Understanding the Critical Design Parameters of Aerial Manipulators During Physical Interaction

J.T. (Teun) Bartelds

MSc Report

Committee:
Prof.dr.ir. S. Stramigioli
Dr.ir. M. Fumagalli
Dr. R. Carloni
Dr. A. Yenehun Mersha

August 2015

022RAM2015

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

Preface

Unmanned aerial vehicles (UAVs) have the potential to revolutionize inspection and maintenance to large structures. Non-contact scenarios, such as visual inspection, are currently being exploited commercially. Nevertheless, many maintenance tasks require physical interaction, which is still a very active area of research.

The AEROWORKS project, a European collaboration of universities and companies, envisions a team of *Aerial Robotics Workers* capable of performing inspection and maintenance tasks with a high level of autonomy. This new generation of UAVs is equipped with dexterous manipulators, novel physical interaction and co-manipulation control strategies, perception systems, and planning intelligence.

This thesis' main part is the article entitled *'Understanding the Critical Design Parameters of Aerial Manipulators During Physical Interaction'*, which will be submitted for publication in IEEE/ASME Transactions on Mechatronics. The article presents a generic method for describing aerial interaction. As a specific case, the scenario of impact absorption onto a vertical wall is studied. Based on the resulting analysis, a preliminary manipulator is designed and validated using experiments.

In addition, two appendices are included in which the other main contributions of the project are covered. The first appendix describes the synthesis of a dynamic model, which was used throughout the project to identify the critical parameters, test manipulator concepts and debug various controller implementations. The second appendix elaborates on the design of a preliminary test setup, consisting of a 1D cart endowed with a manipulator.

Understanding the Critical Design Parameters of Aerial Manipulators During Physical Interaction

Teun Bartelds, Stefano Stramigioli, Matteo Fumagalli

***Abstract*—This article first describes the physical interaction of UAVs from a generic viewpoint, after which it is tailored to the specific case of aerial impacts. The goal throughout the analysis is the identification of critical manipulator design parameters. It is shown that positioning the contact point and incorporating the right impedance properties is crucial in order to successfully absorb an impact. A hybrid manipulator, i.e. one combining active and passive joints, is proposed to adequately fulfil these requirements. The passive joint is suspended by springs, which function as a buffer capable of absorbing the kinetic energy during the impact. This flow of energy is made irreversible by means of a mechanical locking mechanism. The significance of each of the identified parameters and design considerations is demonstrated through experiments that compare the optimal design to other configurations. Moreover, the experiments show that the proposed manipulator is capable of handling the impact, such that the UAV transitions into a stable contact situation.**

***Index Terms*—UAV, aerial manipulation, manipulator design, impact absorption, physical interaction**

I. Introduction

THE number of industrial applications in which unmanned aerial vehicles (UAVs) are used has grown rapidly over the past few years. Aerial vehicles are often solely used as an agile sensing platform, incapable of physically interacting with its environment. This contactless operation has already proven to be very valuable, however a large potential still lies in tasks that do require physical interaction. UAVs with this type of capabilities are addressed as *Aerial Manipulators*. If in addition tasks can be executed at a high level of autonomy, we speak of *Aerial Robotic Workers*, a class of UAVs envisioned by the Aeroworks project [1]. An application scenario that would greatly benefit from such a aerial robotic worker is for example the maintenance of wind-turbines (fig. 1). Both the endangerment of people as well as the large time and monetary cost of such an operation are reduced when using aerial robots, in this particular as well as countless other applications.

A well-covered research topic is the transportation of loads, often focusing specifically on collaborative object carrying [2], [3], slung loads [4] or the combination of both [5], [6]. Furthermore, quasi-static contact scenarios receive considerable attention. In [7] the UAV first docks rigidly to wall, after which a task can be executed. Impacts at a considerable velocity are an inevitable aspect of real, outdoors scenarios where sensing uncertainties, e.g. in localization and state estimation, as well as system disturbances, e.g. wind gusts, strongly limit the performance of the system. Other works, e.g. [8], [9] describe contact scenarios in near-hovering conditions, i.e. small angles and low velocities. However, to our best knowledge, the realistic scenario of highly dynamic, aerial impacts has not been covered in research up to this moment, neither in a design nor in a control perspective.

This article provides a way of analysing the interaction and manipulation from a generic floating-base perspective, which is then applied more specifically to the case of high velocity impacts. The discovered parameters that critically influence the interaction lead to a novel design, which is expected to set an example for a new generation of compliant aerial manipulators. The realized prototype features reconfigurability of the identified critical parameters, which allows for experimental validation of their influence. Moreover, the success of the optimal configuration is demonstrated, approving the design considerations made throughout the analysis.

The article is organized as follows. Section II provides a generic description of the system. In Section III these generic equations are tailored to the scenario of high velocity aerial impacts. Then, Section IV describes the design of the prototype, the developed test procedure and the obtained results. Finally, in Sections V and VI the conclusions are summarized and suggestions for future work are made.

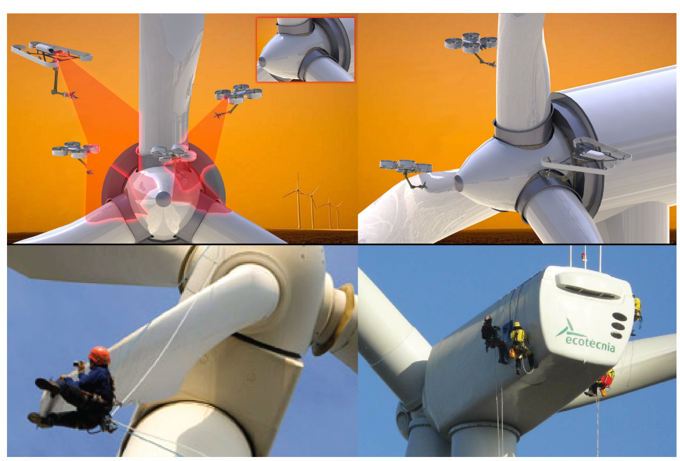


Fig. 1. A typical application scenario where a group of aerial robotic workers collaboratively performs inspection and maintenance tasks to a windmill (top) as compared to the current operations endangering humans (bottom). [1]

This work has received funding from the European Unions Horizon 2020 Research and Innovation Programme under the Grant Agreement named AEROWORKS, no. 644128.

The authors are with the Faculty of Electrical Engineering, Mathematics and Computer Science, CTIT Institute, University of Twente, The Netherlands. Email: J.T.Bartelds@student.utwente.nl, {S.Hamaza, A.Capra, S.Stramigioli, M.Fumagalli}@utwente.nl.

II. Generic methods

A. System description and general notation

Figure 2 depicts the generic system, consisting of a UAV and a manipulator, and introduces the relevant reference frames. Ψ_w, Ψ_b and Ψ_e denote the world, body-fixed and end-effector frame respectively. The body-fixed frame is attached to the UAVs centre of gravity (CoG) and aligned such that the angular rates around the x, y and z-axes correspond to roll, pitch and yaw rotations respectively. Furthermore, the forces and torques acting on the end-effector due to physical interaction, are represented by a wrench $W_c = (F_c, M_c)^T \in \mathbb{R}^6$.

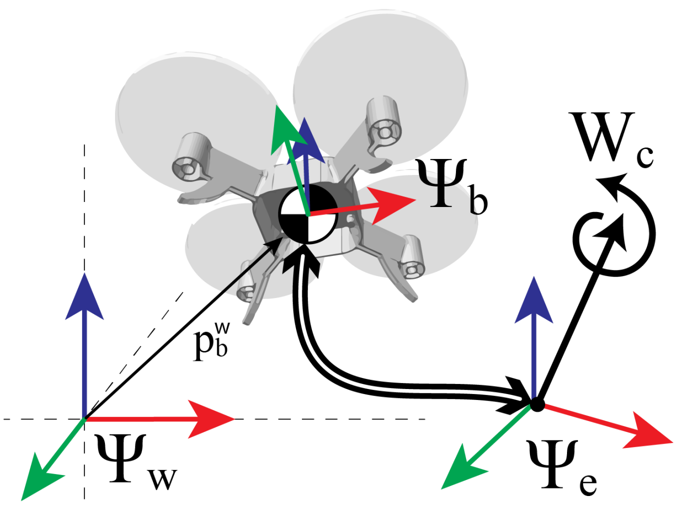


Fig. 2. Generic system sketch.

Some notations are adopted throughout this work, which are described hereafter. Physical quantities, which can often be interpreted geometrically, are denoted without superscripts. Once expressed in a frame Ψ_i it is denoted with a superscript i. For example, the contact wrench W_c denotes a physical quantity. Once expressed in the end-effector or body-fixed frame it is denoted by W_c^e or W_c^b respectively.

Although the orientation is uniquely captured using a set of Euler angles $\theta \in \mathbb{R}^3$, it is often represented by an orthonormal rotation matrix $R \in \mathbb{R}^{3\times 3}$. The rotation that changes the base of a vector from Ψ_i to Ψ_j can be denoted θ_i^j or R_i^j. Both notations represent the same rotation.

Furthermore, the tilde operator $(\tilde{\cdot})$ is used to represent a skew symmetric matrix, which is useful for denoting the cross product of two vectors $a, b \in \mathbb{R}^3$:

$$\tilde{a}b = a \times b \ , \quad \tilde{a} := \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix} \tag{1}$$

B. Dynamic modelling

This section describes the kinetics of the priorly introduced generic system. Here, the goal is to identify critical parameters, without making assumptions on the configuration of the manipulator or specifying the type of interaction.

The equations of motion of the UAV and the manipulator are covered separately in the two subsequent sections. Figures 3 and 4 depict the system sketches of the UAV and manipulator respectively, annotating the wrench $W_m = (F_m, M_m)$ through which the two dynamic systems are coupled.

UAV model:
Figure 3 shows the UAV subsystem, denoting all forces and moments acting on the body with mass m_b and inertia I_b. The configuration of the UAV is uniquely described by its position $p_b^w \in \mathbb{R}^3$ and orientation $\theta_b^w \in \mathbb{R}^3$ with respect to the world frame.

Expressing the translational dynamics in an inertial frame of reference and the rotational dynamics in the body-fixed frame, yields the following equations [10]:

$$\begin{cases} m_b \dot{v}^w = F_g^w + R_b^w F_u^b + R_b^w F_m^b \\ I_b \dot{\omega}^b = \tilde{\omega}^b I_b \omega^b + M_u^b + M_m^b \end{cases} \tag{2}$$

where v denotes the velocity of the CoG w.r.t. the world frame, i.e. $v^w = \dot{p}_b^w$, and ω^b denotes the angular velocity in the body-fixed frame, i.e. the roll, pitch and yaw rates $(\dot{r}, \dot{p}, \dot{y})$. The term $\tilde{\omega}^b I_b \omega^b$ represents the cross coupling of the angular velocity due to gyroscopic effects. The external forces and moments acting on the body are due to gravity (F_g), rotor thrust and drag (F_u, M_u) and the connection to the manipulator (F_m, M_m) and are described in more detail hereafter.

The gravitational acceleration, $g = 9.81 m/s^2$, works at the CoG of the UAV, yielding a pure force contribution:

$$F_{g,b}^w = (0, 0, -m_b g)^T \tag{3}$$

Multirotor vehicles typically suffer from underactuated linear dynamics, which is the case when the propellers axes are parallel. The input space of a vehicle with n aligned rotors,

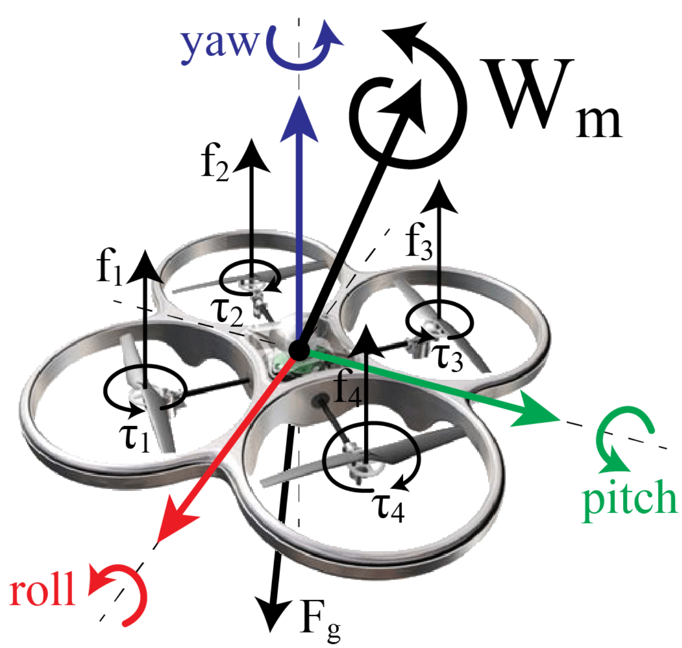


Fig. 3. Forces and moments working on the UAV subsystem.

can be represented by the total thrust and a roll, pitch and yaw torque:

$$\begin{pmatrix} T \\ \tau_r \\ \tau_p \\ \tau_y \end{pmatrix} = \sum_{i=1}^{n} \begin{pmatrix} f_i \\ d_{y,i} f_i \\ d_{x,i} f_i \\ (\text{-}1)^i \tau_i \end{pmatrix} \tag{4}$$

where d_i is the position of rotor i in the body-fixed frame. Finally, the generated input wrench is written as:

$$\begin{cases} F_u^b = (0, 0, T)^T \\ M_u^b = (\tau_r, \tau_p, \tau_y)^T \end{cases} \tag{5}$$

Manipulator model:
Figure 4 shows a sketch of a generic m degrees of freedom manipulator and specifies the variables of interest. Its configuration is uniquely described using a set of generalized coordinates $q_m \in \mathbb{R}^m$.

