Faculty of Aerospace Engineering

Nonlinear Fault Tolerant Flight Control of a Quadrotor Vehicle subjected to Double Rotor Failures

L.M.C. Sijbers

28 June 2018

TU Delft Delft University of Technology

Challenge the future

Nonlinear Fault Tolerant Flight Control of a Quadrotor Vehicle subjected to Double Rotor Failures

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

L.M.C. Sijbers

28 June 2018

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Copyright © L.M.C. Sijbers
All rights reserved.

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Nonlinear Fault Tolerant Flight Control of a Quadrotor Vehicle subjected to Double Rotor Failures "** by **L.M.C. Sijbers** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 28 June 2018

Readers:

Prof.dr.ir. Q. P. Chu

dr.ir. C. C. de Visser

dr.ir. E. Mooij

PhD Student Sihao Sun

Contents

Chapter 1

General Introduction

This thesis is aimed at producing a fault tolerant quadrotor that is capable of trajectory control. The reason for this is two-fold: Firstly, having a controller capable of flying a damaged quadrotor helps improve the overall fault tolerant control and make quadrotors more resilient to damage. Secondly, ongoing research requires flight data from damaged quadrotors in order to estimate flight envelope shrinkage. Current controllers are limited in damage cases or in their capability to reject the nonlinear effects that are inherent with flying quadrotors. In this research, a nonlinear controller is designed to achieve flight with two complete rotor failures.

Contrary with single rotor failures, the double rotor failures have two configurations. The case where adjacent rotors have failed or the case for diagonally opposite rotors. Both will be discussed in this research. The adjacent case is limited to simulations whereas the diagonal case is flown inside wind tunnel.

A controller is synthesized to cope with diagonal rotor damage. An Incremental Nonlinear Dynamic Inversion (INDI) method is used to translate the required rotational rates of the body to required actuator inputs. The outer loop is designed to control the primary axis about which the quadrotor may freely rotate subsequent to rotor failure.

The controller designed in this research does not implement an online fault detection & diagnosis module (FDD) which is required for rejecting in-flight occurring damages. Rather, the damage is applied pre-flight and flight software is switched to the relevant control strategy. This way flight data can still be gathered to support ongoing research. This limitation is introduced due to the complexity of FDD design which can be a thesis topic on its own.

Throughout this report, references will be made to the vehicle that is being used: The Bebop 2 quadrotor. The rotational direction of the rotors, the rotor numbers and axis definitions are presented in fig. 1-1.

This report consists of three parts: A scientific paper, supporting appendices and the preliminary report. These contribute to answering the research questions set at the start of this thesis project. The preliminary report is added to provide background and literature review regarding the subjects in this thesis and the synthesis of the research questions.

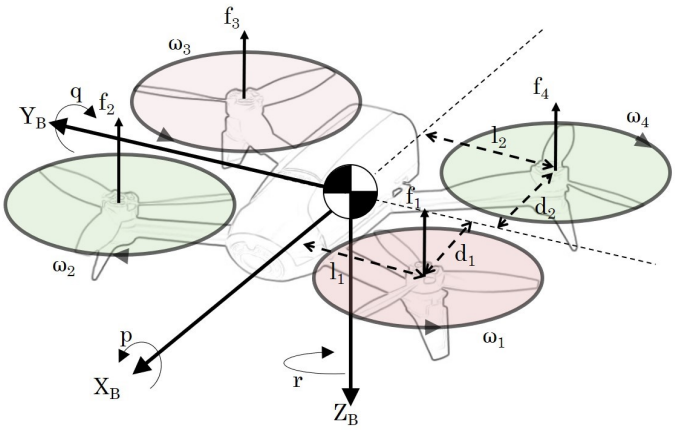

Figure 1-1: The Bebop 2 quadrotor and conventions.

1-1 Research Question

The following research question must be answered:

RG 1 How can we expand the current INDI + primary axis method to allow a 2-rotor failed quadrotor to minimize the deviation from its intended velocity vector?

The main research question is motivated by the idea that the quadrotor will not be capable of controlling all rotation rates when 2 rotors are damaged, but will still be able to control its position and velocity.

Several sub-questions are formulated in order to answer the main research question. The first and second sub-question focus on potential pitfalls and obstacles in designing a controller whilst the third sub-question focuses on the achievable performance.

RQ 1.1 What are the bottlenecks in expanding the current INDI controller for the two opposite rotor fail case?

RQ 1.2 What are the bottlenecks in expanding the current INDI controller for the two adjacent rotor fail cases?

RQ 1.3 What kind of tracking performance can the 2-rotor damaged quadrotor achieve for hover and forward flight?

The paper is mainly concerned with answering research questions RQ 1.1 and RQ 1.3. Appendix A is aimed at answering the remaining sub-questions RQ 1.2. Finally, appendix B plays a supporting role for the main research question.

Part I

Paper

Nonlinear Fault Tolerant Flight Control of a Quadrotor Vehicle subjected to Complete Loss of Two Rotors

Leon M.C. Sijbers*, Sihao Sun†and Coen C. de Visser‡

Delft University of Technology, 2629 HS Delft, The Netherlands

Conventional controllers are incapable of controlling a rotor failed quadrotor in high speed flight. In this paper, a fault tolerant controller based on Incremental Nonlinear Dynamic Inversion is proposed to control a quadrotor subjected to the loss of two rotors. A pseudoinverse is used to control the roll and pitch rate to steer the average thrust direction. The proposed controller is implemented on an asymmetric quadrotor and is flown in high wind speeds. Performance comparisons were made between the loss of one or two rotors. The single rotor failure case performed only slightly better than the double rotor failure case in terms of achieved wind speed. The fastest achieved forward flight with 2 rotors was 8 m/s compared to 9.5 m/s with 3 rotors.

Nomenclature

n	Primary Axis, $-$
A	Surface area, m^2
C	(Friction) coefficient, $-$
d	Position in x, y, z, m
$d_{1,2}$	Distance from x axis, m
F	Friction forces, N
f_c	Cutoff frequency, Hz
f_i	Rotor force for rotor i, N
g	Gravitational constant, $9.81 m/s^2$
H	Transfer function
I	Moment of Inertia tensor, kgm^2
K	Gain
$l_{1,2}$	Distance from y axis, m
M	Moment, Nm
m	mass, kg
p	Roll rate, rad/s
q	Pitch rate, rad/s
R	Rotation matrix
r	yaw rate, rad/s
T	Thrust, N
u	Body velocity in x_b direction, m/s
v	Body velocity in y_b direction, m/s
w	Body velocity in z_b direction, m/s

Subscripts

$\circ$	Placeholder for direction
A	Aerodynamic related

*MSc student, Control and Simulation Division, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands. AIAA member.

†PhD student, Control and Simulation Division, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands. AIAA member.

‡Assistant Professor, Control and Simulation Division, Faculty of Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands. AIAA member.

a	Atittude
B	Body Reference Frame
C	Control related
I	Inertial Reference Frame
i	Number
ref	Reference value
t	Current time instance
$t-1$	Last time instance

Symbols

Δ	Infinitesimal increment
δ	Change in value
η	Body angles, *rad*
ϕ	Roll angle, *rad*
ψ	Yaw angle, *rad*
ρ	Air density, kg/m^3
τ	Time constant, s
θ	Pitch angle, *rad*

Superscripts

$\ddot{\square}$	Second order time derivative
$\dot{\square}$	First order time derivative
T	Transpose operator

I. INTRODUCTION

Small Unmanned Aerial Vehicles (UAV), characterized by their low weight and size, find themselves a increasing presence in our everyday life. UAV are used for various tasks such as surveillance, research platforms, mobile flying cameras and goods delivery. A special type of UAV is the quadrotor system. A quadrotor consist of a frame with four rotors mounted on the corners. The rotors are driven by brushless DC motors and are not capable of reversing direction; only positive thrust can be generated. The key selling points of a quadrotors are their low complexity, lightweight and versatility in take-off and landing. Quadrotors are inherently unstable, meaning they require controllers to provide closed loop stability. Furthermore, they are under-actuated systems; damage to either of the four rotors quickly results in unrecoverable dynamics. This makes it an interesting subject from a control theory perspective as well as a safety perspective.

Flights conducted with damaged quadrotors are used to identify models and establish the flight envelope. Fault Tolerant Controllers are applied to enable, like the name suggests, flight where a conventional controller would not. Various damage cases are explored to provide an insight in flight envelope shrinkage and to discover new modes.

Damage regarding partial loss of rotor effectiveness has been thoroughly researched. A variety of control methods have been used such Gain Scheduled PID,[1] Linear Parameter Varying,[2] PID backstepping[3] and sliding mode control.[4–6] Full rotor failed quadrotors were first stabilized by Lanzon, Freddi & Longhi where the strategy was to eliminate yaw control and control the axis around which the quadrotor spins.[7] Double rotor failures have been investigated using the same technique, which is now called primary axis control.[8,9] These methods were limited to a linear controller; an offline calculation of the solution is needed which is not ideal for flight testing.

The first use of a nonlinear controller to control a rotor failure was by Lu & van Kampen.[10] An Incremental Nonlinear Dynamic Inversion (INDI) was used to control a quadrotor with 3 rotors in a simulation. The first application of the INDI controller to a full rotor failed quadrotor was by Sun, Sijbers & de Visser,[11] which is used as a basis in exploring the double rotor failure cases.

The main contribution of this paper is the design and application of an INDI controller to control a quadrotor subjected to double rotor failures in high speed flight. The INDI controller is improved with a pseudoinversion and is used in conjunction with the primary axis control. The quadrotor is flown inside a wind tunnel to compare single and double rotor failure cases and show the effectiveness of the proposed controller.

Actuator dynamics and the effect of rotor damage has not yet been identified in previous research and can be used to develop controllers. As such, a small contribution in this paper is the estimation of the actuator dynamics.

The remainder of this paper is structured as follows: Section II presents the modeling of an asymmetric quadrotor. Section III presents the control strategy featuring the proposed INDI controller and the Primary axis control loop. The

setup and results of the flights are presented in Section IV. Finally, the conclusion is given in Section V.

II. Modeling of an asymmetric quadrotor

This section presents the definition of the reference frames, the translational and rotational equations of motions.

II.A. Reference frames

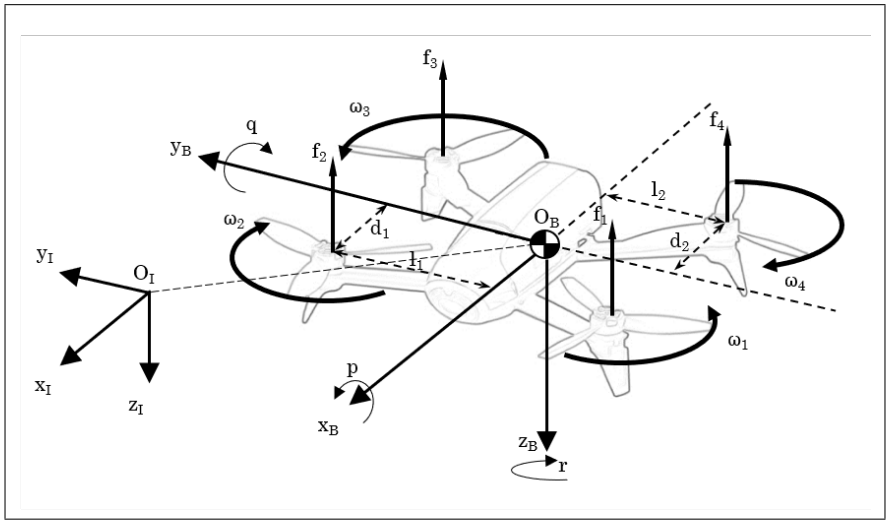

Figure 1. Definition of the body-fixed (B) and intertial (I) reference frames and rotor forces

Define a right-handed orthogonal axis-system as the body-fixed reference frame with the origin O_B in the center of gravity. Both positive X_B and Z_B are in the symmetry plane pointing forward and down, respectively. The positive Y_B points to the right perpendicular to the symmetry plane.

The reference frames are shown in figure 1. The quadrotor moves and rotates in the Inertial reference frame (O_I,x_I,y_I,z_I). Its origin O_I is fixed with respect to inertial space with Z_I pointing down. A relative translation offset between the origins O_I and O_B can be defined as $\mathbf{d} = [x,y,z]^T$. The relative rotations around respectively x_I, y_I, z_I are defined as $\eta = [\phi,\theta,\psi]^T$; the well-known roll, pitch yaw notation.[12]

The rotor force vectors f_i are also shown in figure 1. Mind the direction of the rotor velocity ω_i. Although the quadrotor is asymmetric the hubs are defined symmetrically for simplicity; the distance from the rotor forces to the center of gravity are denoted by d_1, d_2, l_1 and l_2.

The mapping of the inertia frame to the body frame is done through the rotation sequence yaw, pitch, roll which yields rotation matrix $R_{I\to B}$:

$$R_{I\to B} = \begin{bmatrix} C\theta C\psi & C\theta S\psi & -S\theta \\ S\phi S\theta C\psi - C\phi S\psi & S\phi S\theta S\psi + C\phi C\psi & S\phi C\theta \\ C\phi S\theta C\psi + S\phi S\psi & C\phi S\theta S\psi - S\phi C\psi & C\phi C\theta \end{bmatrix} \tag{1}$$

where C and S are shorthand forms for cosine and sine respectively. Note that $R_{B\to I} = R_{I\to B}^T$.

II.B. Equations of motion

The vehicle is modeled as a rigid body with a mass m and four rotors . Each i^{th} rotor can be modeled as a flat disk producing a force f in the $-Z_b$ direction originating from the hub. This gives a simplified total thrust vector $T = [0,0,\sum_{i=1}^{4} -f_i]^T$ acting in the center of gravity. The translational equations of motion in the inertial frame is given by eq. (2). The acceleration vector is given by the second time derivative of $\ddot{d}$ in the inertial frame.

$$m\ddot{d} = RT + mg - RF \tag{2}$$

whereby $g = [0,0,9.81] ms^{-2}$, R the rotation matrix, and F is a diagonal matrix with friction forces for each body velocity component shown in eq. (3).

$$F = \begin{bmatrix} \frac{1}{2} C_x A_x \rho u |u| & 0 & 0 \\ 0 & \frac{1}{2} C_y A_y \rho v |v| & 0 \\ 0 & 0 & \frac{1}{2} C_z A_z \rho w |w| \end{bmatrix} \tag{3}$$

with ρ the air density, $C_\circ$ the friction coefficient, $A_\circ$ the surface area, and u, v and w the body velocities in the respective x_b, y_b and z_b directions. The rotational equations of motion are given by eq. (4). The angular acceleration is given as second time derivative of $\dot{\eta}$. The torques acting on the body are captured in M.

$$I\ddot{\eta} = M - (I \times \dot{\eta})\,\dot{\eta} \tag{4}$$

The moment of inertia I in eq. (4) is given by eq. (5). Specific values can be found in table 1. The references frames, equations of motion and parameters described in this part are used to design a control strategy.

$$I = \begin{bmatrix} I_{xx} & 0 & -I_{zx} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix} \tag{5}$$

Table 1. Quadrotor parameters

$l_1 = l_2$, m	$d_1 = d_2$, m	m, kg	-
0.088	0.115	0.410	-
I_{xx} [kgm^2]	I_{yy} [kgm^2]	I_{zz} [kgm^2]	I_{xz} [kgm^2]
$1.67 \cdot 10^{-3}$	$1.38 \cdot 10^{-3}$	$2.82 \cdot 10^{-3}$	$1.21 \cdot 10^{-6}$

III. Control Strategy

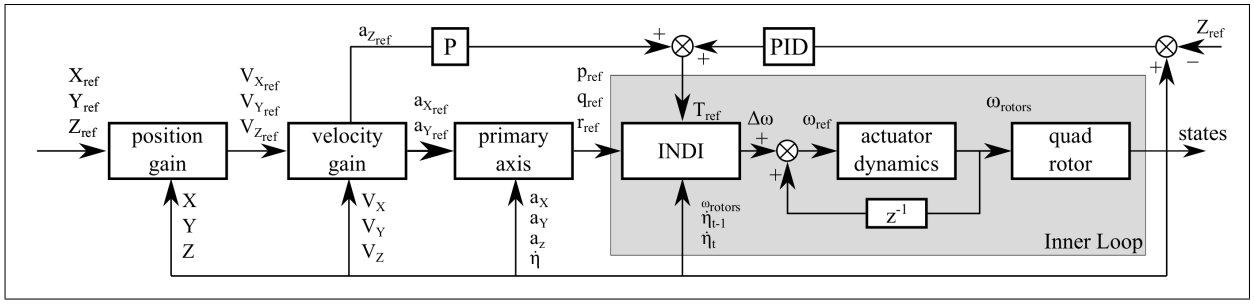

Figure 2. Control diagram for the proposed INDI controller

This section presents the full control structure of the proposed flight controller. First, the primary axis control is explained. Secondly, the derivation of the INDI controller is presented. After that, the actuator dynamics are discussed. Lastly, the control of a rotor failure is explained. The full control diagram is presented in figure 2.

