Design and control of an aerial manipulator with elastic suspension

Arda Yiğit

UNIVERSITÉ DE STRASBOURG

ÉCOLE DOCTORALE MSII 269
Mathématiques, Sciences de l'Information et de l'Ingénieur

Laboratoire des sciences de l'Ingénieur, de l'Informatique et de l'Imagerie
ICube UMR 7357

THÈSE présentée par :

Arda YİĞİT

soutenue le 7 décembre 2021

pour obtenir le grade de : **Docteur de l'université de Strasbourg**

Discipline/ Spécialité : Robotique

Conception et commande d'un manipulateur aérien avec suspension élastique

THÈSE dirigée par :
M. GANGLOFF Jacques Professeur, Université de Strasbourg

RAPPORTEURS :
M. CARO Stéphane Directeur de recherche CNRS, École Centrale de Nantes
M. FRANCHI Antonio Professeur, Université de Twente

AUTRES MEMBRES DU JURY :
Mme FANTONI Isabelle Directrice de recherche CNRS, École Centrale de Nantes
M. CUVILLON Loïc Maître de conférences, Université de Strasbourg
M. DURAND Sylvain Maître de conférences, INSA de Strasbourg

Arda YİĞİT

Conception et commande d'un manipulateur aérien avec suspension élastique

Résumé

Cette thèse s'intéresse à la conception et à la commande d'un Manipulateur Aérien avec Suspension Élastique (MASE), hybride entre un robot à câbles et un drone omnidirectionnel. Un manipulateur aérien est suspendu par un ressort à un robot parallèle à câbles (RPC), profitant du grand espace de travail du RPC et des performances dynamiques importantes du véhicule aérien. La suspension du véhicule aérien permet de compenser la gravité et ainsi limite la consommation énergétique du système. Différentes stratégies de commande dynamiques non-linéaires ont été implémentées afin d'améliorer encore plus l'efficacité énergétique du MASE. Un prototype a été développé afin d'évaluer expérimentalement le potentiel applicatif du MASE. En plus de l'évaluation expérimentale du prototype, cette thèse a donné lieu à des contributions théoriques sur la stabilité de la commande par couples calculés et sur l'élimination de l'erreur statique avec la commande prédictive non-linéaire.

Mots-clés : robotique aérienne, robotique à câbles, commande non-linéaire, commande prédictive

Abstract

This thesis is focused on the design and control of an Aerial Manipulator with Elastic Suspension (AMES), hybrid between a cable robot and an omnidirectional drone. An aerial manipulator is suspended from a cable-driven parallel robot (CDPR) by a spring, taking benefit from the large workspace of a CDPR and the high dynamics of an aerial vehicle. The suspension of the aerial vehicle compensates for the gravity and consequently improves the energy efficiency of the global system. Different control strategies have been tested to improve the energy efficiency of the AMES even more. A prototype has been developed and its applicative potential evaluated experimentally. Besides the experimental evaluation of the prototype, this thesis yielded theoretical contributions on the stability of the computed torque controller and on offset-free nonlinear model predictive control.

Keywords: aerial robotics, cable robotics, nonlinear control, model predictive control

Acknowledgements

I would like to thank first the members of the jury, Isabelle Fantoni, Antonio Franchi and Stéphane Caro, for accepting to devote their time to evaluating my work.

I would also like to express all my gratitude to my supervisors that helped me during this wonderful adventure. Thank you Jacques for sharing your huge experience with me, for your professional and personal advises, for the freedom you have granted me. I am extremely lucky to have the opportunity to work with a supervisor like you. Thank you Loïc for your friendship, for your involvement, for your patience. I have never met someone as available as you to help others. Thank you Sylvain for the trust you had in me, for your encouragements. You always knew how to improve the mood and make work more pleasant.

I also express my sincere gratitude to the interns with whom I have collaborated: Gustave, Johan, Côme, Luc, Quentin, Miguel, Yannick, Jean D., Jean S., Florian.

Thank you to all the PhD students and all the members of the AVR team for the pleasing discussions, very helpful remarks and for sharing their knowledge. Working in such a large team was a perfect opportunity to discover different ways to do science and to open my mind to different points of view.

These final words are for my parents. I can never thank you enough for everything you did for me until now. Thank you for helping me to make this dream come true.

Contents

List of Figures

List of Tables

Acronyms

AMES Aerial Manipulator with Elastic Suspension.

AWG Aerial Wrench Generator.

CDPR Cable-Driven Parallel Robot.

CoM Center of Mass.

CTC Computed Torque Control(ler).

DoF Degree of Freedom.

EKF Extended Kalman Filter.

ESC Electronic Speed Controller.

MPC Model Predictive Control(ler).

NMPC Nonlinear Model Predictive Control(ler).

OCP Optimal Control Problem.

PD Proportional-Derivative.

PID Proportional-Integral-Derivative.

SAM Suspended Aerial Manipulator.

TOW Total Orientation Workspace.

UAM Unmanned Aerial Manipulator.

UAV Unmanned Aerial Vehicle.

VRPN Virtual-Reality Peripheral Network.

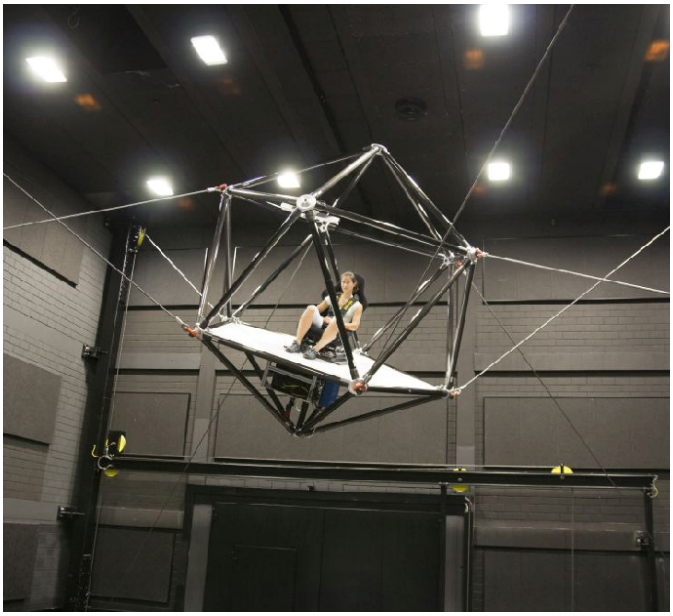
1 Introduction and State of the Art

1.1 Context and Motivations

Cable-driven parallel robots (CDPRs) are parallel robots in which leg segments have been replaced by cables. Therefore, they are lightweight, cost-effective and can work in large workspace. However, their design and control are complex because of many singular configurations and possible collisions with the cables. They also suffer from low stiffness, especially due to long cables. On suspended CDPRs, gravity acts as an additional cable with constant direction and constant tension. Consequently, suspended CDPRs have limited downward dynamics. Figure 1.1 shows two examples of CDPRs, a large-scale motion simulation platform (1.1a) and a cable-suspended camera to broadcast sporting events from a stadium (1.1b).

The LIRMM lab at Montpellier and the ICube lab collaborated on the DexterWide project (ANR-15-CE10-0006). The DexterWide project aimed at equipping a CDPR with an industrial robotic arm for applications involving dexterous tasks over large workspaces. The CoGiRo robot (see Figure 1.2) was built at LIRMM during this project (Gouttefarde et al., 2015). To reduce disturbances created by the robotic arm movements on the CDPR, ICube proposed to use additional actuators on the moving platform to perform active stabilization. The PiSaRo4 robot (see Figure 1.3) was built in order to test the stabilization techniques on a simpler system, a planar CDPR suspended by springs to emulate the flexibility of long cables. Besides three winches, it embeds four propellers that can generate any planar wrench (two forces and one torque). The added value of the additional actuators are assessed experimentally with respect to the winch-only actuation (Khayour et al., 2020).

The PiSaRo4 robot is thereby overactuated, and one may ask whether the cables are



(a) CableRobot simulator (Miermeister et al., 2016).

(b) SkyCam (“SkyCam”, n.d.).

Figure 1.1 – CDPR examples.

Figure 1.2 – CoGiRo robot (Gouttefarde et al., 2015).

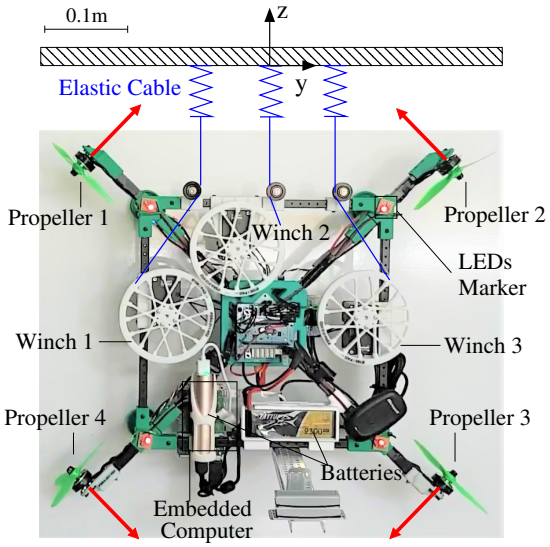


Figure 1.3 – PiSaRo4 robot (Khayour et al., 2020)

still mandatory. Without the cables, this robot can be considered an aerial vehicle under the condition that the propeller thrust suffices to hover, compensating for the gravity. Aerial vehicles suffer from low autonomy (the flight time of a quadcopter is usually less than 30 minutes) since gravity compensation requires important power consumption. Most aerial vehicles use LiPo batteries because of their specific power (the amount of maximum power that can be delivered per unit of mass) high enough to hover, but suffer from relatively low specific energy, limiting the flight time. The Ragone plot from Figure 1.4 illustrates this trade-off and compares different battery technologies.

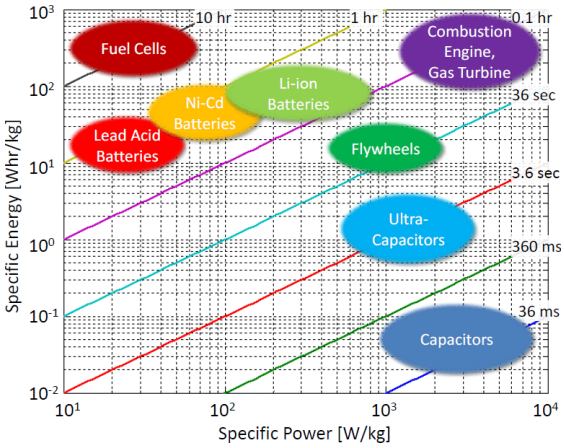


Figure 1.4 – Ragone plot (figure from (Moura et al., 2010)).