As the manipulator is mounted on the UAV, i.e. a floating base, its generalized coordinates are augmented with the UAVs state variables:

$$q = \begin{pmatrix} x \\ \theta \\ q_m \end{pmatrix} \in \mathbb{R}^{m+6} \tag{6}$$

The manipulator wrench is composed of terms due to inertial, fictitious, gravitational and the physical interaction respectively:

$$W_m^b = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + W_{g,m}^b(q) + W_c^b \tag{7}$$

Note that this is not the equation of motion of the manipulator, but purely a description of the wrench experienced at the base of the manipulator due to all external forces and moments. Hence, internal actions, e.g. due to actuation, passive components or friction at joints, are not visible explicitly. They influence the evolution of the manipulator's state and thereby enter into equation 7.

Moreover, $M(q)$ is not to be confused with the positive definite, symmetric mass matrix. More specifically it is not even a square matrix. As it maps the accelerations of the generalized coordinates to a 6-dimensional wrench, it is a $6 \times (m + 6)$ matrix that can be interpreted as the projection of all inertial forces and moments onto the base frame. The same holds for $C(q, \dot{q})$, which is a projection of all fictitious forces and moments onto the base.

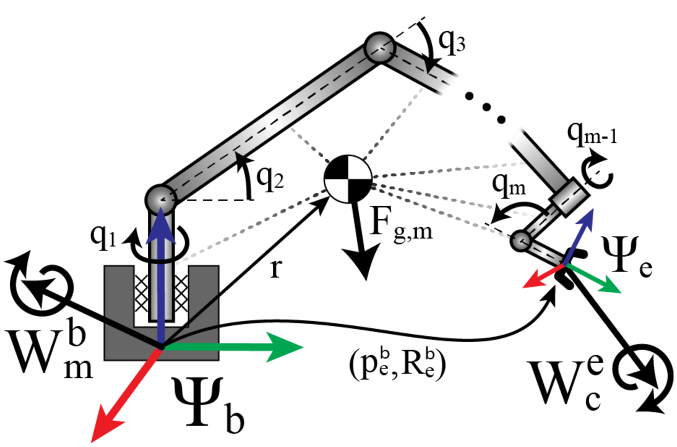


Fig. 4. Sketch of a generic m degrees of freedom manipulator, denoting forces and moments in play. Moreover, the position of the instantaneous centre of gravity of the manipulator and the transformation to the end-effector frame are denoted r and (p_e^b, R_e^b) respectively.

Since the instantaneous CoG of the manipulator is in general not coincident with that of the UAV, the gravitational force will also induce a moment on the UAV base:

$$W_{g,m}^b(q) = \begin{pmatrix} R_w^b F_{g,m}^w \\ \tilde{r}^b R_w^b F_{g,m}^w \end{pmatrix} \tag{8}$$

where $F_{g,m}^w = (0, 0, -m_m g)^T$ and r is the position of the manipulator's CoG w.r.t. the body-fixed frame as depicted in figure 4.

By definition, as it is the effect of an interaction, the contact wrench is not determined solely by the system nor by the environment. Hence, it is not specified further in this generic case. Nevertheless, it is in general more practical to define the contact wrench in the end-effector frame, yielding the following transformation:

$$W_c^b = \begin{pmatrix} R_e^b F_c^e \\ R_e^b M_c^e + \tilde{p}_e^b R_e^b F_c^e \end{pmatrix} \tag{9}$$

where p_e^b and R_e^b denote respectively the translation and rotation of Ψ_e w.r.t. Ψ_b.

C. Generic design considerations

Next, some manipulator design considerations are made for the generic application of aerial interaction. Despite being obvious, or even trivial in some cases, they provide a set of initial guidelines for designing aerial manipulators when made explicit.

In general the UAV's payload provides a severe limitation to the total weight of the carried manipulator. However, also the distribution of mass, both statically and dynamically, are to be taken into consideration. In reference to equation 7, both $M(q)$ and $C(q, \dot{q})$ contribute dynamically, whereas $W_{g,m}$ yields a static contribution. Even if the manipulator is static, i.e. $\dot{q}_m = \ddot{q}_m = 0$, motion of the UAV is still affected by the presence of the manipulator in two ways. Firstly, it adds a force and possibly a moment due to gravity (eq. 8). Secondly, it adds to the effective mass $\bar{m}$ and inertia $\bar{I}$ of the UAV:

$$\begin{cases} \bar{m} = m_b + m_m \\ \bar{I} = I_b + I_m + m_m |r|^2 \end{cases} \tag{10}$$

The design parameter that shows up in the angular part of both equations is r, the location of the manipulator's CoG w.r.t. the UAV. Ideally, minimizing the magnitude of r would reduce both the terms, thereby minimizing the influence of the manipulator on the UAV, however in practice $r \neq 0$. Aligning r with the gravitational z-axis, i.e. keeping the CoG of the manipulator positioned exactly above or below that of the UAV, would minimize the contribution of the cross-product in equation 8. Note that placing mass above the CoG in general reduces the attitude stability, effectively imitating an inverted pendulum.

Dynamic disturbances due to internal manipulator motion, i.e. $M(q)\ddot{q} + C(q, \dot{q})\dot{q}$, can be minimized through minimizing the moving mass. This can be done effectively by using a parallel kinematic structure and placing actuators at the base of the manipulator, preferably right above or below the UAV's CoG. Also the use of motion profiles that minimize $\dot{q}$ and $\ddot{q}$ can make a significant difference to the magnitudes of the disturbances.

As mentioned in the previous section, the contact wrench is only defined as a result of both the interacting bodies. Therefore it is not possible to give a design guideline on how to deal with the resulting force and moment in a generic case. However, it should be noted that, with reference to equation 9, the contact location p_e^b is of importance to the induced moment, i.e. $\tilde{p}_e^b R_e^b F_c^e$. Depending on the scenario, the controllability of this parameter is key in managing the induced moments on the UAV and thus maintaining stability of the system. Moreover, the impedance felt at the end-effector is largely dependent on the design of the manipulator, meaning that it is also considered a critical manipulator design parameter. Both factors are covered in more detail for the impact absorption scenario in Section III.

Since aerial manipulators have strongly diverging goals, care must be taken when assessing optimality of design parameters. In many cases the effects of the manipulator's dynamics on the UAV are to be minimized, as not to disturb the flight controller. However, scenarios can be imagined where the manipulator acts as an additional control input to the UAV, e.g. when the response of the rotors is not fast enough. In this case it might be favourable to place some mass at the end-effector, maximizing the manipulator's inertia.

III. Case study: aerial impacts

A. Scenario description

Next, the generic description of Chapter II is applied to the case of absorbing aerial impacts. Consider the situation depicted in figure 5, in which a UAV approaches a rigid, vertical wall in order to perform a task that requires physical interaction. It is desired that the impact is dealt with adequately by a manipulator, without disturbing the flight controller. The goal is to establish the contact at the target location, without any bounces. After the impact is absorbed, a force should continuously be applied to the environment, such that the UAV remains in contact with the wall even if disturbances are present.

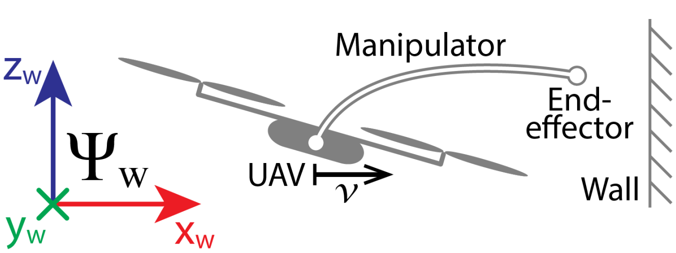


Fig. 5. The impact scenario: a UAV equipped with a manipulator that extends outside the rotor area to absorb the impact with a vertical wall.

The objective in this scenario is thus twofold and can be formalized as:

Design a manipulator that robustly manages the impact of a UAV with a wall, such that:

a) the UAV remains stable and controlled as is, and
b) the end-effector does not lose contact during impact.

In this work we aim to achieve this goal by adequately adapting the manipulator's design, such that the overall system behaves as required. It is noted explicitly that the UAV, including its on-board flight controller, is not modified. This approach is favourable because the developed manipulator can be used to upgrade any UAV in a modular way.

B. Case-specific analysis

In this specific case of impact absorption some simplifying assumptions can be made, which are described hereafter.

The tip of the end-effector is assumed to be small, such that there exists only a single contact point. This implies that the contact wrench, when expressed in the end-effector frame of reference, is a pure force, i.e. $M_c^e = 0$. Equation 9 is updated accordingly:

$$W_c^b = \begin{pmatrix} R_e^b F_c^e \\ \tilde{p}_e^b R_e^b F_c^e \end{pmatrix} \tag{11}$$

It is assumed that the end-effector does not slip along the surface, i.e. the wall can generate any reaction force to keep the end-effector on its planar position and prevents the tip from penetrating the wall. This can be realized in practice by covering the tip of the end-effector with an appropriate material, e.g. rubber or foam.

The environment is static, implying that no energy can be transferred to it. Furthermore, it is assumed that the collisions themselves are fully elastic, i.e. no energy is dissipated in the contact. Note that these assumptions might seem restricting, but actually represent the worst-case scenario. If the environment would absorb part of the impact energy, this would likely contribute to the goal of reducing the linear kinetic energy.

The UAV is commanded to approach at a constant height and pitch angle (p_0) such that the horizontal component of the thrust matches the desired contact force F_d, yielding:

$$p_0 = \tan^{-1}\left(\frac{F_d}{mg}\right) \tag{12}$$

Moreover, the gyroscopic effects are neglected, because the angular velocities are relatively small and they do not contribute to the analysis. Hence, the UAV dynamics (eq. 2), at the moment right before the impact, reduce to:

$$\begin{cases} m_b \dot{v}^w = F_d + R_b^w F_m^b \\ I_b \dot{\omega}^b = M_m^b \end{cases} \tag{13}$$

Let's assume also that the design considerations from Section II-C are implemented. Hence, the moving mass of the manipulator is low, such that the total system's CoG is independent of manipulator motion. Furthermore, the static load due to gravity is well within payload capabilities of the platform. Therefore the manipulator wrench (eq. 7) reduces to:

$$\begin{cases} F_m^b = F_c^b \\ M_m^b = M_c^b \end{cases} \tag{14}$$

The following two sections elaborate on the objectives separately, in order to deduct critical design parameters for each. Subsequently a manipulator design is proposed and validated using simulations and experiments.

C. Induced moments

The first design parameter, which was already identified in a generic context in Section II-C, is the controllability of the end-effector position p_e^b. A more detailed motivation and analysis, specific to the case of aerial impacts, is conducted hereafter.

Due to the underactuatedness of the UAV, its lateral dynamics can only be controlled indirectly, by giving up direct control over the attitude. This introduces a strong coupling between the linear and angular dynamics of the aerial manipulator. Hence, for the UAV to remain stable and controllable as is (objective a, Section III-A), it is crucial that disturbances on the angular dynamics are minimized.