III.A. Primary Axis outer loop

The outer loop translates a desired position in inertial space into a desired body rate for the inner loop. A quadrotor undergoing a full rotor failure is unable to control all rotation rates.[13] A new solution is required in that case.

A revolutionary control law was investigated to control a single full rotor failure by Lanzon, Freddi & Longhi.[7] This strategy — primary axis control — was subsequently employed to control double- and triple- rotor failures by Mueller & D'andrea.[8] The crux of primary axis control is to relinquish yaw rate control and thereby control the direction of the normalized average thrust vector **n**. The required accelerations are transformed into a normalized desired x and x component in the inertial frame. It then provides the required body rates to the inner loop in order to achieve tracking of the $\mathbf{n_{ref}}$ vector.

To start, the translational accelerations can be rewritten into:

$$m\ddot{d} = mg\mathbf{e_3} - R\mathbf{e_3}T \tag{6}$$

where $\mathbf{e_3} = [0,0,1]^T$.

Let $\mathbf{n} = [n_x, n_y, n_z]^T$ then

$$\dot{\mathbf{n}} = -\eta \times \mathbf{n} \tag{7}$$

The desired **n** in the body frame ,n_{ref_B}, can be calculated using the reference acceleration $\ddot{d}_{ref} = [a_{x_{ref}}, a_{y_{ref}}, a_{z_{ref}}]^T$ and the rotational matrix R.

$$\mathbf{n_{ref_B}} = mR^{-1}\frac{\left(\ddot{d}_{ref} - ge_3\right)}{n_z T_c} \tag{8}$$

With the required thrust calculated as:

$$T_c = -m\frac{(a_{z_{ref}} - g)}{c\phi c\theta} \tag{9}$$

The reference acceleration in the inertial frame is calculated by a PI-feedback on position and velocity. All feed-forward ff, proportional P, integrator I and derivative D gains are presented in table 2.

The control of the primary axis is then given by

$$\begin{bmatrix} \dot{n}_x \\ \dot{n}_y \\ \dot{n}_z \end{bmatrix} = \begin{bmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{10}$$

The yaw rate r can be eliminated, as yaw rate control is not required, in eq. (10) and rewriting gives:[10]

$$\begin{bmatrix} p_{ref} \\ q_{ref} \end{bmatrix} = \begin{bmatrix} 0 & -n_z \\ n_z & 0 \end{bmatrix}^{-1} \begin{bmatrix} K_a & 0 \\ 0 & K_a \end{bmatrix} \begin{bmatrix} \dot{n}_x \\ \dot{n}_y \end{bmatrix} + \begin{bmatrix} n_x \\ -n_y \end{bmatrix} r \tag{11}$$

where an additional attitude gain K_a is added.

III.B. INDI inner loop

The INDI is a sensor based controller whereby a incremental control input is calculated to achieve a required difference in acceleration. The INDI method was first published by Ostroff & Bacon.[14]

The rotational equations of motion (eq. (4)) can be rewritten in terms of a infinitesimal time-step to produce eq. (12). The total moment has been separated into an aerodynamic moment M_A and control moment $M_C u$ with M_C the control effectiveness matrix. Here, the change in aerodynamic moment ΔM_A and body rates $\Delta\dot{\eta}$ are negligible and can be omitted. The change in angular acceleration $\Delta\ddot{\eta}$ and control moment $\Delta M_C u$ can be rewritten to eq. (13) to find the increment control input Δu.

$$I\Delta\ddot{\eta} + \Delta\dot{\eta} \times I\dot{\eta} = \Delta M_A + \Delta M_C u \tag{12}$$

$$\Delta u = M_C^{-1} I(\ddot{\eta}_{ref} - \ddot{\eta}_t) \tag{13}$$

When the sampling rate of the IMU is high (512 Hz in this case), the assumption that the angular acceleration of the past time step is equal to the current time step (i.e. $\ddot{\eta}_{t-1} \approx \ddot{\eta}_t$) is valid. $\ddot{\eta}_{t-1}$ can then be calculated by eq. (14) using the finite difference from the measured, and filtered, body rates $\dot{\eta} = [p,q,r]^T$. Filtering is done using a second order Butterworth filter with cutoff frequency of 10 [Hz] similar to the literature found.[15]

$$\ddot{\eta}_{t-1} = \frac{\dot{\eta}_t - \dot{\eta}_{t-1}}{\Delta t} \tag{14}$$

The required angular accelerations are obtained using the output of the outer loop $\dot{\eta}_{ref}$ and the current body rates $\dot{\eta}$ in eq. (15).

$$\ddot{\eta}_{ref} = (\dot{\eta}_{ref} - \dot{\eta})K + \frac{\delta \dot{\eta}_{ref}}{\delta t} \tag{15}$$

The reference thrust is calculated as a separate loop according to:

$$T_{ref} = (g - a_{z_{ref}})K_{ff} + e_z K_P + \int_0^t e_z \delta t K_I + \frac{\delta e_z}{\delta t} K_D \tag{16}$$

Whereby the feed-forward is used to bring the initial reference closer towards the thrust required in equilibrium. This helps with take-off scenario, although it is not necessarily required as a PID feedback is also implemented.

Application of the values found in eq. (14), eq. (14) and eq. (15) into eq. (13), and defining the control effectiveness matrix for the quadrotor gives eq. (17). The distances for the moment arms $d_{1,2}$ and $l_{1,2}$ defined in figure 1. The reaction torque from the rotors acting on the body are modeled as a coefficient C_k. An explicit solution exists as matrix M_C is full-rank.

Table 2. Thrust feedback and forward gains

	Thrust	Position	Velocity	Attitude
K_{ff}	0.65	0	0	0
K_P	300	1	2	10
K_I	85	1	0	0
K_D	20	0	0	0

$$\begin{bmatrix} \Delta f_1 \\ \Delta f_2 \\ \Delta f_3 \\ \Delta f_4 \end{bmatrix} = \underbrace{\begin{bmatrix} d_1 & -d_1 & -d_2 & d_2 \\ l_1 & l_1 & -l_2 & -l_2 \\ C_k & -C_k & C_k & -C_k \\ 1 & 1 & 1 & 1 \end{bmatrix}^{-1}}_{M_C} \begin{bmatrix} \mathbf{I} \\ m \end{bmatrix} \begin{bmatrix} \dot{p}_{ref} - \dot{p} \\ \dot{q}_{ref} - \dot{q} \\ \dot{r}_{ref} - \dot{r} \\ T_{ref} - (\sum_{i=1}^{4} f_i) \end{bmatrix} \tag{17}$$

To transform the forces to rotor speeds, the assumption used is that the thrust varies linearly with the square of the rotor speeds ω according to thrust coefficient C_T.[16] The incremental rotor speed can then be calculated using eq. (18) and is added to the rotor speed from last time step.

$$\Delta\omega = \sqrt{\frac{\Delta f}{C_T}} \tag{18}$$

III.C. Actuator dynamics

An important part of the overall stability of the system are the actuator dynamics. The actuators should be able to follow a command of the same frequency as the yaw rate r without too much delay.

An input-output analysis is done to determine a simple first-order model for the actuators of the form given by eq. (19). The time constant $\tau = \frac{1}{2\pi f_c}$ can be found at the frequency corresponding to a -3 [dB] magnitude in figure 3. The cutoff frequency f_c appears to be 8.5 [Hz] for a nominal rotor. The corresponding phase lag is 57 degrees. Various rotor damages have been investigated as well. Rotors were impaired by reducing the radius; for example, a 75% rotor has 25% of the original length cut off. This results in faster dynamics, but should not be confused with increased performance however. For example, a rotor that is half as effective needs to spin $\sqrt{2} = 41\%$ faster to achieve the same force, see also eq. (20). This is a problem as the actuators are limited to an angular velocity of $\omega_{max} = 1200$ [rad/s].

$$H_{actuator}(s) = \frac{1}{\tau s + 1} \tag{19}$$

$$C_T \omega_1^2 = \frac{1}{2} C_T \omega_2^2 \Leftrightarrow \sqrt{2}\omega_1 = \omega_2 \tag{20}$$

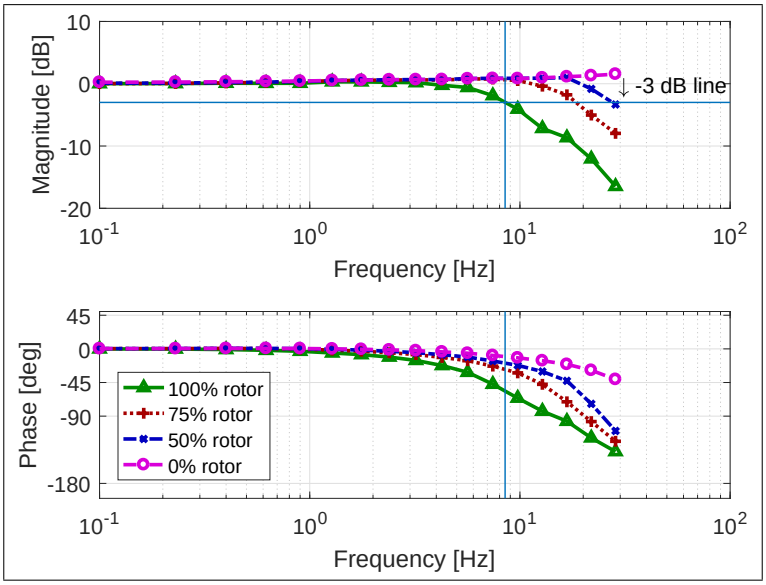

Figure 3. Bode plot for the actuators with various rotor damages

III.D. Single and double rotor failrues

A quadrotor undergoing rotor failure cannot control all states. This can be seen by inspecting the control effectiveness matrix M_C. Without loss of generality it is assumed that left back rotor has failed; rotor index 4. The relaxation of the yaw rate r has been used for single rotor failures. The resulting control effectiveness after elimination of the column corresponding to the broken rotor and the row corresponding to the yaw rate channel is presented in eq. (21).

$$\begin{bmatrix} \Delta\dot{p} \\ \Delta\dot{p} \\ \Delta T \end{bmatrix} = \underbrace{\begin{bmatrix} d_1 & -d_1 & -d_2 \\ l_1 & l_1 & -l_2 \\ 1 & 1 & 1 \end{bmatrix}}_{M_C} \begin{bmatrix} \Delta f_1 \\ \Delta f_2 \\ \Delta f_3 \end{bmatrix} \tag{21}$$

As both a state (yaw) and a rotor can be eliminated from eq. (17) the resulting M_C matrix in eq. (21) is square and has full-rank.

The same procedure can be used for a double rotor failure. Assume rotor 2 and 4 have failed. Elimination of the corresponding yaw rate and rotor channels will result in a tall *3-by-2* matrix. A tall matrix yields no mathematical solution suggesting the elimination of another state. Choosing either of roll rate p, pitch rate q or thrust T will result in an unstable system. The tall M_C matrix for two failed rotors is shown in eq. (22). The original application of the INDI controller is thus not sufficient for double rotor failures.

$$\begin{bmatrix} \Delta\dot{p} \\ \Delta\dot{p} \\ \Delta T \end{bmatrix} = \underbrace{\begin{bmatrix} d_1 & -d_2 \\ l_1 & -l_2 \\ 1 & 1 \end{bmatrix}}_{M_C} \begin{bmatrix} \Delta f_1 \\ \Delta f_3 \end{bmatrix} \tag{22}$$

The proposed INDI controller solves the absence of an inversion solution in eq. (22) by replacing the analytic inversion process in eq. (13) and eq. (17) with the Moore-Penrose inverse. The *3-by-2* tall matrix can then be inverted to provide a reference to both roll rate and pitch rate as well as thrust. The Moore-Penrose pseudoinverse (denoted by superscript $+$) provides a 'least squares error'-solution for the under-actuated quadrotor. This also means the solution is sub-optimal for all states. At the same time, it is expected that the robust nature of the INDI controller can cope with these 'least-squares' inaccuracies in the inverted control effectiveness matrix. For the same reason, inaccuracies with respect to other constant factors in eq. (17) such as moment of inertia $\mathbf{I}$ and mass m should be easily compensated. Therefore, no changes are supplied to the onboard autopilot regarding changed inertia or mass introduced by the loss of rotors. Applying the Moore-Penrose inversion on a double rotor failure results in the required control inputs (eq. (23)).

$$\begin{bmatrix} \Delta f_1 \\ \Delta f_3 \end{bmatrix} = \underbrace{\begin{bmatrix} d_1 & -d_2 \\ l_1 & -l_2 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}^+}_{M_C} \begin{bmatrix} \mathbf{I} \\ m \end{bmatrix} \begin{bmatrix} \dot{p}_{ref} - \dot{p} \\ \dot{q}_{ref} - \dot{q} \\ \dot{r}_{ref} - \dot{r} \\ T_{ref} - (\sum_{i=1}^{4} f_i) \end{bmatrix} \tag{23}$$

IV. Test setup and Results

The proposed control system is validated through flight tests in the Open Jet Facility (OJF) at the Delft University of Technology. The OJF is a large scale low-turbulence wind tunnel that features a 2.85 m aperture and wind speeds of up to 35 m/s

The test platform used in this research is the Parrot Bebop 2 quadrotor. The camera module was removed and battery was replaced to decrease total weight to produce a higher thrust-to-weight ratio. The Bebop 2 has onboard accelerometers and gyroscopes that measure with a 512 Hz sampling rate. The onboard processor runs Paparazzi UAV, an open-source autopilot. The software-branch developed in this resource is available on GitHub.[17]

IV.A. OptiTrack

An OptiTrack motion camera system provides position feedback based on the collection of reflective markers place on the quadrotor. The OptiTrack measures with submillimeter accuracy with a refresh rate of 120 Hz. A Wi-Fi connect is used to send the information to the onboard processor. The OptiTrack system will calculate an erroneous center of gravity based on the location of the reflective markers, which can result in performance degradation in combination with primary axis control. This may be alleviated by filtering the signal. To illustrate, the OptiTrack and on-board measured roll angle are compared in figure 4 where also the effect of filter frequency is shown. A tradeoff must be made between over- and under-fitting.

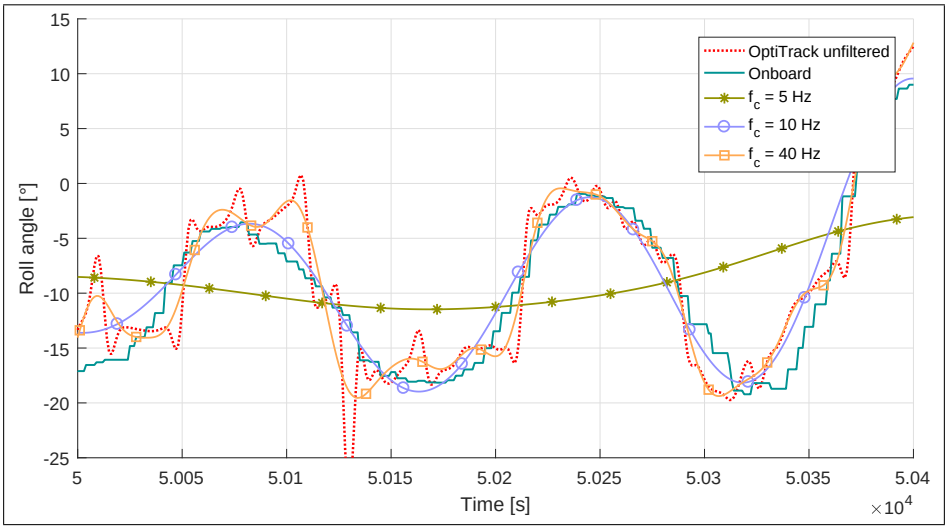

Figure 4. Effect of filtering the OptiTrack signal

IV.B. Flights

Flights are done to determine the forward velocity limits and the stability during forward flight. The quadrotor is given a position hold command in the center of the nozzle whilst the wind speed is incrementally increased.

The motor drivers are incapable of shutting down rotors in-flight. Therefore, the rotors are removed pre-flight and control architecture is switched to the failure mode. Both single (SRF) and (diagonal) double rotor failures (DRF) are tested.

IV.C. State analysis

Approximately 52 flights have been flown of which 10 useful SRF and 12 useful DRF flights. The ensemble averages for both configurations are calculated for wind speeds of 1 m/s and 4 m/s.

The outer loop tracking performance for both single and double rotor failures in 1 m/s wind are shown in figure 5. The reference signal for the DRF cases include a portion of the flights towards the setpoint which is visible for the x and z channel. Both the SRF and the DRF case can track the reference signals with a certain error margin. The largest error occurs for height tracking for the DRF case. The height is controlled separately from x and y. It is suspected the feed forward term in this 'thrust loop' does not handle the loss of actuators correctly. This investigation is left for future work.