We can therefore retain the idea that at least one cable is needed in order to compen-

sate for the gravity, allowing for a higher autonomy. This thesis introduces a new robot, inspired by this idea of suspending an end effector by only one cable and actuating it with propellers.

This thesis was supported by the e-VISER project funded by the French National Research Agency (ANR-17-CE33-0008).

1.2 Aerial Manipulation

1.2.1 Aerial Manipulators

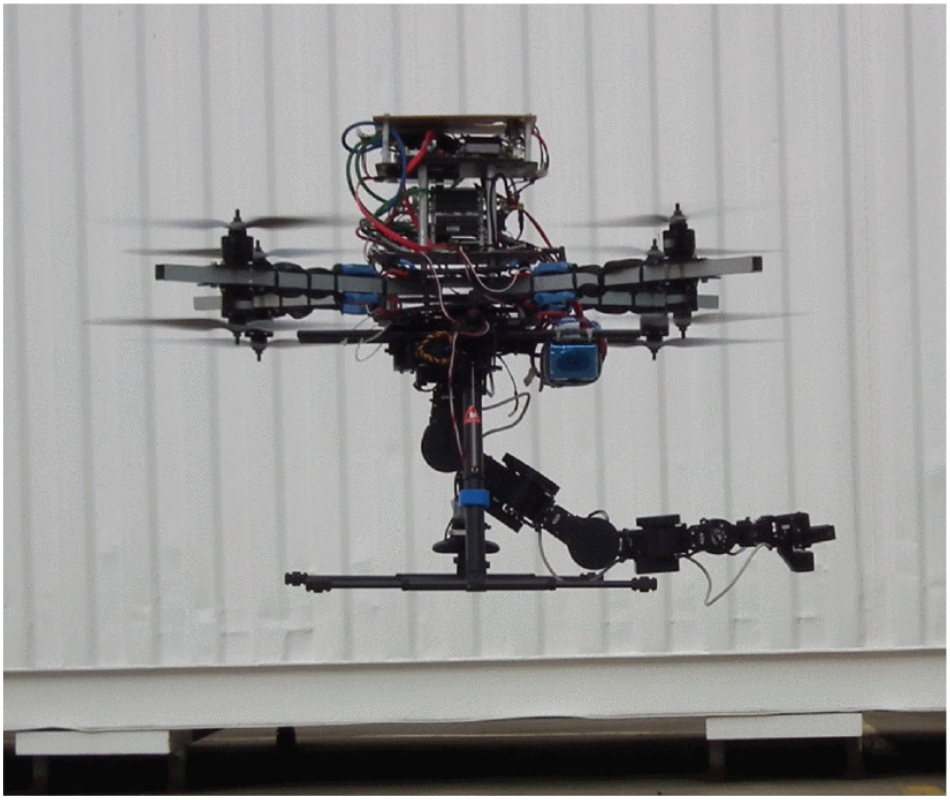
For active robotic tasks such as grasping, positioning or assembly of mechanical parts, unmanned aerial vehicles (UAVs) need to be equipped with appropriate tools. According to Ruggiero et al. (Ruggiero et al., 2018), the two most common solutions are flying hands, where a gripper is mounted directly on the aerial vehicle, and unmanned aerial manipulators (UAMs), where the UAV carries one or more robotic arms (see Figure 1.5).

The robotic arm mounted on the UAV allows for improved dexterity of the UAM. The number of degrees of freedom (DoFs) of the robotic arm can be chosen to fully actuate the end effector. Therefore, a UAV with collinear thrusters, like common quadcopters, would need at least a 2-DoF arm. However, even in this case, ensuring the full actuation of the UAM can be challenging. In practice, more complex robotic arms are used for full actuation (Huber et al., 2013). In addition to guaranteeing full actuation, additional DoFs can help for disturbance rejection by moving links while maintaining the end effector static (Danko & Oh, 2013). Nevertheless, the more complex the robotic arm, the heavier the aerial platform and the higher the energy consumption.

Flying hands often use helicopters or collinear propeller UAVs. While they are useful in scenarios involving simple tasks such as grasping and transporting, they have limited manipulation capabilities since they are underactuated. It is possible to improve their dexterity by increasing the number of DoFs of the UAV (Kawasaki et al., 2015).

Multiple UAVs can cooperate to manipulate an object. They can be linked, together and to the end effector, either by rigid articulations (Six et al., 2018) or by cables (Erskine et al., 2019; Tognon et al., 2018) (see Figure 1.6). Combining three or more quadcopters allows for 6-DoF control of the end effector. Note that a similar concept exists in the field of underwater robotics (El-Ghazaly et al., 2015): the hybrid

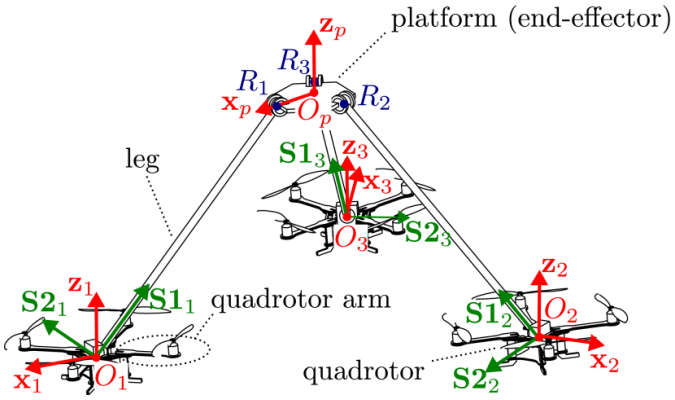
(a) Flying hand (Pounds et al., 2011).



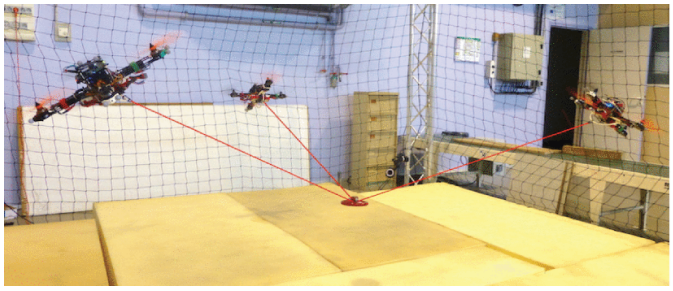
(b) Unmanned aerial manipulator (Heredia et al., 2014).

Figure 1.5 – Aerial manipulator examples.

cable-thruster (HCT)-actuated underwater vehicle-manipulator system (UVMS). The underwater vehicle is actuated both by the vehicle thrusters and cables attached to the offshore platform (see Figure 1.7). The static carrier could also be replaced by multiple moving boats, each holding one cable.



(a) Rigid links (figure from (Six et al., 2018)).



(b) Cable links (Erskine et al., 2019).

Figure 1.6 – UAV cooperation examples.

Linking a UAV to the ground with a tether has many advantages. From a practical point of view, the tether can be used to power the UAV to provide theoretically infinite flight time (Choi et al., 2014). Moreover, for tasks requiring fluids, such as cleaning or painting, the fluid can be fed through the tether (Tognon & Franchi, 2021). The tether can also improve maneuverability, for example enabling landing on sloped surfaces (see Figure 1.8) (Tognon & Franchi, 2017).

1.2.2 Omnidirectional Aerial Vehicles

Omnidirectionality

Commercial UAVs use almost exclusively designs with collinear thrusters such as quadcopters. These UAVs benefit from mechanical simplicity and energetic efficiency.

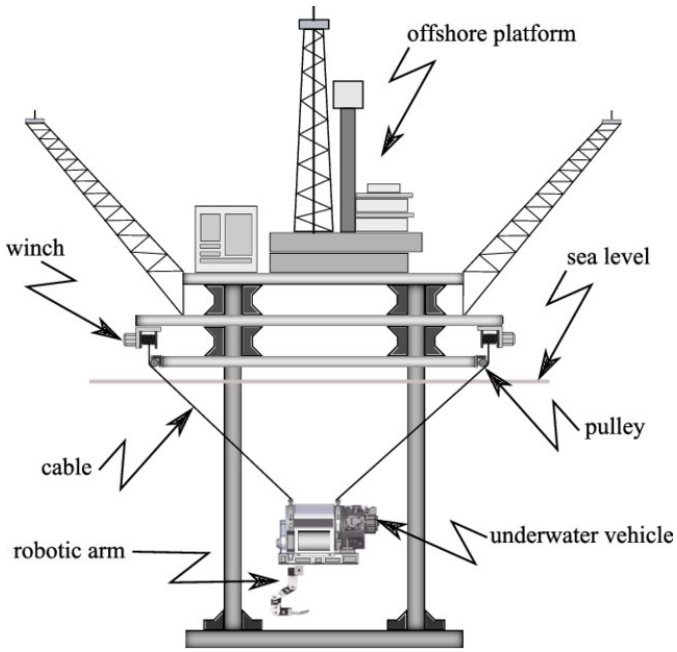


Figure 1.7 – HCT-a-UVMS (figure from (El-Ghazaly et al., 2015)).

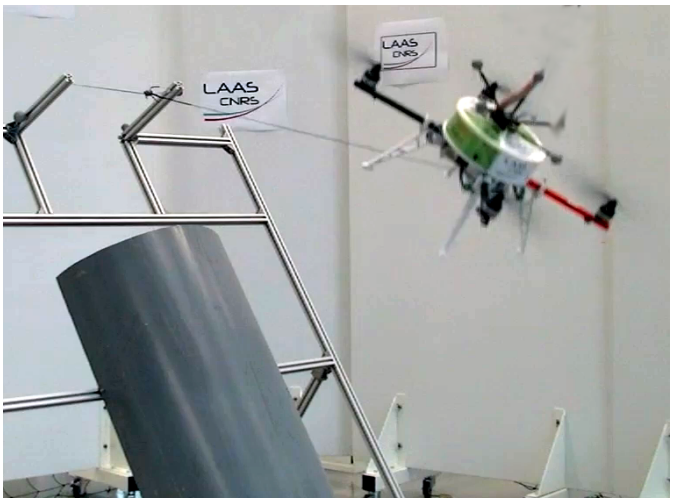


Figure 1.8 – Tethered UAV (Tognon & Franchi, 2017).