Reducing the UAV's linear momentum upon impact requires a serious contact force to be generated. This linear force also influences the angular dynamics, as can be seen when combining the angular parts of equations 11 and 14:

$$M_m^b = \tilde{p}_e^b R_e^b F_c^e \tag{15}$$

Moreover, because of the significant amount of energy associated with the impact, the induced attitude disturbance is likely to be severe. The induced moment can be geometrically interpreted as the cross-product between the end-effector position w.r.t. the CoG and the contact force, which is depicted in figure 6. The generated contact force is directed opposite to the velocity at which the UAV approaches. Hence, minimizing the induced moment is equivalent to:

$$\min \left| \frac{p_e^b}{|p_e^b|} - \frac{v}{|v|} \right| \tag{16}$$

That is, the end-effector should be positioned such that the contact point coincides with the line of momentum, i.e. a line through the CoG in the direction of the UAV's velocity.

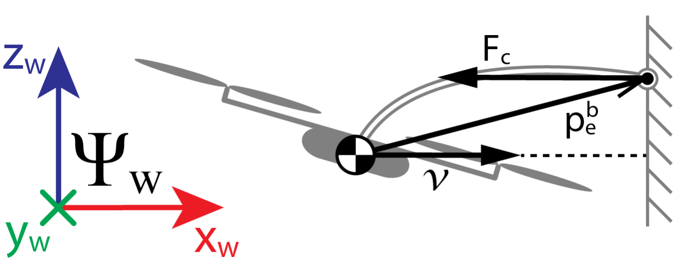


Fig. 6. A sketch of the impact scenario depicting the line of momentum (dashed) and the induced moment $p_e^b \times F_c$ that occurs when the end-effector is not positioned on this line.

In reality positioning errors will be present, which result in an induced moment that acts as a disturbance on the attitude control. The contact force itself cannot be minimized, as a certain impulse is required to slow down the system. However, the frequency spectrum of the contact force is a parameter that can be designed for up to some extend.

Let $\mathcal{S}(s)$ be the attitude's sensitivity function, i.e. the transfer function from the induced moment to the attitude. The sensitivity function provides a measure for the attenuation or amplification of disturbances in the frequency domain:

$$E_D(j\omega) = \mathcal{S}(j\omega) M_m^b(j\omega) \tag{17}$$

where E_D is the attitude error due to the disturbance. That is, the effect of the induced moment on the attitude is filtered by the sensitivity function.

In general the sensitivity function resembles the one sketched in figure 7. Hence, the following approximation holds:

$$\begin{cases} \mathcal{S}(j\omega) \ll 1 & \text{if, } \omega \ll \omega_c \\ \mathcal{S}(j\omega) \approx 1 & \text{if, } \omega \gg \omega_c \end{cases} \tag{18}$$

where ω_c denotes the controllers bandwidth. Therefore, it is desired to elongate the duration of the impact as much as possible, such that the main frequency components of the contact force are within the attitude controller's bandwidth and thus attenuated. This can be achieved by adding a compliance, which is also required for absorbing the impact energy, as will be discussed in the next section.

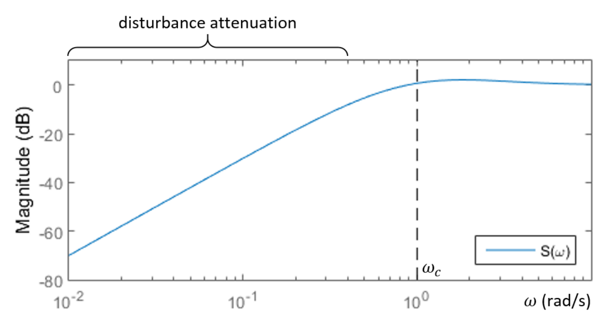


Fig. 7. A sketch of a typical sensitivity function for a floating mass subject to a PD-controller, indicating a high sensitivity to disturbances near and above the controller's bandwidth.

D. Impact absorption

The second design parameter is based on the requirement that the impact must be absorbed in such a way that the system does not bounce away from the wall (objective b, Section III-A).

Let's consider the conservation of the system's energy during the impact, which must hold under the assumptions made in Section III-B. The manipulator is interpreted as a virtual energy buffer, combining internal dissipation, actuation and storage of energy. The system comprises the following energetic storage elements:

- E_t - translational kinetic energy
- E_r - rotational kinetic energy
- V_m - manipulator energy

This implies that the UAV's inputs, besides counteracting gravity, do not influence the energy balance significantly.

Conservation of the total energy can then be expressed as:

$$\frac{\mathrm{d}}{\mathrm{d}t}[E_t + E_r + V_m] = 0 \tag{19}$$

Initially, the system only contains linear kinetic energy, i.e. $E_t(t_0) = E_0$. Moreover, the first objective results in minimization of the induced moments, as described in the previous section, hence $\mathrm{d}E_r/\mathrm{d}t \approx 0$. The goal is to extract the linear kinetic energy, which, in consideration of the set requirements, can only be transferred to the manipulator:

$$\frac{\mathrm{d}}{\mathrm{d}t}V_m = -\frac{\mathrm{d}}{\mathrm{d}t}E_t \tag{20}$$

It should be noted that the desired behaviour is only obtained if this conversion of energy is irreversible. Imagine the manipulator behaving as pure storage element, e.g. a mechanical spring, the flow of energy would be reversed right after the velocity becomes zero, resulting in the aerial manipulator to bounce away from the wall, which is undesirable. Hence, a mechanism that only extracts energy is required, which can be realised in various ways, either actively or passively. An active system can implement any control law that ensures that the exerted force or moment opposes the direction of the motion. Passive solutions can be obtained by including sufficient damping, such that energy is dissipated instead of being returned. Another approach would be to include a locking mechanism that only allows motion in the direction opposite to the force.

IV. Experimental validation

A. Manipulator design

Based on the critical parameters that were identified a manipulator is designed, which is depicted in figure 8.

The manipulator includes an angular degree of freedom, which is position controlled by a servo motor, such that the end-effector can be positioned in accordance to Section III-C. A passive degree of freedom is added in series, which is constrained by two linear bearings and suspended using springs. The end-effector is mounted on a carbon fibre rod that passes through a locking mechanism which blocks the motion in one direction. This ensures the irreversibility of the stored spring energy, as proposed in Section III-D. The design also complies with the generic design considerations (Section II-C), since the manipulator's CoG is positioned above the centre of the UAV. Only a lightweight rod protrudes outside the area covered by the rotors.

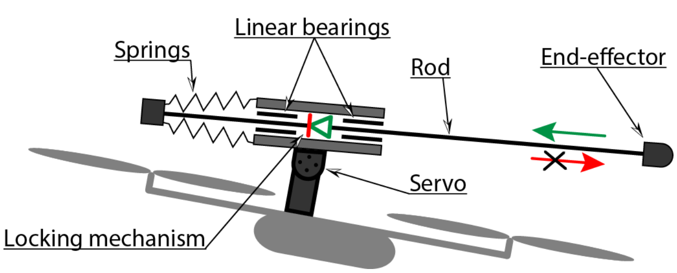


Fig. 8. Sketch of the manipulator prototype, annotating the most important components and the rod's allowed and blocked direction of motion.

Figure 9 shows the design of the locking mechanism, which operates passively based on frictional forces. Two cams on either side of the rod are synchronised through gear teeth protruding above and below the cam profiles. The cams are lightly pretensioned using an elastic band, as to ensure a normal force between the cams and the rod is present. Hence, moving the rod generates a tangential friction force, which induces a moment on the cam. In the allowed direction the cam radius decreases, thereby reducing the normal force and thus the generated friction. In the locked direction, the cam radius increases, causing a strong increase in normal force which in turn tightens the cams even more.

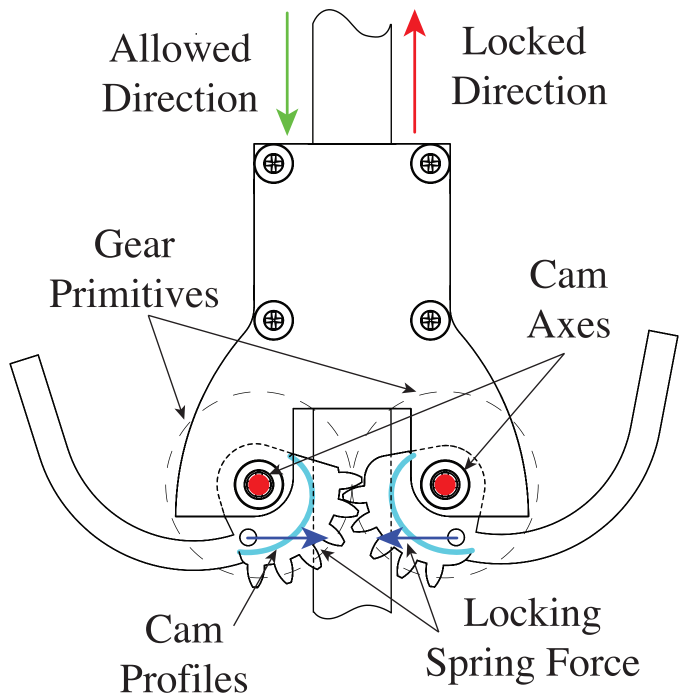


Fig. 9. Detailed drawing of the passive locking mechanism, annotating the important components. Note also the levers on the cams which are used as a manual release mechanism.

The realized aerial manipulator is depicted in figure 10. The manipulator prototype is mounted on a Parrot AR.Drone 2.0, a commercially available quadrotor platform [11]. The UAV's on-board flight controller [12] receives velocity setpoints over a wireless connection. Reflective markers are mounted to allow the UAV and manipulator degrees of freedom to be tracked using an external motion capturing system from OptiTrack [13]. The servo used is a Dynamixel AX-12+ from Robotis [14].

The positioning of the servo is based on the forward kinematics, which can be derived geometrically, and the UAV's attitude measurement, which is either derived from the external motion capturing system or from an on-board estimate. The positioning assumes a fixed location of the CoG, which was estimated precedently. Moreover, the direction of the velocity is not estimated online, but assumed to be in the desired direction of approach.

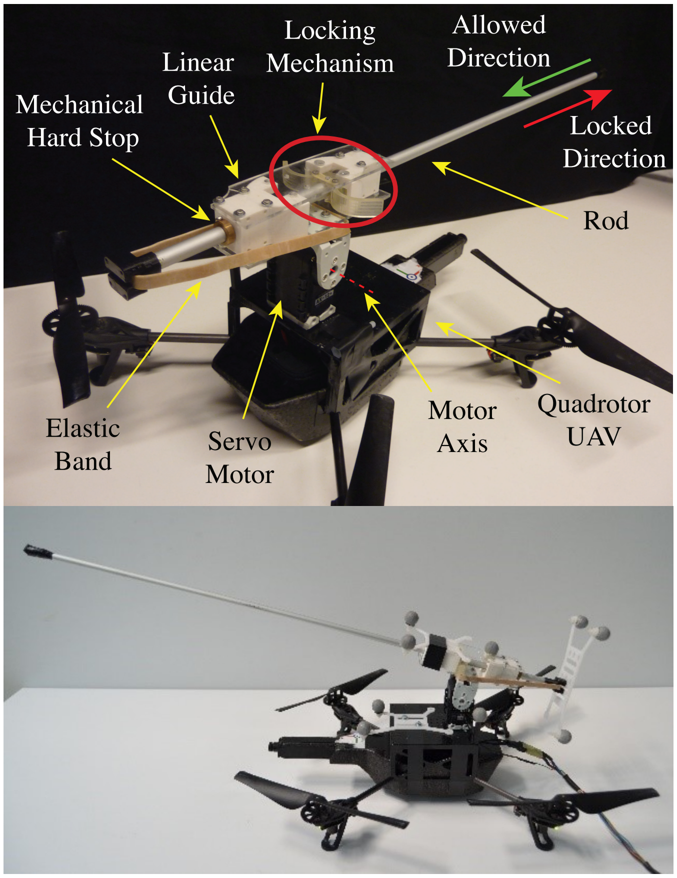


Fig. 10. Close-up of the realized prototype annotating the relevant components (top); The setup equipped with IR-reflective markers (bottom).

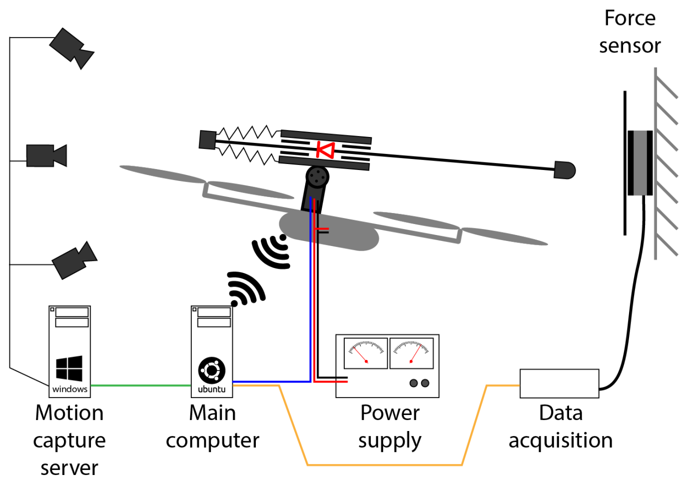


Fig. 11. A sketch of the experimental setup, annotating the relevant subsystems. The UAV is tethered for providing power and manipulator control signals, however the UAV itself is controlled wirelessly. Furthermore, the motion capture system and force sensor are both read out via a wired network connection.