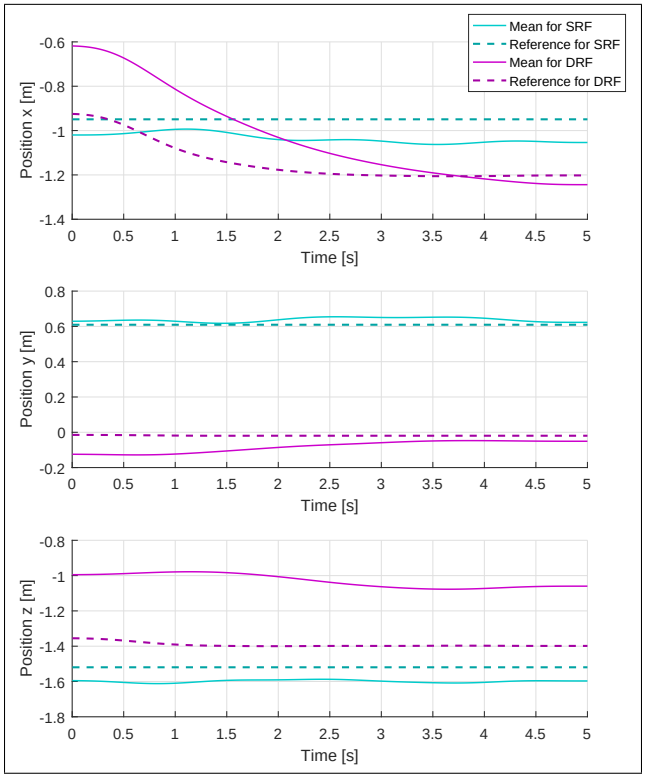

Figure 5. Comparing mean position tracking for single rotor failure (SRF) and double rotor failures (DRF) in 1 m/s wind

The steady state position errors creep in for DRF undergoing higher wind speeds. Clearly, the integral gains are tuned incorrectly. This prevents comparison of the outer loop states x, y and z during higher wind speeds. Instead, the performance of the inner loop can be analyzed (i.e. the roll and pitch rate errors).

The inner loop tracking performance for a 1 m/s and 4 m/s wind speed are shown in figure 6 and 7, respectively. A Butterworth filter with 2 Hz cutoff frequency is used as a way to indicate a moving average.

IV.D. Limit testing

The limits of both configurations were found by incrementally increasing the wind speed until the system crashed. The results for fastest achieved wind speed per configuration are plotted in figure 8. Here, the effect of the lacking integral gains can be clearly seen for the DRF case. The x position starts to drift away. This also results in a lost connection by the OptiTrack position measurement system.

The rotor speeds with respect to the body are shown in figure 10. Only the front rotors are shown to increase readability; rotor 2 for SRF and rotor 1 for DRF. Absolute values for the rotation rate are shown due to the different yaw directions. During flight without wind the required speed difference between two rotors is lower than for high wind condition, which is especially visible in the DRF case. Rotor speed fluctuates with 250 rad/s at a low 2 m/s wind speed and increases to nearly 600 rad/s at 8 m/s. This effect is less visible for the SRF case but still persists. This range widens until both the upper and lower bound of the actuator limits are hit. The average rotor speeds for both

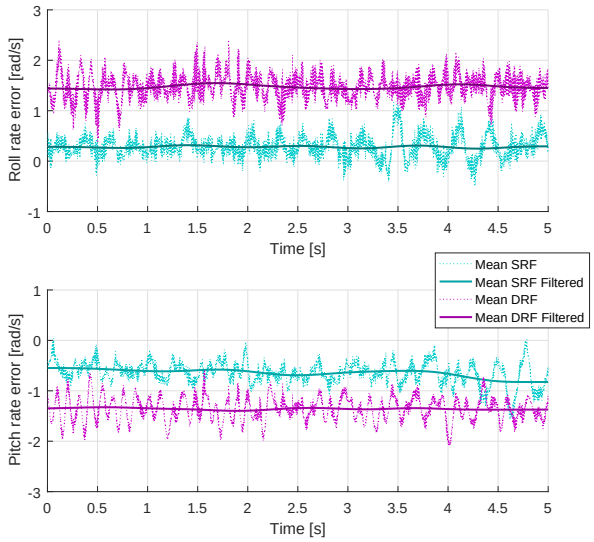

Figure 6. Comparing mean roll and pitch rate tracking error for single rotor failure (SRF) and double rotor failures (DRF) in 1 m/s wind

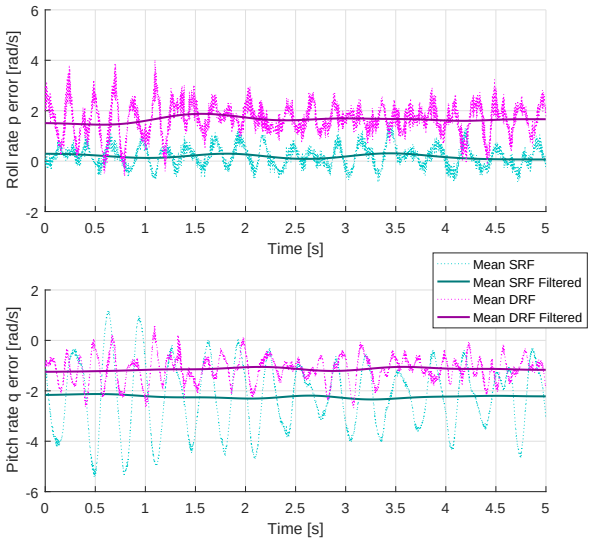

Figure 7. Comparing mean roll and pitch rate tracking error for single rotor failure (SRF) and double rotor failures (DRF) in 4 m/s wind

cases illustrate more clearly the effect of larger excitation. In figure 9 the average observed rotor speeds are shown in relation with time and wind speed for the best flights.

IV.E. discussion

An unexpected result is the pitch rate error for the SRF being larger than the DRF for a wind speed of 4 m/s in figure 7. The variance of the ensemble average can be explained by the coincidental alignment of the sinus-like error signals. However, the time average of the signals, approximated by a 3^{rd}-order Butterworth filter with a cutoff frequency at 2 Hz, is lower than for the DRF. It was expected that the SRF would be better able to control the pitch rate due to the availability of an additional rotor. Inspection of the single flights show larger deviations for the pitch rate for SRF; probably due to the wind speed drag working in the x-direction.

Another possibility is human error. A plotting error has been ruled out by inspection of related states such as the pitch angles, differentiation of the pitch angles and values for the primary axis $\mathbf{n_i}$, which all show the same behavior. Furthermore, videos of the flights and the angular rates in figure 7 do indeed confirm that the SRF had more aggressive rate maneuvers than DRF case. This larger observed body rate could then imply a larger error as the magnitude of the references rates are equal across both cases.

What remains are the differences in the test setup for single and double rotor failures for both the software and the hardware. It was expected that averaging over multiple flights would eliminate small discrepancies due to differences in things such as marker placement, rotor geometry, quadrotor frames and battery weight. It should further be mentioned that the SRF case had a small counterweight attached to the front right leg to assist in takeoff. This was not expected to be an influence in flight, but could be a contributing factor.

Regarding the software, the architectures for both cases are roughly the same but different software 'branches' were used. Not all parameters were monitored and could be inadvertently changed across branches. An important parameter, for example, the filter cutoff frequency used for state filtering could have been different.

Future research should perform compare flights using the same software branch and the same hardware configuration to rule out the unlikely possibility of the double rotor failure software being superior to the SRF case in terms of pitch rate.

V. CONCLUSIONS

The usage of the Incremental Nonlinear Dynamic Inversion (INDI) as an application for quadrotor fault tolerant control was limited to single rotor failures (SRF). This research expands the applicable damage cases to include diagonal double rotor failures (DRF). The loss of an additional rotor necessitates the INDI controller to use a Moore-Penrose inversion instead of a normal inversion. Position control is achieved by relaxing the yaw rate and controlling

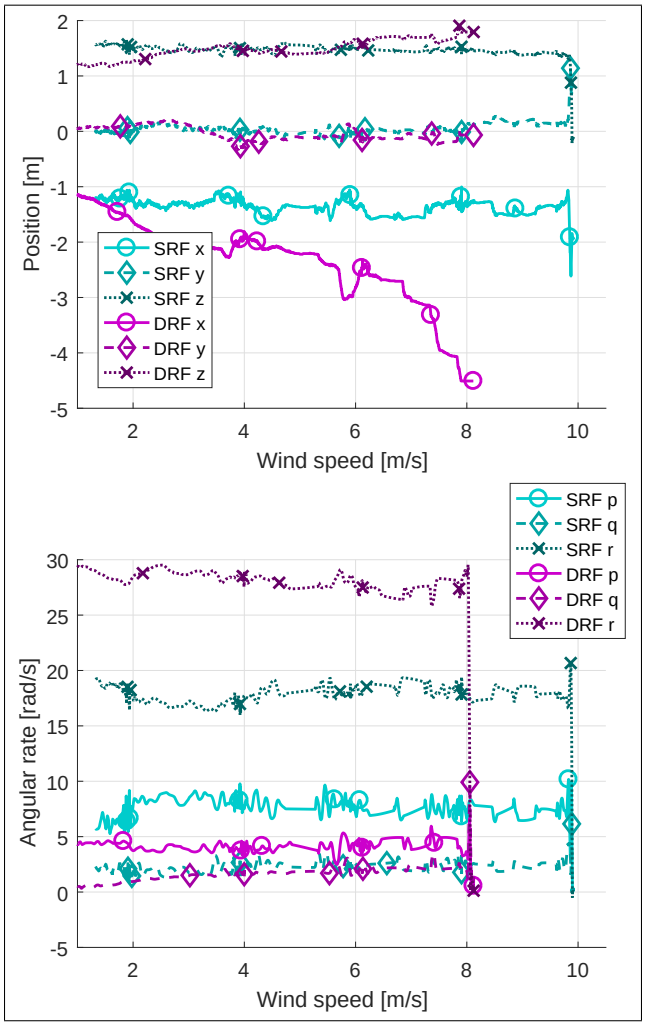

Figure 8. Best flight comparison for position and rotation rates

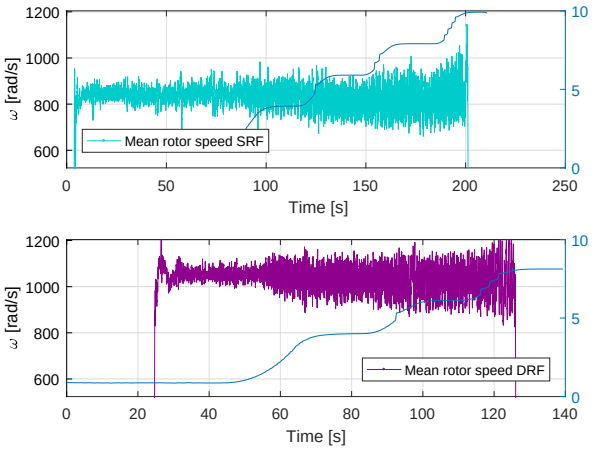

Figure 9. Best flight comparison for front rotor speeds

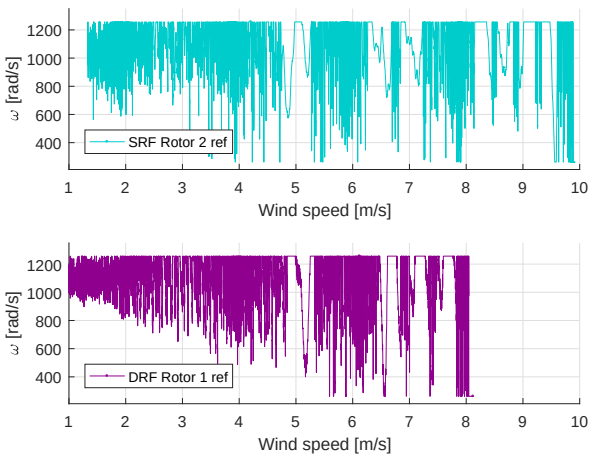

Figure 10. Best flight comparison for front rotor speeds

the average thrust direction in inertial space.

Position hold flights with increasing wind speeds have been performed in a wind tunnel. Comparisons between SRF and DRF cases have been made to assess the impact on flight performance. The average of 10 SRF flights and 12 DRF flights in 1 m/s wind shows that attitude control is worse for the DRF case. An expected result as less control authority was available and a sub-optimal inversion process was used. Strangely, the pitch rate error for SRF surpassed the DRF case for higher wind speeds. The outer loop tracking error for both cases was bounded but did not appear to be asymptotically stable. The fastest achieved flight for the DRF case was 8 m/s which is expected to be worse than the 10 m/s achieved by the SRF. The loss of two rotors dramatically increased yaw rate; there was no counter-rotating rotor to lower the yaw rate equilibrium. Investigation of the yaw rate indicate that it was still within the cutoff frequency of the actuator dynamics and as such is not considered a problem.

A recommendation for future research is the addition of a Fault Detection and Diagnosis module to be able to handle in-flight sustained damage. Also, the authors aim to find a reason for the better pitch rate tracking in the double rotor failure case.

Acknowledgments

The authors thank the MAVLab and High Speed Lab, TU Delft for providing test facilities and equipment.

References

[1]Sadeghzadeh, I., Mehta, A., Chamseddine, A., and Zhang, Y., "Active Fault Tolerant Control of a Quadrotor Uav Based on Gain- Scheduled Pid Control," *Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on*, 2012, pp. 1–4.

[2]Liu, Z., Yuan, C., and Zhang, Y., "Active Fault-Tolerant Control of Unmanned Quadrotor Helicopter Using Linear Parameter Varying Technique," *Journal of Intelligent & Robotic Systems*, , No. April, 2017.

[3]Benrezki, R. R., Tadjine, M., Yacef, F., and Kermia, O., "Passive Fault Tolerant Control of Quadrotor UAV Using a Nonlinear PID *," *IEEE Conference on Robotics and Biomimetics*, 2015.

[4]Sharifi, F., "Fault Tolerant Control of a Quadrotor UAV using Sliding Mode Control," *Conference on Control and Fault Tolerant Systems*, 2010, pp. 239–244.

[5]Khebbache, H., Sait, B., Yacef, F., and Soukkou, Y., "Robust Stabilization of A Quadrotor Aerial Vehicle in Presence of Actuator Faults," *International Journal of Information Technology, Control and Automation*, Vol. 2, No. 2, 2012, pp. 1–13.

[6]Merheb, A. R., Noura, H., and Bateman, F., "Design of Passive Fault-Tolerant Controllers of a Quadrotor Based on Sliding Mode Theory," *International Journal of Applied Mathematics and Computer Science*, Vol. 25, No. 3, 2015, pp. 561–576.

[7]Lanzon, A., Freddi, A., and Longhi, S., "Flight Control of a Quadrotor Vehicle Subsequent to a Rotor Failure," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 2, mar 2014, pp. 580–591.

[8]Mueller, M. W. and D'Andrea, R., "Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers," *Proceedings - IEEE International Conference on Robotics and Automation*, 2014, pp. 45–52.

[9]Lippiello, V., Ruggiero, F., and Serra, D., "Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach," *IEEE International Conference on Intelligent Robots and Systems*, , No. August 2015, 2014, pp. 4782–4788.

[10]Lu, P. and Van Kampen, E. J., "Active fault-tolerant control for quadrotors subjected to a complete rotor failure," *IEEE International Conference on Intelligent Robots and Systems*, Vol. 2015-Decem, 2015.

[11]Sun, S., Sijbers, L., Wang, X., and de Visser, C., "High-Speed Flight of Quadrotor despite Loss of Single Rotor [SUBMITTED]," *IEEE Robotics and Automation Letters*, 2018.

[12]Lu, P., van Kampen, E.-J., de Visser, C., and Chu, Q., "Aircraft fault-tolerant trajectory control using Incremental Nonlinear Dynamic Inversion," *Control Engineering Practice*, Vol. 57, 2016, pp. 126–141.

[13]Tayebi, A. and McGilvray, S., "Attitude stabilization of a VTOL quadrotor aircraft," *IEEE Transactions on Control Systems Technology*, Vol. 14, No. 3, 2006, pp. 562–571.

[14]Ostroff, A. J. and Bacon, B. J., "Enhanced NDI strategies for reconfigurable flight control," *Proceedings of the 2002 American Control Conference*, 2002, pp. 3631–3636.

[15]Bacon, B., Ostroff, A. J., and Joshi, S., "Reconfigurable NDI Controller Using Inertial Sensor Failure Detection & Isolation," *Aerospace and Electronic Systems, IEEE Transactions on*, Vol. 37, No. 4, 2001, pp. 1373–1383.

[16]Mahony, R., Kumar, V., and Corke, P., "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor," *IEEE Robotics & Automation Magazine*, Vol. 19, No. 3, 2012, pp. 20–32.

[17]Sijbers, L., "Available: https://github.com/LeonOfich/paparazzi_windtunnel/tree/New_Guidance_Test4," 2018.

Part II

Preliminary Report

Handling quadrotor double rotor failures through Fault Tolerant Control

Leon Sijbers, 4092643
Control & Simulation

June 18, 2018

Abstract

The effort in making quadrotors more fail-safe requires a flying damaged quadrotor to estimate the safe flight envelope. The current control methods are insufficient for achieving stabilized flight in face of full rotor failed quadrotors. This research is aimed at designing a Fault Tolerant Controller capable of stabilizing a double rotor failed quadrotor. The control method used will be the Incremental Nonlinear Dynamic Inversion in combination with the *primary-axis* method. The latter sacrifices yaw control and subsequently steers the axis about which the quadrotor rotates. The two remaining healthy rotors risk actuator saturation which can be mitigated with tailored generalized pseudo-inverses, Pseudo-Control Hedging or command filtering. The designed controller will be tested for hover and forward flight conditions while being subjected to disturbances.

