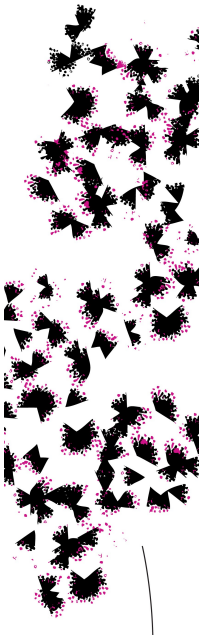
RAM ROBOTICS AND MECHATRONICS



On the Application of Extreme Interaction Torque with an Underactuated UAV

M.J.W. (Max) Snippe

MSc Report

Committee:
Prof.dr.ir. S. Stramigioli
Dr.ir. J.B.C. Engelen
H. Wopereis, MSc
Dr.ir. R.G.K.M. Aarts
Dr.ir. A. Yenehun Mersha

August 2018

033RAM2018
Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE.

MIRA CTIT
BIOMEDICAL TECHNOLOGY AND TECHNICAL MEDICINE

On the Application of Extreme Interaction Torque with an Underactuated UAV

Marcus J.W. Snippe

***Abstract*—This paper aims at setting the initial steps towards the application of extreme interaction torque using an underactuated Unmanned Aerial Vehicle (UAV). Extreme torques are considered torques significantly larger than what UAVs can intrinsically generate.**

An optimization algorithm is designed that uses predetermined constraints, a desired application torque, and UAV parameters to derive an optimal manipulator and input force and torques for the UAV. The optimization minimizes a nonlinearly constrained cost function and returns an optimal homogeneous transformation matrix and input covector. During optimization, the resulting torques and forces are scaled using a weighting function, which allows to assign priority to certain force or torque elements in the application point.

To validate the optimization, the specific use-case scenario of fastening a bolt is used. The UAV parameters and desired application torque are selected for this use-case and the optimization is executed. The weighting function is used to prioritize that undesired torques and forces remain small.

The optimal transformation is validated by means of a static simulation. A certain theoretical maximum torque is considered as reference. This is the maximum that can be reached with the determined UAV weight, maximum thrust, and effective manipulator arm length. The results indicate that the achieved desired torque reaches 67% of its theoretical maximum. The undesired torque reaches only 41% of the value it has when the theoretical maximum of the desired torque is applied. For the chosen UAV, the achieved desired torque is more than 7 times larger than what the UAV can generate intrinsically.

The applicability of the optimization for the chosen use-case is validated by a dynamic simulation with a specifically derived controller. To that end, a dynamic rigid body model of the optimized manipulator on the UAV is derived using screw theory for rigid body dynamics. A Finite State Machine (FSM) is used to solve the problem of fastening a bolt with limited allowed displacement. Fuzzy Inference System (FIS) evaluation based on an estimate of the current damping friction is used to adapt the controller to the unknown friction model of the bolt.

The results show that the chosen controller structure is capable of fastening the bolt to a desired tightening torque. The adaptive FIS PID controller effectively deals with the complex friction torque, but performance can probably be improved by reconsidering the semantic rule-set. The simulation results imply that the chosen motion profile setpoint might not be as effective as a critically damped PID controller with step reference input.

***Index Terms*—UAV, aerial interaction, torque application**

I. Introduction

THE range of applications of UAVs is extensive, due to their mobility and agility. Until shortly, UAVs were mostly used for passive tasks, such as photography/filming, passive inspection, and surveillance. Aerial interaction has increased the range of applications even more. UAV now are able to actively interact with their environment to execute tasks as opening doors, pick-and-placing, and even building rope bridge-like structures [1]–[4].

Although aerial manipulation has been a subject of research for several years, the manner of manipulation is limited to fairly low interaction wrenches. The main reason for this is the fact that UAVs are free floating bodies, contrary to for example a stationary robotic arm with a fixed base. Therefore they are unable to close the force cycle with the environment and thus must intrinsically deliver the reaction forces, which considerably limits the maximum interaction forces and torques for UAVs.

In 2012, a lightweight and versatile manipulator was developed specifically for use on a ducted-fan UAV, which allowed compliant interaction with the environment for non-destructive testing [5]. Scholten, Fumagalli, Stramigioli, *et al.* controlled this manipulator while in contact based on interaction control to follow a path on a surface with the end-effector, whilst simultaneously exerting a force on the surface [6]. For the use in free-flight, a control strategy was derived subsequently that incorporates the dynamics of the manipulator and uses it to improve path tracking performance and maneuverability [7].

To counter the under-actuatedness of UAVs, McArthur, Chowdhury, and Cappelleri endowed a tricopter UAV with an reversible rotor generating thrust that works in the horizontal plane, which adds an additional actuated direction [1]. Several fully-actuated UAVs have been developed by tilting the rotor axes slightly [8]–[10].

More in the direction of applying extreme forces and torques, several control approaches for handling high energy impacts were compared and a manipulator capable of converting the kinetic energy to potential energy and storing it permanently was realized [11], [12]. The results of the latter where used to enable a UAV to perch on a smooth surface to stretch battery-life and consequently the possible air-time of a UAV [13].

UAVs can intrinsically generate limited torques about their $\hat{x}$ and $\hat{y}$ axes that are mainly used for maneuverability. By making UAVs able to execute tasks that demand significantly larger interaction torques than those that UAVs can intrinsically generate, their range of applications can also include tasks like drilling holes, tightening or loosening bolts, or grinding or cleaning surfaces. To that end, this study will attempt to answer the following question:

> *"How can an underactuated UAV be used to apply an extreme torque on an axis perpendicular to a vertical surface?"*

Where '*extreme*' is used to indicate that the torque is significantly larger than what UAVs can generate intrinsically.

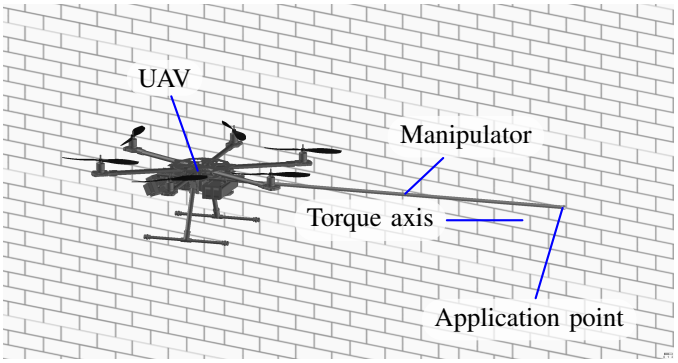


Fig. 1. Example of a UAV endowed with a rigid manipulator that is able to use its thrust to exert a torque on the application point.

A more specific definition of these torques will be provided further on.

Firstly, the problem is elaborated and the proposed approach and used notation are described in the following section. Subsequently, an optimization of a rigid manipulator is derived in section III and a controller is designed for a specific realistic use-case in section IV. The simulated experiments will be described alongside their results in section V. Finally sections VI and VII will discuss and conclude the paper.

II. Problem analysis

A. Problem description

Miniature to small scale UAVs (with a gross take-off weight up to 10 kg and an arm length of up to 0.5 m) can intrinsically generate torques in the order of 0 to ± 15 N m. Therefore, these UAVs are unable to execute tasks with higher required torques, e.g. tightening a metric M10 bolt to its recommended tightening torque requires at least 30 N m, depending on its grade and material[1].

In this paper an attempt is made to derive a method to exert an extreme torque on an axis perpendicular to a vertical surface by using an underactuated UAV of miniature to small scale. A torque is considered extreme if it is at least five times as large in magnitude as the torque that the UAV can intrinsically generate about any of its principle axes. To amplify and concentrate the force and torques that the UAV can generate, the choice is made to endow the UAV with a rigid manipulator. This manipulator can be attached to the UAV either with a joint, or rigidly fixed. Figure 1 shows an example of how the UAV would be positioned with respect to the surface and the application point.

To be able to exert extreme torques using an underactuated UAV, the following points have to be taken into account.

- UAVs in single contact[2] cannot close the force cycle with their environment and therefore must be able to withstand the torques that they apply to maintain an equilibrium state.
- The maximum torques and thrust of an underactuated UAV are limited and coupled, as will be shown in section III-B. The payload of a UAV is limited further by the gravitational acceleration of its own mass and the mass of the intended manipulator.
- The mass and inertia of the manipulator will also contribute to the dynamics of the system. The UAV must still be able to fly in free-flight, without completely saturating its input force and torques.
- The UAV with the intended manipulator will have to be positioned in an optimal state to exert maximum torque. This optimal state must be either directly reachable or reachable through a control procedure.

Additionally, the following assumptions are made in this study.

- Typically, half of the maximum inputs is used as a safety margin to maintain some level of maneuverability. This rule of thumb will also be kept in this study.
- To ensure that the UAV's rotors will not hit the surface, a safe distance has to be kept at all times. This distance will have to be kept between the bounding box of the UAV and the surface.
- This study is confined to applying a torque about an axis perpendicular to a virtual flat vertical surface that stretches out infinitely.
- Phenomena as the ground-, wall-, and ceiling-effect are not considered.
- The manipulator has a uniform mass distribution, thus a constant mass per unit length. Its inertial properties are assumed to be equal to those of a thin rod.
- The manipulator has an end effector with a fixed mass. Its inertial properties are assumed to be equal to those of a point mass.

B. Approach

The displacement and orientation of the UAV imposed by the manipulator as seen from the application point is described with a generic homogeneous transformation. The kinematics of the system are determined based on this transformation. These kinematics are used to determine the transformation of the force and torques applied by and on the UAV to the application point.

An optimization procedure is executed to determine what the optimal transformation is that maximizes the desired torque at the application point.

The constraints of the optimization are determined based on the constraints and limits mentioned in section II-A. These can be ordinary boundary conditions, but could also contain highly non-linear constraints due to the coupling of thrust force and torques that the UAV can generate.

A specific hypothetical UAV with a certain desired torque on the application point is used to validate the optimization. The optimization is solved numerically. The results are used to derive a dynamical model of the UAV with the manipulator, in contact.

The transformation is validated with a static simulation of the dynamic model. For this simulation it is assumed that the application point is unmovable and resists all forces and torques that the UAV applies. The results of this simulation

[1] https://www.blacksfasteners.co.nz/assets/Metric-8_12466_1.pdf
[2] Contact without anchoring to another contact point.

are the measured forces and torques on the application point, which can be compared with the theoretical result.

To validate that the manipulator can be used in a realistic use-case scenario, a custom controller is designed for the use case of tightening a bolt. The friction model of the bolt is assumed unknown to the controller. Multiple levels of damping friction are simulated to verify the generic applicability of the manipulator and controller.

C. Notation

The notation used in this work follows the notation used by Stramigioli and Bruyninckx [14]. Generally, subscripts denote a 'name' indicator and superscripts denote a reference frame (e.g. p_j^k denotes the point j expressed in frame Ψ_k). Quantities without superscript are purely geometric quantities, e.g. (co-)vectors and points, or scalar entities, e.g. power. Capital $P_j \in \mathbb{PR}^3$ is generally used to denote the point p_j in three dimensional projected space, e.g.

$$P_j = \begin{pmatrix} p_j^\intercal & 1 \end{pmatrix}^\intercal,$$

where $^\intercal$ denotes the transpose operator. Due to this notation, any scalar multiplication λP_j where $\lambda \neq 0$ still represents the same point in $\mathbb{R}^3$.

The tilde accent ($\tilde{}$) is used to denote the skew-symmetric matrix representation of three-dimensional (co-)vectors. This skew-symmetric matrix is build from a vector such that $x \wedge y = \tilde{x}y$. Thus

$$\tilde{x} = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix}.$$

Screw theory is used to describe rigid body dynamics. Unless noted otherwise, every rigid body has its coordinate frame defined in its Center of Gravity (CoG). Generalized velocity of a frame is denoted $T \in \mathbb{R}^{6\times1}$ for twist. Twists are composed of the angular velocity of a body and its virtual linear velocity through the origin of the frame in which it is expressed,

$$T_j^{k,l} = \begin{pmatrix} \omega_j^{k,l} \\ v_j^{k,l} \end{pmatrix},$$

where ω and v are three-dimensional vectors. The superscript notation of twists is slightly different from what was mentioned before, i.e. $T_j^{k,l}$ denotes the twist of frame Ψ_j with respect to frame Ψ_l, expressed in frame Ψ_k.

Generalized forces are denoted $W \in \mathbb{R}^{1\times6}$ for wrench. Wrenches are the dual of twists, and are composed of torques and forces, such that $W^k T_\bullet^{k,\bullet} = P$ where P is power as scalar,

$$W^k = \begin{pmatrix} \tau^k & f^k \end{pmatrix},$$

where τ and f are the torques and forces respectively, both are three-dimensional co-vectors.