The locking mechanism can be disabled easily, in order to run comparative studies with a normal, bidirectional compliance. Also a fully rigid mode is supported, which is realised by mounting an end-stop onto the rod. The locking mechanism can only be released manually, allowing for a single impact each run.

B. Experimental setup and test procedure

The goal of the experiment is to compare the influence of the proposed design parameters. The prototype is easily reconfigurable, such that the manipulator mode can be varied between the following three options:

- Rigid: motion of the rod is blocked in both directions,
- Compliant: motion is allowed in both directions,
- Locked: the rod can only move in one direction.

Note that the springs to absorb the energy are only engaged in the compliant and locked modes. Also the end-effector position is chosen to be varied among three options, which can be expressed as the height between the CoG and the contact frame, or more precisely $p^w_{e,z} \in \{10,\ 0,\ \text{-}6\}$ cm for the up, centred and down configuration respectively.

Hence, every experiment can be identified using a 2-letter acronym:

- Manipulator mode $\in \{\mathbf{R}\text{igid}, \mathbf{C}\text{ompliant}, \mathbf{L}\text{ocked}\}$
- End-effector height $\in \{\mathbf{U}\text{p}, \mathbf{C}\text{entre}, \mathbf{D}\text{own}\}$

In order to obtain repeatable experimental data, the test procedure was scripted. Firstly, the UAV takes off and hovers at its initial position in front of the target at a distance of 2.3 meters. Then, the UAV approaches the wall at a fixed pitch angle $p_0 \approx 10°$. After impact the UAV tries to maintain contact for approximately 1 second, before it returns to its initial hovering position and lands.

C. Results

The results of the nine experiments, covering all possible configurations, are presented in figure 12 and interpreted hereafter. Moreover, a summary of the experimental data is presented in table I, providing an overview of several parameters that can be used to asses the success of the different configurations. It tabulates whether the UAV crashed (either during or after the impact), whether one or multiple bounces with the wall are registered, the maximum contact force in Newtons ($\|F_c\|_\infty$) and the maximum induced pitch angle in degrees $\|p - p_0\|_\infty$. A photo of the aerial manipulator in contact with the target is depicted in figure 13. All subsequent remarks throughout this section are in reference to figure 12 and table I.

Note that not all arguments can be derived from the presented results in a irrefutable manner. Nevertheless, with video recordings and full logs of all experiments, which provide a

	Crash	Bounce	$\|F_c\|_\infty$	$\|p - p_0\|_\infty$
LC	0	0	19	13
LD	1	0	15	63
LU	0	1	28	25
CC	0	1	28	19
CD	1	1	22	61
CU	1	0	27	43
RC	1	1	126	47
RD	1	1	86	73
RU	1	0	60	87

TABLE I
SUMMARY OF THE PERFORMED EXPERIMENTS.

good qualitative understanding of the obtained behaviour, each notion is made with confidence. For example, table I states that for the 'LU' configuration a bounce was detected, which is not visible in the plot of the contact forces in figure 12. In this case, the video showed that, after the bounce, the end-effector missed the target. Hence, no contact force is registered in the quantitative data, but the qualitative behaviour, i.e. the fact that the UAV bounced, is indisputable.

Overview of the experiemnts:
The first two columns in table I provide a final verdict on whether objective a and b, as defined in Section III-A, were achieved.

The only configuration in which the UAV fully complies with the stated objectives, i.e. the UAV does not crash and not bounce, is the proposed design: the locked manipulator in centred position. This strongly supports the made design considerations and indicates the importance of the identified parameters.

The only other configurations where the UAV managed to stay airborne, are 'LU' and 'CC'. For the 'LU' run, the UAV pitches back slightly due to which it propels away from the wall. The attitude disturbance stayed within controllable limits, hence the pitch angle is controlled back to its desired value p_0 and resulting in a second impact at a lower approach velocity. In the 'CC' configuration the induced moment is also manageable. Due to the released spring energy, the UAV bounces back more than 0.5 meter before it re-approaches the wall.

The 'LD' impact did not result in a bounce, because the spring energy remained locked in the manipulator. Nevertheless, the pitch angle is disturbed so much that the UAV can no longer generate the required lift to counteract gravity and thus slides all the way down along the wall.

The compliant manipulator crashed for the two non-centred configurations ('CD' and 'CU'). Although the initial behaviour is identical to that of the locked manipulator, the additional induced moment generated when the energy is released from the spring is too severe for the attitude controller to cope with. Moreover, as soon as the contact is lost, the servo's control actions tend to destabilize the system ever further, because the reaction torques on the UAV are in antiphase with the control actions.

The experiments with the rigid manipulator tell a story of their own. Because of the lack of a compliance, the interaction force changes into a short impulse and the magnitude scales accordingly ($\|F_c\|_\infty$). That is, because the total change of momentum is the same, the total impulse, i.e. force integrated over time, must be similar. The change of the interaction behaviour is reflected in the other plots as well. Especially the pitch angles are subject to a much more drastic change, from which the UAV is not able to recover, not even in the centred position.

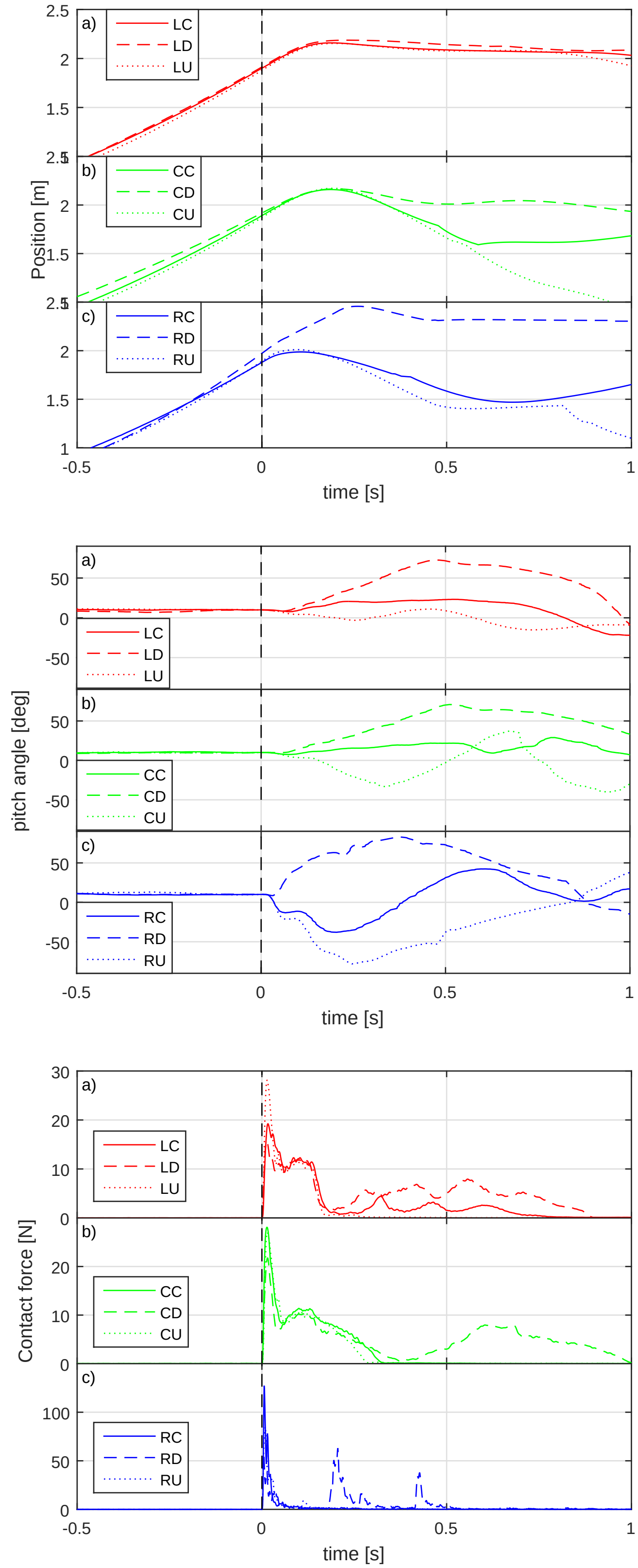


Fig. 12. Experimental results of all the 9 tested configurations, depicting the position, pitch angle and contact force measurements.

Induced moments:
Let's first investigate the induced moments on the UAV by looking at the evolutions of the pitch angles. Based on the prior analysis, it is hypothesised that the up and down configurations ('xU' and 'xD') result in a nose up and nose down pitch respectively, whereas for the centred configurations ('xC') no significant disturbance is to be observed. The results show that, for each of the three manipulator modes, the centred position yields the lowest disturbance and up and down position result in an induced moment if backward and forward direction respectively, supporting the hypothesis.

Some remarks are to be made however. The disturbance in the locked-upper ('LU') configuration is relatively small compared to the other manipulator modes, although still tilts back beyond the horizontal configuration, i.e. $p < 0$. The fact that the disturbance is less severe can be explained by the fact that the manipulator, despite its offset, still extracts a significant amount of energy from the system.

Also the behaviour of the rigid-centred ('RC') configuration might be unexpected at first sight, as it strongly pitches back while the end-effector is supposed to be coincident with line of momentum. Apparently there is a severe momentum induced despite a relatively small positioning error. Referring to the characteristics of the sensitivity function (eq. 18) and with an eye on the plot of the contact forces, it can be explained why the attitude controller is much more sensitive in this configuration as compared to the 'LC' and 'CC' runs.

The locked mechanism shows a duration of the initial impact of 0.2 seconds. The compliant one roughly doubles the total contact duration as it also releases the spring energy. The duration of the initial impact in rigid mode is only around 0.04 seconds, i.e. a factor 5 smaller. Hence, the frequency spectrum of the disturbance is in a range at which the attitude controller is much more sensitive. Since the total impulse to slow down the aerial manipulator is similar, the magnitude of the force is factor 5 larger. Both the shortened duration and the increased magnitude contribute to the severity of the disturbance on the pitch angle. Moreover, the sudden change in attitude results in a large control action from the servo, which generates an additional backward moment.

Impact absorption:
From the trajectories of the UAV position in the direction normal to the wall, it can be clearly observed that in the locked configuration no bouncing occurs as opposed to both the compliant and the rigid modes. This supports the analysis done in Section III-D. All results are consistent, except for the 'CD' and 'RD' configurations, which require additional interpretation.

For the compliant-down ('CD') experiment, the UAV does not bounce back as much as the other compliant configurations ('CC' and 'CU'). Due to the forward pitch that is induced during the impact, the UAV is launched away from the wall in with a vertical velocity component. Since the height is no longer constant the gravitational potential energy should be included in the energy balance. Part of the stored energy is converted into height energy, reducing the amount that is returned as kinetic energy. Hence, the UAV does not bounce back quite as far compared to the other compliant runs. Nevertheless, the end-effector briefly looses contact before a second contact is registered. In the rigid-down ('RD') experiment, this effect is even stronger, resulting on the UAV overshooting the wall and crashing behind it.

The rigid configuration does not provide any means of storing the initial kinetic energy. Hence, part of it is converted to angular kinetic energy and the remaining part is simply reflected as linear momentum again. The abrupt change of velocities results in severe control actions which destabilize the system. The uncontrolled motion in presence to the wall resulted in inevitable crashes. The compliant mechanism does provide a temporary energy buffer. However the results showed that, in combination with the induced moments in the non-centred configurations, still no stable flight behaviour is obtained. Hence, it is concluded that the extraction, or dissipation of the kinetic energy during impact is crucial, as was suggested from the analysis.

V. Conclusions

The generic method derived in Section II lead to a set of global design consideration for aerial manipulators. Despite their possible obviousness, they articulate key aspects that should be optimized relative to the desired goal. Subsequently, the analysis was tailored to the case of aerial impacts, which brought to light two crucial design parameters: the positioning of the contact point w.r.t. effective CoG and the impedance properties of the manipulator in the approach direction. A simple, yet effective prototype was designed that clearly supported these arguments.