However, they suffer from two main issues: they must tilt in order to translate in the horizontal plane and consequently they are unable to apply independently forces and moments to the environment.

Omnidirectional aerial vehicles can generate a 6-DoF wrench, and so are suitable for manipulation tasks without need for an embedded robotic arm (Hamandi et al., 2021). Omnidirectionality is obtained by positioning and orienting n propellers on the platform to generate any force and moment independently. The number n depends on the type of the thrusters: while six bidirectional thrusters are enough to control all DoFs, a redundant design with at least seven thrusters is needed when using unidirectional propellers (Tognon & Franchi, 2018). Note that bidirectional thrusters offer a larger thrust range, but a lower magnitude since symmetric propeller blades have reduced efficiency.

The thrusters no longer being collinear, the efficiency (the power needed to hover) of the omnidirectional vehicle is lower with respect to a collinear design, since the thrust direction of individual propellers is not anymore along the vertical direction and leads to a partial use of the total thrust. This phenomenon is illustrated in Figure 1.9. As an example, collinear propellers need to generate 1 kg thrust each to hover with a 2 kg aerial vehicle, while 2 kg thrust is needed per propeller if the propellers have a 60° angle with respect to the vertical axis, because only the projection of the thrust along the vertical axis matters for hovering. To improve the efficiency to the detriment of mechanical simplicity, it is possible to use servomotors to modify the orientation of the thrusters during the flight (Ryll et al., 2016).

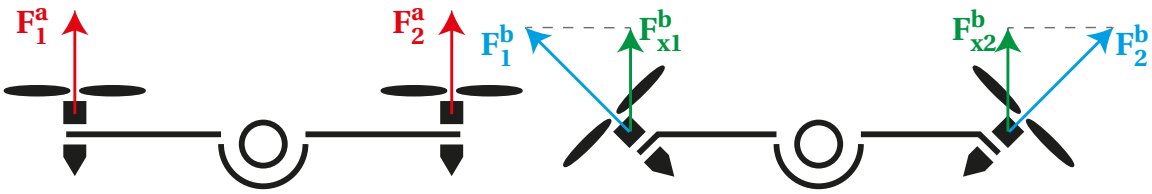


Figure 1.9 – Energy efficiency of an omnidirectional UAV.

Design and Applications

Brescianini and D'Andrea developed an omnidirectional multirotor vehicle actuated by eight nonparallel bidirectional thrusters (see Figure 1.10a) (Brescianini & D'Andrea, 2018). They are positioned in an optimal configuration maximizing the agility, i.e. the highest thrust and torque that can be generated on all directions, while maintaining rotational invariance. The position and the attitude control loops are decoupled. Non-

linearities are canceled with a feedback linearization approach. A control allocation computes the rotor velocity input signal by minimizing the power consumption while limiting the difference between the desired thrust and the actual thrust. An external tachometer is used to regulate the rotor rotational velocity.

The ODAR is another aerial vehicle, designed for omnidirectional aerial wrench generation (see Figure 1.10b) (Park et al., 2018). The eight nonparallel bidirectional thrusters are positioned to maximize the minimum-guaranteed omnidirectional wrench. A control allocation technique using a selective mapping algorithm is proposed to avoid the destabilizing effect of sensorless electronic speed controllers (ESCs) at low speeds.

The OMAV uses 6 pairs of orientable coaxial propellers for actuation (see Figure 1.10c) (Brunner et al., 2020). A nonlinear model predictive controller (NMPC) computes the desired wrench. An allocation problem is then solved at a higher frequency to convert the wrench into actuator control signals. The NMPC handles constraints on the dynamic model of the system, limiting the maximum wrench and its rate of change. The plant model is augmented with constant disturbances corresponding to unmodeled dynamics. An extended Kalman filter (EKF) observes these disturbances, allowing for zero steady-state error. The controller is implemented on an onboard computer with the ACADO framework (Houska et al., 2011a).

Bicego et al. used a more accurate actuator-level model for the NMPC (Bicego et al., 2020). Therefore, they took into account actuator dynamics and imposed more meaningful constraints on the propeller speed and acceleration. With this approach, the allocation problem is included into the NMPC optimization problem. Thereby, it is suitable for a large range of aerial robots, from under-actuated to fully actuated platforms. Moreover, this strategy is compatible with both bidirectional and unidirectional actuators, since the NMPC can handle those constraints. The controller is implemented on a distant computer with the MATMPC framework (Chen et al., 2019) using the qpOASES solver, and the control input is sent to the aerial platform by means of a serial cable. The plant model is discretized at 10 Hz (0.1 s intervals) over a prediction horizon of 1 s. However, the controller runs at frequency higher than 200 Hz. This technique allows for a wider prediction horizon since the computational time increases significantly with the number of discretization nodes.

Jacquet and Franchi improved this previous work (i) by adding perception constraints to the NMPC to ensure the visibility of a chosen feature assuming a pyramidal field of view and (ii) by modifying the cost function to also minimize the perpendicularity error of the camera optical axis with respect to the tracked fiducial marker (Jacquet &

Franchi, 2021). Furthermore, the controller runs at 500 Hz on board, on an embedded computer with an Intel Core i7-8565U.

1.2.3 Suspended Aerial Manipulators

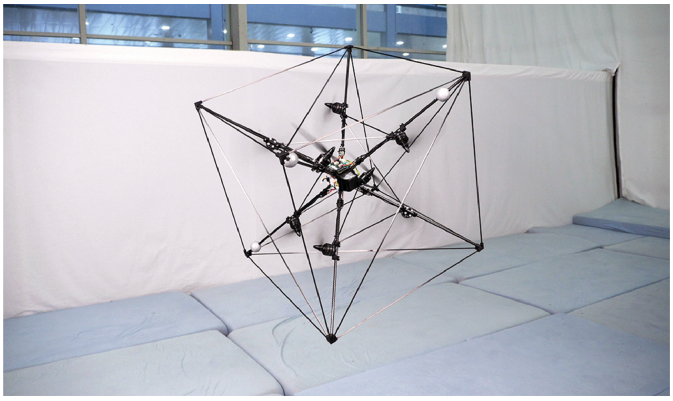
In order to optimize the energy consumption of the aerial vehicle and improve its accuracy, a few works have proposed to use permanently or temporarily a cable. The SpiderMAV stabilizes itself in a spider-inspired way, launching anchors on nearby walls and ceiling to perch and stabilize the platform (see Figure 1.11) (Zhang et al., 2017). This is a low-power way of generating a constant wrench but it relies on nearby compatible anchoring surfaces.

The cable-Suspended Aerial Manipulator (SAM) consists in an omnidirectional UAV, suspended below an aerial carrier by means of cables and carrying a 7-DoF serial manipulator (see Figure 1.12) (Sarkisov et al., 2019). It is equipped with 3 winches, 8 propellers arranged to produce an omnidirectional 6-DoF wrench, a landing gear and various sensors (IMU, GPS and two cameras). The aerial carrier is assumed to be static. The SAM regulates its pose using a cascade control scheme performing proportional-derivative feedback, with the propeller-induced wrench as control input. This omnidirectional aerial vehicle has an Omniplus design (Tognon & Franchi, 2018). The propulsion units are optimally arranged to minimize the conditioning of the thrust allocation matrix, aiming at an equal distribution of the omnidirectional wrench between the propulsion units while guaranteeing a null space of the thrust allocation matrix such that the resulting wrench is canceled if all propellers generate the same thrust.

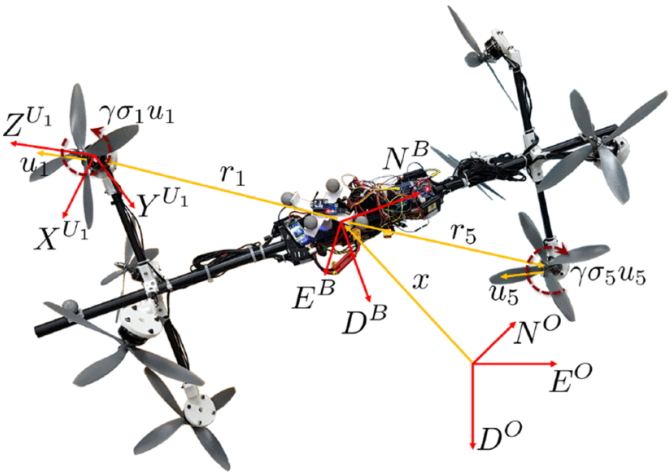
1.3 Contributions

This thesis presents the Aerial Manipulator with Elastic Suspension (AMES), shown in Figure 1.13. An aerial manipulator is suspended at the tip of a crane-like robotic carrier by a spring. The aerial manipulator, which is called here an Aerial Wrench Generator (AWG), is eventually holding a gripper. The AWG generates a 6-DoF wrench at the end effector thanks to propulsion units. So the AMES with its robotic carrier, spring and AWG may be considered a new kind of robot, a hybrid between a serial manipulator and an aerial manipulator with the spring acting as a flexible linkage. We called this robot "dextAIR".

The robotic carrier is optional: it is useful to extend the workspace of the robot, but



(a) Omnidirectional UAV from Brescianini and D'Andrea (Brescianini & D'Andrea, 2018).

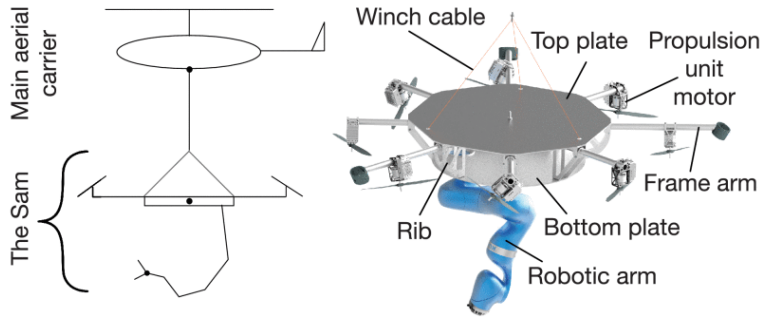


(b) The ODAR (Park et al., 2018).

(c) The OMAV (Brunner et al., 2020).