1 Introduction

Small Unmanned Aerial Vehicles (UAV), characterized by their low weight and size, find themselves a presence in our everyday life more often and this only seems to be increasing. UAVs are used for various tasks such as surveillance, research platforms, mobile flying cameras, goods delivery etc.[1]
Quadrotors (see fig. 1) are perfect for such tasks due to their low complexity, low maintenance, vertical take-off and landing(VTOL), and hover ability.

Figure 1: Example of a quadrotor. (https://www.parrot.com/us/drones/parrot-bebop-2)

[1]Amazon Prime Air will start once regulatory support is achieved: https://www.amazon.com/Amazon-Prime-Air/b?node=8037720011

Quadrotors are inherently unstable, meaning they require controllers to provide closed loop stability. It is able to control rotations around three axes as well upwards thrust. Quadrotors are under-actuated systems; damage to either of the four rotors quickly results in unrecoverable dynamics. This makes it an interesting subject from a control theory perspective as well as a safety perspective.

A trivial solution is adding hardware components to solve the under-actuation. For example, by adding extra rotors to reach a multirotor configuration. [1, 2] Tilting the rotors is a solution as well. [3] Saied *et al.* solved this underactuation problem by employing coaxial double-rotors. [4] The main problem with increasing hardware redundancy is the inherent increase in complexity and thus maintenance cost; a key advantage of quadrotors for their utility in the first place. Therein lies the added value in designing control architecture capable of controlling the quadrotor when damaged. This is called Fault Tolerant Control.

Research is done to estimate the global flight envelope of damaged quadrotors. The flight envelope being the combination of states the quadrotor is capable of reaching. This global envelope can be calculated using a model; the model can be built using flight data. The problem arises in gathering the flight data as the damaged quadrotor cannot fly with its current controller. That is where Fault Tolerant Control helps by enabling extended flight.

Previous research was limited to partial rotor failure [5–10] or a single-rotor failure [11–13]. Double-rotor failure was investigated, but was limited to linear control methods. [14, 15] This research aims to develop a nonlinear controller and apply it to a double-rotor failed quadrotor. In achieving this the body of knowledge regarding the safe flight envelope of damaged quadrotors can expanded.

The remainder of this paper is structured as follows: the historical perspective as well as the most recent developments regarding Fault Tolerant Control for quadrotors are discussed in section 2. In the same section it is also shown how this research contributes to the current body of knowledge and how it can be used for future research. The research goal, objectives and questions are formulated in section 3. The methodology, which will be used to answer the research questions, will be explained in section 4. Finally, the conclusions are presented in section 5.

2 State-of-the-Art Review

In this section the current advancements on the Fault-Tolerant Control (FTC) for quadrotors will be discussed. First, faults and failures are defined in section 2.1. Then a short introduction is given about Passive and Active Fault Tolerant Controllers, their shortcomings and advantages in section 2.2. Subsequently, the application of various FTC strategies are discussed in section 2.4. After rotor failure a different strategy should be employed: the reduced attitude control. This can be used in combination with adaptive incremental nonlinear dynamic inversion to robustly control the nonlinear dynamics of the quadrotor. The modeling of an asymmetric quadrotor is done in section 2.3. Here, the reference frames are defined. The equations of motions are presented for the quadrotor. The actuators are discussed section 2.5 and shown how these influence the moments on the body. It becomes apparent that a rotor failure increases the risk of actuator saturation which can be mitigated using specialized generalized inversion and Pseudo-Control Hedging. This section will be concluded by specifying how this research contributes to the body of knowledge in section 2.6.

2.1 Defining faults and failures

In this paper the definitions for faults and failures will be borrowed from the book *Fault-diagnosis systems: An introduction from fault detection to fault tolerance* by R. Isermann.

"A fault is an unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard condition. [16]

A fault may cause a reduction or loss in the system performance, but it may not have an effect on the correct functioning of the system. For example, a bent skid on a helicopter is not as strong or stable as a healthy skid, but the helicopter will still be able to land on the skid.
Faults may be numerous and small and have no effect. In other cases a fault is so severe that it causes the occurrence of a failure.

"A failure is a permanent interruption of a system's ability to perform a required function under specified operating conditions." [16]

Specifically, a failure event is the inability of a functional unit to perform its tasks. Going back to the helicopter example, the skid may have broken off asymmetrically which means the helicopter would roll over when it tries to land.

2.2 Fault Tolerant Control

A Fault Tolerant Flight Controllers' (FTC or FTFC) task is to automatically maintain stability and acceptable performance even if faults or failures occur in the system. [17] The ways FTC achieve this can be divided into two branches: Passive FTC (PFTC) and Active FTC (AFTC).

In PFTC the controller architecture is fixed and designed to be robust to variation in parameters. The advantage is a single reliable controller with relatively low hardware and software requirements that reacts faster than AFTC. [18] Another advantage is that PFTC can be made more reliable than AFTC according to classical reliability theory. [19]

A fault or failure can be so severe that the designed robustness of the PFTC is not sufficient. Active Fault Tolerant Control solves this by re-configuring itself when a fault occurs. This logically brings with it that a fault must be detected and diagnosed. The module dedicated makes the control architecture more complex, but also allows a wider variety of faults to be controlled.

2.2.1 FDD module

An essential part of AFTC is this Fault Detection and Diagnosis module (FDD). It must be accurate such that it does not generate false positives, or worse, false negatives. The former introducing decreased performance whilst the latter fails to stabilize the system with potentially catastrophic outcome. The FDD must also be quick. The time it takes between the occurrence and detection of a fault must not be longer than the time it takes for the system to destabilize. FDD becomes more accurate with time so there is a trade-off to be made. [18]

Efforts are made to combine a fast-responding passive FTC with an active FTC. The PFTC tries to stabilize the post-fault system which buys the FDD valuable time to accurately diagnose a fault. The AFTC then takes over and ensures performance. [20] [21] [22] Although this solves the time/accuracy-sensitivity, the disadvantage of having a PFTC non-optimally control the nominal system remains. The preferred method in that case will still be a single AFTC.

Fault Detection and Diagnosis and Fault Tolerant Control are symbiotic; their performance depend on each others output. [17] In order to abstractly test and prove the stability of a FTC, it is often assumed the FDD is perfect and has detected the fault. [23] [24] [25] [11] The design of a FDD module is considered out

of the scope and as such it is assumed the FTC has switched controller configurations.

2.2.2 FTC module

Once the fault/failure is detected and diagnosed the Fault-Tolerant Controller can do its job. A supervision or switching mechanism can switch to the controller designed for a specific failure case. [17] The field of Fault-Tolerant Control dates back to the 1980s when NASA started doing dedicated research into reconfigurable control techniques. [26]. More inteliggent FTC was researched in the 1990s. [27] Since then, many different mathematical design tools have been developed. Some are pre-computed control laws (e.g. Gain Scheduling or Linear Parameter Varying) others focus on online automatic redesign such as Dynamic Inversion or Sliding Mode Control *et cetera*. A thorough overview of various FTC designs is given by Zhang. [28]

Failure cases for quadrotors

The quadrotor system is subjected to various forms of faults, most occur in the sensors or actuators. In this paper, the research will be limited to actuator failures and structural failures to the rotors. That is to say, the amount of thrust a rotor is able to produce will be diminished or non-existent. In literature this is called loss of effectiveness (LOE).

The LOE is defined as the gain applied to the desired rotor thrust and the actual rotor thrust. This results in decreased performance as shown in Figure 2. A LOE value of 0 denotes a healthy rotor whereas LOE = 1 means total failure; anything in between 0 and 1 is considered a partial failure.

The four possible failure cases for single- and triple-rotor failures are generally equal. Double-rotor failures have two distinct cases: the adjacent-case or the opposite-case. This is visually explained in Figure 3. The distinction is important due to the rotational direction of the remaining healthy rotors and the force balance around the center of gravity.

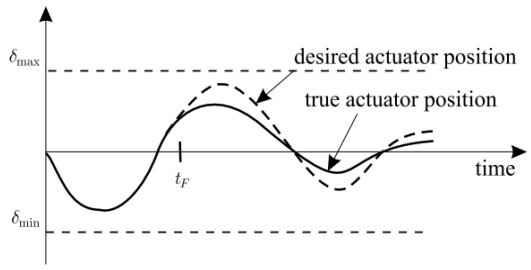

Figure 2: Loss of effectiveness (LOE) example. [17]

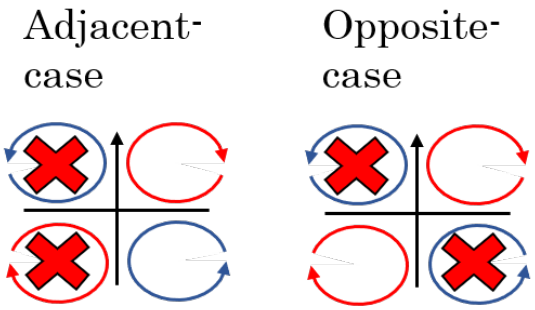

Figure 3: Example of adjacent- and opposite-case for double-rotor failure

2.3 Modeling an Asymmetric Quadrotor

The application of a controller requires some modeling. A quadrotor model is derived in this section. The derivation presented here is an adapted combination of the one given in Mahony, Kumar & Corke in [29] and the one by Lu & van Kampen in [12]. It has been adapted as the quadrotor used in this research has a different axis of symmetry compared to the literature. Prior to that the reference frames and axis definition is given.

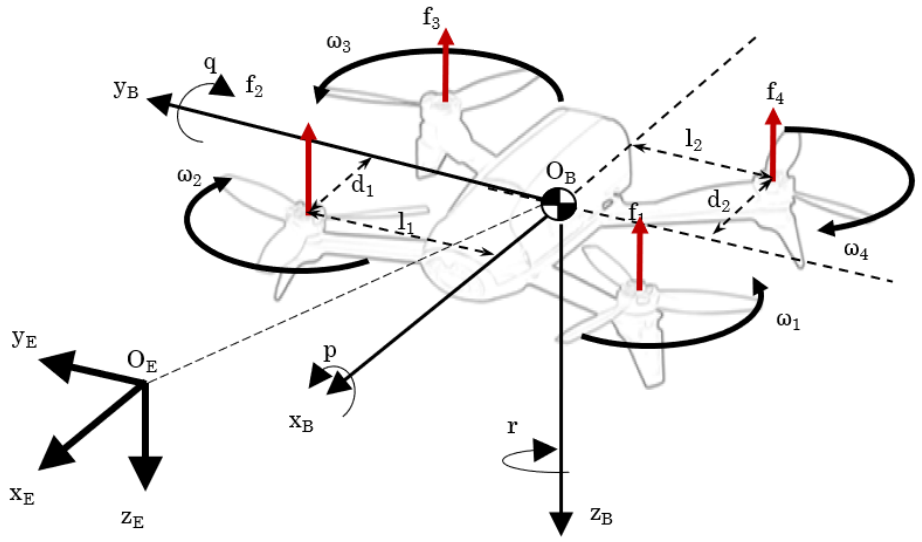

Figure 4: Bebop 2 vehicle model in the Body Frame. All rotors produce a force parallel to the Z-axis. Rotors 1 and 3 rotate clockwise, whilst rotors 2 and 4 rotate counterclockwise. The vehicle is symmetric about the $X_B Z_B$-plane.

2.3.1 Reference frames

The motion of the quadrotor can be studied using a body-fixed and space-fixed reference frame.

The body-fixed reference frame (O_B,x_B,y_B,z_B) has its axis fixed with the orientation of the vehicle body. The origin O_B is fixed to the center of gravity. The Bebop 2's symmetry will be used to define the X-axis. Following aircraft dynamic conventions and right-hand orthogonal rules the body reference frame can be defined as positive x_B out of the nose, positive y_B to the right when facing in the positive x_B direction and positive z_B down. This vehicle model is illustrated in fig. 4

The quadrotor moves and rotates in the Earth reference frame (O_E,x_E,y_E,z_E). Its origin O_E is fixed with respect to Earth and Z_E points down. A relative translation offset between the origins O_E and O_B can be defined as $\boldsymbol{d} = [x, y, z]^T$. The relative rotations around respectively $x_E, y_E z_E$ are defined as $\boldsymbol{\eta} = [\phi, \theta, \psi]^T$; the well-known roll, pitch, yaw notation. [30]

The mapping of the Earth frame to the Body frame is done through the rotation sequence yaw, pitch, roll which yields rotation matrix $\boldsymbol{R}_{E\to B}$:

$$\boldsymbol{R}_{E\to B} = \begin{bmatrix} C\theta C\psi & C\theta S\psi & -S\theta \\ S\phi S\theta C\psi - C\phi S\psi & S\phi S\theta S\psi + C\phi C\psi & S\phi C\theta \\ C\phi S\theta C\psi + S\phi S\psi & C\phi S\theta S\psi - S\phi C\psi & C\phi C\theta \end{bmatrix} \tag{1}$$

where C and S are shorthand forms for cosine and sine respectively. The body angles are roll ϕ, pitch θ and yaw ψ. Note that $\boldsymbol{R}_{B\to E} = \boldsymbol{R}^T_{E\to B}$.

2.3.2 Modeling quadrotor

The vehicle is modeled as a rigid body with four rotors with a mass m. Each rotor can be modeled as a flat disk producing a force in the $-Z_b$ direction originating from the hub. This gives a simplified total thrust vector $\boldsymbol{T} = [0, 0, \sum_{i=1}^{4} f_i]$. The translational equation of motion in the Earth frame is given by eq. (2).

$$\ddot{\boldsymbol{d}} = -\frac{1}{m}\boldsymbol{R}\boldsymbol{T} + \boldsymbol{g} - \frac{1}{m}\boldsymbol{R}\boldsymbol{F} \tag{2}$$

whereby $\boldsymbol{g} = [0, 0, 9.81] ms^{-2}$, $\boldsymbol{R}$ the rotation matrix, and $\boldsymbol{F}$ is a diagonal matrix with friction forces for each velocity component.

$$\boldsymbol{F} = \begin{bmatrix} \frac{1}{2}C_x A_x u\,|u| & 0 & 0 \\ 0 & \frac{1}{2}C_y A_y v\,|v| & 0 \\ 0 & 0 & \frac{1}{2}C_z A_z w\,|w| \end{bmatrix} \tag{3}$$

The relation of angular velocity in Earth and body is given by the following equations:

$$\dot{\phi} = p + qS\phi T\theta + rC\phi T\theta \tag{4}$$

$$\dot{\theta} = qC\phi - rS\phi \tag{5}$$

$$\dot{\psi} = \frac{1}{C\theta}(qS\phi + rC\phi) \tag{6}$$

Restating the trigonometric abbreviations for clarity: $S, C, T, \rightarrow sin(\cdot), cos(\cdot), tan(\cdot)$ respectively. The angular velocity $\boldsymbol{\omega} = [p, q, r]^T$ and its derivative $\dot{\boldsymbol{\omega}}$ are related to the moments through the rotational equations of motion (eq. (7)).

$$\boldsymbol{I}_B\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \boldsymbol{I}_B\boldsymbol{\omega} = \boldsymbol{M}_{torque} + \boldsymbol{M}_{aero} \implies \dot{\boldsymbol{\omega}} = \boldsymbol{I}_B^{-1}[\boldsymbol{M}_{torque} - \boldsymbol{\omega} \times \boldsymbol{I}_B\boldsymbol{\omega} + \boldsymbol{M}_{aero}] \tag{7}$$

with $\boldsymbol{M}_{torque}$ the torques produces by the actuators and $\boldsymbol{M}_{aero}$ the moments produced by the body. [31] The Moment of Inertia tensor of the Bebop 2 is given by $\boldsymbol{I}_B$

$$\boldsymbol{I}_B = \begin{bmatrix} I_{xx} & 0 & -I_{zx} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix} \tag{8}$$

2.4 Application of FTC to quadrotors

The failure cases have been discussed in section 2.2.2. The first step in designing FTC is making it able to handle small deviations from nominal situations. Therefore, partial LOE is the first logical step in assessing the effectiveness of a FTC.

The partial LOE of one or more rotors has been thoroughly investigated using passive and active FTC. [5–10] All these control strategies are designed to reject errors for all states: attitude and position. None are able to compensate for a total rotor failure, making this total rotor failure an important step in research.

A total rotor failure and the full attitude control seems an impossible combination. A recent study by Merheb, Noura & Bateman in 2017 solved the problem of a full rotor failure by proposing a weight shift

which allows the quadrotor to be controlled as a *trirotor* configuration. [13] The yaw rate error was bounded, but not asymptotically stable; a good accomplishment in itself. However, this method requires an additional actuation system that shifts the Center of Gravity during flight. This is adding hardware redundancy which makes the quadrotor heavier and more complex.

2.4.1 Reduced Attitude control

A revolutionary control law for handling a full rotor failure was proposed by Lanzon, Freddi & Longhi in [11]. The researchers showed that sacrificing the ability to control yaw allows full control of the other states. Essentially this is controlling the reduced attitude representation; The controller steers the unit vector, or *primary axis* about which the system rotates.