The position and orientation of frame Ψ_k with respect to frame Ψ_l, can be expressed in a transformation matrix $H_k^l \in SE(3)$. Transformation matrices also conform to the sub- and superscript rules as mentioned above. Additionally, they are used as mapping from one frame to another, e.g. $P^j = H_k^j P^k$. For twists, this mapping is defined as

$$T_j^{k,l} = \mathrm{Ad}_{H_m^k}\, T_j^{m,l},$$

where

$$\mathrm{Ad}_{H_m^k} = \begin{pmatrix} R_m^k & 0 \\ \tilde{p}_m^k R_m^k & R_m^k \end{pmatrix}.$$

For the wrench, due to duality, the mapping is

$$\left(W^k\right)^\intercal = \mathrm{Ad}_{H_k^m}^\intercal \left(W^m\right)^\intercal. \tag{1}$$

H-matrices are constructed as

$$H_j^k = \begin{pmatrix} R_j^k & o_j^k \\ 0 & 1 \end{pmatrix},$$

where $o_j^k \in \mathbb{R}^3$ denotes the origin of Ψ_j expressed in Ψ_k and $R_j^k \in SO(3)$ is the rotation matrix that describes the rotation of Ψ_j with respect to Ψ_k. A more detailed introduction to screw theory in robotics is provided by Stramigioli and Bruyninckx [14].

All coordinate systems used in this study consist of an origin o and the $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes. The coordinate systems are right-handed and orthonormal. The hat ($\hat{}$) accent is used for (co-)vectors of unit magnitude. The same notation is used for unit twists and in section IV-A the definition of a unit twist will be given.

In the schematic drawings shown in this study, $\longrightarrow$ are forces, $\twoheadrightarrow$ are torques, and $\hat{x}$-, $\hat{y}$- and $\hat{z}$-axes are drawn as $\longrightarrow$, $\longrightarrow$, and $\longrightarrow$ respectively, unless noted otherwise.

The maximum value of an arbitrary variable x is denoted $\bar{x}$, the minimum value is denoted $\underline{x}$.

III. Optimization

To optimize the transformation imposed by the manipulator, first the kinematics are determined in section III-A. Then the constraints and limits for the optimization are determined in section III-B. A new coordinate frame Ψ_f is defined in section III-C, in which the resulting optimal wrench is expressed during the optimization. This additional frame simplifies the cost function. Subsequently, a cost function is derived that will be minimized in section III-D. Finally, the results of the optimization of the specific UAV are presented in section III-E.

A. Kinematics

In this study the system is considered as shown in figure 3. The frames introduced in figure 3 are:

- Ψ_0 inertial non-moving frame which coincides with the application point;
- Ψ_b UAV body-fixed frame placed in its CoG and aligned with its principal axes;
- Ψ_m similar to Ψ_b but for the manipulator;
- Ψ_{gb} UAV's gravity frame, also placed in its CoG but aligned with the axes of Ψ_0 such that the gravitational force always points along the negative $\hat{z}^{\mathrm{gb}}$-axis;
- Ψ_{gm} similar to Ψ_{gb} but for the manipulator.

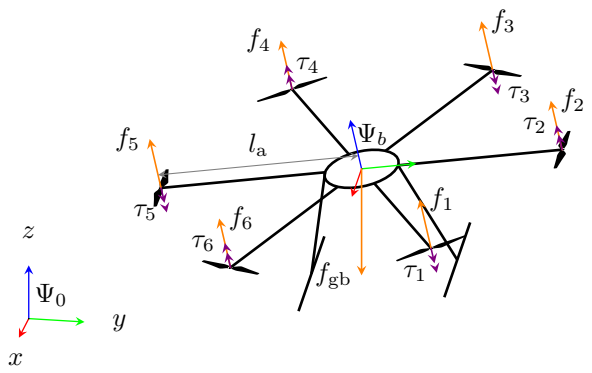


Fig. 2. Forces and torques generated by-, and gravity working on an under-actuated UAV. The length of the UAV arms is denoted l_a.

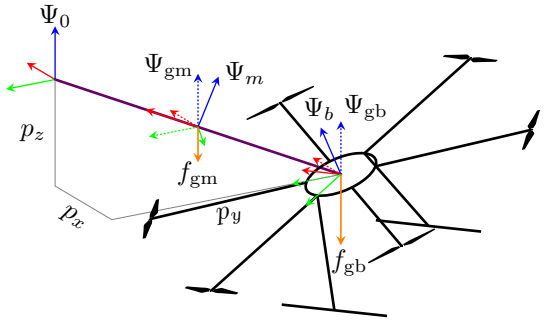


Fig. 3. Schematic representation of the UAV with generic manipulator. The dashed axes compose the frames for the gravitational forces, the solid axes the body fixed frames. The violet line represents the manipulator.

The UAV is treated as a single rigid body that generates the forces and torques as shown in figure 2. These forces and torques can be described as a single thrust force f_t along the body's z-axis and three torques about the three body axes τ_x, τ_y, and τ_z, following

$$K \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \frac{1}{2} & 1 & \frac{1}{2} & -\frac{1}{2} & -1 & -\frac{1}{2} \\ -1 & 0 & 1 & 1 & 0 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{pmatrix} = \begin{pmatrix} f_\mathrm{t} \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix}, \tag{2}$$

where

$$K = \mathrm{diag}\left\{1 \quad l_\mathrm{a} \quad \tfrac{\sqrt{3}}{2} l_\mathrm{a} \quad k_\mathrm{d}\right\},$$

and $k_\mathrm{d} \ll 1$ is the ratio between generated thrust and drag torque of the rotors, which is assumed constant and equal for all rotors. l_a denotes the length between the rotors and the CoG. For every rotor $i \in \mathbb{Z} \cap [1, 6]$, the thrust force of that rotor is $f_i \in \mathbb{R} \cap [0, \bar{f}_\mathrm{rot}]$, where $\bar{f}_\mathrm{rot}$ is the maximum thrust that *one* rotor can generate. The total thrust f_t thus is $f_\mathrm{t} \in \mathbb{R} \cap [0, 6\bar{f}_\mathrm{rot}]$ and $6\bar{f}_\mathrm{rot} = \bar{f}_\mathrm{t}$.

The torques and force on the right hand side of equation (2) are the controllable inputs of the system.

Besides the inputs, two separate gravitational wrenches are acting in the system. All forces and torques add to the total system wrench as three separate wrenches, expressed as follows

$$\begin{aligned} W_b^b &= \begin{pmatrix} \tau_x & \tau_y & \tau_z & 0 & 0 & f_\mathrm{t} \end{pmatrix}, \\ W_\mathrm{gb}^\mathrm{gb} &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -f_\mathrm{gb} \end{pmatrix}, \\ W_\mathrm{gm}^\mathrm{gm} &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -f_\mathrm{gm} \end{pmatrix}, \end{aligned} \tag{3}$$

where f_gb and f_gm are the gravitational forces acting on the UAV body and the manipulator respectively. The frames in which these wrenches are expressed are shown in figure 3.

The generic rigid manipulator is used to transform the wrench generated by the UAV to the application point. To that end, the manipulator can be defined as a generic transformation which can be fully described with the H_b^0-matrix. This matrix contains the position p_b^0 and orientation R_b^0 of the application point (Ψ_0) in the body fixed coordinate system Ψ_b. p_b^0 will also transform the gravitational wrench,

$$H_\mathrm{gb}^0 = \begin{pmatrix} I_3 & p_b^0 \\ 0 & 1 \end{pmatrix},$$

and the gravitational wrench of the manipulator will be transformed by half p_b^0

$$H_\mathrm{gm}^0 = \begin{pmatrix} I_3 & \frac{1}{2} p_b^0 \\ 0 & 1 \end{pmatrix}.$$

Using equation (1), the wrenches from equation (3) can be transformed to the application point Ψ_0. When expressed in the same frame, the wrenches can be summed element-wise, giving one total wrench W_tot^0.

$$(W_\mathrm{tot}^0)^\intercal = \mathrm{Ad}_{H_0^\mathrm{gm}}^\intercal (W_\mathrm{gm}^\mathrm{gm})^\intercal + \mathrm{Ad}_{H_0^\mathrm{gb}}^\intercal (W_\mathrm{gb}^\mathrm{gb})^\intercal + \mathrm{Ad}_{H_0^b}^\intercal (W_b^b)^\intercal.$$

To clearly show the dependencies of W_tot^0 on inputs and the gravitational forces, it can be expressed as the following matrix product

$$\left(W_\mathrm{tot}^0\right)^\intercal = G \begin{pmatrix} \tau_x & \tau_y & \tau_z & f_\mathrm{t} & f_\mathrm{gb} & f_\mathrm{gm} \end{pmatrix}^\intercal, \tag{4}$$

where G is a non-linear function of ϕ, θ, ψ, p_x, p_y, and p_z and will be provided in section III-D in simplified form.

B. Constraints and limits

It must be taken into account that the forces in equation (4) are not of the same order of magnitude. f_gb is proportional to the UAV's mass, thus fixed for a chosen UAV. f_gm is proportional to the length of the manipulator, determined by p_x, p_y, and p_z and thus cannot be controlled directly. The maximum torques $\bar{\tau}_x$ and $\bar{\tau}_y$ and the maximum thrust $\bar{f}_\mathrm{t}$ are mutually coupled. This coupling is described elaborately in appendix A. In short, this coupling allows the co-vector $\begin{pmatrix} \tau_x & \tau_y & f_\mathrm{t} \end{pmatrix}$ to be in the (approximated) 3D space shown in figure 13b, assuming that τ_z is zero. This is captured in the following inequality constraints

$$f_\mathrm{t} + \frac{2}{3 l_\mathrm{a}} \tau \leq 6\bar{f}_\mathrm{rot}, \qquad \frac{2}{3 l_\mathrm{a}} \tau \leq f_\mathrm{t}, \qquad \tau \leq \frac{3 l_\mathrm{a}}{2} \bar{f}_\mathrm{rot},$$

where

$$\tau = \sqrt{\tau_x^2 + \tau_y^2}$$

and $\bar{f}_\mathrm{rot}$ is the maximum thrust of a single rotor.

The UAV must be able to maintain in-flight stability with the manipulator equipped. To that end, the additional load torque due to W_{gm} and the gravitational force acting on the end-effector cannot be greater than the torque that can be generated with the thrust needed to maintain hovering. To maintain maneuverability, an additional safety factor of $1/2$ is used, which places the following constraint on the length and mass of the manipulator

$$l_{\mathrm{m,xy}}\mathrm{g}\left(m_{\mathrm{ee}}+\frac{1}{2}m_m\right)\leq\frac{3}{4}l_{\mathrm{a}}m_{\mathrm{tot}}\mathrm{g},$$

where $m_{\mathrm{tot}}=m_{\mathrm{ee}}+m_m+m_b$, m_{ee} is the end-effector mass, m_m is the manipulator mass, m_b is the UAV body mass, and $l_{\mathrm{m,xy}}$ is the length of the manipulator projected on the xy-plane of the world frame, thus the effective arm gravity makes from the end effector's CoG and the manipulator's CoG to Ψ_b.

Additionally, the total gravitational force may not exceed $\bar{f}_{\mathrm{t}}/2$, again to maintain maneuverability. This places the following constraint

$$\mathrm{g}m_{\mathrm{tot}}\leq\frac{1}{2}\bar{f}_{\mathrm{t}}.$$

Another maneuverability constraint is defined to limit the moment of inertia of the manipulator, including the mass of the end-effector. Due to the relatively low maximum yaw torque about $\hat{z}^b$, the additional moment of inertia due to the end effector and the manipulator should not reduce the maneuverability too much. When the UAV is in free-flight with the manipulator endowed, there is no need for aggressive maneuvering, but an acceptable angular acceleration must still be feasible. Because angular acceleration is inversely proportional with the total moment of inertia, the maximum moment of inertia about $\hat{z}^b$ is chosen to be limited to twice the moment of inertia of the UAV about $\hat{z}^b$

$$J_{\mathrm{ee}}+J_m\leq 2J_{\mathrm{zz},b},$$

such that the total moment of inertia about $\hat{z}^b$ remains smaller than $3J_{\mathrm{zz},b}$. J_{ee} and J_m are the moments of inertia of the end-effector and the manipulator respectively, both about $\hat{z}^b$, and $J_{\mathrm{zz},b}$ is the third diagonal element of the UAV's inertia matrix.