Figure 1.10 – Omnidirectional UAV examples.

Figure 1.11 – SpiderMAV (Zhang et al., 2017).



(a) Concept (figure from (Sarkisov et al., 2019)).

(b) Prototype.

Figure 1.12 – Cable-suspended aerial manipulator (Sarkisov et al., 2019).

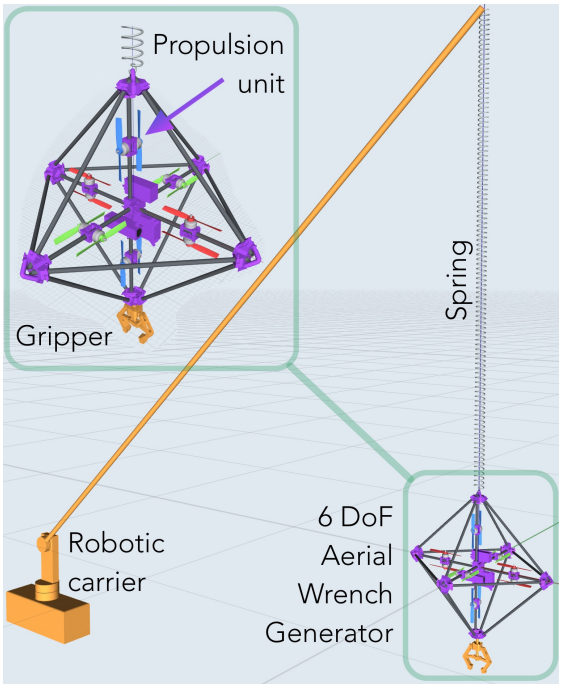


Figure 1.13 – AMES concept.

with a static anchoring point of the spring, the workspace may already be large enough for many tasks. The spring creates an equilibrium position for the AWG even when motors are off, compensating for the gravity, and therefore little energy is required to hover. The robotic carrier moves slowly the equilibrium point of the AWG to the average position of the current task. A mobile anchoring point may be available in many scenarios: weeding, pesticide delivery or harvesting in association with agricultural robots, pick and place of parcels in logistic hubs with a gantry crane, 3D construction printing in combination with a tower crane...

This thesis yielded several scientific contributions and software developments.

Software Developments

- Main developer of Teensyshot with my supervisor Jacques Gangloff. Teensyshot is a firmware running on Teensy microcontroller units to allow for speed control of drone motors thanks to telemetry data available from electronic speed controllers (ESCs).
- Contributor to the Raspberry Pi Simulink Coder toolkit (RPIt) developed by Jacques Gangloff (Gangloff et al., 2020). RPIt uses Simulink Coder to generate

embedded C code from a Simulink diagram for a Debian target and communicates with the target in runtime to visualize signals and modify parameters. The contributions consisted in debugging and developing new Simulink blocks to pair RPIt with new devices.

- Contributor to acados (Verschueren et al., 2019). acados is a software package that generates efficient low-level C code that can be deployed on embedded platforms to solve nonlinear optimal control and estimation problems. The contributions consisted in debugging the Matlab interface.

Scientific Contributions

- Design of the Aerial Wrench Generator (AWG),
- Performance characterization of the AWG using computed torque controller (CTC) with proof of stability using singular perturbation theory,
- Performance improvement of the AWG using nonlinear model predictive controller (NMPC),
- Combined control of the AWG with a CDPR carrier using (i) a partitioned control law and (ii) NMPC with energy minimization to handle redundancy.

1.4 Related Publications

During this thesis, following articles were published.

1.4.1 Conference

- Presenter of (Yiğit et al., 2020) at ICRA 2020: preliminary results on the AWG with CTC. Associated video available at https://youtu.be/Nbs62fteUgs.
- Presenter of (Yiğit, Arpa Perozo, Cuvillon, et al., 2021a) at ICRA 2021: improving the dynamics of the AWG using NMPC. Associated video available at https://youtu.be/6a4gE4A6bLU.
- Presenter of (Yiğit, Arpa Perozo, Ouafo, et al., 2021) at IROS 2021: combined control of the AWG with its CDPR carrier using a partitioned controller. Associated video available at https://youtu.be/NxJjCoystsA.

- Co-author of (Khayour et al., 2020), published at IROS 2020: MPC of the PiSaRo4 robot. Associated video available at https://youtu.be/tqbiuCRztxQ.
- Co-author of (Bertrand et al., 2020), published in EBCCSP 2020: development of an event-based pose estimation algorithm using a Dynamic Vision Sensor (DVS).

1.4.2 Journal

- Author of (Yiğit, Arpa Perozo, Cuvillon, et al., 2021b), published in IEEE Robotic and Automation Letters: performance characterization of the AWG using CTC with proof of stability using singular perturbation theory. This work was also presented during the ICRA 2021 conference and received the **best video award** during the JJCR 2020 (Journée des Jeunes Chercheurs en Robotique, France). The associated video is available at https://youtu.be/Ey88RkaMrNc.

1.5 Outline of the Thesis

The AWG is first introduced in Chapter 2. Its dynamic model is derived before assessing experimentally the performance of computed torque control and nonlinear model predictive control laws. In Chapter 3, the AWG is suspended from a CDPR. The dynamic model of the combined system is derived. A partitioned controller and an NMPC are compared experimentally. Finally, Chapter 4 concludes this thesis and discusses some perspectives.

2 Aerial Wrench Generator

2.1 Introduction

This chapter focuses on the Aerial Wrench Generator (AWG) with a static anchoring point. Indeed, the carrier is assumed to be slow, and hence, in a first approximation, the contribution of the carrier to the dynamics of the Aerial Manipulator with Elastic Suspension (AMES) is neglected.

Performance characteristics of the AWG are, to a large extent, independent of the choice of the control law. Yet, assessing them is useful to tune adequately the controllers and to identify what has to be improved for the next iterations of the AWG design. First, a well-known control law with decent performance, namely a computed torque control (CTC), is proposed to characterize the AWG performance. Then, a more sophisticated controller, a nonlinear model predictive controller (NMPC), is implemented to further improve dynamic capabilities of the AWG.

In the following sections, a generic dynamic model is introduced which is valid for any aerial vehicle with non-steerable thrusters. Then, a prototype of the AWG, able to generate a 6-DoF wrench without overactuation, is presented and tested experimentally using both CTC and NMPC.

2.2 Dynamic Modeling

The dynamics of the AWG may be derived using two different levels of simplification. The simplified approach neglects the inertiae of the rotating parts (rotors and propellers). Therefore, the AWG is considered a single rigid body with n wrenches acting on it, n being the number of propulsion units, besides the gravity and the

spring restoring force. A more accurate approach considers $n+1$ rigid bodies, thus taking into account inertial effects due to the rotating parts: the gyroscopic effect and the acceleration torque. The most common approach found in the literature is the simplified one since aerial vehicles with collinear propulsion units use pairs of contra-rotating propellers, compensating for inertial effects. In the following, the simplified approach is used to model the AWG dynamics.

In case of significant disparities with the simplified model and the experiments, the inertial effects can be included as external wrenches rather than with additional rigid bodies, thus without adding too much complexity to the model.

2.2.1 Parametrization

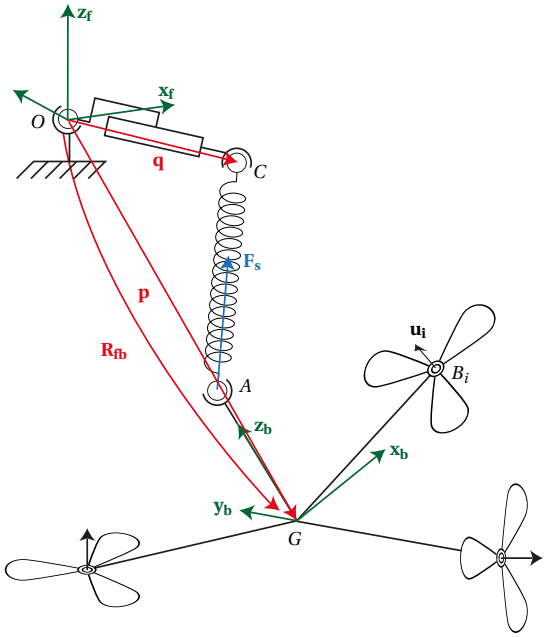


Figure 2.1 – AWG parameters.

The geometric parameters of the system are shown in Figure 2.1. In order to be as generic as possible, the following model makes no assumptions on the number of propulsion units (can be under or overactuated), nor on their directionality (unidirectional or bidirectional).

First let us introduce some of the notations used in the remaining of this manuscript. Let $\mathbf{u}$ and $\mathbf{v}$ be vectors and $\mathscr{R}_q$ a reference frame. The projection of $\mathbf{v}$ in $\mathscr{R}_q$ is written ${}^q\mathbf{v}$. The cross product of $\mathbf{u}$ and $\mathbf{v}$ is denoted $\mathbf{u}\times\mathbf{v}$ and can be computed in any reference frame: ${}^q(\mathbf{u}\times\mathbf{v}) = {}^q\mathbf{u}\times{}^q\mathbf{v}$. The cross product matrix $[\cdot]_\times$ is defined such that ${}^q\mathbf{u}\times{}^q\mathbf{v} =$

$[{}^q\mathbf{u}]_\times {}^q\mathbf{v}$.

Let $\mathscr{R}_f = (O, \mathbf{x_f}, \mathbf{y_f}, \mathbf{z_f})$ be an inertial frame with O its origin and $\mathscr{R}_b = (G, \mathbf{x_b}, \mathbf{y_b}, \mathbf{z_b})$ a moving frame attached to the center of mass (CoM) G of the AWG with its $\mathbf{z_b}$ axis pointing toward the on-board anchoring point of the spring A. The rotation matrix $\mathbf{R_{fb}} \in SO(3)$ describes the orientation of $\mathscr{R}_b$ with respect to $\mathscr{R}_f$, the angular velocity vector of this rotation is denoted $\boldsymbol{\omega} \in \mathbb{R}^3$. The AWG has n propulsion units, positioned in an arbitrary way. The position of the center of the i-th propulsion unit is B_i, and $\mathbf{u_i}$ is the unit thrust direction vector. The carrier corresponds to the second anchoring point of the spring C. The position vectors of the CoM G and of the carrier end effector C are respectively $\mathbf{p}$ and $\mathbf{q}$. The restoring force applied by the spring on the AWG is noted $\mathbf{F_s} \in \mathbb{R}^3$.