The Reduced Attitude control, or *primary axis* control, controls the average thrust vector. This vector $\boldsymbol{n}$ is aligned with the primary axis about which the vehicle's body rotates. For this application the yaw axis is chosen such that $\boldsymbol{n} = [0, 0, 1]^T$.

The translational accelerations can be rewritten into:

$$\ddot{\boldsymbol{d}} = g\boldsymbol{e}_3 - \boldsymbol{R}\boldsymbol{e}_3\frac{T}{m} \tag{9}$$

where $\boldsymbol{e}_3 = [0, 0, 1]^T$.
Let $\boldsymbol{n} = [n_x, n_y, n_z]^T$ then

$$\dot{\boldsymbol{n}} = -\boldsymbol{\omega}^B \times \boldsymbol{n} \tag{10}$$

The outer loop controls the position by generating a desired direction for the primary axis

$$\boldsymbol{n}_d = m\boldsymbol{R}^{-1}\frac{\left(\ddot{\boldsymbol{d}}_d - g\boldsymbol{e}_3\right)}{n_z T_c} \tag{11}$$

With the required thrust calculated as:

$$T_c = -m\frac{(\ddot{z}_d - g}{c\phi c\theta} \tag{12}$$

The control of the primary axis is then given by

$$\begin{bmatrix}\dot{n}_x\\ \dot{n}_y\\ \dot{n}_z\end{bmatrix} = \underbrace{\begin{bmatrix}0 & -n_z & n_y\\ n_z & 0 & -n_x\\ -n_y & n_x & 0\end{bmatrix}}_{B}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{13}$$

The B matrix is singular which will be important later.

Mueller and D'Andrea presented a control strategy that is similar to the one by Lanzon *et al.*, but added a degree of freedom by controlling the ratio of thrust force. [14] The proposed strategy was implemented on an actual quadrotor using a Linear Quadratic Regulator controller around predetermined solutions for rotor thrust. The one- and two-rotor failure cases were explored using a high performance symmetric quadrotor. The three-rotor loss case was investigated in a separate study by Zhang, Mueller & D'Andrea whereby a single-rotor flying vehicle was built. [32] The downside of this method is that the solution has to be determined *a priori*. This limits the application of this method as it will not be able to adapt to the various disturbances it encounters while flying.

Lippiello, Ruggiero & Serra proposed to turn off the rotor opposing the broken one which essentially turns the quadrotor into a birotor. Subsequently, this birotor was controlled using a PID-control as well as backstepping control. It was shown that such a birotor is able to reach any point in Cartesian space. The study was limited to a simulation environment; application to a real testbed was left as future work. [33] The controller used in this study is a linear controller which makes it more difficult to handle uncertainties and disturbances.

Crousaz, Farshidian, Neunert & Buchli showed it is possible to employ Sequential Linear Quadratic control to control a quadrotor with a single or double (opposite) rotor failure. This method is also capable of handling aggressive manoeuvres using slung load and follow a trajectory for the failure cases. The experiment is limited to a simulation environment without any disturbances. [15]

2.4.2 Nonlinear control techniques

The literature presented in section 2.4 and section 2.4.1 were linear control techniques. These techniques are great when dealing with a single solution e.g. hover in place. However, a quadrotor is a highly nonlinear vehicle. Therefore, the performance of linear controllers degrade when operating over a range of states. This requires gain scheduling between linear controllers, or even better, the use of nonlinear control techniques such as nonlinear dynamic inversion (NDI).
Nonlinear dynamic inversion requires full knowledge of the system model and states to make a linearized model. Should there be any model inaccuracies or errors then the model cannot be fully linearized. This makes NDI method itself fault-sensitive.

2.4.3 Nonlinear Dynamic Inversion

The method of Nonlinear Dynamic Inversion (NDI), also called feedback linearization, is algebraic transformation of the nonlinear system dynamics into a linear one. This is fundamentally different from Jacobian linearization as it is no longer a linear approximation. [34]

The NDI method will be explained here using the nonlinear dynamics as presented in section 2.3.2.

The first steps are the rotational equations of motion which were already presented in eq. (7) and reiterated here:

$$I\dot{\omega} + \omega \times I\omega = \mathbf{M} \tag{14}$$

We can write $\mathbf{M}$ as a polynomial with respect to the aerodynamic moment of the body $\mathbf{M_{aero}}\mathbf{x}_a$ and the torque supplied by the rotors $\mathbf{M_{torque}} = [\tau_p, \tau_q, \tau_r]^T = \mathbf{M_c}\delta$ as

$$\mathbf{M} = \mathbf{M_a}\mathbf{x}_a + \mathbf{M_c}\delta \tag{15}$$

Here $\boldsymbol{x}_a$ is the state vector, δ the actuator forces for each rotor and $\mathbf{M_a}$ and $\mathbf{M_c}$ the aerodynamic and control effectiveness matrices respectively.

Replacing $\dot{\omega}$ in eq. (14) with virtual control ν and rewriting eq. (15) for δ gives

$$\delta = (\mathbf{M_c})^{-1}(I\nu + \omega \times I\omega - \mathbf{M_a}\mathbf{x}_a) \tag{16}$$

The block diagram for a typical NDI controller is given in Figure 5.

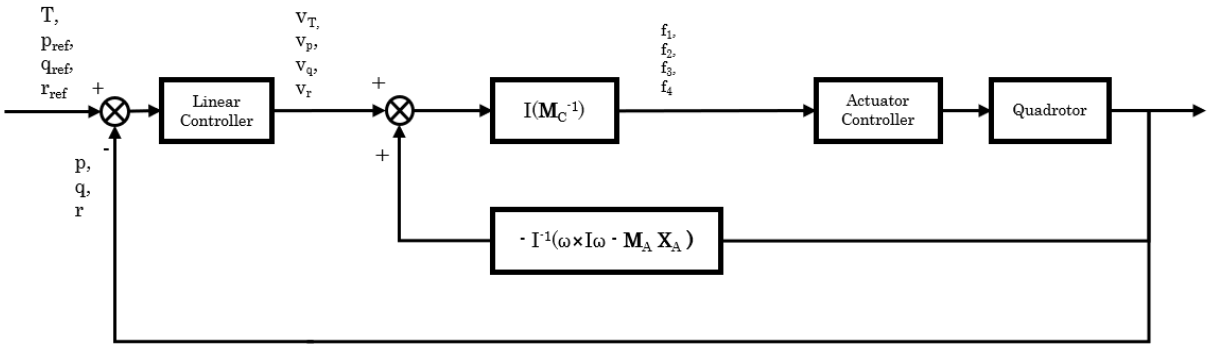

Figure 5: Control diagram for Nonlinear Dynamic Inversion control

The limitations of eq. (16) is that it still requires a linear outer-loop controller. The closed loop behavior of the virtual control v and the aircraft states can be tuned with a single integrator since angular velocities (ω) can simply be measured.

This NDI method assumes constant moment of inertia (I) and geometry which might not be the case in a structural failure. Also, the aerodynamic moments ($\mathbf{M}$) are around the center of gravity, which might change. The assumptions mean the nonlinear dynamics cannot be perfectly canceled.

2.4.4 Incremental NDI

A downside of NDI is the dependency on accurate estimates of the actual aerodynamic coefficients in $\mathbf{M_A}$. Incremental Nonlinear Dynamic Inversion works with less information about the model which makes it more robust when compared with NDI. Instead of calculating the required deflection (or rotor velocity) it calculates the required *incremental* deflection compared to the current situation, hence the name. The INDI method was first published by Ostroff & Bacon in [35]. The derivation given here is based on the paper by Sieberling, Chu & Mulder [36].

The INDI control, like NDI, again starts with the rotational equations of motion.

$$I\dot{\omega} + \omega \times I\omega = \mathbf{M} = \mathbf{M_a x_a} + \mathbf{M_c}\delta \tag{17}$$

The moment $\mathbf{M}$ within eq. (17) changes over time. In an infinitesimal time-step the angular rates $\Delta\omega = 0$ do not change, neither do the aerodynamic moments such that $\Delta\mathbf{M_a} = 0$. The angular accelerations $\dot{\omega}$ do change. This isolates the difference solely to the control moments. Equation (17) can be reduced to eq. (18) with $\mathbf{M_c}$ the control effectiveness matrix:

$$\Delta(\mathbf{M_c}\delta) = I\,(\Delta\dot{\omega}) = I\,(\dot{\omega}_{new} - \dot{\omega}_{old}) = \mathbf{M_c}\,(\Delta\delta) \tag{18}$$

The angular accelerations of the old time-step are measured by the Inertial Measurement Unit and are used for $\dot{\omega}_{old}$ in eq. (18). This makes the INDI a sensor-based approach, but requires filtering of the measured states.

Rewriting eq. (18) to find $\Delta\delta$ and inputting the required angular accelerations $\dot{y} = \dot{\omega}_{new}$ gives:

$$\Delta\delta = \mathbf{M_c}^{-1} I\,(\dot{\mathbf{y}} - \dot{\omega}_{old}) \tag{19}$$

The incremental deflection calculated in eq. (19) is added to the current deflection to produce the required deflection. The resulting control diagram is given in Figure 6.

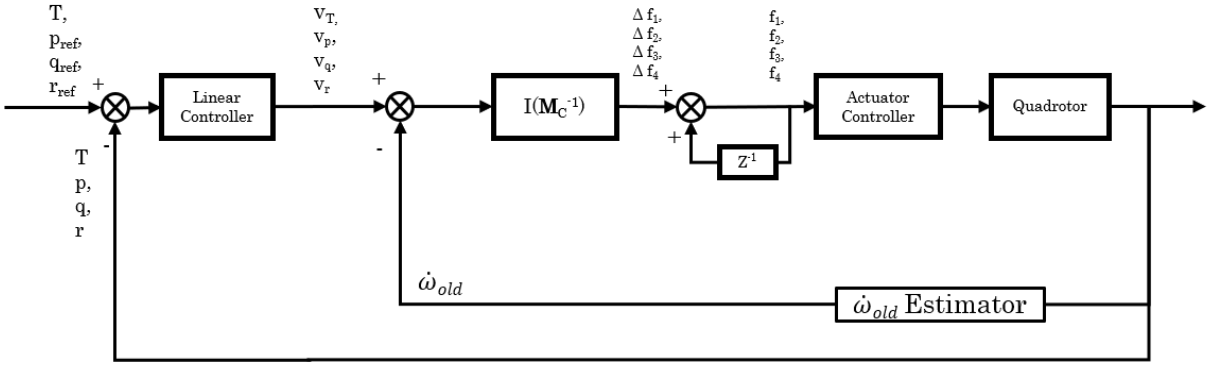

Figure 6: Control diagram for Incremental NDI control

The INDI method was proven to be effective at controlling the attitude and position of a quadrotor. Application of this method on damaged quadrotors was done by Lu & van Kampen [12]. Lu & van Kampen showed that a Fault Detection and Isolation module can effectively be implemented with the primary axis control strategy proposed by Lanzon *et al.* and Mueller & D'Andrea. The control architecture was switched to an Incremental Nonlinear Dynamic Inversion (INDI)+primary axis controller after detection of a failure.

The paper by Lu & van Kampen [12] used feedback linearization for the position controller for the outer loop. The inversion process requires the (impossible) inversion of the singular B matrix in eq. (13). To circumvent this problem the equation was rewritten into:

$$\begin{bmatrix} \dot{n}_x \\ \dot{n}_y \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -n_z \\ n_z & 0 \end{bmatrix}}_{B_2} \begin{bmatrix} p \\ q \end{bmatrix} + \begin{bmatrix} n_y \\ -n_x \end{bmatrix} r \tag{20}$$

Inverting B_2 gives the desired p and q for the inner loop control.

The INDI method only requires the effectiveness coefficients of the actuators in $\mathbf{M_c}$ in eq. (18). This means any inaccuracies that were within $\boldsymbol{M}_a$ are mitigated. Unfortunately, this method is still sensitive to deformations in the $\mathbf{M_c}$ matrix. An online adaptation of the $\mathbf{M_c}$ could solve this.

2.4.5 Adaptive INDI

The control effectiveness can physically change with time. This can be due to unexpected faults such as blade deformations or structural damage. Another significant one may the power supplied by the battery which fluctuates with duration and temperature. Such changes in the control effectiveness may be slow changing or abrupt. Either way, by applying an online parameter identification these changes can be estimated. The result is a time-varying $\mathbf{M_c}$ in eq. (19).

The online parameter identification takes the input and the output of the system to estimate the parameters that would fit this system. A simple algorithm that accomplishes this is the Least Squares Parameter estimation. This estimation method was also implemented in the Adaptive INDI controller by Smeur, Chu & Croon. [37] Its adaptive abilities were tested by flying with and without extra aerodynamic drag in the

form of 'bumpers'. Smeur *et al.* continued in [38] by showing that the gust disturbance rejection of the INDI method on a healthy quadrotor was significantly improved compared with the classical PID-control.

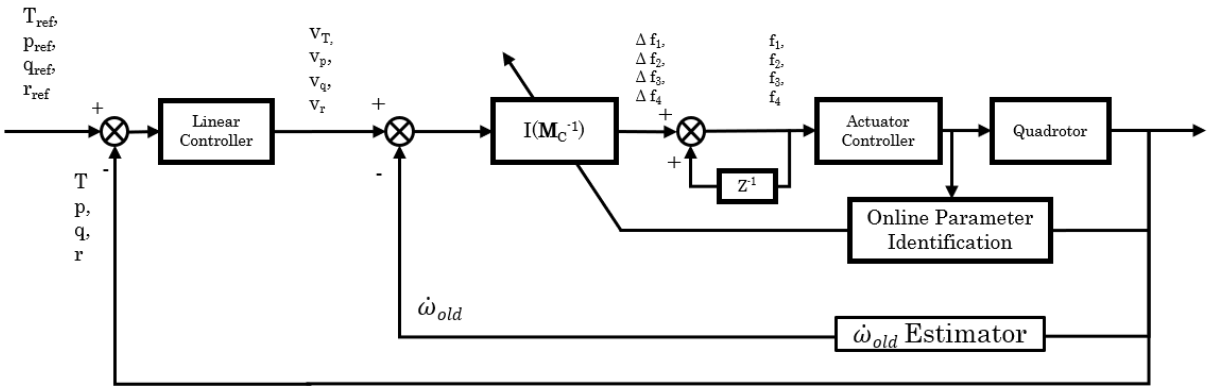

Figure 7: Control diagram for Adaptive INDI control

A disadvantage of adaptive methods is that during initialization the estimated parameters might be incorrect which results in unattainable control commands. The actuators have physical limits which should be safeguarded during flight. Furthermore, after a rotor failure the remaining actuators have to work harder and are thus closer to their physical limits. Control allocation methods can be used to prevent these problems.

2.5 Control allocation after rotor failures

Control allocation is used to translate the required torques into forces which should be produced by the actuators. By inverting the control effectiveness matrix these can easily be calculated. Rotor failure makes this inversion process more difficult. This section is aimed at finding a solution for the inversion problem as well as providing a prevention method for actuator saturation.

2.5.1 Modeling actuators

The Bebop 2's four rotors are actuated by brushless direct current (BLDC) motors and Pulse-Width Modulation controllers. An added advantage of BLDC motors is a high-frequency accurate RPM readout of all motors. [2] The actuators dynamics do not follow the reference signal instantly and can typically be modeled as a first-order system where the rotors' actual and commanded angular velocity are:

$$\omega_{actual_i} = \frac{1}{\tau s + 1}\omega_{commanded_i} \tag{21}$$

with τ a time-constant. [39] The constraints on angular velocity are:

$$\begin{cases} \omega_{actual_i} \geq \omega_{min}, \forall i \\ \omega_{actual_i} \leq \omega_{max}, \quad \forall i \end{cases} \tag{22}$$

The rotors themselves can be modeled as rigid-bodies as well. This simplifies the mathematical model for control purposes, but opens up potential disturbances such as rotor flapping. [11] [29].

[2] https://www.parrot.com/us/spareparts/parrot-bebop-2-motor-kit# parrot-bebop-2-motor-kit-details

The thrust force f_i of each rotor in hover can simply be modeled as a linear relation with angular velocity as eq. (23). [29]

$$f_i = C_f \omega_i^2 \tag{23}$$

Likewise, the resultant torque τ_i has a linear relation with the force. According to the rotation of the rotor the torque is either positive (clockwise) or negative (counterclockwise); this is modeled through the $(-1)^{1+i}$ term. [14]

$$\tau_i = (-1)^{i+1} C_k f_i \tag{24}$$

The constants C_f and C_k can be determined offline through static tests.

The 4 rotors can influence the total force u_f and the torques on the body $\boldsymbol{\tau}$ through the control allocation. This allocation is derived based upon Figure 4.

A pitch up maneuver is achieved by having differential thrust in rotors 3+4 and rotors 1+2. Likewise, roll is performed between differential thrust in rotors 1+3 and rotors 2+4. Yaw control is obtained by difference in rotation direction of the rotors; clockwise rotating rotors exert a counterclockwise torque on the body frame and vice versa.