The UAV needs to have a certain distance from the surface on which the torque is applied to safeguard the rotors from hitting the surface. The bounding box of the UAV is approximated by a cylinder shape with a radius of $l_{\mathrm{a}}+r_r$, where l_{a} is the length of the UAV's arms and r_r the radius of the rotors. An additional safety margin is determined to be 5 cm. The minimum absolute distance $|p_x|$ from the UAV CoG to the surface is therefore constraint as

$$|p_x|\geq l_{\mathrm{a}}+r_r+0.05. \tag{5}$$

Finally, the maximum angle that the UAV is able to make with the horizontal plane will have to be constraint. For applications in which the manipulator can suddenly break loose from the application point, the UAV must maintain controllability. To that end, the maximum angle is limited by constraining the pitch angle θ and roll angle ϕ to $\pi/8=22.5°$ as follows

$$\sqrt{\theta^2+\phi^2}\leq\frac{\pi}{8}. \tag{6}$$

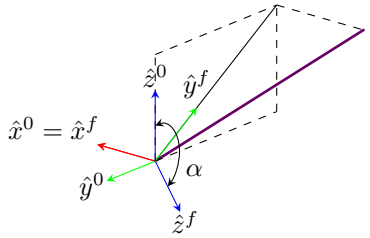


Fig. 4. The manipulator (again in violet) without UAV to show how Ψ_f is defined.

C. Additional wrench frame

Any force generated by the UAV will be directly transferred to the application point. In addition, the transformation of torques from Ψ_b to Ψ_0 cannot cancel forces in Ψ_0. Therefore, minimizing the force in Ψ_b directly minimizes the force in Ψ_0. This will bring the optimal manipulator back to where *only* τ_x is exerted on the application point and the thrust is used to cancel gravity.

To properly define a constraint on the force, it must constrain the components of the force that are not contributing to the desired torque. To that end, the wrench can be expressed in a virtual frame Ψ_f, which shares its origin with Ψ_0 and aligns its $\hat{y}$-axis with the projection of the manipulator on the yz-plane of Ψ_0. To transform W^0_{tot} to Ψ_f, equation (1) can be used with $\mathrm{Ad}_{H^0_f}$.

H^0_b can be parameterized with six parameters[3]. The orientation is parameterized with three angles ψ, θ and ϕ corresponding to the sequential rotations about the $\hat{z}$-, $\hat{y}'$- and $\hat{x}''$-axis respectively. These angles are Tait-Bryan angles, commonly known as yaw, pitch, and roll in aviation. The three position parameters are p_x, p_y and p_z.

Due to the rotational symmetry of the UAV as explained in appendix A, under the assumption that τ_z is kept zero, ψ is not influencing the value of W^0_{tot}. Therefore, during the optimization ψ is taken 0 rad to decrease the needed computations. Afterwards, ψ is taken such that the manipulator always lies in the xz-plane of Ψ_b, where x is positive. This way, the manipulator is always pointing 'forward' which will ease flying in free-flight.

When p_y and p_z are known, H^f_0 can be determined with the atan_2 function with p_y and p_z as arguments. See equation (27) in appendix D for the definition of atan_2 used in this paper. H^f_0 can then be expressed as

$$H^f_0=\begin{pmatrix}1&0&0&0\\0&\cos\alpha&-\sin\alpha&0\\0&\sin\alpha&\cos\alpha&0\\0&0&0&1\end{pmatrix},$$

where

$$\alpha=\mathrm{atan}_2(p_z,p_y).$$

[3]Provided that $\theta\neq\pi/2$ but with the constraint in equation (6), θ is not allowed to reach $\pi/2$.

D. Cost function

The goal of the manipulator is to transform the UAV wrench in such a way that the total wrench in Ψ_f conforms to the following:

- τ_x^f is maximized, because this is the goal of this study;
- τ_y^f, τ_z^f, and f_y^f are minimized, because these torques are undesired and can damage the end-effector or the application point;
- $f_x^f = f_x^0$ is also minimized, but must be positive, because a positive force along $\hat{x}^0$ is needed to prevent the UAV from drifting from the application point.

In this, f_z^f is not constrained, because it directly contributes to τ_x^f.

Due to the fact that $k_\mathrm{d} \ll 1$, the maximum $\bar{\tau}_z$ is significantly smaller than $\bar{\tau}_x$ and $\bar{\tau}_y$. Increasing τ_z is also very costly in terms of thrust. Therefore, we choose to neglect τ_z as a potential input. With $\tau_z = 0$, equation (4) loses a column and can be expressed as

$$\left(W_\mathrm{tot}^0\right)^\intercal = G\begin{pmatrix}\tau_x & \tau_y & f_\mathrm{t} & f_\mathrm{gb} & f_\mathrm{gm}\end{pmatrix}^\intercal,$$

where G is

$$G = \begin{pmatrix} \mathrm{c}_\theta & \mathrm{s}_\phi\mathrm{s}_\theta & p_z\mathrm{s}_\phi + p_y\mathrm{c}_\phi\mathrm{c}_\theta & -p_y & -\frac{p_y}{2} \\ 0 & \mathrm{c}_\phi & -\mathrm{c}_\phi\left(p_x\mathrm{c}_\theta - p_z\mathrm{s}_\theta\right) & p_x & \frac{p_x}{2} \\ -\mathrm{s}_\theta & \mathrm{c}_\theta\mathrm{s}_\phi & -p_x\mathrm{s}_\phi - p_y\mathrm{c}_\phi\mathrm{s}_\theta & 0 & 0 \\ 0 & 0 & \mathrm{c}_\phi\mathrm{s}_\theta & 0 & 0 \\ 0 & 0 & -\mathrm{s}_\phi & 0 & 0 \\ 0 & 0 & \mathrm{c}_\phi\mathrm{c}_\theta & -1 & -1 \end{pmatrix},$$

where $\sin i$ and $\cos i$ are denoted s_i and c_i respectively.

An optimal manipulator can be determined by minimizing an objective function, taking into account the (non-linear) constraints set on the decision variables. The objective function can be defined as

$$\min_{q_d} \left\| QU\left(O(q_d) - \check{O}\right)\right\|_2,$$

where

$$O(q_d) = \begin{pmatrix}\tau_x^f & \tau^f & f_x^f & f_y^f\end{pmatrix}^\intercal,$$

$$\tau^f = \sqrt{\left(\tau_y^f\right)^2 + \left(\tau_z^f\right)^2}, \tag{7}$$

q_d is a vector containing the decision variables, $Q \in \mathbb{R}^{4\times4}$ is a constant diagonal weighing function, and $U \in \mathbb{R}^{4\times4}$ is a diagonal matrix that scales only the force entries of $O(q_d)$ with l_m. The latter is used to obtain equal units for all entries, which allows to sum them in calculating the norm.

The desired value of $O(q_d)$ is given in $\check{O}$:

$$\check{O} = \begin{pmatrix}\tau_{x,\mathrm{des}}^b & 0 & 0 & 0\end{pmatrix}^\intercal,$$

where $\tau_{x,\mathrm{des}}^b$ is the desired target value for τ_x^0.

The following decision variables are combined in q_d:

$$q_d = \begin{pmatrix}f_\mathrm{t} & \tau_x & \tau_y & p_x & p_y & p_z & \phi & \theta\end{pmatrix}.$$

Two types of constraints are used in the optimization, constant boundary constraints, and inequality constraints. Additionally, some intermediate variables are used to make the calculations more uncluttered.

The only constant boundary constraint that is of importance is the minimum length of p_x, from equation (5).

$$p_x < -(l_\mathrm{a} + r_r + 0.05),$$

where l_a is the length of the UAV arm and r_r the radius of its rotors. This constraint is limiting the *maximum* of p_x, because the positive $\hat{x}^0$-axis points into the wall.

For the following, it is assumed that the manipulator has uniform mass distribution and a mass density per length unit of ρ_m, the end-effector can be represented as point mass, and the manipulator as a slender rod.

The intermediate variable values that are calculated are τ^f as in equation (7) and

$$\begin{aligned} l_m &= \sqrt{p_x^2 + p_y^2 + p_z^2}, \\ f_\mathrm{gm} &= \mathrm{g}\rho_\mathrm{m} l_m, \\ J_\mathrm{ee} &= m_\mathrm{ee} l_m^2, \\ J_m &= \frac{1}{3} m_m l_m^2, \end{aligned} \tag{8}$$

where g is the gravitational acceleration constant.

With the constraints from section III-B and the intermediate variables from equation (8), the following inequality constraints are set. The inequality constraints are limiting the maximum torque in the xy-plane with equation (9a), the thrust based on figure 13b with equations (9b) and (9c), the total system weight with equation (9d), the maximum roll/pitch angle combined, with equation (6), and the manipulator moment of inertia, including the end-effector mass m_ee with equation (9f).

$$\tau - \frac{3}{2} l_\mathrm{a} \bar{f}_\mathrm{rot} \leq 0 \tag{9a}$$

$$\frac{2}{3l_\mathrm{a}}\tau - f_\mathrm{t} \leq 0 \tag{9b}$$

$$f_\mathrm{t} - 6\bar{f}_\mathrm{rot} + \frac{2}{3l_\mathrm{a}}\tau \leq 0 \tag{9c}$$

$$f_\mathrm{gb} + f_\mathrm{gm} - 3\bar{f}_\mathrm{rot} \leq 0 \tag{9d}$$

$$\sqrt{\phi^2 + \theta^2} - \frac{\pi}{8} \leq 0 \tag{9e}$$

$$J_\mathrm{ee} + J_m - 2J_{\mathrm{zz},b} \leq 0 \tag{9f}$$

The optimization is carried out by using MATLAB® [15], in particular the function `fmincon`[4]. This function is chosen to allow for non-linear constraints, using an interior point algorithm for nonlinear programming [16].

E. Results

The numerical values used in this optimization are listed in table I. The value for ρ_m corresponds to the weight per unit length of a typical carbon fiber tube. The remainder of values correspond to typical values for a small-scale hexacopter.

For the chosen use case, the desired value of $\tau_{x,\mathrm{des}}^b$ is chosen to be $10\bar{\tau}$ which equals $15 l_\mathrm{a} \bar{f}_\mathrm{rot}$. Substituting the numerical values from table I, this equals 60.75 N m, which provides the UAV with enough torque to fasten an M10 metric bolt to its recommended fastening torque.

[4]See https://mathworks.com/help/optim/ug/fmincon.html for the documentation of the MATLAB® `fmincon` function.

TABLE I
NUMERICAL VALUES USED IN THE OPTIMIZATION FUNCTION.

Parameter	Value	Unit
$\bar{f}_{\mathrm{rot}}$	13.5	N
m_b	2.5	kg
l_{a}	0.3	m
r_r	0.1	m
ρ_{m}	0.06	kg m^{-1}
g	9.806 65	m s^{-2}
$J_{\mathrm{zz},b}$	0.1	N m s^2 rad^{-1}

TABLE II
OPTIMAL VALUES OF THE DECISION VARIABLES.

Parameter	Value	Unit
f_{t}	60.909	N
p_x	−0.45	m
p_y	1.1807	m
p_z	0	m
ϕ	0	rad
θ	0	rad
τ_x	2.1636	N m
τ_y	−5.6767	N m

The goal of the optimization is to maximize τ_x^f, but keep the other elements of $O(q)$ close to zero. Because it is unknown on beforehand what a reachable maximum for τ_x^f is, the focus is put on keeping the other elements close to zero. To this end, the weighing function Q is chosen such that less focus is given to the first term of $O(q) - \check{O}$.

$$Q = \mathrm{diag}\{\tfrac{1}{2} \quad 1 \quad 1 \quad 1\}$$

Interestingly, setting the first element of Q, henceforth referred to as $Q_{[1]}$, to any value above 0.1, will not change the decision variables that define H_b^0. The decision variables that change are f_{t}, τ_x^b, and τ_y^b. A balance is found between using thrust to increase τ_x^f and consequently increase τ_y^f or using τ_x^b and τ_y^b to decrease τ_y^f. When setting $Q_{[2]}$ to 0, the maximum thrust will be used to generate τ_x^f, because τ_y^f is no longer considered.

The resulting optimal values of the decision variables are listed in table II. These result in a transformation H_b^0 of

$$H_b^0 = \begin{pmatrix} 0.35614 & 0.93443 & 0 & -0.45 \\ -0.93443 & 0.35614 & 0 & 1.1807 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{10}$$

and

$$H_m^0 = \begin{pmatrix} 0.35614 & 0.93443 & 0 & -0.225 \\ -0.93443 & 0.35614 & 0 & 0.59035 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{11}$$

which result in the frames shown in figure 5 as Ψ_b and Ψ_m respectively.

As expected, the optimal transformation matrix places the thrust perpendicular to the manipulator arm to maximize its contribution to τ_x^f. This also ensures that f_y^f is zero.