More specifically, the following conclusions for handling aerial impacts can be drawn based on the conducted experiments:

- In order to ensure flight stability the induced moments on the UAV must be minimized,
- A compliant manipulator is a necessity in order to survive small positioning errors,
- It is crucial to prevent the stored impact energy from re-entering the system as to prevent bounces,

Fig. 13. The aerial manipulator establishing a stable contact during one of the experiments.

Moreover, it is concluded that the realized prototype was capable of achieving a stable transition into the contact situation. The importance of the identified design parameters are made clear, since only the proposed optimal configuration, i.e. locked-centre ('LC'), was successful. A photo of the system during impact is depicted in figure 13.

In the end, the approach of handling the aerial impact through the design of the manipulator was successful. Despite the additional payload and introduced dynamic disturbances, the UAV was capable of handling the impacts without any further modifications. Nevertheless, we are aware of the improvements that can be made if the UAV's flight controller is aware of the carried manipulator. More specifically, including feed-forward control actions, based on the derived dynamic equations, is expected to improve the flight performance and stability.

VI. Future work

For the developed methods to become profitable in a real scenario, some extensions to the carried out experiments are envisioned. To allow docking into any surface, also inclined or curved ones, an estimation of the surface normal of the target location is required. It is expected that existing mapping techniques can be adapted to provide this functionality. Moreover, an online estimator of the aerial manipulator's CoG and momentum can be employed to determine the optimal end-effector location more accurately.

Besides the improvements to the specific case of impact absorption, we plan to use the developed methods on analysing the contact scenario. Changing the contact constraints from a quasi-static to a fully dynamic perspective should make it possible to employ the full agility of the UAV and perform more demanding tasks.

Combining these developments paves the way for true *aerial robotic workers*, a new class of UAVs, capable of performing physical tasks in hard-to-reach locations at a high level of autonomy.

References

[1] *AEROWORKS*. [Online]. Available: http://www.aeroworks2020.eu/

[2] G. Gioioso, A. Franchi, G. Salvietti, S. Scheggi, and D. Prattichizzo, "The flying hand: a formation of uavs for cooperative aerial telemanipulation," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4335–4341.

[3] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed autonomous robotic systems*. Springer, 2013, pp. 545–558.

[4] M. Bisgaard, A. la Cour-Harbo, and J. D. Bendtsen, "Adaptive control system for autonomous helicopter slung load operations," *Control Engineering Practice*, vol. 18, no. 7, pp. 800–811, 2010.

[5] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, 2011.

[6] M. Bernard and K. Kondak, "Generic slung load transportation system using small size helicopters," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 3258–3264.

[7] A. Albers, S. Trautmann, T. Howard, T. A. Nguyen, M. Frietsch, and C. Sauter, "Semi-autonomous flying robot for physical interaction with environment," in *Robotics Automation and Mechatronics (RAM), 2010 IEEE Conference on*. IEEE, 2010, pp. 441–446.

[8] M. Fumagalli, R. Naldi, A. Macchelli, F. Forte, A. Q. Keemink, S. Stramigioli, R. Carloni, and L. Marconi, "Developing an aerial manipulator prototype: physical interaction with the environment," 2014.

[9] S. Bellens, J. De Schutter, and H. Bruyninckx, "A hybrid pose/wrench control framework for quadrotor helicopters," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2269–2274.

[10] B. Erginer and E. Altuğ, "Modeling and pd control of a quadrotor vtol vehicle," in *Intelligent Vehicles Symposium, 2007 IEEE*. IEEE, 2007, pp. 894–899.

[11] *AR.Drone 2.0*, Parrot. [Online]. Available: http://ardrone2.parrot.com/

[12] P.-J. Bristeau, F. Callou, D. Vissiere, N. Petit *et al.*, "The navigation and control technology inside the ar. drone micro uav," in *18th IFAC world congress*, vol. 18, no. 1, 2011, pp. 1477–1484.

[13] *OptiTrack*. [Online]. Available: http://www.optitrack.com/

[14] *Dynamixel AX-12+*, Robotis. [Online]. Available: http://en.robotis.com/

Appendix I
Planar Simulations

The planar dynamical model described in this appendix functioned as the backbone for simulating the behaviour during the impact under various design configurations. Many adaptation were made to try out, debug or verify different conceptual designs. The models are represented as bond-graphs [1], which were developed, simulated and animated using 20-sim [2].

Note that the double-lined arrows and multibonds represent a triple of signals and power-conjugated variables respectively. That is, each double-lined arrow is associated with the variables (s_x, s_y, s_r) and each multibond with $\{(e_x, e_y, e_r), (f_x, f_y, f_r)\}$ denoting respectively the efforts and flows. The subscripts denote the orthogonal planar translations (x, y) and the planar rotation (r).

The developed model is cut into 3 parts which are successively introduced hereafter. Then, a small selection of simulation results is presented as to demonstrate how the models can be used in the design process. Finally, the limitations of this model are discussed and several extensions which can be made in future work are discussed.

I-A. Main kinematic structure

The central part of the model is the kinematic structure depicted in figure I-1. The one-junctions on the left represent the position and orientation of the UAV's CoG, expressed in either the body-fixed or the world frame. A modulated transformer (`MTF`) implements the conversion between the reference frames, which in this causality results in the following equations:

$$\texttt{MTF}: \begin{pmatrix} \cos(r) & -\sin(r) & 0 \\ \sin(r) & \cos(r) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{I-1}$$

Note that the modulation variable comes from the body-fixed frame. Since this side has a flow-out causality, also the corresponding q-variable is available (which can be extracted using a q-sensor element that outputs the integral of the flow).

The inertia of the UAV is modelled using an I-element, which is constant in (and thus expressed in) the body-fixed frame:

$$\texttt{I}: \begin{pmatrix} m_b & 0 & 0 \\ 0 & m_b & 0 \\ 0 & 0 & I_b \end{pmatrix} \tag{I-2}$$

The gravitational pull is expressed in the world frame and only affects the y-direction:

$$\texttt{Se}: (0, -m_b g, 0)^T \tag{I-3}$$

Moreover, the UAV's control inputs are incorporated using a modulated source of effort (MSe), which in this case also includes the saturation that might occur. To do so, the commanded thrust and torque (T_u, M_u) are decomposed in the individual rotor forces f_1 and f_2:

$$\begin{cases} f_1 = T_u/2 - M_u/2d \\ f_2 = T_u/2 + M_u/2d \end{cases} \tag{I-4}$$

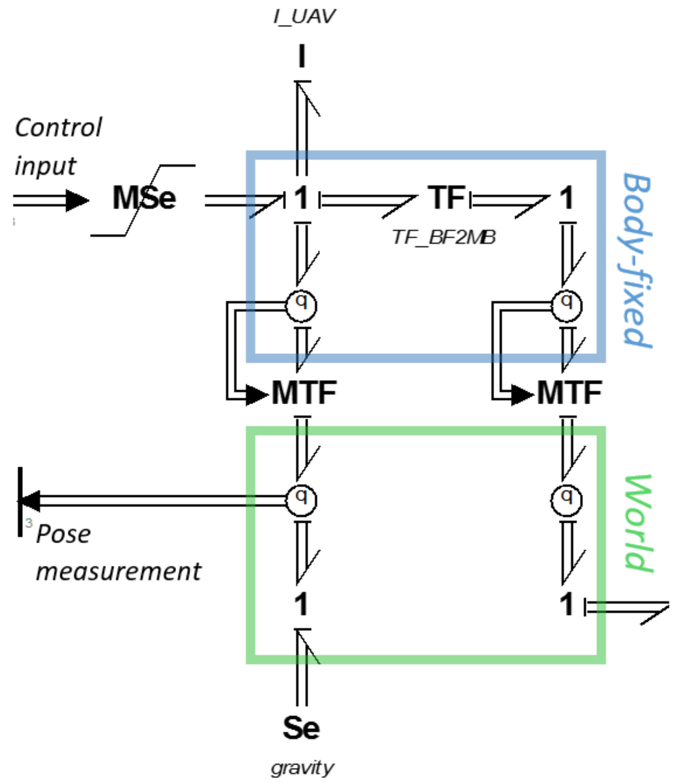


Fig. I-1. Backbone of the model including the UAV kinematics and the interfaces to the controller (fig. I-2) and interaction model (fig. I-3). The body-fixed frame and world frame, in which the variables are expressed, are clearly annotated.

Where d is the half of the distance between the rotors. Each force is limited to the maximum propeller thrust:

$$\begin{cases} \hat{f}_1 = \text{limit}(f_1, \text{-}f_{max}, f_{max}) \\ \hat{f}_2 = \text{limit}(f_2, \text{-}f_{max}, f_{max}) \end{cases} \tag{I-5}$$

Subsequently, the total thrust and moment after saturation are reconstructed, yielding the output of the element:

$$\texttt{MSe}: \begin{pmatrix} 0 \\ \hat{f}_1 + \hat{f}_2 \\ d(\hat{f}_2 - \hat{f}_1) \end{pmatrix} \tag{I-6}$$

The other transformer (`TF_BF2MB`) models the offset between the UAV's CoG and the manipulator's base position, i.e. p_m^b. As this offset yields a fixed transformation in the body-fixed frame, it is not modulated. The associated transformation matrix is given by:

$$\texttt{TF}: \begin{pmatrix} 1 & 0 & p_{m,x}^b \\ 0 & 1 & p_{m,y}^b \\ 0 & 0 & 1 \end{pmatrix} \tag{I-7}$$

Also the manipulator's base location is converted to world coordinates, using the same modulated transformer (eq. I-1). The unconnected bond in the bottom-right enters the interaction submodel, which is described in Section A-C.

I-B. Controller

In order to simulate the scenarios, we need a (simple) flight controller as well. Figure I-2 depicts the controller model, consisting of an altitude and an attitude loop. Note that the lateral position is not controlled directly.

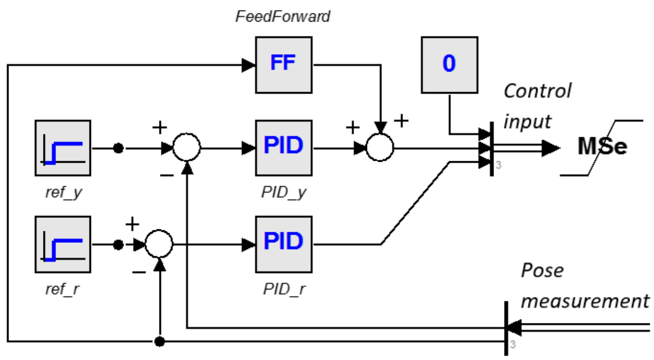


Fig. I-2. The UAV's controller, consisting of two separate loops for the altitude and attitude.

The altitude controller consists of both a feedback (PID) and a feed-forward (FF) action. The latter is applied to compensate for the effect of gravity, which is results in:

$$F_{\text{FF}} = \frac{mg}{\cos(r)} \tag{I-8}$$

The reference signal is simply the desired height in world coordinates, which is taken constant throughout most simulations.

The attitude controller employs only a feedback action. In general it does not make sense to manually prescribe an attitude signal. However, in this case we want to approach and contact the wall at a constant angle of attack, which can be done easily. The final goal is to push against the wall with a certain constant force (F_c), which can only be generated by the UAV as the horizontal component of the total thrust. The total thrust is already determined by the altitude controller, which in absence of disturbances converges to the feed-forward part. Hence, the following attitude setpoint (r_d) can be derived:

$$r_d = \tan^{-1}\left(\frac{F_c}{mg}\right) \tag{I-9}$$

Both feedback loops include a PID-controller which is parametrized under the assumption that the plant behaves as a freely floating mass, i.e. $P(s) = 1/ms^2$ [3]. Let's consider a PD-controller in its serial form as it allows for easy analysis in the frequency domain:

$$C(s) = K_p \frac{\tau_z s + 1}{\tau_p s + 1}, \quad \tau_z > \tau_p \tag{I-10}$$

The following analysis parametrizes the controller, based on a desired closed-loop bandwidth and the inertia experienced at the output. That is, we derive K_p, τ_z and τ_p as a function of ω_c and m.