Given d_1, d_2, l_1 and l_2 for the moment arms and the yaw torque coefficient C_k there is a control allocation from rotor force f_i to the total force u_f and the torques $\boldsymbol{\tau}$ presented in the form of $\boldsymbol{m} = \boldsymbol{\Gamma u}$ in eq. (25).

$$\begin{bmatrix} u_f \\ \tau_p \\ \tau_q \\ \tau_r \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ d_1 & -d_1 & -d_2 & d_2 \\ l_1 & l_1 & -l_2 & -l_2 \\ C_k & -C_k & C_k & -C_k \end{bmatrix}}_{\Gamma} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{25}$$

A steady hover would require $u_f = mg$ and $\tau_p, \tau_q, \tau_r = 0$.

From eq. (25) it becomes apparent that full attitude control is not possible if either of the actuators is not fully working; Γ would not be full rank. Finding a solution to the control allocation problem is done by inverting matrix Γ in eq. (25). More on this in section 2.5.2.

Furthermore for altitude control, losing actuators also means:

$$u_f \leq (4 - \#\text{ failed rotors}) \cdot f_{max} \tag{26}$$

which pushes the remaining actuators closer to saturation. Saturation of the actuators may lead to oscillations or instability and should be avoided at all times.

A double rotor-failed quadrotor will not be able to handle the same control objectives that a healthy quadrotor can. This makes the damaged quadrotor even more susceptible to actuator saturation. A strategy should be implemented that prioritizes stability over performance. Several possible solutions are explored here.

The first method is aimed at preventing the system from relaying unattainable desired moments into unattainable actuator forces (and thus saturations happen). This can be accomplished by tailoring the inversion matrix for Γ to include the saturation limits. This is explained in section 2.5.2.

The second method, in section 2.5.3,tries to prevent the outer loop from desiring unattainable moments. This method is called Pseudo-Control Hedging.

The last method, in section 2.5.4, involves inserting the actuator dynamics into the system as part of the INDI loop or through filtering the INDI output.

2.5.2 Generalized Inverses and the Moore-Penrose Pseudo-inverse

In the nominal quadrotor case the Γ-matrix is a square nxn. By inverting Γ the explicit solution is derived for the controls. After rotor-failure the Γ-matrix becomes a 'tall' nx$(n-1)$-matrix. Tall matrices have no inverse and thus no exact solution. Generalized inverses, also called pseudo-inverses, are an approximation for this inversion process. The derivations for the generalized inverse and Moore-Penrose pseudo-inverse are cited, and slightly adapted to fit this thesis, from the book *Aircraft Control Allocation* by Durham, Bordignon & Beck. [40]

Generalized inverse A generalized inverse is a matrix P (m x n) that together with control effectiveness matrix Γ (n x m) produces an identity matrix of size nxn. The generalized inverse present a

$$\Gamma P = I \tag{27}$$

and

$$\boldsymbol{u}_p = P\boldsymbol{m}_{des} \tag{28}$$

such that

$$\Gamma \boldsymbol{u}_p = \boldsymbol{m}_{des} \tag{29}$$

with $\boldsymbol{u}_p = [f_1, f_2, f_3, f_4]^T$ the control force and the desired moments $\boldsymbol{m}_{des} = [u_f, \tau_p, \tau_q, \tau_r]^T$.

All such generalized inverses may be represented as

$$P = N\left(\Gamma N\right)^{-1} \tag{30}$$

with arbitrary N.

A generalized inverse matrix P does not hold any information on the constraints of the actuators or the efficiency of the solution. In the process of deriving matrix P it is possible to select certain properties. One such properties is to minimize the control vector $\boldsymbol{u}$. This can be done through enforcing a norm-function on $\boldsymbol{u}$ such that it is minimum. A popular norm is the Euclidean norm, or 2-norm, which applied to a generalized inverse will produce the Moore-Penrose pseudo-inverse.

In general the Moore-Penrose pseudo-inverse is used for overactuated systems. The derivation given by Durham, Bordignon & Beck is no different. However, it can also be used for underactuated systems. This was shown by Bojadah who applied the Moore-Penrose pseudo-inverse for feedback linearization of an underactuated spacecraft. [41]

Moore-Penrose pseudo-inverse Minimize $\boldsymbol{u}^T\boldsymbol{u}$ subject to $B\boldsymbol{u} = \boldsymbol{m}$.
Using LaGrange multipliers we define the scalar function

$$\mathcal{H}\left(\boldsymbol{u}, \boldsymbol{\lambda}\right) = \frac{1}{2}\boldsymbol{u}^T\boldsymbol{u} + \boldsymbol{\lambda}^T\left(\boldsymbol{m} - B\boldsymbol{u}\right) \tag{31}$$

Here $\boldsymbol{\lambda}$ is a n-vector of LaGrange multipliers. $\mathcal{H}$ will be a minimum when

$$\frac{\delta \mathcal{H}}{\delta \boldsymbol{u}} = 0, \frac{\delta \mathcal{H}}{\delta \boldsymbol{\lambda}} = 0 \tag{32}$$

which yields

$$\frac{\delta \mathcal{H}}{\delta \boldsymbol{u}} = \boldsymbol{u}^T - \boldsymbol{\lambda}^T B = 0 \tag{33}$$

$$\frac{\delta \mathcal{H}}{\delta \boldsymbol{\lambda}} = \boldsymbol{m} - B\boldsymbol{u} = 0 \tag{34}$$

From eq. (33) it follows that

$$\boldsymbol{u} = B^T \boldsymbol{\lambda} \tag{35}$$

eq. (35) inserted into eq. (34), will yield

$$\boldsymbol{m} = B\boldsymbol{u} = BB^T \boldsymbol{\lambda} \tag{36}$$

Since BB^T is square it is also invertible. Thus

$$\boldsymbol{\lambda} = \left(BB^T\right)^{-1} \boldsymbol{m} \tag{37}$$

Replacing $\boldsymbol{\lambda}$ in eq. (35) with eq. (37) gives

$$\boldsymbol{u} = B^T \left(BB^T\right)^{-1} \boldsymbol{m} \tag{38}$$

This yields the P matrix from eq. (28) with $N = B^T$.

Weighted Moore-Penrose pseudo-inverse The Moore-Penrose pseudo-inverse gives the minimum vector length $\boldsymbol{u}$, but this does not guarantee that the control allocation is within the saturation bounds as given by eq. (22). A weight matrix W can be inserted in the derivation of the pseudo-inverse to try to enforce certain constraints. This would lead to the minimization of $\boldsymbol{u}^T W^T W \boldsymbol{u}$. Following a similar derivation as for the unweighted Moore-Penrose pseudo-inverse

$$P = W^T B^T \left(BW^T B^T\right)^{-1} \tag{39}$$

Although the weighting matrix W will help in avoiding saturation it cannot guarantee that the constraints are met. One is a linear scaling of the control vector.

The most saturated element in the control vector $\boldsymbol{u}$ may be scaled down such that it is nearly saturated. The direction of the control vector can be preserved by applying this same scaling factor for all elements.

2.5.3 Pseudo Control Hedging

The generalized inverses do not guarantee that the actuators do not saturate. The outer loop can still demand unattainable control forces given the circumstances. Maintaining physically possible control commands requires the prioritization of stability over performance. A more recent solution that does this is Pseudo-Control Hedging (PCH).

Pseudo-Control Hedging was first proposed in 2000 by Johnson, Calise, El-Shirbiny & Rysdyk to prevent an adaptive controller to inadvertently give incorrect reference commands which may lead to actuator saturation. [42] The term 'hedging' is used in the financial industry as a synonym for insurance or protection; in a likewise fashion the virtual control signal (or pseudo-control) is enhanced to protect against saturation.

This is done through computing a reference model (RM). The RM calculates the part that is unattainable by the actuators and subtracts it from the unattainable input signal. The added benefit is that actuator wind-up is negated as the control signal is achievable by the actuators.

Johnson & Kannan went on to show the use of PCH in the outer-loop position control for autonomous helicopters. [43] Futhermore, the bandwidth of the outer-loop can be made closer to alleviate timescale separation requirements. [44] Simplício, Pavel, van Kampen & Chu showed the PCH method can be effectively combined with INDI by applying it on a helicopter based UAV. [45] This shows the PCH method is a promising method in preventing actuator saturation on a quadrotor as well.

The pseudo-control from the reference model is computed as [45]

$$\boldsymbol{\nu}_{rm} = K_{rm} \left(\boldsymbol{\Omega}_{ref} - \boldsymbol{\Omega}_{rm} \right) \tag{40}$$

with K_{rm} being a diagonal gain matrix. Together with the virtual control hedge

$$\boldsymbol{\nu}_h = \boldsymbol{I}^{-1} \boldsymbol{M}_c \left(\boldsymbol{u}_{ref} - \boldsymbol{u}_{actual} \right) \tag{41}$$

When no saturation occurs the $(\boldsymbol{u}_{ref} - \boldsymbol{u}_{actual}) = 0 \implies \boldsymbol{\nu}_h = 0$. This makes the RM behave like a low-pass filter with bandwidth K. Subsequently, the reference model produces a reference signal with a hedge for actuator saturation.

$$\boldsymbol{\Omega}_{rm} = \frac{1}{s} \left(\boldsymbol{\nu}_{rm} - \boldsymbol{\nu}_h \right) \tag{42}$$

Expanding the control diagram in fig. 7 with the PCH results in section 2.5.3.

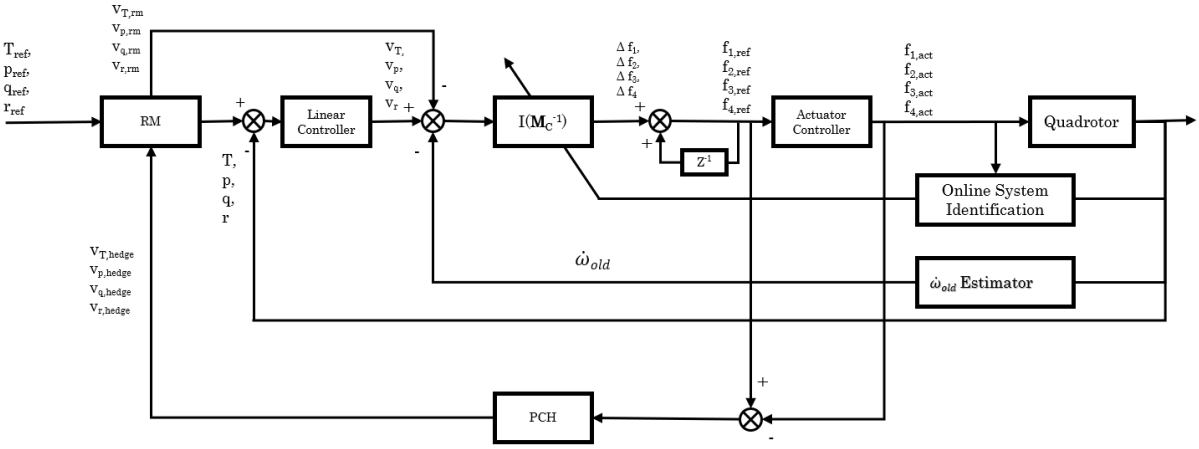

Figure 8: Control diagram for Adaptive INDI with Pseudo-Control Hedge

2.5.4 Command Filtering

The Pseudo-Control Hedging method seems promising, but practical implication might be constrained by the thesis project time. A simpler solution would be to incorporate the actuator dynamics. Two possible applications exist with each their own merits.

Applying actuator dynamics into the INDI controller One way is to insert the actuator dynamics into the INDI controller. The INDI can then better control the spin-up of the actuators which should give better response. An example is given below.

As was mentioned section 2.5 the angular accelerations are related to the current angular velocities and the rotor speeds:

$$\begin{bmatrix} \int \dot{T} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} \int T \\ p \\ q \\ r \end{bmatrix} + \boldsymbol{B} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \tag{43}$$

with $\boldsymbol{A}$ approaching 0 during the infinitesimal time step and

$$\boldsymbol{B} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ d_1 & -d_1 & -d_2 & d_2 \\ l_1 & l_1 & -l_2 & -l_2 \\ C_k & -C_k & C_k & -C_k \end{bmatrix} \tag{44}$$

The required rotor angular velocities are acquired by inverting $\boldsymbol{B}$. This can be augmented with the rotors' angular accelerations according to the actuators dynamics as given by eq. (21).

$$\begin{bmatrix} \int \dot{T} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{T} \end{bmatrix}}_{G_1} \begin{bmatrix} \int T \\ p \\ q \\ r \\ \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\tau} & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix}}_{G_2} \begin{bmatrix} \omega_{1_{ref}} \\ \omega_{2_{ref}} \\ \omega_{3_{ref}} \\ \omega_{4_{ref}} \end{bmatrix} \tag{45}$$

The control allocation in $\boldsymbol{B}$ is moved to the first part of the right side of the equation. The actuator time constants are given by $\boldsymbol{T} = diag(-\frac{1}{\tau}, -\frac{1}{\tau})$.

The downside of this method is that during the infinitesimal time step the G_1 matrix is no longer negligible and should be part of the inversion process together with matrix G_2.

Applying a filter to the reference command The INDI output is fed into the actuator driver and is subject to the actuator dynamics. The rotors are not capable of an instantaneous response that is required. A first- or second-order filter can be applied to the reference command as supplied by the INDI controller. This filter can have the same structure as the actuator dynamics as given by eq. (21).

2.6 This research

This research will expand the INDI technique to control double-rotor failed quadrotors. It will use the INDI control technique as this is better able to reject disturbances for more states than a linear controller would be able to. This INDI controller will control the reduced attitude, or *primary axis*, to enable control for total rotor failure. The combination of the INDI controller and a double-rotor failure is completely novel. The double-rotor failure is expected to make the inversion process difficult. Furthermore, past research is dedicated to proper symmetrical high-performance quadrotors. This research will show that the designed control techniques are also viable for asymmetrical commercially-off-the-shelf quadrotors.

A expected result of the two-rotor impaired quadrotor is its inability to fully control all the states. Therefore, a prioritization for a minimal velocity error must be made such that potential crashes will have minimal

impact.

The resulting flying quadrotor can be tested in order to estimate how the flight envelope shrinks when compared with the nominal case. This allows the improvement of the safety of quadrotors by having better models and a more complete understanding about how quadrotors behave. A more detailed overview of the research objectives is given section 3.

3 Research Plan

The literature review has showed the historical background as well as state-of-the-art regarding fault tolerant control for quadrotors. The research plan can be defined with that knowledge.

The research objective is to:
Design an INDI controller to control a two-rotor failed quadrotor and evaluate its control and stability by executing test flights in the Cyberzoo.

3.1 Research Question

To achieve the stated research objective the following research question must be answered:

RG 1 How can we expand the current INDI + primary axis method to allow a 2-rotor failed quadrotor to minimize the deviation from its intended velocity vector?

The main research question is motivated by the idea that the quadrotor will not be capable of full control when 2 rotors are damaged and that it prioritizes a safe landing such that impact damage will be minimal.

This main research question is answered by answering the sub-questions. The first and second sub-question focus on potential pitfalls and obstacles in designing a controller whilst the third sub-question focuses on the achievable performance.

RQ 1.1 What are the bottlenecks in expanding the current INDI controller for the two opposite rotor fail case?

RQ 1.2 What are the bottlenecks in expanding the current INDI controller for the two adjacent rotor fail cases?

RQ 1.3 What kind of tracking performance can the 2-rotor damaged quadrotor achieve for hover and forward flight?

Several objectives and milestones have to be reached in order to answer the research questions. These are summarised in section 3.2.

3.2 Research Objectives and Milestones

1. Design phase
   (a) Investigate a way to expand current controller to handle a double-rotor fail cases
   (b) Identify the technical feasibility of double-rotor fail cases
   (c) Verify controller for various failures in Matlab/Simulink simulation environment
2. Implementation phase
   (a) Synthesis of software in C language
   (b) Debug, tune and validate controller on quadrotor in Cyberzoo
3. Testing phase
   (a) Test quadrotor for hover situation
   Test hover with disturbances

(b) Test quadrotor for a waypoint-follow situation
Test waypoint-follow with disturbances

4. Post-test Analysis
   (a) Data clean up/filtering/normalizing
   (b) Write paper

4 Methodology

The methods that shall be used to achieve the objectives, as stipulated in section 3.2, are explained in this section.

Design phase The design phase consists of building the current working controller in the Matlab/Simulink environment. Here, the new controller can be safely tested and verified. The inversion problem will be investigated using pseudo-inverse. Saturation problems are experimented on using the weighted pseudo-inverses, Pseudo-Control Hedging and command filtering.

Implementation phase TU Delft owns several commercially-off-the-shelf quadrotors which are used for research purposes: Parrot ARdrone, Bebop and Bebop 2. In this research the Parrot Bebop 2 (fig. 1) shall be used to conduct flight tests.

The flights will be performed within the indoor 'Cyberzoo'. This is a 10mx10mx10m cube dedicated to safely conduct UAV flight tests. It separates the vehicles from any persons using nets. The ground is covered with protective mats to soften the inevitable crashes.