To simplify the notations, since the position vectors $\mathbf{p}$ and $\mathbf{q}$ are always expressed in the inertial frame $\mathscr{R}_f$ and the angular velocity vector $\boldsymbol{\omega}$ in the body frame $\mathscr{R}_b$, the superscript is omitted: $\mathbf{p} = {}^f\mathbf{p}$, $\mathbf{q} = {}^f\mathbf{q}$ and $\boldsymbol{\omega} = {}^b\boldsymbol{\omega}$.

2.2.2 Propulsion Unit Modeling

During the rotation of a propeller, the blades are subject to aerodynamic forces normal to their surfaces. The force $\mathbf{F_{aero}}$ generated on each blade has two components: one along the axis of the propeller $\mathbf{F_{thrust}}$ and one orthogonal to the axis $\mathbf{F_{drag}}$ (see Figure 2.2). The components along the axis contribute to a resulting force, called thrust, and the ones orthogonal to the axis are responsible for a torque called drag, both oriented along the rotation axis. When dealing with UAV propellers, both the thrust and the drag are considered proportional to the square of rotational velocity. The proportionality coefficients, a for the thrust and b for the drag, can be either derived using fluid mechanics (Khan & Nahon, 2013) or identified experimentally.

The most rigorous way to deal with the inertial effects of the propellers on the flying platform requires a multibody dynamic model. We propose here a simplified approach, similar to Brescianini and D'Andrea's one (Brescianini & D'Andrea, 2018), that allows for taking into account the inertial effects. Let J_p be the moment of inertia of the rotating parts about the rotation axis and w_i the rotational velocity of the i-th propulsion unit. We consider the propulsion units as wrench generators that encompass the gyroscopic torque $J_p w_i \mathbf{u_i} \times \boldsymbol{\omega}$, the acceleration torque $J_p \dot{w}_i \mathbf{u_i}$ (reaction torque on the AWG body to accelerate the rotating parts), the thrust $a w_i^2 \mathbf{u_i}$ and the drag $b w_i^2 \mathbf{u_i}$. Therefore, the i-th propulsion unit behaves as a wrench generator with

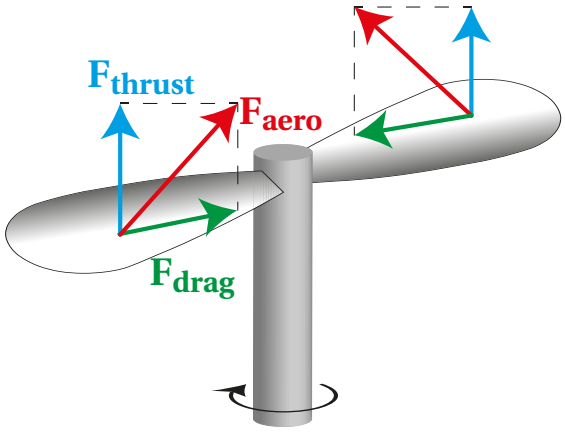


Figure 2.2 – Forces on propellers.

force $\mathbf{F_i} = a w_i^2 \mathbf{u_i}$ and torque $\mathbf{N_i} = -\left(b w_i^2 \mathbf{u_i} + \boldsymbol{\omega} \times J_p w_i \mathbf{u_i} + J_p \dot{w}_i \mathbf{u_i}\right)$.

2.2.3 AWG Dynamics

As discussed above, the AWG is modeled as a single rigid body. Therefore, its dynamics can be obtained using Newton-Euler's equations:

$$\begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_b \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \boldsymbol{\omega} \times \mathbf{J}_b \boldsymbol{\omega} \end{pmatrix} + \begin{pmatrix} -m^f \mathbf{g} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} {}^f\mathbf{F_s} \\ {}^b\mathbf{N_s} \end{pmatrix} = \begin{pmatrix} {}^f\mathbf{F} \\ {}^b\mathbf{N} \end{pmatrix} \tag{2.1}$$

with $m > 0$ the total mass of the platform, $\mathbf{J}_b \in \mathbb{R}^{3\times3}$ its inertia tensor at the CoM expressed in $\mathscr{R}_b$, $\mathbf{g} \in \mathbb{R}^3$ the gravity acceleration, $\mathbf{F_s} \in \mathbb{R}^3$ the restoring force of the elastic link on the AWG and $\mathbf{N_s} \in \mathbb{R}^3$ the associated torque at the CoM, $\mathbf{F} \in \mathbb{R}^3$ the resulting force generated by the propellers and $\mathbf{N} \in \mathbb{R}^3$ the associated torque at the CoM. Additionally, $\mathbf{0}$ and $\mathbf{I}$ denote respectively the zero and the identity matrices with appropriate dimensions.

Let $\boldsymbol{\eta} = (\psi\ \theta\ \phi)^T$ be a set of Euler angles describing the orientation of the body frame $\mathscr{R}_b$ with respect to the inertial frame $\mathscr{R}_f$. The analytical Jacobian matrix $\mathbf{S}(\boldsymbol{\eta})$ maps the time derivative of $\boldsymbol{\eta}$ to the angular velocity ${}^b\boldsymbol{\omega}$:

$${}^b\boldsymbol{\omega} = \mathbf{S}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} \tag{2.2}$$

The behavior of the elastic link is modeled as a simple linear spring with stiffness k

and free length l_0:

$$ {}^{f}\mathbf{F_s} = -k\left(\|\mathbf{CA}\| - l_0\right)\frac{{}^{f}\mathbf{CA}}{\|\mathbf{CA}\|} \tag{2.3} $$

where

$$ {}^{f}\mathbf{CA} = \mathbf{p} + \|\mathbf{GA}\|\,\mathbf{R_{fb}}\,{}^{b}\mathbf{z_b} - \mathbf{q} \tag{2.4} $$

Therefore, the associated torque is

$$ {}^{b}\mathbf{N_s} = \|\mathbf{GA}\|\,{}^{b}\mathbf{z_b} \times \mathbf{R_{fb}}^{T}\,{}^{f}\mathbf{F_s} \tag{2.5} $$

In the most simplified approach, the propulsion units are simply considered thrust and drag generators. Let $\mathbf{W}_b \in \mathbb{R}^{6\times n}$ be the matrix that maps the column matrix $\mathbf{w_2} = (\cdots\ w_i|w_i|\ \cdots)^T$ of signed squared propeller rotational velocities w_i to the wrench they apply on the platform expressed in the body frame $\mathscr{R}_b$:

$$ \begin{pmatrix} {}^{b}\mathbf{F} \\ {}^{b}\mathbf{N} \end{pmatrix} = \mathbf{W}_b\mathbf{w_2} \tag{2.6} $$

$$ \mathbf{W}_b = a\begin{pmatrix} \cdots & {}^{b}\mathbf{u_i} & \cdots \\ \cdots & {}^{b}\mathbf{GB_i} \times {}^{b}\mathbf{u_i} & \cdots \end{pmatrix} - b\begin{pmatrix} \cdots & \mathbf{0} & \cdots \\ \cdots & {}^{b}\mathbf{u_i} & \cdots \end{pmatrix} \tag{2.7} $$

Let $\mathbf{X} = (\mathbf{p}^T\ \boldsymbol{\eta}^T)^T$ be the pose coordinates of the AWG. Combining (2.1), (2.6) and (2.2), the equations of motion of the robot can be written in the Cartesian-space canonical formulation:

$$ \underbrace{\begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T\mathbf{J}_b\mathbf{S} \end{pmatrix}}_{\mathbf{M(X)}}\begin{pmatrix} \ddot{\mathbf{p}} \\ \ddot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T(\mathbf{J}_b\dot{\mathbf{S}} + [\mathbf{S}\dot{\boldsymbol{\eta}}]_\times\mathbf{J}_b\mathbf{S}) \end{pmatrix}}_{\mathbf{C(X,\dot{X})}}\begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} -m\,{}^{f}\mathbf{g} - {}^{f}\mathbf{F_s} \\ -\mathbf{S}^T\,{}^{b}\mathbf{N_s} \end{pmatrix}}_{\mathbf{G(X)}} = \underbrace{\begin{pmatrix} \mathbf{R_{fb}} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T \end{pmatrix}\mathbf{W}_b\mathbf{w_2}}_{\tilde{\mathbf{W}}\mathbf{(X)}} \tag{2.8} $$

2.2.4 Rotation Representation

In the remaining of this manuscript, roll (ψ), pitch (θ) and yaw (ϕ) angles describe the rotation of the body frame $\mathscr{R}_b$ with respect to the inertial frame $\mathscr{R}_f$. This corresponds to a rotation ψ about the $\mathbf{x_f}$ axis, followed by a rotation θ about the $\mathbf{y_f}$ axis and a rotation ϕ about the $\mathbf{z_f}$ axis:

$$\mathscr{R}_0 = \mathscr{R}_f \xrightarrow{\text{Rot}(\mathbf{x_f},\psi)} \mathscr{R}_1 \xrightarrow{\text{Rot}(\mathbf{y_f},\theta)} \mathscr{R}_2 \xrightarrow{\text{Rot}(\mathbf{z_f},\phi)} \mathscr{R}_3 = \mathscr{R}_b$$

Note that these rotations are all about the axes of the inertial frame $\mathscr{R}_f$ (which is static). The rotation matrix is obtained by left-multiplying the rotation matrices in the order of the rotations:

$$\mathbf{R_{fb}} = \mathbf{R}_{\mathbf{z},\phi}\mathbf{R}_{\mathbf{y},\theta}\mathbf{R}_{\mathbf{x},\psi} \tag{2.9}$$

where $\mathbf{R}_{\mathbf{u},\alpha}$ is the matrix corresponding to a rotation α about the $\mathbf{u}$ axis, and $(\mathbf{x},\mathbf{y},\mathbf{z})$ is the canonical basis of $\mathbb{R}^3$. An equivalent representation can be obtained using rotations about the axes of the intermediate rotating frames:

$$\mathscr{R}_0 = \mathscr{R}_f \xrightarrow{\text{Rot}(\mathbf{z_f},\phi)} \mathscr{R}'_1 \xrightarrow{\text{Rot}(\mathbf{y_1},\theta)} \mathscr{R}'_2 \xrightarrow{\text{Rot}(\mathbf{x_2},\psi)} \mathscr{R}_3 = \mathscr{R}_b$$