The resulting wrench in the application point then is

$$W_{\mathrm{tot}}^0 = \begin{pmatrix} 44.69 & 10.53 & 0 & 0 & 0 & 35.65 \end{pmatrix}. \tag{12}$$

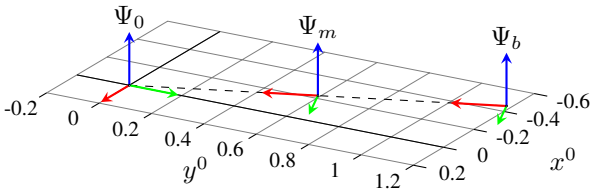


Fig. 5. The optimal transformation of Ψ_b and consequently of Ψ_m.

The second element can never be zero when the thrust is used to generate additional $\hat{x}^0$ torque, due to the minimum length of p_x. Additionally, in this configuration the last element of W_{tot}^0 is aligned with $\hat{z}^f$ and is therefore not constrained. The first element is nearly 7.3 times as big as $\bar{\tau}$, which comes close to the desired 10 times and is still large enough to fasten an M10 bolt.

The optimal input covector is

$$\begin{pmatrix} \tau_x^b & \tau_y^b & f_{\mathrm{t}} \end{pmatrix} \approx \begin{pmatrix} 2.2 & -5.7 & 60.9 \end{pmatrix}$$

which is also plotted as • in figure 13b. It shows that the endpoint is placed on the surface of the potential space, indicating that the input is completely saturated when applying the maximal torque.

IV. CONTROL

Most possible use cases that require high torques, require it to obtain a certain displacement at high damping or to overcome stiction. The former needs a controller that allows movement and the latter needs a certain amount of caution when increasing the applied torque, due to a possible overshoot when the friction suddenly decreases drastically. This illustrates that the control of a UAV in torque applying mode is not straightforward and demands a dedicated controller. To achieve this, first the dynamic behavior of the system in the chosen use case is determined. Next a FSM is introduced to overcome the problem of rotating back and forth in the chosen use case. Controllers are tuned for the separate states, and finally the problem of stiction is treated by introducing a FIS to determine an additional controller gain.

A. Dynamics

A dynamic model has been developed to validate the kinematics of the optimal manipulator. Both the UAV and the manipulator are considered rigid bodies, rigidly fixed to each other. For the use case of fastening a bolt, the Degree of Freedom (DoF) that is to be controlled is the rotation about $\hat{x}^0$, which is denoted q_1. The formerly static transformation matrices are now functions of q_1 in this use case. The inputs are f_{t}, τ_x and τ_y, again assuming τ_z to be kept zero.

Additionally, a ratchet-like joint is assumed between the manipulator and the bolt. This joint ensures that the UAV is able to move back after tightening the bolt, to keep the needed stroke length small.

To determine the Jacobian function that maps from the change rate of the generalized coordinates to the rigid body

twists, the unit twist of q_1 is used. The unit twist of a rotational joint is defined as

$$\hat{T} = \begin{pmatrix} \hat{\omega} \\ \tilde{r}\hat{\omega} \end{pmatrix},$$

where $\hat{\omega}$ denotes the unit angular velocity of the joint and $\tilde{r}$ the tilde representation of the arm r that reaches from the origin to the rotation axis of ω. Note that this can be any point on that axis [14].

The generalized coordinate q_1 only rotates about $\hat{x}^0$, which gives a unit twist of

$$\hat{T}_{q_1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\intercal$$

and a constant Jacobian mapping of

$$T_i^{0,0} = J\dot{q}_1 = \hat{T}_{q_1}\dot{q}_1.$$

Now this Jacobian can be used to express the kinetic co-energy E^* of the two bodies as follows:

$$\begin{aligned} E_i^* &= \frac{1}{2}\left(T_i^{i,0}\right)^\intercal \mathcal{I}^i T_i^{i,0} \\ &= \frac{1}{2}\left(T_i^{0,0}\right)^\intercal \operatorname{Ad}_{H_0^i}^\intercal \mathcal{I}^i \operatorname{Ad}_{H_0^i} T_i^{0,0} \\ &= \frac{1}{2}\dot{q}^\intercal J^\intercal \operatorname{Ad}_{H_0^i}^\intercal \mathcal{I}^i \operatorname{Ad}_{H_0^i} J\dot{q}, \end{aligned} \tag{13}$$

where i can be either m for manipulator or b for the UAV and $\mathcal{I}$ denotes the inertia tensor. The choice of expressing every rigid body frame in the CoG of that body with its axes aligned with the principal inertial axes of the body, ensures that $\mathcal{I}^i$ is diagonal and can be composed as

$$\mathcal{I}^i = \begin{pmatrix} J_i & 0 \\ 0 & m_i I_3 \end{pmatrix}.$$

Besides the kinetic co-energy, the potential energy $V_i(q)$ of every body can be determined as follows

$$\begin{aligned} V_i(q) &= m_i g h \\ &= m_i g \begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix} H_i^0(q) \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^\intercal. \end{aligned} \tag{14}$$

With equations (13) and (14) the system's Lagrangian equation can be determined as

$$\begin{aligned} \mathcal{L}(q_1, \dot{q}_1) &= E^*(q_1, \dot{q}_1) - V(q_1) \\ &= \frac{1}{2}\dot{q}_1^\intercal J^\intercal \operatorname{Ad}_{H_0^m}^\intercal \mathcal{I}_m^m \operatorname{Ad}_{H_0^m} J\dot{q}_1 \\ &\quad + \frac{1}{2}\dot{q}_1^\intercal J^\intercal \operatorname{Ad}_{H_0^b}^\intercal \mathcal{I}_b^b \operatorname{Ad}_{H_0^b} J\dot{q}_1 \\ &\quad - m_m g \begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix} H_m^0(q_1) \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^\intercal \\ &\quad - m_b g \begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix} H_b^0(q_1) \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^\intercal \end{aligned}$$

and to that end H_0^m and H_0^b must be expressed in terms of q_1. This can be done by transforming the matrices in equations (10) and (11) as follows:

$$H_0^b(q_1) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos q_1 & -\sin q_1 & 0 \\ 0 & \sin q_1 & \cos q_1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \left(H_b^0\right)^{-1},$$

$$H_0^m(q_1) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos q_1 & -\sin q_1 & 0 \\ 0 & \sin q_1 & \cos q_1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \left(H_m^0\right)^{-1},$$

where

$$\left(H_b^0\right)^{-1} = \begin{pmatrix} (R_b^0)^\intercal & -(R_b^0)^\intercal p_b^0 \\ 0 & 1 \end{pmatrix}$$

and likewise for $\left(H_m^0\right)^{-1}$.

Euler-Lagrangian equations can be solved for the complete system to find the equations of motion. Substituting the numerical values provided by the optimization the Euler-Lagrangian results in

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_1}\right) - \frac{\partial \mathcal{L}}{\partial q} &= \tau_x^0 + \tau_{\mathrm{td}} \\ 3.57\ddot{q}_1 + 29.39\cos q_1 &= \tau_x^0 + \tau_{\mathrm{td}}, \end{aligned} \tag{15}$$

where τ_{td} denotes the auxiliary torques due to friction.

Due to the aforementioned ratchet joint, the bolt friction torque τ_{b} is only considered when the bolt is tightened, i.e. when $\dot{q}_1 > 0$. Additionally, when the bolt is completely fastened (when the head of the bolt touches the surface) it will start elongating, adding a spring element to the dynamics. Consequently, the head of the bolt is pulled towards the wall by its elongation, increasing the normal friction of the head. This also increases the friction torques significantly.

$$\tau_{\mathrm{td}} = \begin{cases} 0, & \text{if } \dot{q}_1 \leq 0, \\ \tau_{\mathrm{b}}, & \text{if } \dot{q}_1 > 0 \text{ and } x_{\mathrm{bolt}} \leq 0, \\ \tau_{\mathrm{b}} + \tau_{\mathrm{e}}, & \text{if } \dot{q}_1 > 0 \text{ and } x_{\mathrm{bolt}} > 0, \end{cases} \tag{16}$$

where τ_{b} denotes the torque due to friction of the bolt and τ_{e} denotes the additional torques due to reaching the end of the bolt. Therefore, there are three separate system models with significantly different dynamic behavior.

To ensure a realistic bolt model, the friction torque τ_{b} cannot be considered merely viscous. Using a discontinuous friction model that includes a static friction term, can be problematic when using numeric simulation techniques, therefore the continuous friction model from Specker, Buchholz, and Dietmayer is used [17]. Figure 6 shows how the friction force of this rotation can be build from three separate terms, viz. viscous-, Coulomb-, and Stribeck friction. The separate terms can be expressed as:

$$\begin{aligned} \tau_{\mathrm{v}}(\omega) &= D\omega, \\ \tau_{\mathrm{c}}(\omega) &= \tau_{\mathrm{c,c}} \tanh\left(\frac{\omega}{\omega_t}\right), \\ \tau_{\mathrm{st}}(\omega) &= g(\omega)\left[\tau_{\mathrm{st,c}} - \tau_{\mathrm{c}}(\omega_{\mathrm{sp}}) - \tau_{\mathrm{v}}(\omega_{\mathrm{sp}})\right], \end{aligned}$$

where

$$g(\omega) = \frac{\omega}{\omega_{\mathrm{sp}}} \exp\left[-\left(\frac{\omega}{\sqrt{2}\omega_{\mathrm{sp}}}\right)^2 + \frac{1}{2}\right]$$

and the total bolt friction torque is

$$\tau_{\mathrm{b}}(\omega) = \tau_{\mathrm{v}}(\omega) + \tau_{\mathrm{c}}(\omega) + \tau_{\mathrm{st}}(\omega). \tag{17}$$

In the above $\tau_{\mathrm{c,c}}$ and $\tau_{\mathrm{st,c}}$ denote the Coulomb- and Stribeck friction constants respectively, D denotes the viscous damping constant, and ω_t determines the steepness of the $\tanh$ function in the expression for the Coulomb friction. Similarly, ω_{sp} determines the region in which the Stribeck friction is present. Note that ω_{sp} *is not* the Stribeck velocity ω_{st}, but is typically chosen $\omega_{\mathrm{sp}} \approx \omega_{\mathrm{st}}/3$ [17].

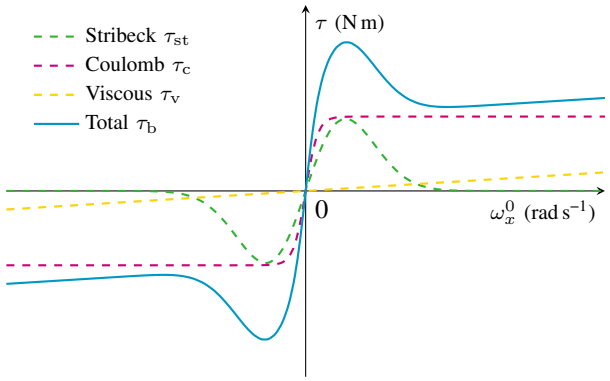


Fig. 6. Description of a continuous model for the friction forces τ_{b} as a function of $\omega_x^0 = \dot{q}_1$[17].

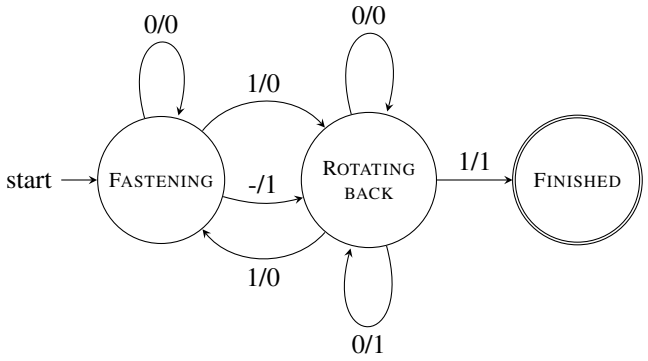


Fig. 7. Finite state machine structure of the controller. The conditions are shown in the form ⟨desired angle q_1 reached⟩/⟨desired torque τ_x^0 reached⟩, e.g. 1/0 denotes that the end of the position stroke is reached without reaching the desired torque and 1/1 means that both the desired position and the desired torque are reached. A dash (-) denotes that the concerning state transition can happen with either value for that condition.

B. Finite state machine

Due to the significantly different dynamic models resulting from equations (15) and (16), the control approach is chosen to contain a FSM with separately tuned controllers. In state FASTENING, q_1 will start at 0 and rotate toward the desired angle. When this angle is reached or when the desired application torque is reached, the state will change to ROTATING BACK, in which the q_1 setpoint is set to 0 again. When q_1 is back at 0, the state will change to FINISHED if the desired application torque was reached the last time in FASTENING and back to FASTENING if the application torque was *not* reached. Note that the desired application torque must be smaller than or equal to the maximum torque deliverable.