Moreover, the Cyberzoo has an integrated position tracking system which can be used instead of a Global Positioning System. Preferably, the controller may rely on the on-board sensors making it capable for full autonomous flying, but these signals may be too noisy and biased.

Testing phase Testing will be done in the aforementioned Cyberzoo. Here the quadrotor is tested for hovering and flight. Additional disturbances can be added to test for robustness. Disturbances can consist of, for example, a high-powered industrial fans that simulate gusts of wind. Several of such fans are available at the Cyberzoo. These can be positioned to blow from an arbitrary direction as the quadrotor tries to hold its position or reach its waypoint.

Post-test analysis The data will be gathered using the 3D position tracking system. Furthermore, the quadrotor has several sensors which can be logged online and be gathered after flying. Combining the position tracking system's and on-board sensors' data results in an accurate and complete measurement set.

5 Conclusion

The proposed research aims at developing a nonlinear Fault Tolerant Control in order to control a double-rotor failed quadrotor. Past research showed it was already possible to do this using linear control methods. A nonlinear controller allows the estimation of the damaged quadrotor model for various states and disturbances.

The Incremental Nonlinear Dynamic Inversion (INDI) method allows the on-board sensors to be used to establish the required control forces. The only model still required is the control effectiveness which can be made adaptive using Least Squares.

The INDI method applied to a rotor-failed quadrotor requires the adoption of a Reduced Attitude control strategy called the Primary axis control. The requirement for yaw control is relaxed in order to control the other states. Furthermore, the inversion required for the INDI process is impossible if a rotor is removed. The use of pseudo-inverses can provide a best-possible solution in that case.

This research uses a commercially-off-the-shelf quadrotor which, when subjected to rotor failures, is sensitive to actuator saturations. The added benefit of using pseudo-inverses is that it can be tailored to try to prevent saturation from occurring. However, this is not a guarantee. Pseudo Control Hedging may be used to prevent the system of unattainable control efforts or perhaps a simpler solution in the form of command filtering will suffice.

A two-rotor impaired quadrotor is incapable of full control. As such this research asks the question: *How can we expand the current INDI + primary axis method to allow a 2-rotor failed quadrotor to minimize the deviation from its intended velocity vector?* In order to answer this question along with its sub-questions, the designed controller will be implemented on a quadrotor and tested for position control while experiencing disturbances.

The result of this research can be used to identify new damaged quadrotor models which can subsequently be used for flight envelope estimation.

References

[1] Duc-tien Nguyen David Saussi. Robust Self-Scheduled Fault-Tolerant Control of a Quadrotor UAV. pages 5926–5932, 2017.

[2] Moses Bangura and Robert Mahony. Thrust Control for Multirotor Aerial Vehicles. *IEEE Transactions on Robotics*, 33(2):390–405, 2017.

[3] J. Escareno, A. Sanchez, O. Garcia, and R. Lozano. Triple tilting rotor mini-UAV: Modeling and embedded control of the attitude. *Proceedings of the American Control Conference*, (1):3476–3481, 2008.

[4] Majd Saied, Benjamin Lussier, Isabelle Fantoni, Clovis Francis, Hassan Shraim, and Guillaume Sanahuja. Fault diagnosis and fault-tolerant control strategy for rotor failure in an octorotor. *Proceedings - IEEE International Conference on Robotics and Automation*, 2015-June(June):5266–5271, 2015.

[5] Farid Sharifi. Fault Tolerant Control of a Quadrotor UAV using Sliding Mode Control. pages 239–244, 2010.

[6] Hicham Khebbache, Belkacem Sait, Fouad Yacef, and Yassine Soukkou. Robust Stabilization of A Quadrotor Aerial Vehicle in Presence of Actuator Faults. *International Journal of Information Technology, Control and Automation*, 2(2):1–13, 2012.

[7] I. Sadeghzadeh, A. Mehta, A. Chamseddine, and Youmin Zhang. Active Fault Tolerant Control of a Quadrotor Uav Based on Gain- Scheduled Pid Control. *Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on*, pages 1–4, 2012.

[8] Y. M. Zhang, A. Chamseddine, C. A. Rabbath, B. W. Gordon, C. Y. Su, S. Rakheja, C. Fulford, J. Apkarian, and P. Gosselin. Development of advanced FDD and FTC techniques with application to an unmanned quadrotor helicopter testbed. *Journal of the Franklin Institute*, 350(9):2396–2422, 2013.

[9] Qiang Shen, Danwei Wang, Senqiang Zhu, and Eng Kee Poh. Fault-Tolerant Attitude Tracking Control for a Quadrotor Aircraft. *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 6129–6134, 2014.

[10] Abdel Razzak Merheb, Hassan Noura, and Francois Bateman. Design of Passive Fault-Tolerant Controllers of a Quadrotor Based on Sliding Mode Theory. *International Journal of Applied Mathematics and Computer Science*, 25(3):561–576, 2015.

[11] Alexander Lanzon, Alessandro Freddi, and Sauro Longhi. Flight Control of a Quadrotor Vehicle Subsequent to a Rotor Failure. *Journal of Guidance, Control, and Dynamics*, 37(2):580–591, mar 2014.

[12] Peng Lu and Erik Jan Van Kampen. Active fault-tolerant control for quadrotors subjected to a complete rotor failure. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2015-Decem, 2015.

[13] Abdel Razzak Merheb, Hassan Noura, and Francois Bateman. Emergency control of AR drone quadrotor UAV suffering a total loss of one rotor. *IEEE/ASME Transactions on Mechatronics*, 22(2):961–971, 2017.

[14] Mark W. Mueller and Raffaello D'Andrea. Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 45–52, 2014.

[15] De Crousaz, Farbod Farshidian, Michael Neunert, and Jonas Buchli. Unified Motion Control for Dynamic Quadrotor Maneuvers Demonstrated on Slung Load and Rotor Failure Tasks. (ICRA):2223–2229, 2015.

[16] Rolf Isermann. *Fault-diagnosis systems: An introduction from fault detection to fault tolerance*. Springer, Berlin, 2006.

[17] Guillaume Ducard, J.J. *Fault-tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*. Springer, London, 2009.

[18] H.M. Smaili, T. J. J. Lombaerts, Christopher Edwards, Michel Verhaegen, Stoyan Kanev, Redouane Hallouzi, Colin Jones, Jan MacIejowski, and Hafid Smail. *Fault tolerant flight control - A survey*, volume 399. 2010.

[19] Jakob Stoustrup and Vincent D. Blondel. Fault Tolerant Control: A Simultaneous Stabilization Result. *IEEE Transactions on Automatic Control*, 49(2):305–310, 2004.

[20] Yue Wang, Jun Pan, Bin Jiang, and Ning-yun Lu. Hybrid Model based Fault Tolerant Control for Quadrotor Helicopter with Structural Damage. pages 5933–5938, 2016.

[21] Abdel Razzak Merheb, Francois Bateman, Hassan Noura, and Younes Al Younes. Hierarchical fault-tolerant control of a quadrotor based on fault severity. *Conference on Control and Fault-Tolerant Systems, SysTol*, 2016-Novem:666–671, 2016.

[22] Cen Zhaohui and Hassan Noura. A composite Fault Tolerant Control based on fault estimation for quadrotor UAVs. *Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications, ICIEA 2013*, pages 236–241, 2013.

[23] M. W. Mueller and R. DAndrea. Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles. *The International Journal of Robotics Research*, 35(8):873–889, 2016.

[24] Hasan Basak and Emmanuel Prempain. Switching recovery control of a quadcopter UAV. *2015 European Control Conference, ECC 2015*, (2):3641–3646, 2015.

[25] Vincenzo Lippiello, Fabio Ruggiero, and Diana Serra. Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach. *IEEE International Conference on Intelligent Robots and Systems*, (September):4782–4788, 2014.

[26] R J Montoya, W E Howell, W T Bundick, Aaron J Ostroff, and R M Hueschen. Restructurable Controls. *NASA Conference Publication 2277*, 1983.

[27] R. F. Stengel. Intelligent Failure-Tolerant Control. IEEE Control Systems, 1990.

[28] Youmin Zhang and Jin Jiang. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, 32(2):229–252, 2008.

[29] Robert Mahony, Vijay Kumar, and Peter Corke. Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor. *IEEE Robotics & Automation Magazine*, 19(3):20–32, 2012.

[30] Peng Lu, Erik-Jan van Kampen, Cornelis de Visser, and Qiping Chu. Aircraft fault-tolerant trajectory control using Incremental Nonlinear Dynamic Inversion. *Control Engineering Practice*, 57:126–141, 2016.

[31] Sammy Omari, Minh Duc Hua, Guillaume Ducard, and Tarek Hamel. Hardware and software architecture for nonlinear control of multirotor helicopters. *IEEE/ASME Transactions on Mechatronics*, 18(6):1724–1736, 2013.

[32] Weixuan Zhang, Mark W. Mueller, and Raffaello D'Andrea. A controllable flying vehicle with a single moving part. *Proceedings - IEEE International Conference on Robotics and Automation*, 2016-June:3275–3281, 2016.

[33] Vincenzo Lippiello, Fabio Ruggiero, and Diana Serra. Emergency landing for a quadrotor in case of a propeller failure: A PID approach. *IEEE International Conference on Intelligent Robots and Systems*, (August 2015):4782–4788, 2014.

[34] Jean-Jacques Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice-Hall Inc, New Jersey, 1991.

[35] Aaron J. Ostroff and Barton J Bacon. Enhanced NDI strategies for reconfigurable flight control. *Proceedings of the 2002 American Control Conference*, pages 3631–3636, 2002.

[36] S. Sieberling, Q. P. Chu, and J. A. Mulder. Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction. *Journal of Guidance, Control, and Dynamics*, 33(6):1732–1742, 2010.

[37] 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*, 39(3):450–461, 2015.

[38] E. J. J. Smeur, Guido C. H. E. de Croon, and Qiping Chu. Gust Disturbance Alleviation with Incremental Nonlinear Dynamic Inversion. 2016.

[39] Peng Lu, Erik-jan Van Kampen, and Qiping P Chu. Advances in Aerospace Guidance, Navigation and Control. pages 81–98, 2015.

[40] Roger Durham, Wayne; Bordignon, Kenneth A.; Beck. *Aircraft Control Allocation*. John Wiley & Sons, 2017.

[41] Abdulrahman H. Bajodah. Stabilization of Underactuated Spacecraft Dynamics Via Singularly Perturbed Feedback Linearization. *IFAC Proceedings Volumes*, 40(7):377–382, 2007.

[42] Eric Johnson, Anthony Calise, Hesham El-Shirbiny, and Rolf Eysdyk. Feedback linearization with Neural Network augmentation applied to X-33 attitude control. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2000.

[43] EN Johnson and SK Kannan. Adaptive flight control for an autonomous unmanned helicopter. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2002.

[44] Eric N. Johnson and Suresh K. Kannan. Adaptive Trajectory Control for Autonomous Helicopters. *Journal of Guidance, Control, and Dynamics*, 28(3):524–538, 2005.

[45] P. Simplício, M. D. Pavel, E. van Kampen, and Q. P. Chu. An acceleration measurements-based approach for helicopter nonlinear flight control using incremental nonlinear dynamic inversion. *Control Engineering Practice*, 21(8):1065–1077, 2013.

Part III

Appendices

Appendix A

Adjacent Double Rotor Failures

In this chapter the solutions will be explored when two adjacent — i.e. counter-rotating — rotors simultaneously fail on the quadrotor. Even though the quadrotor used in this research is not fully symmetric, it is assumed here that there is a generality between all possible permutations of failures. For visualization and readability it is assumed in this chapter that the back rotors have failed, rotors 3 and 4. This means the vehicle is only capable of producing a positive pitch rate.

A-1 Yaw rate relaxation

The strategy for single and diagonal rotor failures can be applied to the adjacent case: to no longer control yaw rate but instead generate a stable and periodic hover solution using a constant rotor force.

Finding a solution with constant rotor force for the adjacent rotor case creates some problems. Equal thrust by the two remaining front rotors (# 1 and # 2) generates no yaw torque and thus the yaw rate will decrease due to the aerodynamic drag. This eventually causes the system to flip and crash. Therefore, at least some yaw torque is required.

A maximum yaw rate is reached when only one rotor is producing force. This also means that the single rotor should counteract gravitational forces. Not desirable when the goal is to minimize vertical velocity. Furthermore, a single rotor is not powerful enough if the quadrotor is not specifically built for this case [1]. A differential thrust to rotors 1 and 2 whereby there still exist some yaw torque is desirable.

In the study by Mueller & D'Andrea it was shown that the periodic solution for 2 adjacent rotor failure is similar to the three rotors failed. The ratio of thrust for two rotors was established to be approximately $\frac{\omega_1}{\omega_2} = 0.2$ [2].

A-2 Pitch rate relaxation

Another solution exists whereby the thrust from the rotors equal each other but are periodic in nature. This strategy makes the system no longer time-invariant in the body-fixed frame. A simple strategy would be to apply full thrust when a thrust component works against gravity; the upward phase of the full rotation. This also induces a positive pitch rate. Zero thrust would be applied on the downward phase of the rotation. Pitch rate damping prevents the vehicle from increasing its pitch rate *ad infinitum* and forces an equilibrium periodic pitch rate.

The downward velocity should be reduced in order to limit crash impact. This depends largely on the actuator dynamics and aerodynamic friction. As the pitch rate increases, so does the frequency of the pulses the actuators need to generate. An equilibrium pitch rate will be established due to aerodynamic friction.

The ability of the system to hold level, or at least reduce the fall velocity, can be estimated with a simulation. A simplified controller can be constructed using an open-loop strategy on the pitch angle θ. This is presented in eq. (A-1). This produces an average acceleration in the negative z-direction which works against gravity.

$$\omega_{rotors} = \begin{cases} MAX, & \text{if } -90^\circ < \theta < 90^\circ \\ 0, & \text{otherwise} \end{cases} \tag{A-1}$$

In a simulation it can be shown that the current quadrotor vehicle is not capable of reducing its vertical velocity. This is presented in fig. A-2. A comparison with free fall is difficult as the simulation model does not incorporate velocity drag. From a qualitative point of view it is still interesting to see how much better the rotors must perform in order to stabilize. Therefore, the actuator dynamics are made faster by reducing the time constant. These faster dynamics are also plotted in fig. A-2.

The pitch rate equilibrium shown in fig. A-3 is also increased when the rotors are more instantaneous. This is to be expected as the maximum thrust force is achieved earlier in the rotation of the body.

The corresponding accelerations are visualized in fig. A-1. The average acceleration for the different time constants are plotted as a dotted line. It becomes apparent that increasing actuator dynamics will not result in a negative average acceleration. The maximum rotational rate of the rotors does not provide enough thrust on one half of the cycle to overcome the gravitation that works on the full cycle.

In fig. A-4 the effect of maximum rotor speed and actuator dynamics on vertical velocity are plotted. It becomes apparent that multiple combinations of actuator dynamics and max rotor speed can results in a continuous negative velocity. For example, with 40 times faster actuator dynamics only a max rotor speed increase of 75 % required whereas a 25 times faster dynamics requires a 150 % increase.

High performance motors are capable of providing between 3459 and 3789 [rad/s][1] which is an increase of 188 % with respect to the Bebop 2 motors so that shows the increased rotor

[1] `https://www.miniquadtestbench.com/assets/components/motordata/motorinfo.php?uid=217`

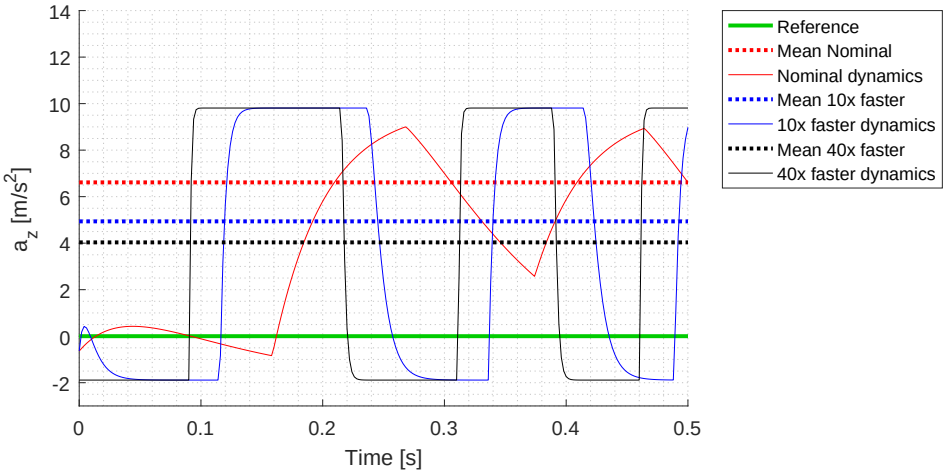

Figure A-1: The effect of different of actuator dynamics on acceleration output.

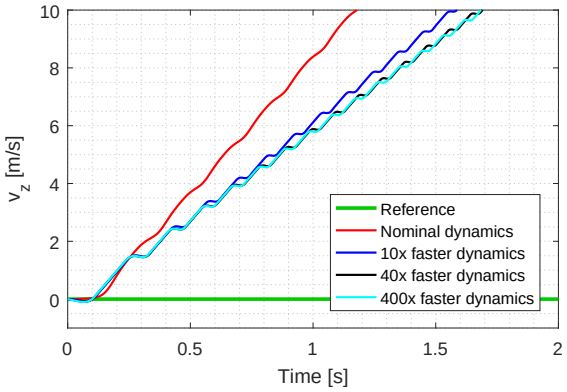

Figure A-2: The vertical velocity for different actuator dynamics.