Therefore, the angular velocity vector $\boldsymbol{\omega}$ is obtained by composition of rotations:

$$\boldsymbol{\omega} = \dot{\psi}\mathbf{x_b} + \dot{\theta}\mathbf{y_1} + \dot{\phi}\mathbf{z_f} \tag{2.10}$$

The projection of the angular velocity vector $\boldsymbol{\omega}$ in the body frame $\mathscr{R}_b$ yields the analytical Jacobian matrix from equation (2.2):

$$\mathbf{S}(\boldsymbol{\eta}) = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\psi & \sin\psi\cos\theta \\ 0 & -\sin\psi & \cos\psi\sin\theta \end{pmatrix} \tag{2.11}$$

The chosen representation of rotations, namely the roll-pitch-yaw angles, suffers from gimbal lock like any other Euler angles. The gimbal lock is illustrated in Figure 2.3: the yellow and gray rings have the same rotation axis, it is not possible to rotate along the axis normal to the surface of the rings. This corresponds to a loss of one degree of freedom. The specific configurations are $\theta = +\frac{\pi}{2}$ and $\theta = -\frac{\pi}{2}$. The rotation matrix $\mathbf{R_{fb}}$

for both cases is shown below:

$$\mathbf{R_{fb}}\left(\theta = +\frac{\pi}{2}\right) = \begin{pmatrix} 0 & -\sin(\phi-\psi) & \cos(\phi-\psi) \\ 0 & \cos(\phi-\psi) & \sin(\phi-\psi) \\ -1 & 0 & 0 \end{pmatrix} \tag{2.12}$$

$$\mathbf{R_{fb}}\left(\theta = -\frac{\pi}{2}\right) = \begin{pmatrix} 0 & -\sin(\phi+\psi) & -\cos(\phi+\psi) \\ 0 & \cos(\phi+\psi) & -\sin(\phi+\psi) \\ 1 & 0 & 0 \end{pmatrix} \tag{2.13}$$

This singularity cannot happen if the range of the pitch angle θ is limited to $\pm 90°$, which will be the case in practice for the AWG attitude. Indeed, a pitch angle θ superior to 90° is not relevant for the considered manipulation tasks and prohibited due to possible interference with the suspension spring.

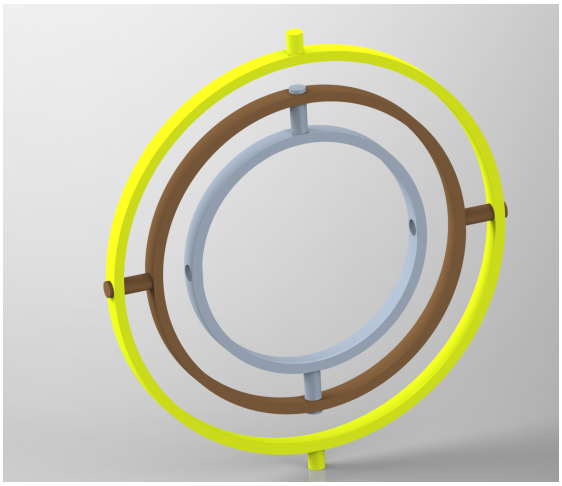


Figure 2.3 – Gimbal lock.

2.3 Design

2.3.1 Mechanical Design and Actuation

In addition to theoretical considerations, practical limitations need to be taken into account while designing the AWG, such as the simplicity of the control allocation and the safety. Our AWG has the same structure as an omnidirectional multirotor vehicle with bidirectional thrusters described in (Brescianini & D'Andrea, 2018) (see Figure 2.5). The relative placement of propulsion units is optimized to maximize the agility,

i.e. the highest thrust and torque that can be generated in all directions. The selected optimal configuration for $n = 6$ propulsion units (Figure 2.4) can generate a 6-DoF wrench without overactuation, simplifying the thrust allocation problem. Intuitively, it is easy to verify that any force and any torque can be generated independently if the drag is neglected (which is lower than the torque that can be generated using thrusts by many orders of magnitude). For example, propulsion units on the $\mathbf{z_b}$ axis generate a force along the $\mathbf{x_b}$ axis if their thrust is in the same direction and a torque about the $\mathbf{y_b}$ axis if the thrusts are opposed.

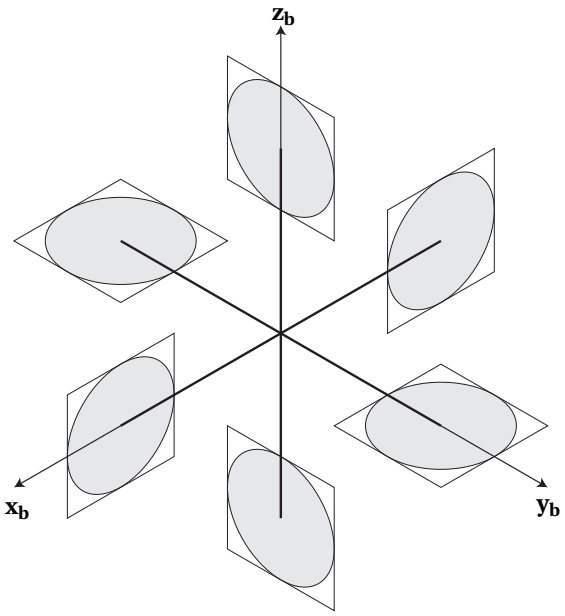


Figure 2.4 – Propeller configuration (figure inspired by (Brescianini & D'Andrea, 2018)).

The propulsion units are attached to the vertices of a regular octahedron built using 10 mm carbon fiber tubes (see Figure 2.4). As a consequence, AWG is as rotationally invariant as possible and its inertia matrix is close to the identity matrix. Additional carbon fiber tubes link vertices to improve the rigidity and also to improve safety by enfolding propellers in a convex structure.

Depending on its mass, the transverse vibrations of the spring can significantly disturb the AWG. That can be avoided by mounting the spring horizontally and suspending the AWG by a cable through a pulley (see Figure 2.6).

Two solutions have been tested in order to generate a bidirectional thrust on a propulsion unit, both using a pair of propellers mounted on two coaxial motors (see Figure 2.5).

The first solution uses symmetrical propellers (Graupner 3D 6"), also called 3D pro-

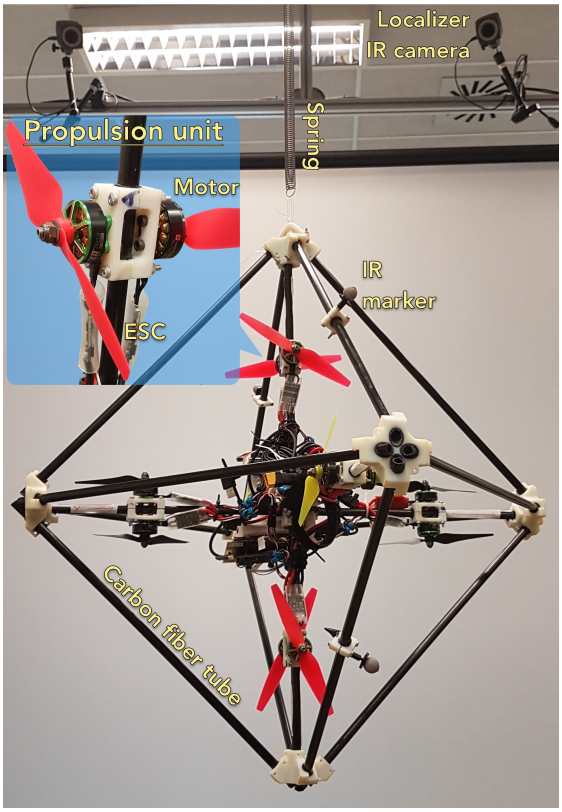


Figure 2.5 – Aerial Wrench Generator.

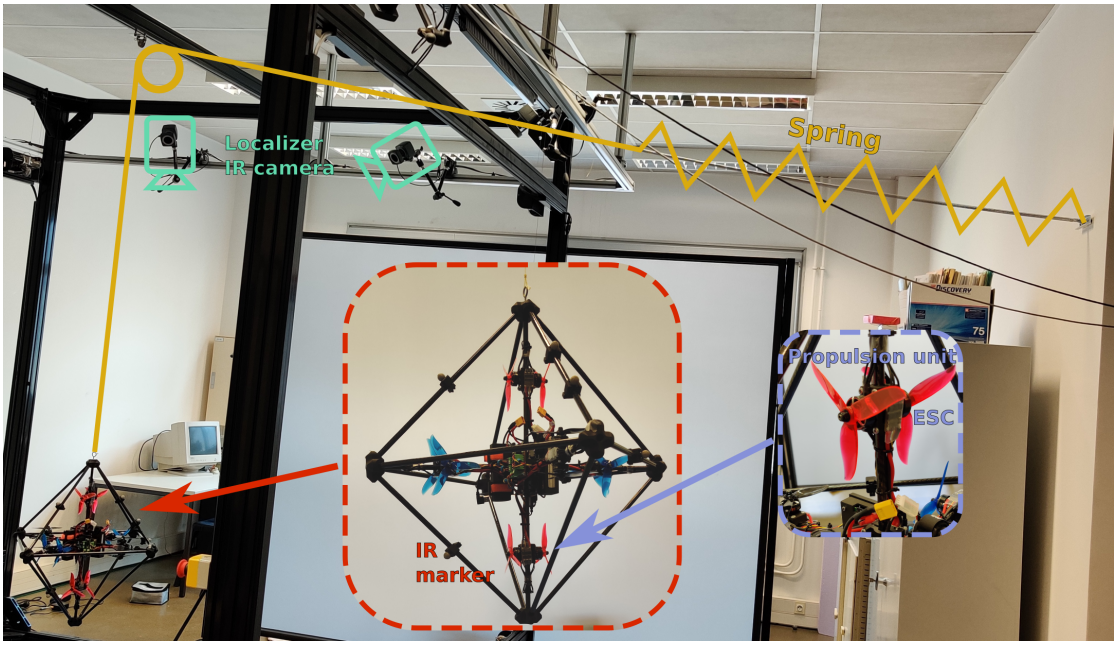


Figure 2.6 – AWG experimental setup.

pellers, that can generate thrust in both directions by changing the rotation direction of the motor. The two propellers of the propulsion unit are contra-rotating to cancel the drag, the acceleration torque and the gyroscopic effect. Experiments with and without contra-rotating propellers have been carried out, however, they did not allow for distinguishing the gyroscopic effect, which was hidden by other unmodeled phenomena such as turbulence. With electronic speed controllers (ESCs) for sensorless brushless DC motors, when an inversion of the rotation direction occurs, the settling time is higher than the settling time of a throttle step without direction change. This is referred as "ESC induced singularity" by Park et al. (Park et al., 2018). This nonlinearity results from brushless motor velocity zero-crossing. Indeed, ESCs use the back EMF (electromotive force) to estimate the speed of the rotor, which is not possible at low speeds. For a standard multirotor drone, this may not be an important issue since propellers need to generate a force constantly in order to hover, but it is a limiting factor for the AWG which is aimed to work mainly around its equilibrium point. Indeed, propellers may need to change the rotation direction many times per second (for example, the natural frequency for roll/pitch angles of the robot is higher than 3 Hz).