We know the total system will behave as a mass-spring-damper, which is a second order system. The resonance peak limits the achievable bandwidth and is crucial in terms of stability. The maximum phase-lead ($\angle_{\max}$) is obtained at the logarithmic centre between the pole and the zero, which where we want our crossover frequency to be:

$$\omega_c := \omega_{\angle_{\max}} = \frac{\log(1/\tau_z) + \log(1/\tau_z)}{2} = \sqrt{\frac{1}{\tau_z \tau_p}} \tag{I-11}$$

The amount of phase-lead is directly related to the distance between the pole and the zero. Hence, we introduce the tameness factor β to parametrize this distance:

$$\beta := \tau_z / \tau_p \tag{I-12}$$

Note that since $\tau_z > \tau_p$ we must choose $\beta > 1$. As a rule of thumb we take $\beta = 10$, which in general provides sufficient damping.

From the definition of the crossover frequency and substitution of eq. I-10 through I-12 we find:

$$\begin{aligned} 1 =: |L(j\omega_c)| &= |C(j\omega_c)P(j\omega_c)| = \\ &= K_p \frac{\tau_z j\omega_c + 1}{\tau_p j\omega_c + 1} \frac{1}{m\omega_c^2} \\ &= K_p \sqrt{\beta} \frac{1}{m\omega_c^2} \end{aligned} \tag{I-13}$$

In conclusion, the following parameters were derived for the PD-controller:

$$\begin{cases} K_p = m\omega_c^2 / \sqrt{\beta} \\ \tau_z = \sqrt{\beta} / \omega_c \\ \tau_p = 1 / \sqrt{\beta}\omega_c \end{cases} \tag{I-14}$$

Adding an I-action requires an additional constraint. Since we do not want the integrator to deteriorate the previous loop-shaping, we separate them in the frequency domain by a factor 3, i.e.:

$$\tau_i := 3\tau_p \tag{I-15}$$

I-C. Interaction

The interaction model is depicted in figure I-3. Let's go through the model from left to right, starting at the incoming bond which is connected on the right-hand side of figure I-1, i.e. the manipulator base in world coordinates.

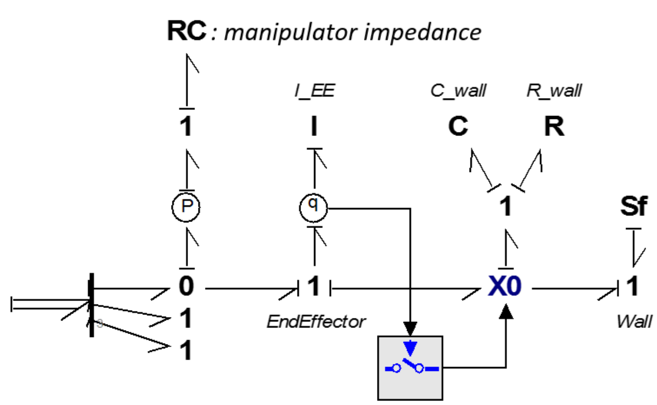


Fig. I-3. The interaction model consisting of the manipulator dynamics and the environment.

The manipulator is assumed to remain horizontal independent of the attitude of the UAV. The control actions required for this in reality are not modelled explicitly, instead the multibond is 'demuxed' and the y and r bonds are connected to an open one-junction. This means that they can impose any flow without receiving any effort in reaction.

The x-direction is connected to the end-effector via a zero-junction, which computes the relative velocity between the base of the manipulator and the end-effector. This relative velocity is used to model the manipulator's impedance, which is depicted as an `RC`-element, but can in fact model any impedance function. Many different implementations were tested, of which a few are described in Section A-D. A power sensor is included to easily access the power extracted or inserted by the manipulator.

The end-effector is modelled as a mass. This is not only to model the effects of the moving mass of the manipulator, which is not very relevant in these simulations, but also to provide the right causality to the `RC`-element, which is a function of the flow-variable.

The collisions with the wall are included as a conditional zero-junction (`X0`), which is switched based on the end-effector position. If enabled, i.e. $x_{\text{EE}} > x_{\text{wall}}$, the `R` and `C` elements are engaged, which generate the walls reaction forces:

$$\begin{aligned} &\texttt{C\_wall}: k_w(x_{\text{EE}} - x_{\text{wall}}), && \text{if } (x_{\text{EE}} - x_{\text{wall}}) > 0 \\ &\texttt{R\_wall}: R_w(\dot{x}_{\text{EE}} - \dot{x}_{\text{wall}}), && \text{if } (\dot{x}_{\text{EE}} - \dot{x}_{\text{wall}}) > 0 \end{aligned} \tag{I-16}$$

Note that the second condition is not automatically satisfied by the conditional zero-junction. Nevertheless, it is crucial since it assures that the wall is not 'sticky'. Without it, the damping might result in a negative force, i.e. towards the wall, which not realistic in most situations.

Lastly, the wall is constrained through a zero-valued source of flow, ensuring zero motion.

I-D. Results: simulation and animation

In order to demonstrate how the model can be used throughout the design process, a few of the simulations that were made are described hereafter. Of course many other variations have been simulated which are not included here. Also an animation was made that graphically represents the output of the model. Stills of this animation are included here and videos are made available together with this report.

Firstly, the height of the manipulator with respect to the UAV's CoG is varied, to demonstrate the induced moments on the UAV. The impedance profile for this simulation resembles a perfectly tuned spring-damper. Figures I-4 and I-5 display the results of this simulation, which indeed show the expected behaviour. A positive offset, i.e. on top of the UAV, induces a backwards rotation, causing the UAV to propel away from the wall. A negative offset has the opposite effect, resulting in the UAV crashing into the wall. If the contact point is perfectly centred, no rotation is induced at all. Furthermore, it can be seen that a well-tuned spring-damper is capable of absorbing the impact without any bounces. However, the positive offset does bounce, which is completely due to the induced moment that tilts the UAV away from the wall. Only after the controller recovers from this disturbance it approaches the wall again.

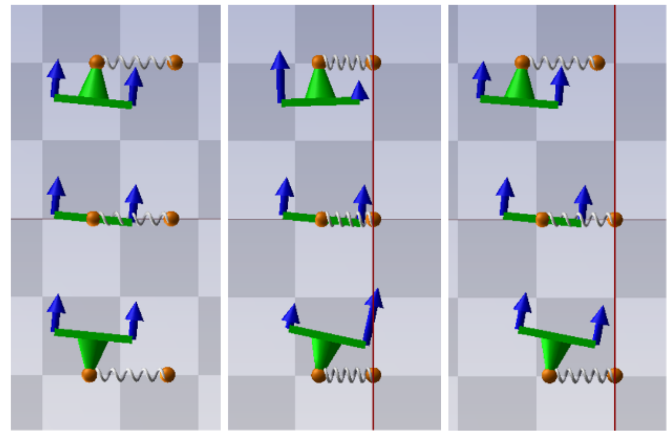


Fig. I-4. Stills of an animation of the comparative height simulation at $t = \{1.5, 1.9, 2.3\}s$ from left to right. The offset varies from top to bottom. The controller's reactions to the induced moment are clearly visible in the centre snapshot.

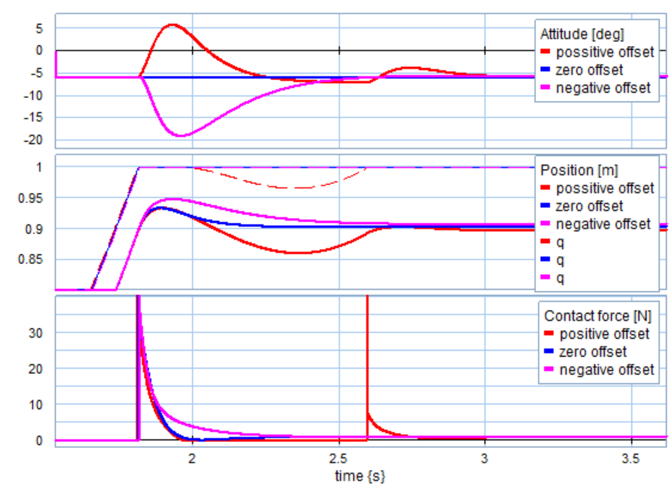


Fig. I-5. Plots of the comparative simulation. The first plot shows the attitude, with negative values corresponding to a nose-down pitch. The second plot shows the UAV (solid) and end-effector (dashed) positions. The last plot shows the contact force.

Secondly, the results of a comparative simulation in which the impedance mode is varied are depicted in figures I-6 and I-7. Note that the manipulator modes used here do not correspond to the ones described in the paper, but resemble the ones discussed in the next appendix.

The first run (red) has a rigid manipulator, which bounces several times. The contact force is a short impulse ($< 0.01s$) of more than $1kN$. Note that the collision model used here is not perfectly elastic, hence some energy is lost at every bounce. Since the manipulator is perfectly positioned at the height of the CoG, no moments are induced and the attitude remains perfectly constant throughout the simulation. Therefore the acceleration towards the wall remains constant as well, yielding a saw-tooth shape for the velocity.

The second run (green) features a compliant manipulator which is perfectly damped. Hence, the results correspond to

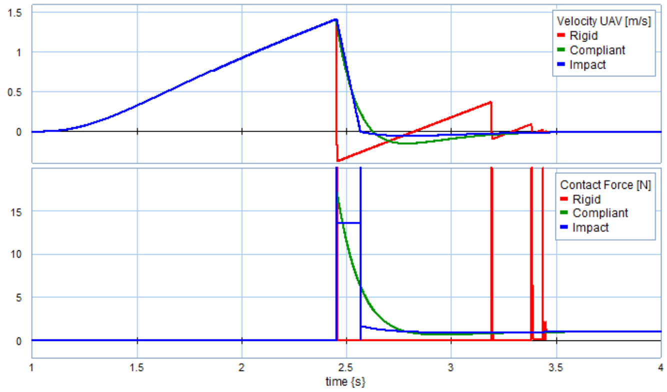


Fig. I-6. Plots of the comparative impedance simulation. The first plots shows the UAV's velocity. The second plot shows the contact forces.

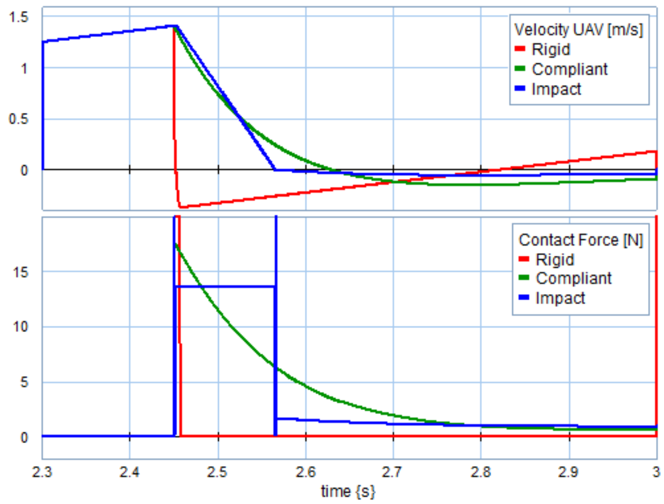


Fig. I-7. Close-up of the plots in figure I-6.

the centred cases of the previous simulation experiment. A close-up of the contact force shows the high initial contact force. This is due to the high relative velocity at impact, for which the behaviour is dominated by the damping component.

The third run (blue) shows the constant force impact controller, which is described in more detail and motivated in the next appendix. The velocity plot clearly shows the linear deceleration. Also the maximum required contact force is lower than for the compliant manipulator, whereas roughly the same behaviour is obtained. After about $0.1s$ the velocity falls below the threshold and the controller switches to the 'contact-case', which is identical to the spring-damper behaviour of the compliant manipulator mode.

I-E. *Discussion and conclusion*

The model turned out to be very useful in analysing the contact scenarios and simulating the effect of different design parameters. To gain further understanding of what happens when using a real manipulator, e.g. one that is not perfectly aligned horizontally, the model should be extended. To do so, I would advice to model the real kinematics and the appropriate controllers of the manipulator explicitly. Most likely, this would be done in the body-fixed frame, hence the interconnecting bond would move to the top one-junction in figure I-1

The current model is easily reusable and extendible for as long as the main characteristics can be captured in a planar setting. I expect that most of the future work can be done so. When considering a full 6D model, it can be structured similarly, however some major adaptations would be required throughout the model.

References

[1] G. Golo, v. d. A. Schaft, P. C. Breedveld, B. M. Maschke, R. Johansson, and A. Rantzer, "Hamiltonian formulation of bond graphs," *Nonlinear and Hybrid Systems in Automotive Control*, pp. 351–372, 2003.

[2] *20-sim*, Controllab Products. [Online]. Available: http://www.20sim.com/

[3] R. Munnig Schmidt, G. Schitter, and J. van Eijk, "The design of high performance mechatronics," 2011.

