 & c\Psi_0 c\Theta_0 c\Phi_0 T_0 & c\Psi_0 s\Theta_0 c\Phi_0 + s\Psi_0 s\Phi_0\\(-s\Psi_0 s\Theta_0 s\Phi_0 - c\Psi_0 c\Phi_0)T_0 & s\Psi_0 c\Theta_0 c\Phi_0 T_0 & s\Psi_0 s\Theta_0 c\Phi_0 - c\Psi_0 s\Phi_0\\ -c\Theta_0 s\Phi_0 T_0 & -s\Theta_0 c\Phi_0 T_0 & c\Theta_0 c\Phi_0\end{bmatrix}}_{\boldsymbol{G}(\Phi_0,\Theta_0,\Psi_0,T_0)} \tag{2.32}$$

and

$$\boldsymbol{u} = \begin{bmatrix} \Phi \\ \Theta \\ T \end{bmatrix} \tag{2.33}$$

$\boldsymbol{G}(\Phi_0, \Theta_0, \Psi_0, T_0)$ is the control effectiveness matrix which is computed based on the attitude and thrust values from the previous step. $\boldsymbol{u}$ is a vector of control variables which are passed to the inner loop.

The linear acceleration $\dot{\boldsymbol{v}}_0$ which is obtained from accelerometer measurements is usually noisy. Applying the same filter as in the case of the inner loop derivation, the Equation 2.31 is given by

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{v}}_f + m^{-1}\boldsymbol{G}(\Phi_0, \Theta_0, \Psi_0, T_0)(\boldsymbol{u} - \boldsymbol{u}_f) \tag{2.34}$$

Solving Equation 2.34 for $\boldsymbol{u}$ and substituting $\dot{\boldsymbol{v}}$ with the equivalent control variable $\boldsymbol{\nu}_{lin}$, the control law of the outer loop is obtained

$$\boldsymbol{u_c} = \boldsymbol{u} = \boldsymbol{u}_f + m\boldsymbol{G}^{-1}(\Phi_0, \Theta_0, \Psi_0, T_0)(\boldsymbol{\nu}_{lin} - \dot{\boldsymbol{v}}_f) \tag{2.35}$$

Figure (TODO:show Figure) shows the block diagram of the outer loop INDI. Using the same approach as with the inner loop, the outer loop which controls linear acceleration is augmented with two closed loops consisting of two gains. These gains represent two controllers which control linear velocity and position.

TODOs:
- 2.1 Source (Probably Holzapfel script)
- 2.3.2 Figure with Crazyflie lever arms and motor numbers
- 2.3.2 Figure with block diagram of the inner loop controller - 2.3.3 Figure with block diagram of the outer loop controller

3 Implementation

3.1 Research Quadrotor

- Some facts about Crazyflie hardware (foto, uC frequency, weight, length (foto of lever arms), onboard sensors, additional sensors/HW (optoflow, sd-card))

3.2 Implementation in Simulink (Simulink Model)

3.2.1 Purpose

- Estimation of the relevant for the HW implementation components (which terms make a huge impact on the controller performance: G1, G1 etc)
- Estimate where value explosion can occur
- Testing the behaviour of the filter
- Testing of the PD-gains

3.2.2 Structure

- Describe some assumptions: eg newton second law frame transformation from B to O for an easy representation
- Present each equation of forces with explaining made assumptions
- Parameter choice (k_F, k_M, ...)
- Describe the actuator dynamics modelling
- Describe the filter transfer function
- Images of the Simulink model

3.2.3 Simulation Results

- Plot differences with or without some terms (G1, G2_yaw)

3.3 Implementation on the Hardware

3.3.1 Parameter Estimation

- Actuator Dynamics (Estimation of the time constant), plot response with estimated constant
- Thrust dynamics estimation
- Estimation of the control effectiveness parameters G1, G2 for inner INDI (describe performed flight to log data), plot the curve with contribution of G1, G2 to the fitting
- PD gain tuning for inner INDI (using pd_inner_cs(), show plots with different D-gains (e.g 25, 10 and 3) to see different damping behaviour), Before explaining gain tuning present all relevant transfer functions of the closed and open loops
- Outer loop (show new diagram of the controller)

3.3.2 Structure of the Code

3.3.3 Testing with contact Forces and Moments?

4 Results

- To make the controller work a minimal knowledge of the system dynymics is need. Nevertherless G1, G2 still have to estimated accuratly because this 2 parameters do have inpact on the controller performance.
- Plain step responses of the outer loop
- Step responses with disturbance of the outer loop

5 Discussion

References

[1] J.J.E. Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice-Hall, Inc, Upper Saddle River, New Jersey 07458, 1991.

[2] J.F. Horn. Non-linear dynamic inversion control design for rotorcraft. *https://www.mdpi.com/journal/aerospace*, 2019.

[3] S. Sieberling, J.A. Mulder, and Q.P. Chu. Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction. *Journal of Guidance Control and Dynamics, 33.2010 no.6*, 2010.

[4] Eduardo Simões Silva. Incremental nonlinear dynamic inversion for quadrotor control. 2015.

[5] E.J.J Smeur, Q.P. Chu, and G.C.H.E. de Croon. Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles. *Journal of Guidance Control and Dynamics, Vol. 39 no.3*, 2015.

[6] E.J.J Smeur, Q.P. Chu, and G.C.H.E. de Croon. Cascaded incremental nonlinear dynamic inversion control for mav disturbance rejection. *Control Engineering Practice, Vol. 73, p. 79-90*, 2018.

Appendix