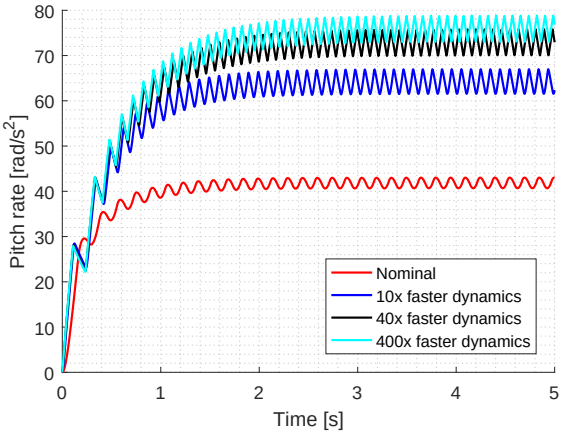

Figure A-3: Pitch rate equilibrium for different actuator dynamics.

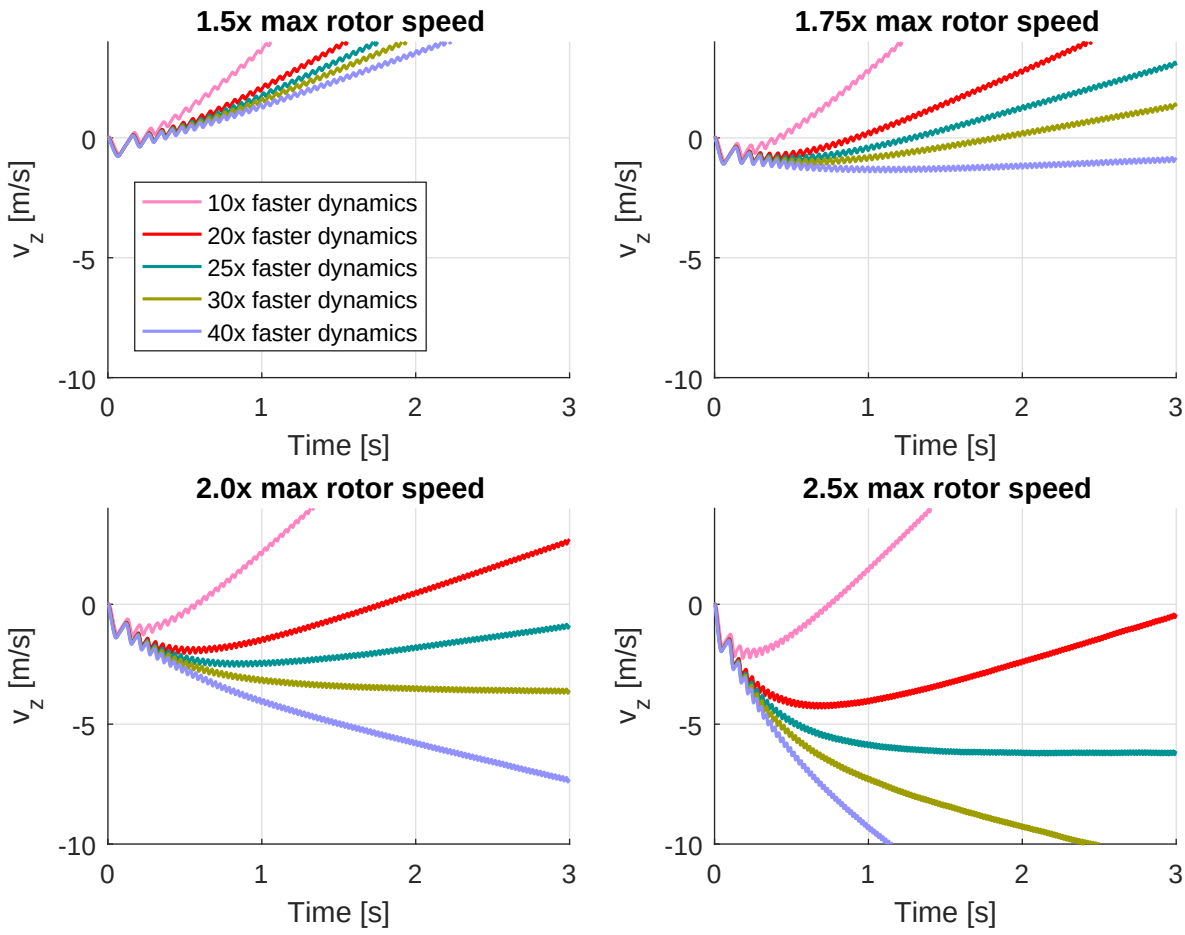

Figure A-4: The effect of different of actuator dynamics and max rotor speeds on velocity output.

speed requirement is in the realm of possibilities. Unfortunately, no information regarding actuator dynamics for these motors are found. The yaw rate solution would be preferred in the case when actuator dynamics are the determining factor as this solution does not require improved dynamics.

A-3 Conclusion

The adjacent double rotor failure case can be split up into two different strategies whereby either the yaw rate or the pitch/roll rate suffers degradation.

The primary axis method can be used by applying an unequal but constant thrust force for both rotors. This allows the quadrotor to stabilize itself whilst having a non-zero yaw rate. The rotors on the quadrotor cannot produce enough force in real flight. Other research has already done thorough simulations to establish a solution.

A solution whereby the pitch rate is chosen to be non-zero is investigated here. The pitch rate solution for an adjacent rotor failure with the current actuator capabilities is limited to a lowered free-fall velocity. To determine the actual terminal velocity the simulation should be appended with velocity drag forces and an overall higher fidelity aerodynamic model.

An equilibrium can be reached by not only making the actuators more instantaneous but also increase the total thrust force. For the same geometry blades, the actuator dynamics should be at least 20 times faster than nominal whilst also increasing the maximum rotor velocity. The latter being achievable through higher quality racing motors.

It can therefore be concluded that for the near future the pitch rate solution is an unfeasible solution for commercially available quadrotors and a yaw rate solution should be adopted instead.

Appendix B

Inner Loop Protection

The common theme in all crashes for both double and single rotor failures at this point seem due to destabilization of the inner loop. Moreover, the result is a sharp increase in downward velocity. The preference with regards to safety is to give up outer loop control in favor of the inner loop. Thereby, keeping its attitude, slowing the quadrotor down to a more manageable velocity and allowing it to descend more gently.

B-1 Implementation of the Inner Loop Protection

Various methods exist to prevent the outer loop from asking unattainable states from the inner loop. A good example is Pseudo Control Hedging (PCH) whereby reference to the inner loop is matched with the ability of the inner loop to follow the reference. This method was used to prevent an adaptive controller from learning based on an impossible command and thereby jeopardizing the system stability. In this paper, such adaptive controller is not present. As the PCH method is quite complex, a new idea is proposed which will be addressed as the ILP .

The basis of the ILP is to decrease the outer loop gains as a means to lower the reference to the inner loop when the actuators are close to saturation. This will help the system to give more control authority to maintaining attitude instead of position. The question remains what the definition of saturation is in this case. The rotors behave as an overdamped system and never reach their full limits due to the system's yaw rate. To solve this inexactness a region is established within which a specific rotor is deemed partially saturated. A rotor with an angular velocity lower than this bound is said to be not saturated and likewise is said to be fully saturated when it is larger than this bound. The outer loop gain is lowered proportionally with the highest saturation level of either of the two working rotors.

The ILP method has similarities with fuzzy gain scheduling; it is also rule-based and does online gain-tuning. [3, 4] Fuzzy gain scheduling has even been deployed as an approximately nonlinear controller for a system subjected to actuator saturation. To the extent of the

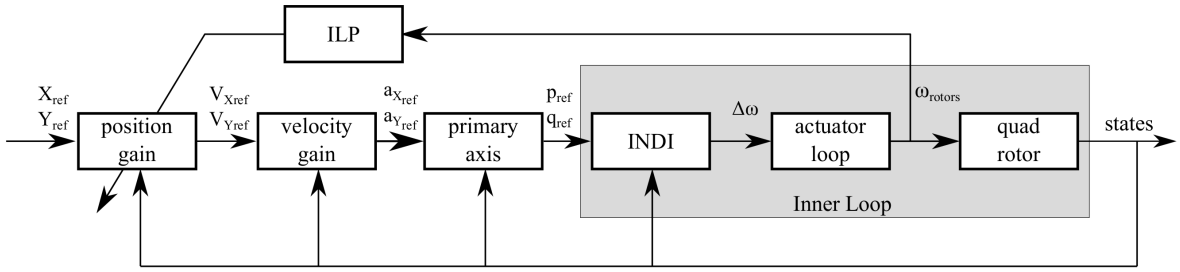

Figure B-1: A control diagram of the proposed Inner Loop Protection system

author's knowledge the ILP method has not yet been applied to prevent actuator saturation based on the actual actuator levels.

The control diagram for the ILP is depicted in fig. B-1. The position gain is tuned online according to eq. (B-1) and can therefore be described as an adaptive control method.

$$K_{position} = K_{original} - c \cdot K_{safety} \tag{B-1}$$

Whereby $K_{original}$ is the originally tuned gain and K_{safety} the extra gain which would be fully subtracted from the $K_{position}$ when one of the actuators is fully saturated. $c \in [0, 1]$ is then defined using

$$\begin{aligned} c &= 1, && max(\omega_i) > L_{upper} \\ c &= \frac{(max(\omega_i) - L_{lower}}{L_{upper} - L_{lower}}, && max(\omega_i) \in [L_{lower}, L_{upper}] \\ c &= 0, && max(\omega_i) > L_{lower} \end{aligned}$$

according to the rotor with the highest angular velocity $max(\omega_i)$ and the bounds defined by upper and lower bounds set by L. For this research the bounds were set at the empirical maximum from previous flights $L_{upper} = 30.77$ [rad/s] and the lower bound at an arbitrary 90 % from the upper bound $L_{lower} = 27.59$ [rad/s]

All variables and bounds introduced in the proposed ILP can be tuned. The original proportional position gain $K_{original}$ was tuned to be 0.8 [-]. The value for K_{safety} was chosen such that it will reduce — when c equals 1 — $K_{position}$ to nearly zero. This is done with regards to practical safety concerns in the test setup.

B-2 Results from the ILP

The ILP was implemented on the double rotor failure case. A position hold command is given whilst the windspeed is steadily increased. This is done until the quadrotor is crashed. The reached windspeeds were between 4 and 8.5 [m/s] and is comparable with the normal double rotor failure case.

A fully functional ILP would enable the control authority to be fully used to minimize the n-axis error in the inertial frame. The performance can be measured using the Euclidean

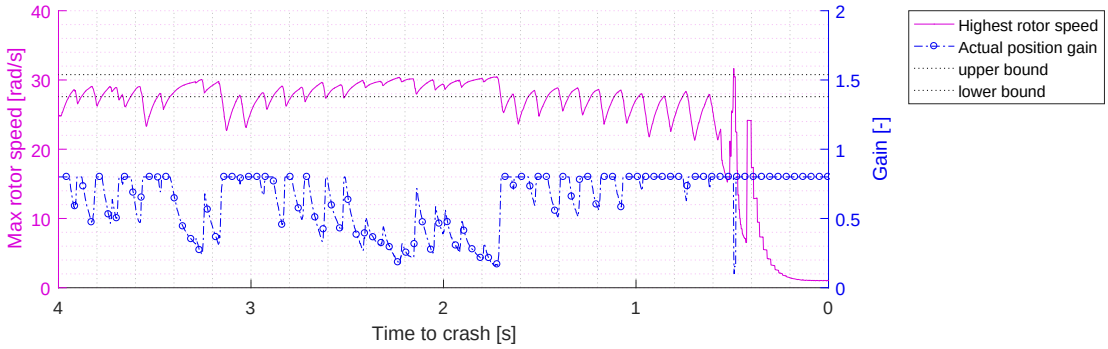

Figure B-2: The effect of the actuator saturation

norm of the n-axis error for both the x and y directions as calculated using eq. (B-2).

$$\|\epsilon_{\mathbf{n}}\|_2 = \sqrt{\epsilon_{\boldsymbol{n}_x}^2 + \epsilon_{\boldsymbol{n}_y}^2} \tag{B-2}$$

A comparison is made between DRF with and without augmentation of ILP in fig. B-3. Six DRF and five ILP flights are averaged to obtain an estimate of the mean. The error for best and worst flights are filtered and plotted to give an indication of outlier performance.

It can be clearly seen in fig. B-3 that the ILP does not increase the inner loop performance. Especially so, knowing that the windspeed limits are alike for both cases. This seems to support the conclusion that the ILP did not have the intended effect.

In an effort to see why the ILP did not perform as expected the effect of actuator saturation on the outer loop gain is investigated, see fig. B-2. The current implementation of the ILP does effectively decrease the outer loop gains. Once the actuators are no longer saturated the outer loop gains are once again increased, but this time the position offset is larger which effectively creates an even larger reference for the inner loop. This vicious circle is frequent enough that the controller seems to work despite the ILP . A recommendation would therefore be to have the ILP decrease the gains for a longer period of time or even until safe landing. Another possibility is to apply a high-order filter to the position command such that the position offset does not translate to extreme reference values.

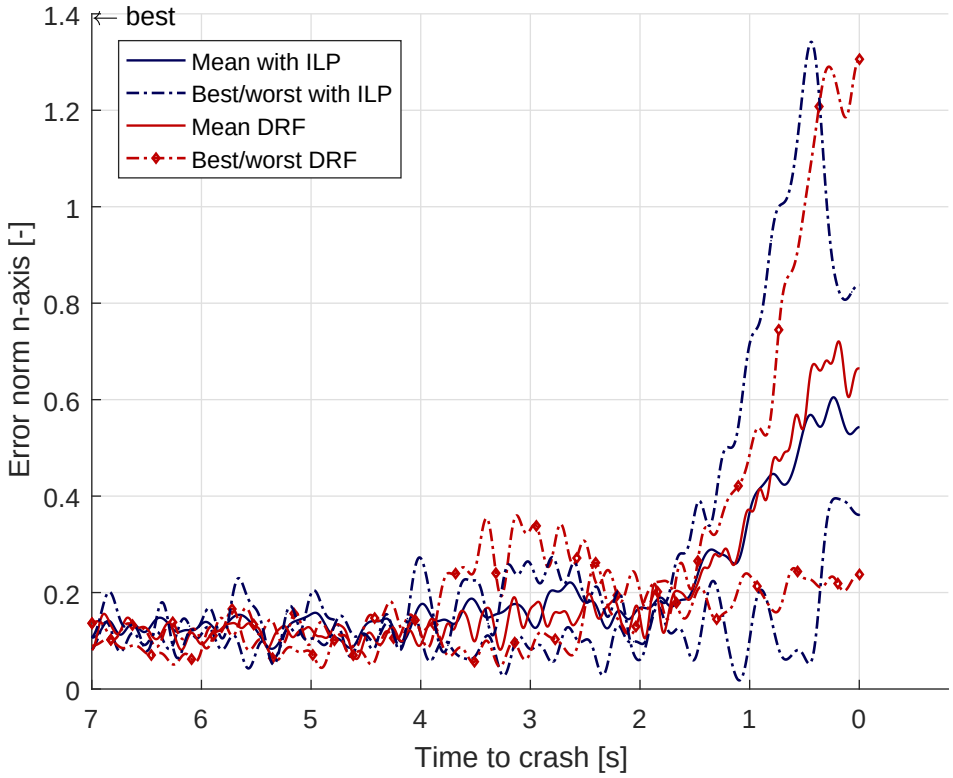

Figure B-3: Average n-axis error for a double rotor failure (DRF) and augmented with the Inner Loop Protection (ILP)

B-3 Conclusion

A variant of an online gain tuning controller was implemented as a means to protect actuator saturation. The position control was sacrificed such that the full control authority can be used for inner loop control.

An oversight in this proposed inner loop protection controller is the effect of unfiltered feedback. The resulting position gain will rise and drop sharply along with the rotor saturations. Filtering the position command or decreasing the position gains for extended periods of time may solve this issue.

Bibliography

[1] Weixuan Zhang, Mark W. Mueller, and Raffaello D'Andrea. A controllable flying vehicle with a single moving part. *Proceedings - IEEE International Conference on Robotics and Automation*, 2016-June:3275–3281, 2016.

[2] Mark W. Mueller and Raffaello D'Andrea. Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 45–52, 2014.

[3] Zhen Yu Zhao, Masayoshi Tomizuka, and Satoru Isaka. Fuzzy Gain Scheduling of PID Controllers. *IEEE Transactions on Systems, Man and Cybernetics*, 23(5):1392–1398, 1993.

[4] N. Cherrat, H. Boubertakh, and H. Arioui. Adaptive fuzzy PID control for a quadrotor stabilisation. *IOP Conference Series: Materials Science and Engineering*, 312(1):1–6, 2018.