The second solution for the propulsion unit is to use two unidirectional propellers (DALPROP 5045) to generate the bidirectional thrust. According to the sign of the force to be generated, the corresponding propeller rotates at the desired speed while the second one is idling, i.e. rotating at its lowest speed (1500 rpm, equivalent to 0.03 N). Therefore, the "zero-crossing" problem is avoided, however only half of the motors are active at the same time, which reduces the power over weight ratio. All the experimental results that are presented in this thesis use the second solution. Some experiments using bidirectional propellers are shown in (Yiğit et al., 2020).

2.3.2 Electronics

Each propeller is actuated by a brushless DC motor (T-Motor F-40 Pro III Kv2400). The speed of the propeller is controlled by an ESC. However, there is only a limited number of commercial ESCs providing effective speed regulation and most of them are designed for helicopters and do not have fast dynamics. Most ESCs exhibit only a coarse linear mapping, dependent on the driven propeller, between the ESC input signal (also called throttle) and the propeller velocity. The ESC firmware can be modified to add speed regulation using an external tachometer (Brescianini & D'Andrea, 2018). A sensorless solution also exists for BL-Ctrl boards (Franchi & Mallet, 2017). Here, we developed an open-source firmware called "Teensyshot"

(https://github.com/jacqu/teensyshot) for the speed controlling of brushless DC motors without additional sensors. Teensyshot can be used with up to 6 ESCs compatible with the DShot600 protocol. A fast anti-windup proportional-integral-derivative (PID) speed regulation loop, using real-time ESC telemetry data acquired through 115 200 bps serial link, runs at 500 Hz on a Teensy 4.0 micro-controller board (see Figure 2.7). Further details on the firmware are available in Appendix B. KISS 32A ESCs are used for the experiments. The controller is tuned such that, in closed loop, the propeller rotational velocity behaves like a first-order system with time constant $t_m = 25\,\text{ms}$:

$$\dot{\mathbf{w}} = \frac{1}{t_m}(\mathbf{w_{ref}} - \mathbf{w}) \tag{2.14}$$

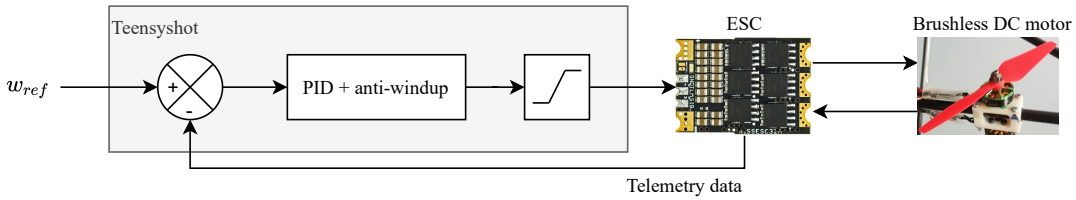


Figure 2.7 – Propeller speed regulation.

The AWG is autonomous: it carries its own energy source, a 2300 mA h, 11.1 V / 3S lithium polymer battery pack (TATTU 3S1P). It also has an on-board CPU (Raspberry Pi 4B) running high-level control algorithms and communicating with a ground station through Wi-Fi TCP/IP sockets thanks to the open-source Simulink toolbox RPIt developed in our lab (Gangloff et al., 2020). The Raspberry Pi is connected by USB to 2 Teensy boards regulating the velocity of a total of 12 propellers.

The AWG is also equipped with infrared (IR) markers that are tracked by a 6-camera Vicon Bonita motion capture system (or MoCap). The data from the cameras are acquired at a refresh rate of 240 Hz on a distant computer and are processed to obtain the 6-DoF pose of the AWG. The distant computer hosts a virtual-reality peripheral network (VRPN) server that answers requests from a VRPN client running on the on-board computer. A complementary filter combines the derivative of the orientation acquired by the motion capture system and the angular velocity acquired by an on-board gyroscope sensor (MPU-9150). Note that the accelerometer measurements cannot be used because of a bad signal-to-noise ratio induced by the high rotational velocity of the propellers.

The electronic architecture of the setup for experimental performance characteriza-

tion of the AWG is illustrated on Figure 2.8. The experiments are carried out using a CTC.

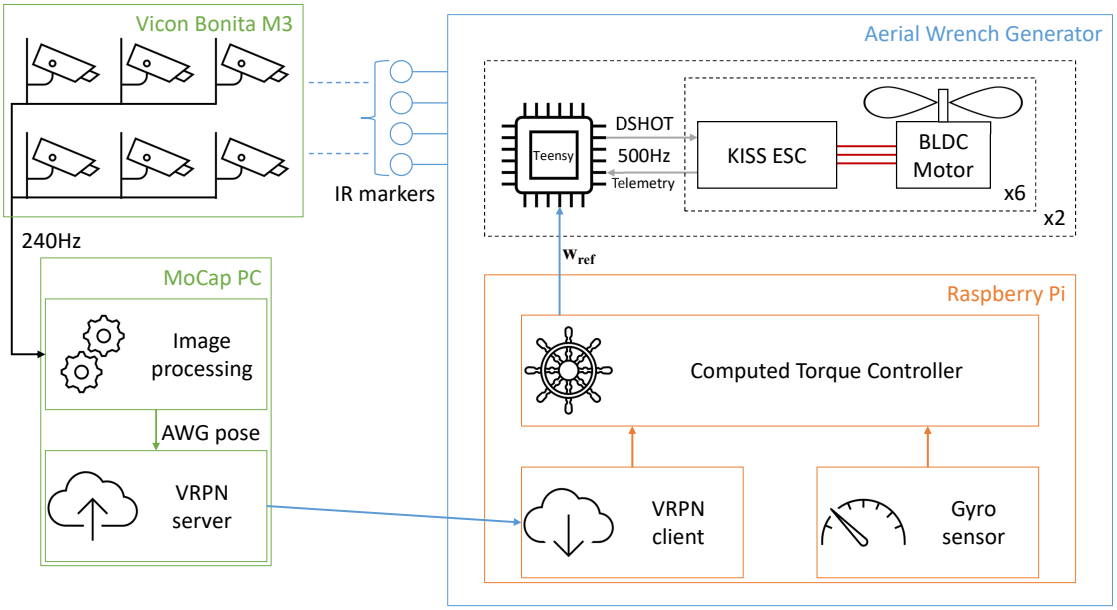


Figure 2.8 – AWG electronic architecture for CTC experiments.

2.4 Experimental Performance Characterization

In the following section, the experimental performance characterization of the AWG is presented. These experiments use a computed torque control law, whose stability is proven using singular perturbation theory.

2.4.1 Computed Torque Control Law

Let us recall the dynamic model (2.8) in its canonical form:

$$\mathbf{M}(\mathbf{X})\ddot{\mathbf{X}} + \mathbf{C}(\mathbf{X},\dot{\mathbf{X}})\dot{\mathbf{X}} + \mathbf{G}(\mathbf{X}) = \tilde{\mathbf{W}}(\mathbf{X})\mathbf{w_2} \tag{2.15}$$

where $\tilde{\mathbf{W}}$ is always invertible since the AWG actuators are not redundant and there is no singular configuration.

A computed torque control (Murray et al., 1994) defines a control law for the actuator torques, here the yielding wrench $\tilde{\mathbf{W}}(\mathbf{X})\mathbf{w_2}$, such that the robot follows a desired trajectory $\mathbf{X_{ref}}$. This control law, also called inverse dynamics control, is designed to transform the nonlinear system of the robot dynamics into a linear closed-loop

system (Spong et al., 2005). It is a special case of feedback linearization of a nonlinear system. The resulting linear system can then be controlled with an outer-loop linear controller.

Let us consider the control $\hat{\mathbf{W}}\mathbf{w_2}$ that compensates for the dynamics of the AWG:

$$\hat{\mathbf{W}}\mathbf{w_2} = \hat{\mathbf{M}}(\ddot{\mathbf{X}}_{\mathbf{ref}} + \mathbf{u}) + \hat{\mathbf{C}}\dot{\mathbf{X}} + \hat{\mathbf{G}} \tag{2.16}$$

where $\hat{\mathbf{W}}$, $\hat{\mathbf{M}}$, $\hat{\mathbf{C}}$ and $\hat{\mathbf{G}}$ are the estimates of $\tilde{\mathbf{W}}$, $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{G}$ in (2.15), and $\mathbf{u}$ the outer-loop feedback control signal (see Figure 2.9).

Then, substituting this computed torque control law in the dynamic model (2.15) yields the following linear system:

$$\ddot{\mathbf{X}} = \ddot{\mathbf{X}}_{\mathbf{ref}} + \mathbf{u} \tag{2.17}$$

Defining $\mathbf{e} = \mathbf{X}_{\mathbf{ref}} - \mathbf{X}$, the trajectory tracking error on the AWG pose, the resulting linear system is rewritten:

$$\ddot{\mathbf{e}} = -\mathbf{u} \tag{2.18}$$

To ensure convergence to zero of the tracking error $\mathbf{e}$, a feedback PD controller can be chosen for the outer loop:

$$\mathbf{u} = \mathbf{k_d}\dot{\mathbf{e}} + \mathbf{k_p}\mathbf{e} \tag{2.19}$$

with $\mathbf{k_p}$ and $\mathbf{k_d}$ respectively proportional and derivative definite positive gain matrices. The gain matrices are tuned for disturbance rejection and desired trajectory tracking performance.