The inertia of the UAV makes that its angular velocity cannot be changed instantaneously, which means that the switching between states will lower average velocity. Increasing the stroke length of the motion thus increases the time efficiency of the control approach. Therefore, the desired position is the maximum q_1 provided that the conditions in section III-D still hold. To that end, the reference setpoint is determined to be $\pi/8$ rad, which is the limit of the UAV's roll/pitch angle.

If the UAV will descend past $q_1 = 0$ and reach the end of the bolt *before* being rotated back to $q_1 = 0$, it is impossible for the system to return and thus it is stuck in an impasse. To prevent this, the minimum allowed angle is set to 0 and the maximum to $\pi/8$. To ensure that overshoot is minimal and movement is smooth, the reference input of the system is chosen to be a third order motion profile.

C. Synthesis

The dynamic equation of motion of the system was derived in section IV-A and is

$$3.57\ddot{q}_1 + 29.39\cos q_1 = \tau_x^0 + \tau_{\mathrm{td}}. \tag{18}$$

The nature of this equation allows the non-linearity of the differential equation to be canceled by an input component. To accomplish that, a feed-forward controller is used. This reduces equation (18) for state ROTATING BACK to

$$3.57\ddot{q}_1 = \tau_x^0,$$

representing a moving mass, i.e. a double integrator plant. For the chosen third order motion profile and the considered moving mass, PD controller gains can be determined based on path following criteria [18], [19]. The maximum tracking error is chosen to be 5% of the maximum stroke. The cross-over frequency ω_{pd} of the PD controller is then determined following

$$\omega_{\mathrm{pd}} = \sqrt{\frac{\bar{\ddot{q}}_1\sqrt{\beta}}{\bar{e}_{\mathrm{t}}}},$$

where $\bar{e}_{\mathrm{t}}$ is the desired maximum error and $\bar{\ddot{q}}_1$ is the maximum acceleration of the third order motion profile, which can be determined as follows:

$$\bar{\ddot{q}}_1 = \frac{8h_m}{t_m^2},$$

where h_m and t_m denote the amplitude and the duration of the motion profile respectively.

The continuous PD controller can be described with the following transfer function

$$C_{\mathrm{pd}}(s) = K_c\frac{\check{\tau}_z s + 1}{\check{\tau}_p s + 1}$$

and the parameters can be determined with[5]

$$K_c = \frac{m_{\mathrm{eq}}\omega_{\mathrm{pd}}^2}{\sqrt{\beta}}, \qquad \check{\tau}_z = \frac{\sqrt{\beta}}{\omega_{\mathrm{pd}}}, \qquad \check{\tau}_p = \frac{1}{\sqrt{\beta}\omega_{\mathrm{pd}}}, \tag{19}$$

where $\beta = 10$ is a tameness factor and m_{eq} is the equivalent mass as seen by the controller (equal to 3.57) [18].

The resulting PD controller is

$$C_{\mathrm{pd}}(s) = \frac{1.147s + 5.437}{0.02109s + 1},$$

which is tuned to state ROTATING BACK, where the UAV is moving back with (ideally) zero friction.

Considering that in state FASTENING the bolt adds a complex friction component to the dynamics of the system, a different controller is needed. Firstly, there is the possibility

[5]The accents of $\check{\tau}_z$ and $\check{\tau}_p$ have no other function than to distinguish the time constants from the various torque variables.

that the end of the bolt is reached *just* before the end of the motion profile is reached. The small remaining error will lead to a small plant input, but the large friction torque τ_e will immobilize the UAV when only using a PD controller. To reach the end of the stroke or saturate the input without reaching the end of the stroke, an integral action is added.

For a PID controller the desired crossover frequency ω_pid can be determined with

$$\omega_\mathrm{pid} = \sqrt[3]{\frac{\dddot{\bar{q}}_1 2\beta}{\bar{e}_\mathrm{t}}},$$

where $\dddot{\bar{q}}_1$ is the maximum jerk of the third order motion profile [19]. The maximum jerk can be determined with

$$\dddot{\bar{q}}_1 = \frac{32 h_m}{t_m^3}.$$

The PID controller can be described with

$$C_\mathrm{pid}(s) = K_c \frac{\check{\tau}_z s + 1}{\check{\tau}_p s + 1} \frac{\check{\tau}_i s + 1}{\check{\tau}_i s}.$$

The parameters from equation (19) can be recalculated for the PID controller by substituting ω_pd for ω_pid and additionally $\check{\tau}_i$ can be chosen

$$\check{\tau}_i \geq 2\check{\tau}_z$$

to prevent that the integral action deteriorates the phase advance of the D action[18]. This results in the following PID controller:

$$C_\mathrm{pid}(s) = \frac{0.6045 s^2 + 4.174 s + 5.881}{0.01028 s^2 + 0.5069 s}. \tag{20}$$

To prevent that the state transitions de-stabilize the system, the start of the motion profile when entering a state is set to the current position. Additionally, the I wind-up of the PID controller is reset every state transition.

D. Fuzzy inference system

To generalize the PID controller for state FASTENING for multiple possible bolt friction shapes, an additional configuration gain K_a is used to make the controller adaptive. By considering the rate of change of q_1 and measured torque on the application point, a convenient choice for K_a is made based on a set of rules that command a FIS. This will allow the controller to e.g. lower K_a when a small force results in a large rate of change and increase K_a when a large force results in a small rate of change $\dot{q}_1$.

K_a will scale the controller in equation (20) with a value between 0.25 and 1.75. The value for K_a as function of $\dot{q}_1$ and τ is shown in figure 8. The FIS is elaborately discussed in appendix C.

V. SIMULATION EXPERIMENTS

A. Simulation description

To validate the kinematics of the model derived in section IV-A, a static simulation was done with q_1 fixed. The inputs of the model were set to the values in table II. Figure 9 shows a 3D render of the UAV in the optimal configuration.

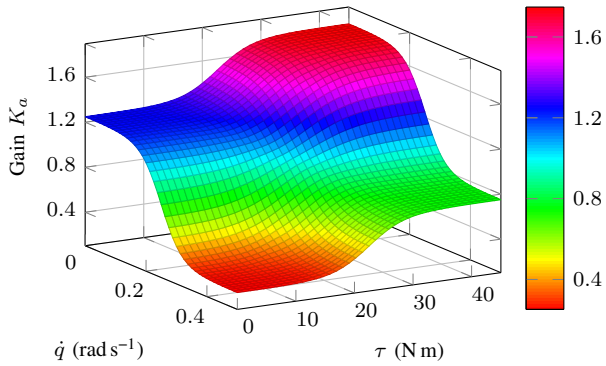


Fig. 8. Evaluation of the FIS with the rules as in appendix C and membership functions as in figure 20.

The total wrench applied to the application point during the static test was

$$W_\mathrm{tot}^0 = \begin{pmatrix} 45.16 & 10.73 & 0.00 & 0.00 & 0.00 & 35.71 \end{pmatrix}, \tag{21}$$

which indeed is very close to the optimal value shown in equation (12).

For a dynamic test, the parameters mentioned in equation (17) must first be determined. The needed accuracy for these parameters is low, therefore it is sufficient to make a rough approximation of the order of magnitude of the parameters. By initializing the UAV in the optimal state and setting the input to zero, it should intuitively behave as a damped pendulum. The parameters have been set such that the response was roughly equal to that of a physical damped pendulum. It will be shown later that a high accuracy for these parameters is not significant for the validation of the optimal manipulator and derived controller.

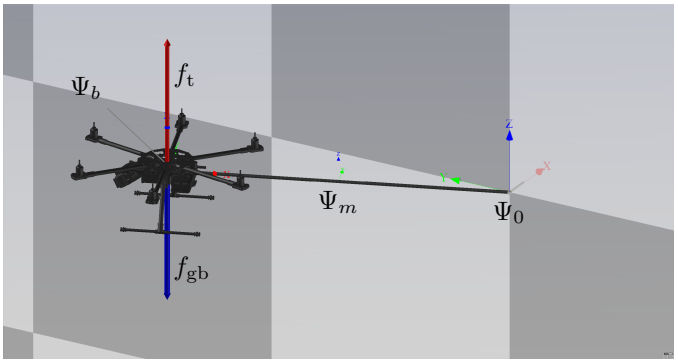
To ensure that the FSM controller works as intended for a variety of possible bolts, the bolt friction torque τ_b is scaled between 1/2 and 3/2 times its initial value.

B. Results

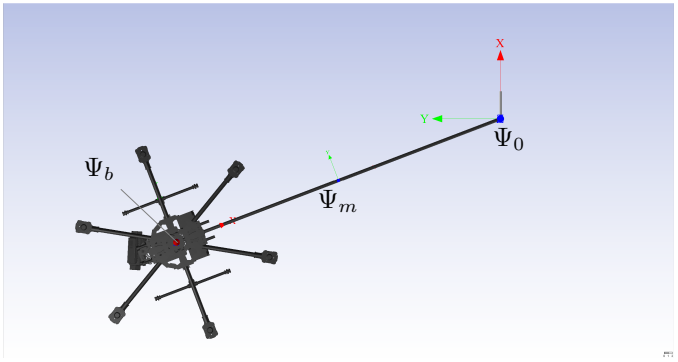
Three simulations have been run, with τ_b scaled by 1/2, 1, and 3/2 respectively. Figure 10 shows the results of these three simulations, i.e. in- and output of the plant u and q_1, the reference angle $q_{1,\mathrm{des}}$, the error between reference and output e, the applied torque τ_x^0 and the additional controller gain K_a as returned by the FIS. The latter is set to 1 if the FSM is not in state FASTENING. Note that the input u is normalized with the maximum torque from equation (12).

As can be seen in figure 10, the differences between the different scalings of the bolt friction torque are marginal. The simulation with $\frac{3}{2}\hat{\tau}_\mathrm{b}$ shows that the increased Stribeck friction causes the FIS to return a higher additional gain to overcome this friction. The simulation with $\frac{1}{2}\hat{\tau}_\mathrm{b}$ shows that overall the additional gain is lower.

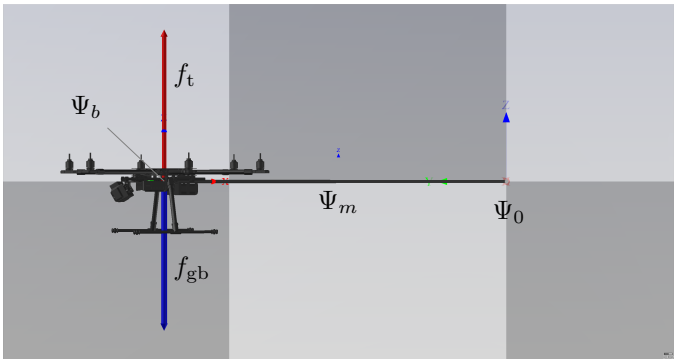
For all three simulations, at t^* the end of the bolt is reached and the elongation and additional friction enforce that the UAV can no longer follow the motion profile accurately. This causes the error signal to rise, consequently increasing u until it is saturated. When u is saturated for some nonzero time, the controller switches to ROTATING BACK, moves back to $q_1 = 0$ and stoppes the simulation by switching to state FINISHED.



(a)



(b)



(c)

Fig. 9. A 3D rendering of the UAV with the optimal manipulator, during the static simulation test. (a) shows an overview, (b) shows a top view from $\hat{z}^b$, and (c) a front view from $-\hat{x}^0$.

VI. Discussion

Without constraints on ϕ and θ, the optimal state would be when the thrust f_t and gravity f_gb pointed in the same direction, thus with the UAV placed upside down. In this study it is assumed that this state would be an unacceptable risk for the system and therefore ϕ and θ have been limited.