APPENDIX II
PRELIMINARY EXPERIMENT: 1D CART

In order to better understand the absorption of impact energy and the influence of the manipulator's impedance profile, a preliminary test setup was developed. The experiment yields the ultimate simplification of the aerial impact case, restraining it to a single dimension. This appendix firstly introduces the simplifications made and the consequential analysis. Subsequently, the overall test setup is sketched, after which the mechanical design and software implementation are described in more detail. Lastly, a qualitative interpretation of the obtained results is made.

II-A. Simplification and analysis

When reduced to the bare minimum, the impact of the aerial manipulator can be viewed of as 2 floating masses, representing the UAV and the end-effector mass as sketched in figure II-1.

In the real system, the UAV's lateral dynamics are only indirectly controllable, due to the underactuated nature of multirotor vehicles. This can be included in the source of force that represents the UAV control, e.g. using a transfer function that mimics the rotational inertia. However, as the goal is to characterize the impact dynamics, it is assumed that the UAV can be force-controlled in the lateral direction. Note that the floating-base behaviour is conserved, since no position control is implemented on the mass that represents the UAV.

The manipulator is also represented by a source of force, which is modulated based on the stroke, i.e. the relative position of the end-effector with respect to the UAV. Because of using impedance control, i.e. an active solution as opposed to passive components such as springs and dampers, different impedance profiles can tested and compared easily.

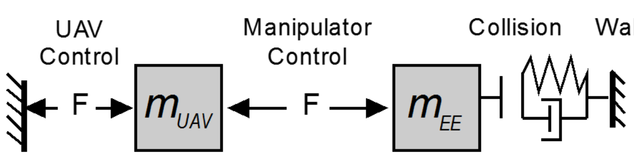


Fig. II-1. A minimalistic representation of the aerial impact scenario.

II-B. Hardware design

The test setup consists of a 1 DOF cart, representing the UAV, equipped with a 1 DOF manipulator. Both DOFs are linear and oriented in parallel, such that the setup represents the proposed simplified model.

A $80 \times 40\ mm^2$ Boikon profile was used as a rails to guide the cart, such that a linear DOF is obtained that allows for a long stroke ($\approx 1\ m$). An inventive design, which could be prototyped rapidly by using only lasercutted parts and off-the-shelve bearings, was made to properly constrain the cart to the track (fig. II-2).

The linear DOF of the manipulator is obtained using a sheet-drum transmission (fig. II-3). It is inspired on a rack-pinion

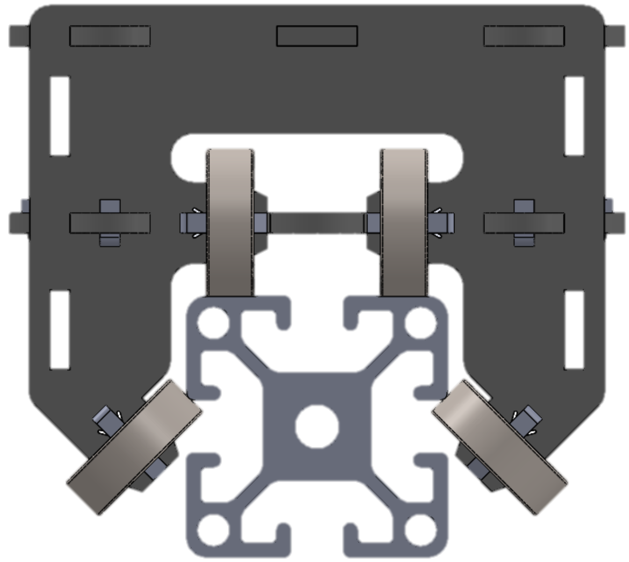


Fig. II-2. Front view of the cart mounted on the rails.

type of transmission, where the rack is replaced by a slider and the pinion by a slider. A thin high-strength steel strip is attached to the front of the slider, then wraps around the drum and is finally connected to the pre-tensioning mechanism on the rear of the slider. The slider is guided by 4 flange bearings. This mechanism provides a rotation-translation transmission that:

- assures the no-slip condition,
- eliminates backlash,
- has low friction,
- does not suffer from wear or ageing.

The impedance-based controller requires the drive chain to be back-drivable. Hence, a Maxon DC-motor is placed on the drum without an intermittent gearbox. The motor is strongly over-dimensioned in terms of power, as to generate sufficient torque in this direct drive configuration. The cart is driven through a timing-belt which is lead back through the Boikon profile with a pulley on either side. Again a Maxon DC-motor is placed in direct drive on one of the pulleys. The motors are controlled using a Pololu motor driver (18v15), with a peak

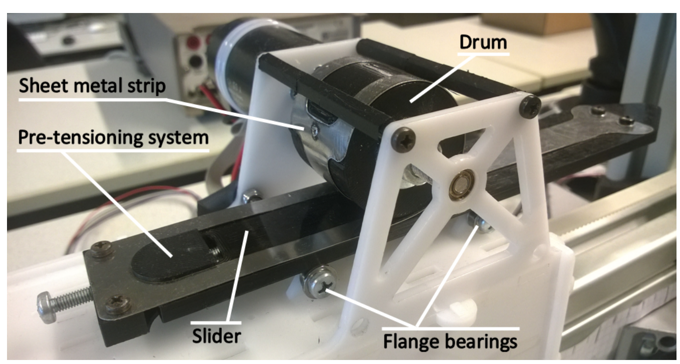


Fig. II-3. A close-up of the sheet-drum transmission, annotating the relevant components.

output current of 21 Amperes.

Both motors are equipped with an encoder for position measurements. An Atmel AVR microprocessor on board the Arduino Mega 2560 closes the control loop. Furthermore, a Ubuntu desktop is used to run ROS, which provides quick access to many useful features, e.g. hooking up a joystick to control the cart, communication to the Arduino and dynamic parameter updates.

II-C. Software design

The AVR microprocessor is used to read out the encoders, estimate positions and velocities, control the motor drivers and implement a safety layer with pre-defined end stops. Moreover, it communicates over a serial connection with ROS to receive setpoints or update parameters.

Timing issues:
Some effort was made to mechanically minimize friction throughout the setup. Moreover, using the motors in direct drive and the usage of lightweight materials, lead to a highly dynamic setup. This requires to decrease the controllers loop time accordingly, in order to guarantee stability.

The developed software prioritizes the tasks at hand. The first priority is to register encoder ticks, which is handled in an interrupt service routine, thereby getting the required priority. The second priority is assigned to the control loop, which must run at a high rate ($\approx$ 1 kHz) as near to real-time as possible. This achieved by executing the control actions in another interrupt service routine, which is triggered by one of the on-board hardware timers. Since the encoder interrupts are of higher priority, they are explicitly re-enabled in the control routine. All other tasks, in this case the serial communication, are executed in the main loop, which serves as the lowest priority task. During the experiments it ran up too 400 Hz, however with much larger deviations from the average loop time. Throttling down the main loop to e.g. 100 Hz provided more consistently timed communication, still at a sufficient rate for this experiment.

A code snippet that act as an example is included hereafter. A more detailed tutorial is also available on the Aeroworks trac-page [1].

```
/*
*  Setup hardware timer
*/
cli(); // disable global interrupts

TCCR5A = 0;  // set entire TCCR3A register to 0
TCCR5B = 0;  // set entire TCCR3B register to 0
TCNT5  = 0;  // initialize counter to 0
TCCR5B |= (1 << WGM52);  // CTC mode
TCCR5B |= (1 << CS50);   // Prescaler = 1
TIMSK5 |= (1 << OCIE5A); // Timer compare interrupt

//Set compare match register to desired timer count:
OCR5A = 16*LOOPTIME;           //16(MHz)*LOOPTIME(us)

sei(); // enable global interrupts
}

/*
*  Interrupt service routine
*/
ISR(TIMER5_COMPA_vect)
{
  //allow the encoder interrupts to be handled
  sei();

  //update position and velocity estimates
  computePosVel();

  //direct force control on the cart
  pwm[CART] = ref[CART];
  setPWM(CART);

  //PD-control on the manipulator
  pwm[MANI] = Kp*(pos[MANI] + ref[MANI])
             + Kd*vel[MANI];
  setPWM(MANI);
}
```

Encoder based velocity estimation:
The motors are equipped with incremental encoders, which provide a quantized position measurement. With a resolution of 500 ticks per revolution, this yields a position resolution of approximately 4 ticks per millilitre, which is sufficient for this experimental setup. The velocity can be estimated in a discrete way by dividing the difference of the positions measurement by the time step of the control loop (Ts). However, the results are not very satisfactory, especially on low velocities.

To indicate the problem, let's crunch some numbers. Assume that the loop runs at 1 kHz. For a velocity of 50 mm/s, this results in approximately 200 encoder ticks per second. Every loop (i.e. every millisecond) we compute the position difference, which will be 0.2 encoder ticks per loop. However, this encoder measurement is quantized, so instead of measuring 0.2 at every loop, we get a measurement of 1 followed by 4 measurements of zero. The velocity (v=delta/Ts) will thus give a spike of 1000 ticks (or 250 mm) per second, once every 5 loops. On average this is correct, but for control purposes it is deficient.

One solution is to compute a running average over the last N-samples. Figure II-4 depicts this method and clearly shows the introduced delay that impairs this method. The plot shows the computed finite-difference velocity (v1) and running averages of 10 (v2) and 100 (v3) samples. The issue is that you would like to increase the width of the filter (such that your resolution increases) when the velocity is low, but decrease N for high velocities to reduce the amount of delay introduced by the averaging. A technique named 'adaptive window filtering' finds the optimal solution to this trade-off, and is described in [2]. However, this method is too computationally expensive to run at such a high rate on the Arduino.

The approach pursued here is based on an inverse time method: instead of computing the position difference at a fixed rate, the time between subsequent encoder ticks is measured. On the 16 MHz Arduino Mega the function micros() has a resolution of 4 microseconds, resulting in a velocity resolution of up to 250000 ticks per second, or 62.5 m/s in case of our example. Note that we are now talking about an upper limit, because for high velocities the time measurement deteriorates. This range is more than sufficient for the setup at hand. A

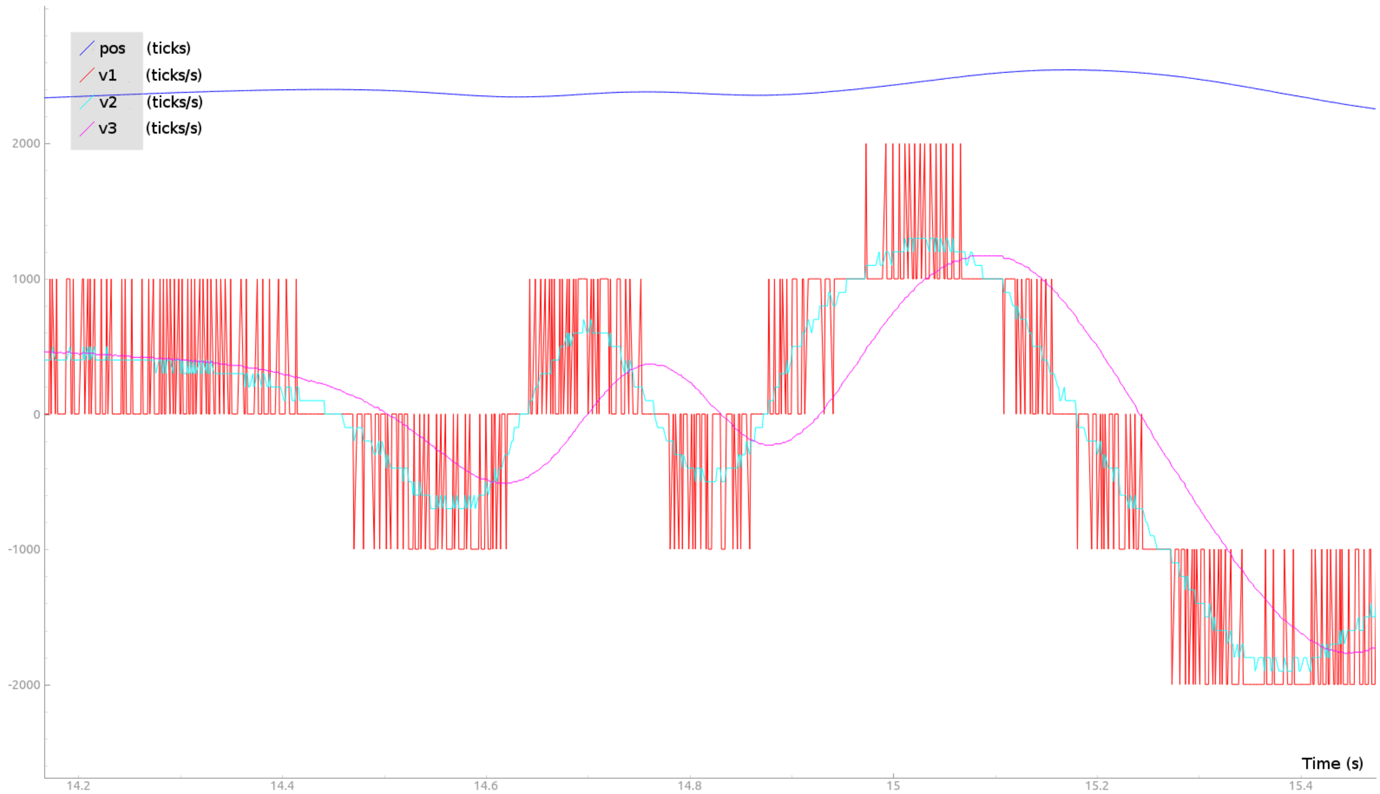


Fig. II-4. Encoder based velocity estimates using a running average filter. Note the trade-off between resolution and introduced delays, making this method deficient for the setup at hand.