The second element of the optimal wrench in equation (12), i.e. the applied τ_y^0, is nearly one fourth of the desired τ_x^0. Due to the maneuverability constraints, the length of the manipulator is limited. Additionally, the bounding box of the UAV places a minimum on p_x. The combination of these two constraints leads to the significant τ_y^0 that can only be canceled by the UAV's τ_y^b or τ_x^b. These torques again are limited due to the coupling of the thrust and the torques as described in appendix A. Consequently, to decrease τ_y^0, τ_x^0 will have to

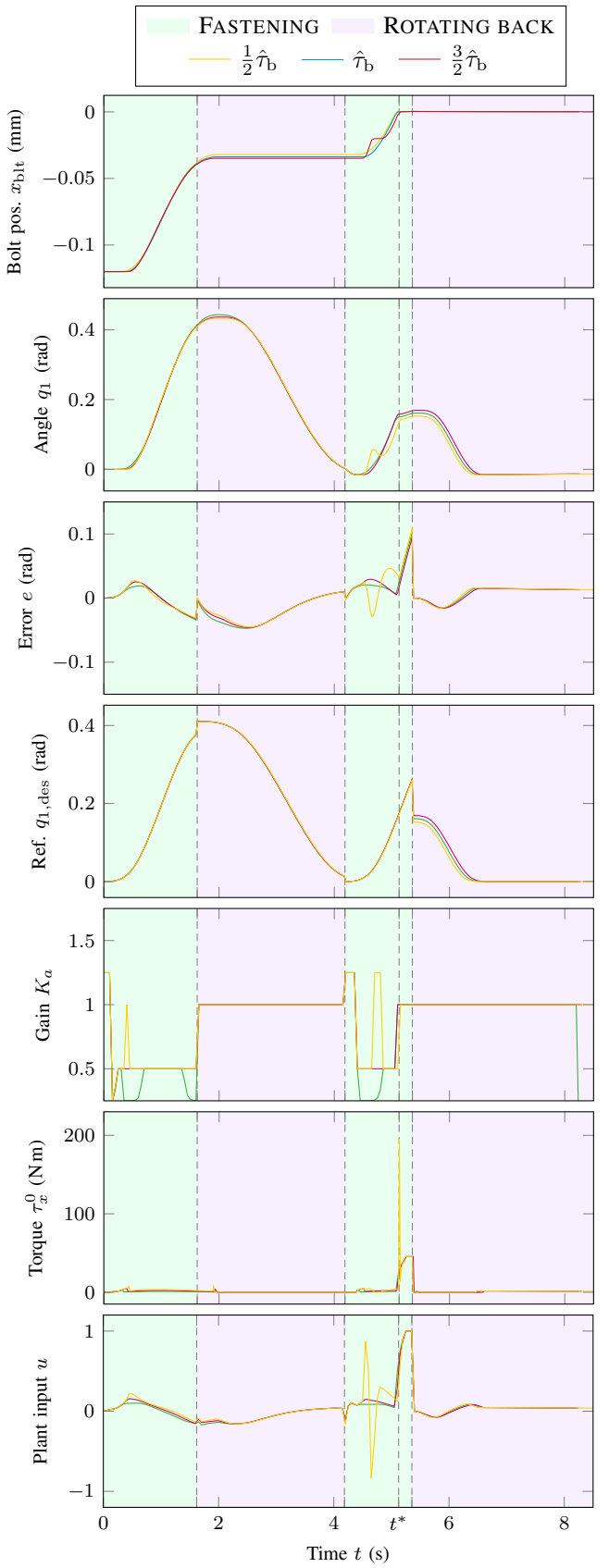


Fig. 10. Simulation results of the dynamic simulations. Three tests have been conducted, with $\frac{3}{2}\hat{\tau}_\mathrm{b}$, $\hat{\tau}_\mathrm{b}$, and $\frac{1}{2}\hat{\tau}_\mathrm{b}$. The jumps in the reference signal are due to the switching between states. The applied torque shows a similar spike for all three simulations, although the one for $\frac{3}{2}\hat{\tau}_\mathrm{b}$ is slightly higher than the other two.

decrease too. A proper balance between these torques can be found by adjusting the weighting function Q. For the use case considered in this study, the value of τ_y^0 is acceptable.

Interestingly, the optimal transformation matrix H_b^0 stays the same for any value above 0.1 for $Q_{[1]}$. Even when changing $Q_{[2]}$ to 0, thus omitting the penalty on τ_y^0, results in the same H_b^0. It seems that the transformation matrix is optimized to a global minimum of the cost function, within the set constraints. The optimal roll ϕ and pitch θ angles are zero, which positions the thrust direction perpendicular to the effective arm of the manipulator. Consequently this ensures that $f_y^f = 0$ and that all thrust force is contributing to τ_x^0.

When setting $Q_{[2]}$ to 0, τ_x^b and τ_y^b will no longer be used and the maximum thrust is used to generate τ_x^0, because τ_y^f is no longer considered. With the optimization results in section III-E, the target torque of 60.75 N m is not reached, but the obtained torque is near to being considered extreme, following the definition in section II-A.

As shown in figure 13b, the endpoint of the input covector is placed on the surface of the potential space. More specifically, it is placed on the top face, which indicates that the input covector is fully saturated. When changing Q, the endpoint will move over this face but will remain coincident. Only when $Q_{[1]}$ is smaller than 0.1, R_b^0 will be identity and p_b^0 will be $\begin{pmatrix}-0.45 & 0 & 0\end{pmatrix}^\mathsf{T}$. The thrust is then used to cancel the gravitational force and only the intrinsically generated τ_x^b is transferred to the application point. The endpoint of the input covector is still located on the surface of the potential space in figure 13b, but on the middle cylinder section.

Considering the maximum thrust of the specific UAV and the length of the optimal manipulator perpendicular to the $\hat{x}^0$ axis, the theoretically maximum torque τ_x^0 would be

$$(\bar{f}_\mathrm{t} - f_\mathrm{gb})\sqrt{p_y^2 + p_z^2} \approx 66.7\,\mathrm{N\,m}$$

and the unwanted maximum torque τ_y^0 would be

$$(\bar{f}_\mathrm{t} - f_\mathrm{gb})\sqrt{p_x^2 + p_z^2} \approx 25.4\,\mathrm{N\,m}.$$

The achieved torque τ_x^0 of 44.7 N m and τ_y^0 of 10.7 N m reach approximately 67% of the theoretical maximum τ_x^0 and only 41% of the undesired theoretical maximum τ_y^0.

Considering the controller, the FIS to determine K_c adds complexity to the control system, but also contributes greatly to the performance. In comparison with a conventional PD controller, tuned for both state FASTENING and ROTATING BACK, the FIS FSM shows less overshoot, a better tracking of the motion profile, and a better response on stick-slip behavior.

On the other hand, the structure of the controller is still P(I)D tuned for a relatively slow motion profile. An alternative might be to tune a P(I)D controller such that it is critically damped when applied to the considered system. Supplying a step reference input to the controlled system would prevent overshoot and offer high immediate gain due to a large immediate error. The slow initial increase of the currently applied motion profile leads to a longer time period where the Stribeck friction is active. Due to this slow start the controller has to catch up when the Stribeck friction is overcome, still causing some overshoot.

The torque curve in figure 10 shows three minor peaks: one at the start and two more at the two first state transitions. This additional torque is probably required to overcome the Stribeck friction component. The higher Stribeck friction for the simulation where $\tau_\mathrm{b} = \frac{3}{2}\hat{\tau}_\mathrm{b}$, seems to cause a stick-slip behavior after the second state transition. This in turn causes K_a to increase quickly, but considering the curves of the plant input u and the error e, it seems to overreact.

At the point in time when the bolt hits the end of its thread, $t = t^*$ in figure 10, a significant spike is detected in the applied torque τ_x^0 which exceeds the torque of the optimal wrench excessively. This could indicate that even higher application torques could be achieved by building up the UAV's momentum and transferring it in a short time interval to the application point. In a real-world test this spike will probably be tempered by the internal damping of the bolt, which is not considered in this simulation.

The additional gain K_a was expected to change smoothly with varying damping in state FASTENING due to its continuous shape shown in figure 8, but in simulation it changes erratic. This might be due to the inputs of the FIS. One input is the measured torque τ_x^0, which changes almost instantaneously with adjusting K_a. The other input is the rate of change $\dot{q}_1$, which needs time to change due to the UAV's inertial properties. This could cause the following development of K_a during simulation:

1) τ_x^0 is normal and $\dot{q}_1$ is still zero initially
2) This causes K_a to increase, in turn increasing τ_x^0 instantaneously
3) $\dot{q}_1$ does not change directly, but the measured τ_x^0 is larger than during the former time step
4) This causes K_a to increase more, increasing τ_x^0 more too
5) Repeating 2-4 causes K_a to quickly reach its limit

The same reasoning can be followed for when $\dot{q}_1$ is too small and K_a is quickly decreased.

VII. CONCLUSION

In this paper the initial steps have been taken towards the application of extreme interaction torques with underactuated UAVs. The kinematics have been derived of a generic transformation of the UAV. Constraints for the optimization algorithm have been determined based on input saturation, maneuverability, and safety margins. A optimization algorithm has been designed that takes the desired torque τ_x^0, the UAV parameters, and a weighting function as input and derives the optimal H_b^0 matrix and input covector to achieve that torque taking into account the weighting of the desired and the undesired forces and torques.

To validate the algorithm, an optimal rigid manipulator has been derived for a specific use case scenario of fastening a bolt. The manipulator has been validated by means of a dynamic UAV model, derived using screw theory for rigid body dynamics. Subsequently the dynamic model has been used to apply the manipulator in the realistic use-case scenario. To that end, a Finite State Machine (FSM) controller was designed with a PD controller for one state and a PID controller for the other

state, both in addition to a feed-forward controller canceling the gravitational influence. The PID controller is adaptive by means of an additional gain, which is determined by means of a Fuzzy Inference System (FIS) based on current angular velocity and torque.

The controlled system has proven to be able to fasten a variety of bolts with different friction parameters. The FIS has proven to successfully counteract the complex nonlinearity of the bolts friction model, but its performance can probably be improved with other semantic rules that prevent the quickly increasing or decreasing resulting K_a.

The optimal manipulator and input covector of the UAV balance the desired and undesired wrench elements in Ψ_0 and achieve a τ_x^0 of 67% of its theoretical maximum and a τ_y^0 of only 41% of its theoretical maximum. For the specific UAV, the achieved τ_x^0 is more than 7 times larger than the torque that it can generate intrinsically.

A. Future work

This study was focused on achieving a static maximum torque. Future work can use the optimization of the manipulator as presented and possibly further increase the achievable torque by utilizing the dynamic properties of the UAV. For example, the inertia and mass of the UAV can be used as a momentum storage, increasing momentum by building up (angular) velocity. This momentum can be transferred to the application point in a short time interval, to achieve a resulting torque that can be added to the static torque achieved in this study.

Another possible addition to this area of study could be the application of a powerful external actuator on the end effector of the manipulator. Using the powerful actuator and the 'slowness' of the UAV's inertia, a short powerful torque could be applied to the application point.

Appendix A
Coupling between thrust and torque

Underactuated UAVs use the thrust from multiple parallel rotors to directly control their altitude and attitude. The altitude can be directly controlled by increasing or decreasing all thrusts. To control attitude, torques about the body fixed $\hat{x}^b$, $\hat{y}^b$, and $\hat{z}^b$ can be generated, which follow from the rotor thrusts and the UAV motor layout.

For a hexacopter as shown in figure 11, these four forces are expressed in the rotor thrusts as follows:

$$T = f_1 + f_2 + f_3 + f_4 + f_5 + f_6 \tag{22}$$

$$\tau_x = l_\mathrm{a}(f_2 - f_5) + \frac{1}{2} l_\mathrm{a}(f_1 + f_3 - f_4 - f_6) \tag{23}$$

$$\tau_y = \frac{\sqrt{3}}{2} l_\mathrm{a}(f_3 + f_4 - f_1 - f_6) \tag{24}$$

$$\tau_z = k_\mathrm{d}(f_1 - f_2 + f_3 - f_4 + f_5 - f_6) \tag{25}$$

Three (symmetric) limits can be recognized when expressing the thrust T as a function of τ_x and τ_y, as shown in figure 12. The point (0,0) in figures 12a and 12b can be found by noticing from equation (22) that at zero thrust, all f_i must be zero, thus from equations (23) and (24) follows that τ_x and τ_y are also zero. The line segment to point (1.5,2) can be found by maximizing the torque with the minimum amount of thrust. The segment from point (1.5,2) to point (1.5,4) follows from the fact that the torque is kept the same if the forces on both sides of the axis are increased equally. The maximum thrust in (0,6) can be found by increasing all forces to their maximum. The total thrust is then $6\bar{f}_\mathrm{rot}$, but the forces in equations (23) and (24) cancel out so there is no torque. The limits can be mirrored in $\hat{z}$, completing the symmetric shape wherein T and τ_x can be controlled.

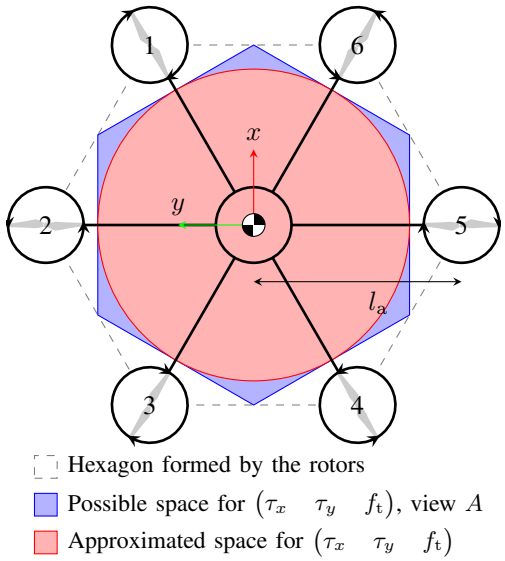


Fig. 11. UAV motor layout for a generic hexacopter.