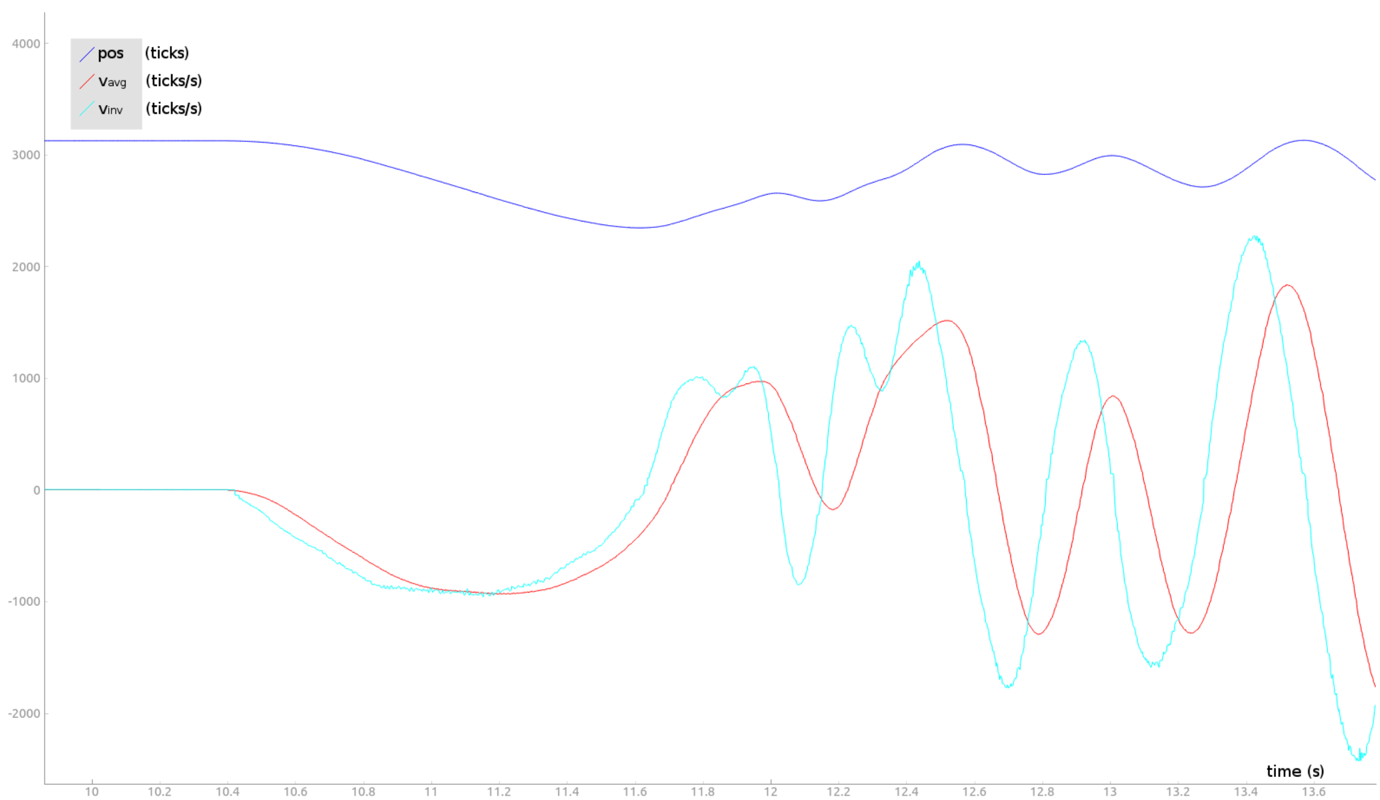


Fig. II-5. Encoder based velocity estimates using a running average filter and the proposed inverse time filter. Note that the latter is much more responsive to changes in velocity without giving in on the resolution.

standard implementation of the inverse time method is given by:

$$\hat{v} = \frac{dx}{\Delta t} \tag{II-1}$$

where dx denotes the position resolution of the encoder, i.e. the stroke per encoder tick. The duration of the last encoder step $\Delta t = t_{enc}^{k-2} - t_{enc}^{k-1}$, where t_{enc}^{k-1} and t_{enc}^{k-2} denote the time at which the last two encoder ticks are registered.

However, one important practical issue is overlooked in this implementation, which occurs when the velocity is (almost) zero? In this situation there are no new encoder ticks registered, so the encoder timestamps remain unchanged. This means that the algorithm 'remembers' the last non-zero velocity estimate, which is undesired if the estimate is used for control purposes. The issue can be resolved in a neat way by incorporating a limited validity of each estimate. If within Δt after the last encoder tick no new interrupt was generated, this means the velocity must have decreased. The time difference between the last encoder tick and the current time (t_{now}) therefore gives an upper bound to the allowed velocity estimates. In the situation that no new encoder ticks are registered, the velocity estimate would asymptotically approach zero as t_{now} goes to infinity. This notion can be included as follows:

$$\hat{v} = \frac{dx}{\max(\Delta t, t_{enc}^{k-1} - t_{now})} \tag{II-2}$$

Figure II-5 depicts the velocity estimates obtained by the running average with 100 samples (Vavg) and the inverse time method (Vinv). Also in the control action, the inverse time method clearly outperformed the other approaches.

II-D. Results

The experiment provided a great initial understanding of impact behaviour. However, simulations pointed out that the coupling to the rotational dynamics of the UAV are crucial to guarantee stability. Hence, the current setup oversimplifies the scenario for a proper, quantitative analysis of the results. Nevertheless, several impedance profiles were tested and their influence on the impact dynamics were compared. Two of them are briefly discussed hereafter.

PD control:
A PD controller, i.e. spring-damper, gave acceptable results, however required to be tuned correctly to the dynamic parameters of the system, as well as the impact velocity. This might be infeasible in practice, especially if the behaviour is obtained by passive mechanical components. Recent developments of variable stiffness actuators might suit the requirement for online tuning of the compliance. However, adding an independently variable damping action would make the system undesirably complex. The current approach of fully active impedance control is also infeasible for the aerial manipulator, as the required motors are way outside the payload capabilities of the UAV at hand.

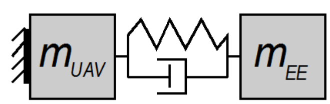
Constant force:
The weight of the motors is thus a problem for airborne applications. If the required torque would be lower, smaller motors can be selected which might result in a feasible solution. This initialized the idea of absorbing the impact by generating a constant force. Moreover, it would gradually slow down the UAV, which is expected to minimize disturbances on the flight control.

Let's first divide the scenario into 3 subsequent stages;

- **Free-flight:** the UAV behaves as a freely floating mass without any contact forces while approaching the wall,
- **Impact:** the manipulator establishes contact with the wall, while the UAV still has a significant kinetic energy,
- **Contact:** the energy of the system is reduced, system remains in contact.

The dynamics of the system change significantly when transitioning from free flight to the contact situation. Therefore the impedance properties should be adapted accordingly, yielding a separate control goal for each stage.

During free flight, i.e. no-contact mode, the controller simulates a compliant spring/damper. The low impedance in this mode increases safety while flying in unknown environments. Moreover, the manipulator can be used to estimate the UAVs impact velocity, assuming that the end-effector remains in contact with the wall, providing an estimate of the kinetic energy at impact. Since in general $m_{UAV} \gg m_{EE}$, the following model approximation holds: The stiffness is denoted



k_1 and the damping parameter is chosen such that the end-effector mass (m_{EE}) is critically damped:

$$F_{m,\text{free-flight}} = k_1 x + 2\sqrt{m_{EE} k_1}\ \dot{x} \tag{II-3}$$

During impact the controller switches to a mode that aims at absorbing the impact energy in an effective way. The energy absorbed by the manipulator is given by:

$$W = \int F_m(x) dx \tag{II-4}$$

In order to minimize the required actuation force, the impact is absorbed using a constant force. The integral can then be omitted:

$$W = F_m \Delta x \tag{II-5}$$

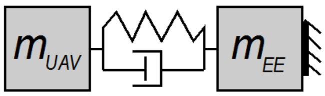
The allowed travel of the manipulator is limited by its workspace. Assume that the total impact energy should be absorbed in 80% of the length of the manipulator (L_m), one finds:

$$F_{m,\text{impact}} = \frac{\frac{1}{2} m_{UAV} v_0^2}{0.8 L_m} \tag{II-6}$$

Finally, once the kinetic energy of the system is reduced below a threshold, the controller switches to PD control again. The stiffness in contact mode is chosen such that the equilibrium position while applying the desired contact force (F_d) is in the centre of the manipulators workspace ($0.5L_m$):

$$k_2 = \frac{F_d}{0.5L_m} \tag{II-7}$$

This ensures the availability of sufficient range of motion to reject disturbances in both directions. The damping parameter is again chosen such that critical damping is obtained, however due to the contact constrained the model simplifies to:



Hence, the damping parameter depends in this case on the mass of the UAV:

$$F_{m,\text{contact}} = k_2 x + 2\sqrt{m_{UAV} k_1}\ \dot{x} \tag{II-8}$$

The switching between the controller modes is handled by the state machine depicted in figure II-6. The UAV starts in free-flight mode, transitioning only if the manipulator is compressed by some percentage x. The system always returns the free-flight condition if this condition no longer holds. Moreover, to prevent multiple rapid transitions between two states, a hysteresis threshold (h) is applied. When establishing contact, the system always transitions through the impact state, in which it tries to reduce the kinetic energy of the system. When the energy is below a certain value y, the system enters the contact mode. In practice this condition is evaluated on the estimated velocity, since $E_k = \frac{1}{2} m\, v_0^2$.

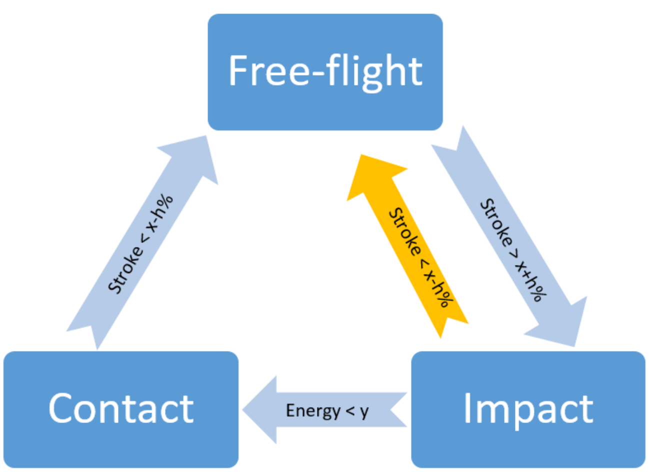


Fig. II-6. State machine implemented to switch between the control modes appropriately.

The proposed control law in combination with the developed state machine yielded a stable controller and the results on the preliminary setup were satisfactory. However, for simplicity and payload considerations the aerial experiments were carried out using a passive mechanism. Once a fully actuated prototype of the aerial manipulator is realized, this approach can be further tested.

References

[1] *RAM trac*, Robotics and Mechatronics, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente. [Online]. Available: https://trac.ram.ewi.utwente.nl/aeroworks/wiki/Devices/ArduinoTutorial

[2] F. Janabi-Sharifi, V. Hayward, and C.-S. J. Chen, "Discrete-time adaptive windowing for velocity estimation," *Control Systems Technology, IEEE Transactions on*, vol. 8, no. 6, pp. 1003–1009, 2000.