For τ_y this shape is similar, but stretched out in horizontal direction. This is due to the fact that the average arm length at maximum τ_x is only $3/2 l_\mathrm{a}$ which is slightly smaller than the average arm length at maximum τ_y of $\sqrt{3} l_\mathrm{a} \approx 1.73 l_\mathrm{a}$.

A virtual axis running through rotor 1 and 4, or through rotor 3 and 6, has the same thrust-torque space as the $\hat{y}$ axis. Similarly, the axes perpendicular to these mentioned axes have the same thrust-torque space as the $\hat{x}$ axis. By linearly interpolating between these six planes, the three-dimensional thrust-torque space can be defined as shown in figure 13a.

The space in figure 13a can be approximated by a rotationally symmetric space, which is obtained by rotating the shape in figure 12a about the $\tau_x = 0$ line. This gives the largest rotationally symmetric space that fits in the space of figure 13a.

Appendix B
Bond graph model

Modeling the dynamic equations of section IV can be done graphically by means of (multi) bond graphs in conjuncture with screw theory. For an introduction in bond graph theory, readers are referred to [20]. The models in this section were derived using bond graph terminology and the simulations from section V have been done using 20sim® [21] in combination with the controller in Matlab® [15].

In the figures provided in this section, the port (s1) denotes a signal (s) input (circle) numbered 1 and [p2] denotes a power

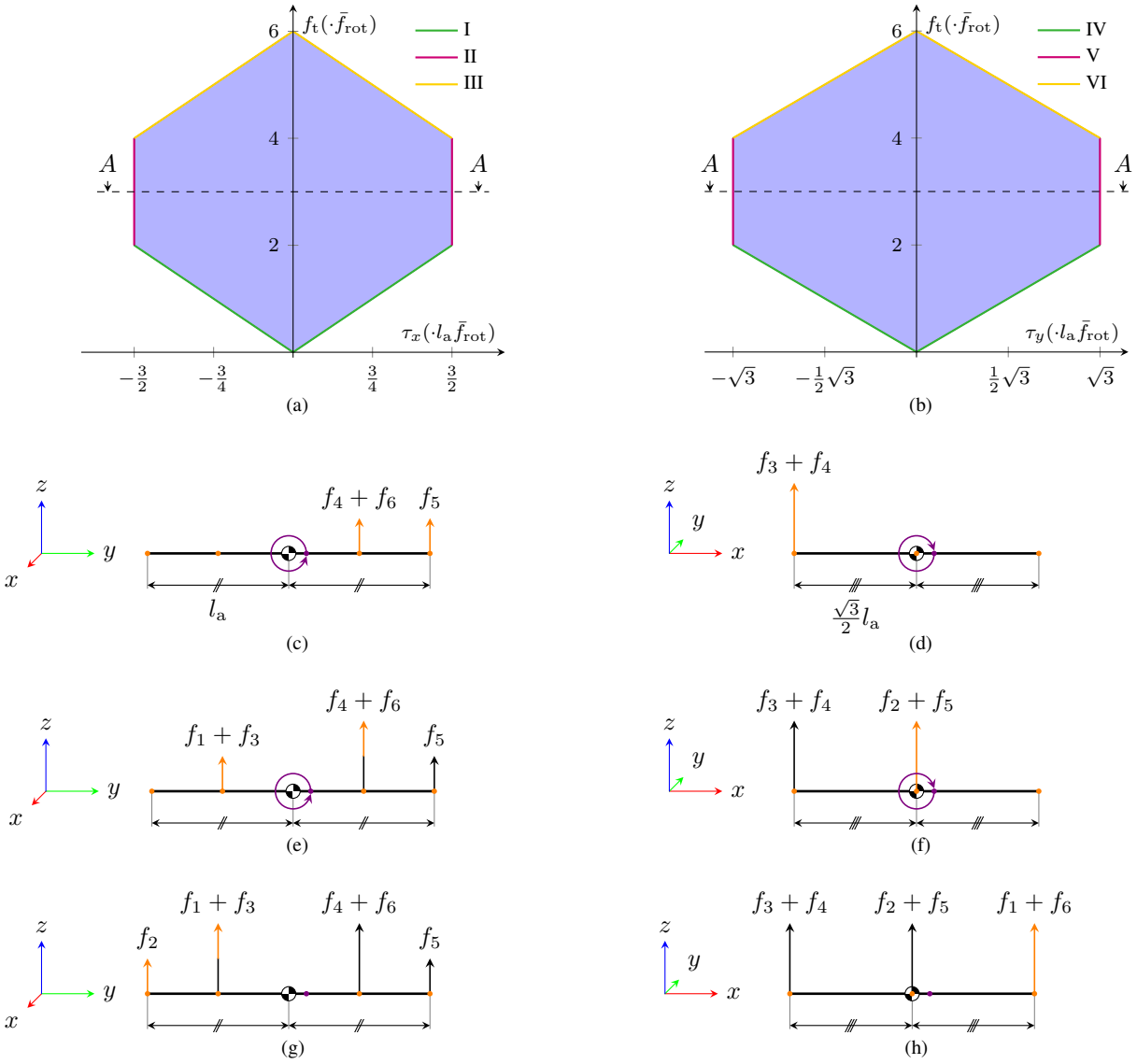


Fig. 12. Graphical representation of the dependency of T on τ_x and τ_y. The graph in (a) shows the plotted relation. (c), (e) and (g) show (in orange) the thrust changes that follow the line segments of I, II, and III in (a) for positive τ_x respectively, and (in violet) the corresponding torque change. For negative τ_x, the thrusts are identical, but mirrored in $\hat{z}$. Similarly, (d), (f) and (h) show the thrust changes that follow the line segments of IV, V, and VI in (b) for positive τ_y respectively, and the corresponding torque change. Note that the maximum τ_y is slightly bigger than τ_x due to the longer average arm length.

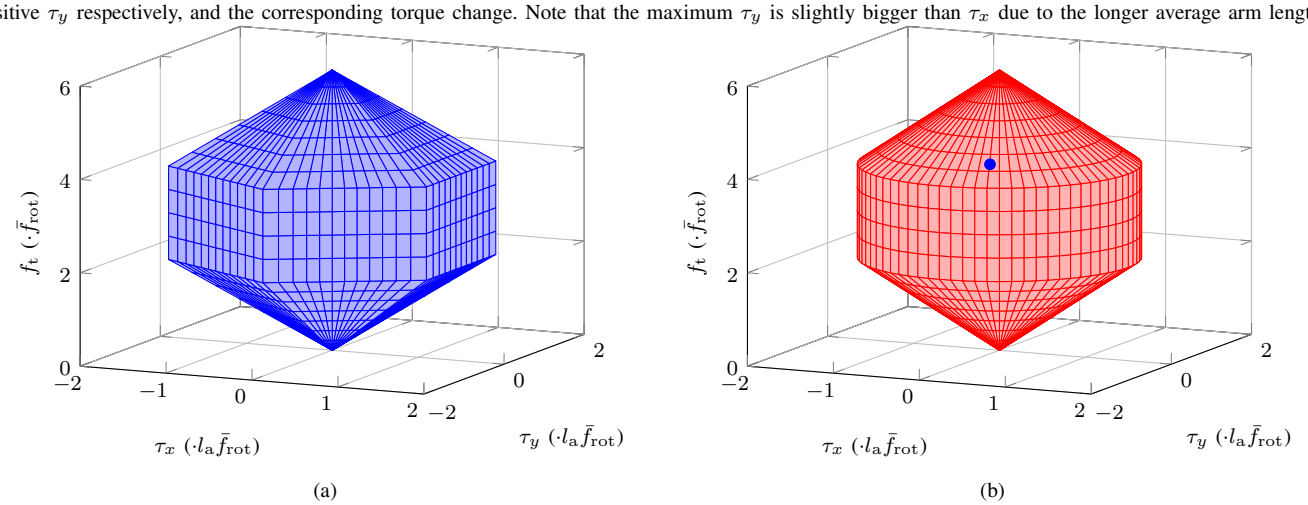


Fig. 13. (a) Possible space for $\begin{pmatrix} \tau_x & \tau_y & f_t \end{pmatrix}$, parameterized in maximum thrust per rotor $\bar{f}_{rot}$ and length of the UAV's arms l_a. A greater torque magnitude can be achieved, but this also result in a net yaw torque. (b) Thrust space approximated as rotationally symmetric space. The mark ● indicates the endpoint of the covector after the optimization in section III-E.

(p) output (square) numbered 2. These ports have no dynamic behavior and only transmit either a signal or a (multi) flow-effort pair.

The high level model in figure 14 consists of a `UAV`-block, which holds a rigid body bond graph structure with a modulated source of effort for the input force and torques [22]. The `Joint`-block contains a bond graph generic joint, but without degrees of freedom, effectively fixing the manipulator in the `Body`-block to the `UAV` with a high stiffness and high damping.

The `Bolt`-block is shown in figure 15 and the `Ratchet`-block is shown in figure 16. The former consists of a switching `X1`-junction that switches closed when the end of the bolt's thread is reached. The latter is used as sensory input for the controller and holds a dissipative element that is switched on when $\dot{q}_1 > 0$ and switched off otherwise.

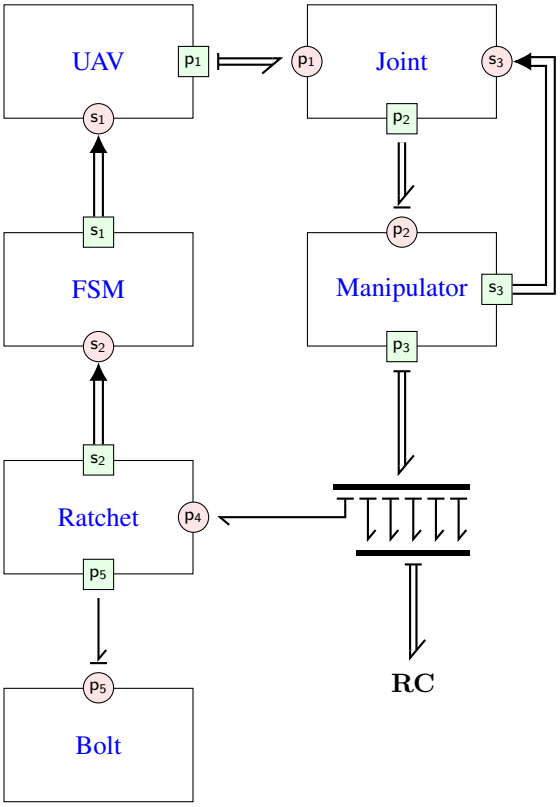


Fig. 14. The high level block diagram of the dynamic model.

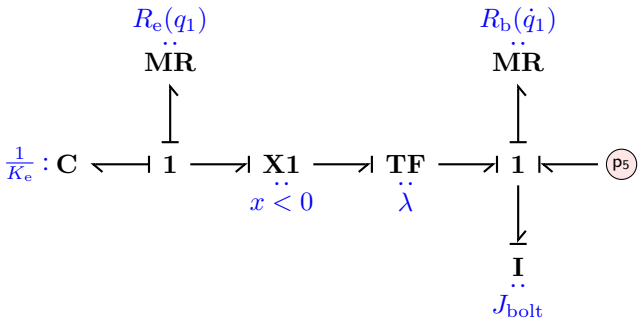


Fig. 15. Bond graph of the `Bolt`-model.

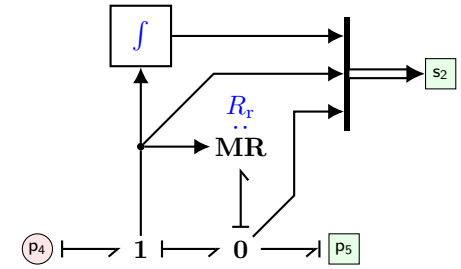


Fig. 16. Bond graph of the `Ratchet`-model.

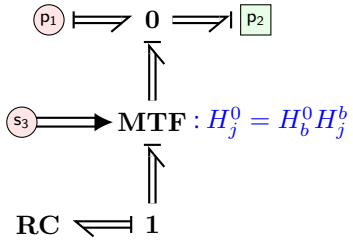


Fig. 17. Bond graph of the `Joint`-model.

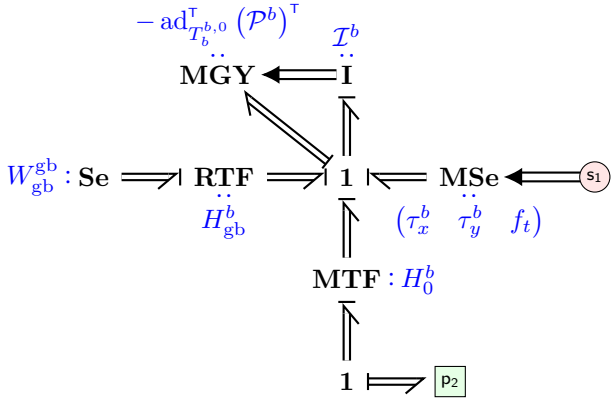


Fig. 18. Bond graph of the `UAV`-model.

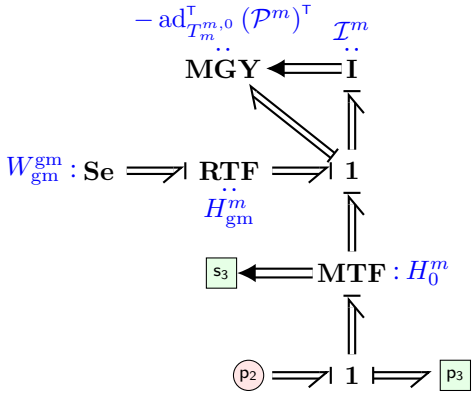


Fig. 19. Bond graph of the `Manipulator`-model.

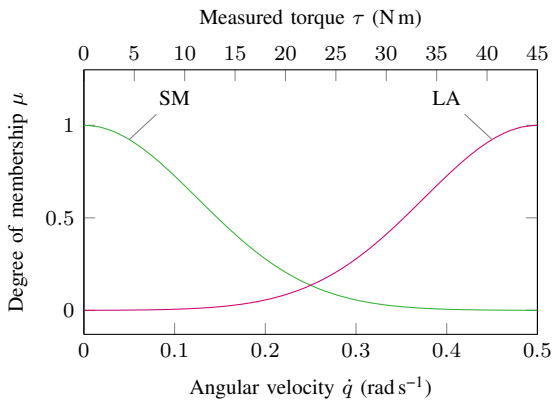


Fig. 20. Membership functions of the input of the Fuzzy Inference System (FIS).

APPENDIX C
FUZZY INFERENCE SYSTEM

A Fuzzy Inference System (FIS) is used to adjust the proportional gain of the PID controller in state FASTENING, due to the unknown friction of the bolt when tightening. The friction cannot be measured directly, therefore the gain is adjusted based on the rate of change of q_1 and the measured torque on the application point. Four straightforward semantic rules are used to determine K_a, which is used to scale the gain K_c of the PID controller. These rules are of the structure `IF (antecedent) THEN (consequent)`:

1) If $\dot{q}_1 = \texttt{LA}$ and $\tau = \texttt{SM}$ then $K_a = 0.25$
2) If $\dot{q}_1 = \texttt{LA}$ and $\tau = \texttt{LA}$ then $K_a = 0.75$
3) If $\dot{q}_1 = \texttt{SM}$ and $\tau = \texttt{SM}$ then $K_a = 1.25$
4) If $\dot{q}_1 = \texttt{SM}$ and $\tau = \texttt{LA}$ then $K_a = 1.75$

In order of the list above, if the current velocity is LA, although the measured torque is SM, the friction of the bolt is most probably small. In that case the control signal can and should be lowered to prevent that the UAV will overshoot its setpoint. If both the measured torque and the current velocity are LA, the gain can be slightly lowered, because the velocity should be limited. If both signals are SM, the gain is slightly increased to help overcome the Stribeck friction of the bolt. If on the other hand the current velocity is SM and the measured torque is LA, the friction is probably high. In that case, the control signal can be increased within the limits of what the UAV is able to apply.

The membership functions that correspond to the values SM (small) and LA (large) for the input signals are shown in figure 20. Note that the scale for the measured torque is shown on the *top* x-axis. The surface plot in figure 8 shows the evaluation of the FIS for the possible combinations of τ and $\dot{q}_1$.

The Gaussian membership functions are described by

$$\mu(x) = \exp\left(\frac{-(x-c)^2}{2\sigma^2}\right), \tag{26}$$

where σ is chosen $\bar{\tau}/4 = 11.25$ for the torque membership functions and $\bar{\dot{q}}_1/4 = 0.125$ for the rate of change membership functions. The constant c is the value of x where the center of the Gaussian function is located. For SM this is at $x = 0$, for LA this is at $x = \bar{x}$.

A. Example

For a given pair $(\dot{q}_1, \tau)$, the corresponding value for K_a can be determined as follows. Firstly, the degree of membership μ for all membership functions is determined as follows:

$$\mu_{\tau,\texttt{SM}}(\tau) = \exp\left(\frac{-\tau^2}{2(\frac{\bar{\tau}}{4})^2}\right) = \exp\left(\frac{-\tau^2}{253.125}\right),$$
$$\mu_{\tau,\texttt{LA}}(\tau) = \exp\left(\frac{-(\tau-\bar{\tau})^2}{2(\frac{\bar{\tau}}{4})^2}\right) = \exp\left(\frac{-(\tau-45)^2}{253.125}\right),$$
$$\mu_{\dot{q}_1,\texttt{SM}}(\dot{q}_1) = \exp\left(\frac{-\dot{q}_1^2}{2(\frac{\bar{\dot{q}}_1}{4})^2}\right) = \exp\left(\frac{-\dot{q}_1^2}{0.03125}\right),$$
$$\mu_{\dot{q}_1,\texttt{LA}}(\dot{q}_1) = \exp\left(\frac{-(\dot{q}_1-\bar{\dot{q}}_1)^2}{2(\frac{\bar{\dot{q}}_1}{4})^2}\right) = \exp\left(\frac{-(\dot{q}_1-0.5)^2}{0.03125}\right).$$

Then the fuzzy logic value of the antecedents in the FIS rule-set are determined. The fuzzy logic equivalent of the logical AND is to take the minimum of the two degrees of membership as follows:

$$a_1 = \min(\mu_{\dot{q}_1,\texttt{LA}}, \mu_{\tau,\texttt{SM}}), \qquad a_2 = \min(\mu_{\dot{q}_1,\texttt{LA}}, \mu_{\tau,\texttt{LA}}),$$
$$a_3 = \min(\mu_{\dot{q}_1,\texttt{SM}}, \mu_{\tau,\texttt{SM}}), \qquad a_4 = \min(\mu_{\dot{q}_1,\texttt{SM}}, \mu_{\tau,\texttt{LA}}).$$

Then the resulting value of K_a can be determined with

$$K_a = \frac{\sum_{i=1}^{4} a_i c_i}{\sum_{i=1}^{4} a_i},$$

where c_i are the consequents as given by rule 1 to 4.

APPENDIX D
DEFINITION OF ATAN$_2$

In this study, the following definition for the atan_2 function is used:

$$\mathrm{atan}_2(y,x) = \begin{cases} \mathrm{atan}\left(\frac{y}{x}\right) & \text{for } x > 0, \\ \mathrm{atan}\left(\frac{y}{x}\right) + \pi & \text{for } x < 0 \text{ and } y \geq 0, \\ \mathrm{atan}\left(\frac{y}{x}\right) - \pi & \text{for } x < 0 \text{ and } y < 0, \\ \frac{\pi}{2} & \text{for } x = 0 \text{ and } y > 0, \\ -\frac{\pi}{2} & \text{for } x = 0 \text{ and } y < 0, \\ 0 & \text{for } x = 0 \text{ and } y = 0 \end{cases} \tag{27}$$

REFERENCES

[1] D. R. McArthur, A. B. Chowdhury, and D. J. Cappelleri, "Design of the I-Boomcopter UAV for environmental interaction," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 5209–5214. DOI: 10.1109/ICRA.2017.7989611.

[2] G. Heredia, A. E. Jimenez-Cano, I. Sanchez, D. Llorente, V. Vega, J. Braga, J. A. Acosta, and A. Ollero, "Control of a multirotor outdoor aerial manipulator," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2014, pp. 3417–3422. DOI: 10.1109/IROS.2014.6943038.

[3] A. Mirjan, F. Gramazio, M. Kohler, F. Augugliaro, and R. D'Andrea, "Architectural fabrication of tensile structures with flying machines," *Green Design, Materials and Manufacturing Processes*, pp. 513–518, 2013.

[4] F. Augugliaro, A. Mirjan, F. Gramazio, M. Kohler, and R. D'Andrea, "Building tensile structures with flying machines," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 3487–3492. DOI: 10.1109/IROS.2013.6696853.

[5] A. Q. L. Keemink, M. Fumagalli, S. Stramigioli, and R. Carloni, "Mechanical design of a manipulation system for unmanned aerial vehicles," in *2012 IEEE International Conference on Robotics and Automation*, May 2012, pp. 3147–3152. DOI: 10.1109/ICRA.2012.6224749.

[6] J. L. J. Scholten, M. Fumagalli, S. Stramigioli, and R. Carloni, "Interaction control of an UAV endowed with a manipulator," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 4910–4915. DOI: 10.1109/ICRA.2013.6631278.

[7] A. Y. Mersha, S. Stramigioli, and R. Carloni, "Exploiting the dynamics of a robotic manipulator for control of UAVs," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1741–1746. DOI: 10.1109/ICRA.2014.6907086.

[8] R. Voyles and G. Jiang, "Hexrotor UAV platform enabling dextrous interaction with structures — preliminary work," in *2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Nov. 2012, pp. 1–7. DOI: 10.1109/SSRR.2012.6523891.

[9] G. Jiang and R. Voyles, "Hexrotor UAV platform enabling dextrous interaction with structures — flight test," in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Oct. 2013, pp. 1–6. DOI: 10.1109/SSRR.2013.6719377.

[10] R. A. M. Rashad Hashem, P. Kuipers, J. Engelen, and S. Stramigioli, "Design, modeling, and geometric control on SE(3) of a fully-actuated hexarotor for aerial interaction," *arXiv.org*, Sep. 2017.

[11] J. T. Bartelds, H. W. Wopereis, S. Stramigioli, and M. Fumagalli, "A comparison of control approaches for aerial manipulators handling physical impacts," in *2016 24th Mediterranean Conference on Control and Automation (MED)*, Jun. 2016, pp. 646–652. DOI: 10.1109/MED.2016.7535915.

[12] J. T. Bartelds, A. Capra, S. Hamaza, S. Stramigioli, and M. Fumagalli, "Compliant aerial manipulators: Toward a new generation of aerial robotic workers," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 477–483, Jan. 2016, ISSN: 2377-3766. DOI: 10.1109/LRA.2016.2519948.

[13] H. W. Wopereis, T. D. van der Molen, T. H. Post, S. Stramigioli, and M. Fumagalli, "Mechanism for perching on smooth surfaces using aerial impacts," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Oct. 2016, pp. 154–159. DOI: 10.1109/SSRR.2016.7784292.

[14] S. Stramigioli and H. Bruyninckx, "Geometry and screw theory for robotics," in *Proceedings IEEE International Conference on Robotics and Automation*, vol. 2001, Citeseer, 2001, p. 75.

[15] MathWorks, *MATLAB*, version 9.4.0.857798 (R2018a) Update 2. Natick, Massachusetts, 2018. [Online]. Available: http://www.mathworks.com/.

[16] R. H. Byrd, J. C. Gilbert, and J. Nocedal, "A trust region method based on interior point techniques for nonlinear programming," *Mathematical Programming*, vol. 89, no. 1, pp. 149–185, 2000.

[17] T. Specker, M. Buchholz, and K. Dietmayer, "A new approach of dynamic friction modelling for simulation and observation," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 4523–4528, 2014, 19th IFAC World Congress, ISSN: 1474-6670. DOI: 10.3182/20140824-6-ZA-1003.01711. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1474667016423116.

[18] J. van Dijk and R. Aarts, "Analytical one parameter method for PID motion controller settings," in *Proceedings of the IFAC Conference on Advances in PID Control*, ser. WeC2.4, University of Brescia, Mar. 2012, pp. 1–6.

[19] T. S. Tadele, T. de Vries, and S. Stramigioli, "PID motion control tuning rules in a damping injection framework," in *2013 American Control Conference*, Jun. 2013, pp. 4957–4962. DOI: 10.1109/ACC.2013.6580607.

[20] J. F. Broenink, "Introduction to physical systems modelling with bond graphs," University of Twente, Tech. Rep., 1999. [Online]. Available: https://www.ram.ewi.utwente.nl/bnk/papers/BondGraphsV2.pdf (visited on 03/09/2018).

[21] Controllab Products, *20-sim*, version 4.6.2 build:7308. 2016. [Online]. Available: http://www.20sim.com/.

[22] J. Engelen and G. Folkertsma, "Control for UAVs," course manual, Jun. 2018.