Finally, the computed torque control law becomes:

$$\mathbf{w_2} = \hat{\mathbf{W}}^{-1}\left(\hat{\mathbf{M}}(\ddot{\mathbf{X}}_{\mathbf{ref}} + \mathbf{k_d}\dot{\mathbf{e}} + \mathbf{k_p}\mathbf{e}) + \hat{\mathbf{C}}\dot{\mathbf{X}} + \hat{\mathbf{G}}\right) \tag{2.20}$$

composed of (i) a feedforward term $\hat{\mathbf{M}}\ddot{X}_{\mathbf{ref}} + \hat{\mathbf{C}}\dot{\mathbf{X}} + \hat{\mathbf{G}}$ that provides the wrench to drive the system along the desired trajectory $\mathbf{X}_{\mathbf{ref}}$ by dynamic model inversion and (ii) a feedback term ($\mathbf{k_p}\mathbf{e} + \mathbf{k_d}\dot{\mathbf{e}}$) to remove the residual error due to the initial condition, since $\ddot{\mathbf{X}} = \ddot{\mathbf{X}}_{\mathbf{ref}}$ does not guarantee that the system follows the desired velocity $\dot{\mathbf{X}}_{\mathbf{ref}}$ or the desired position $\mathbf{X}_{\mathbf{ref}}$.

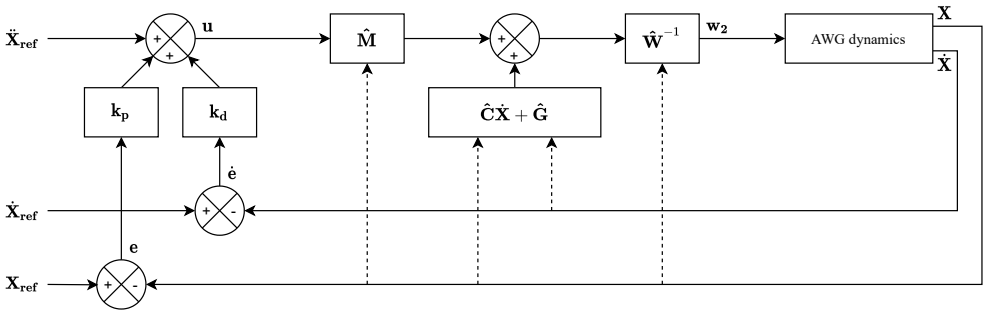


Figure 2.9 – Computed torque control block diagram.

Stability Analysis

In a classical computed torque control, as its name indicates, the output of the controller is directly the torque (or force) input reference of the actuators. When dealing with standard UAV ESCs, this is not possible. Indeed, the ESC throttle signal is proportional to the armature voltage of the equivalent DC motor. Thus, the rotor velocity depends on this voltage but also on rotor frictions or on the resistive torque. Even if the relationship between throttle and velocity could in theory be identified, variations between different motors and propeller characteristics, or even fluctuations of these characteristics during the experiment, make this approach very difficult to implement in practice. Since propeller thrust is related to its squared rotational velocity, implementing a low-level velocity loop at the actuator level allows for a far more precise control of the thrust. However, by taking into account the dynamics of this loop, the stability of the computed torque control law is no more guaranteed. A similar problem was studied for the control of a CDPR via feedback linearization (Begey et al., 2019), where a position loop is implemented at the winch-actuator level. The stability was proven by means of singular perturbation theory.

In the following, the stability of the reduced model (i.e. neglecting the velocity-loop dynamics of the propulsion units) is proven. Then, the robustness with respect to model parameter uncertainties and the influence of an integral action are discussed. Finally, taking into account the propeller dynamics, the proof of stability of the control law is presented using singular perturbation theory.

Stability of the Reduced Model Assuming the plant is perfectly modeled, the proof of stability is straightforward. Substituting equation (2.20) in equation (2.15) yields

the closed-loop error dynamics of the reduced model:

$$\ddot{\mathbf{e}} + \mathbf{k_d}\dot{\mathbf{e}} + \mathbf{k_p}\mathbf{e} = \mathbf{0} \tag{2.21}$$

The stability of the reduced model can be proved using different approaches. For example, in the specific case where the gain matrices are diagonal, the error dynamics are independent for each degree of freedom and the second-order differential equations can be solved analytically. Here, a more generic proof based on the Lyapunov theory for autonomous systems is proposed. Let $\mathbf{x_r} = (\mathbf{e}^T \ \dot{\mathbf{e}}^T)^T$ be the state vector of the reduced model. According to the Lyapunov stability criterion (Theorem 4.1 from (Khalil, 2002)), the reduced model is asymptotically stable at the origin $\mathbf{x_r} = \mathbf{0}$ if there exists a continuously differentiable scalar function $V = V(\mathbf{x_r})$, called Lyapunov function, such that:

- $V(\mathbf{0}) = 0$ and $V(\mathbf{x_r}) > 0$ for $\mathbf{x_r} \neq \mathbf{0}$,
- $\dot{V}(\mathbf{x_r}) < 0$ for $\mathbf{x_r} \neq \mathbf{0}$.

In other words, we are looking for a positive function whose derivative is negative definite. It is common to use the energy as a Lyapunov function candidate for mechanical systems. In our case, we can be inspired from a mass-spring-damper system:

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{2.22}$$

The total mechanical energy of the system (2.22) is:

$$E = \frac{1}{2}m\dot{x}^2 + \frac{1}{2}kx^2 \tag{2.23}$$

By identification, let us define the Lyapunov function candidate:

$$V(\mathbf{x_r}) = \frac{1}{2}\dot{\mathbf{e}}^T\dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^T\mathbf{k_p}\mathbf{e} \tag{2.24}$$

Substituting equation (2.21) in the time derivative of the Lyapunov function candidate yields:

$$\dot{V}(\mathbf{x_r}) = -\dot{\mathbf{e}}^T\mathbf{k_d}\dot{\mathbf{e}} \tag{2.25}$$

The Lyapunov function candidate is trivially positive (except at the origin), but its derivative is only semidefinite: if $\dot{\mathbf{e}} = \mathbf{0}$, then $\dot{V}(\mathbf{x_r}) = 0$ for any value of $\mathbf{e}$. The condition $\dot{V}(\mathbf{x_r}) = 0$ can remain true only if the system evolves on the line $\dot{\mathbf{e}} = \mathbf{0}$, which corresponds to zero speed trajectory in the mass-spring-damper analogy. However, the only state trajectory of the system (2.21) verifying $\dot{\mathbf{e}} = \mathbf{0}$ through time is the origin point $\mathbf{x_r} = \mathbf{0}$. We can conclude that the function $V(\mathbf{x_r}(t))$ is necessarily decreasing through time for any $\mathbf{x_r} \neq \mathbf{0}$. Therefore, the system is asymptotically stable at the origin $\mathbf{x_r} = \mathbf{0}$. Note that LaSalle's theorem (corollary 4.1 from (Khalil, 2002)) formalizes this last argument for asymptotic stability. The origin is asymptotically stable if:

- the origin $\mathbf{x_r} = \mathbf{0}$ is stable,
- the Lyapunov function candidate V is positive definite, i.e. $V(\mathbf{0}) = 0$ and $V(\mathbf{x_r}) > 0$ for $\mathbf{x_r} \neq \mathbf{0}$,
- the derivative of the Lyapunov function candidate $\dot{V}$ is negative semidefinite, i.e. $\dot{V} \leq 0$,
- the set $\dot{V}(\mathbf{x_r}) = 0$ contains no other trajectory than the origin $\mathbf{x_r}(t) = \mathbf{0}$, $t > 0$.

Robustness with Respect to the Spring Stiffness Parameter values may be different between the real plant and the model due to identification or measurement errors. Simulations show that the spring stiffness is a very sensitive parameter: a small overestimation may yield instability. Let us restrict, without loss of generality, the movement of the AWG to the vertical axis and let $\hat{k}$ be the estimated stiffness. The system is modeled as a harmonic oscillator with an external force input F:

$$m\ddot{z} + kz = F \tag{2.26}$$

Let us assume $z_{ref} = 0$ and hence $e = -z$. The computed torque control input is then calculated as follows:

$$F = m(-k_d\dot{z} - k_p z) + \hat{k}z \tag{2.27}$$

Substitution yields:

$$\ddot{z} + k_d\dot{z} + \left(k_p + \frac{k - \hat{k}}{m}\right)z = 0 \tag{2.28}$$

Therefore, the closed loop is stable if all coefficients are positive, i.e. if $k_p > \frac{\hat{k}-k}{m}$. As a consequence, underestimating the stiffness ensures stability for any positive values of k_p and k_d, and makes easier the initial experimental tuning of the controller.

Integral Action The exponential stability of the reduced model does not guarantee offset-free steady state due to discrepancy between the plant and its model in the control law. To illustrate this, it can be intuitively understood that a PD controller cannot compensate for dry friction. To eliminate steady-state error, it is common to add an integrator in the controller (Begey et al., 2019). However, actuators being saturated, integral windup may occur. Standard anti-windup technique that freezes the integrator when the PID output reaches saturation value of the system input can not be used since there is no direct mapping between the output of the PID controller (corresponding to a wrench) and the actuator saturation (see Figure 2.10). Such a mapping is difficult to establish due to the control allocation and feedforward input of the CTC taking place between these two signals. Therefore, important overshoots may occur during sharp and large variations of the reference signal (for example during a large step reference) or close to the boundaries of the AWG workspace.

Stability Proof Using Singular Perturbation Theory We propose an approach, similar to (Begey et al., 2019), to prove the stability of the closed-loop system when actuator dynamics are considered. We assume that the propeller rotational velocity control loop is significantly faster than the AWG dynamics. A first-order system with time constant t_m models the evolution of the actual velocity $\mathbf{w}$ according to the desired velocity $\mathbf{w_{ref}}$ as in equation (2.14).

Let us assume that $t_m = \varepsilon$, with ε a small parameter. The computed torque controlled AWG with its actuator dynamics can be written as a standard singular perturbation model (Khalil, 2002), with:

$$\begin{cases} \dot{\mathbf{x}} = \begin{pmatrix} \dot{\mathbf{X}} \\ \mathbf{M}(\mathbf{X})^{-1}(\tilde{\mathbf{W}}(\mathbf{X})\mathbf{w_2} - \mathbf{C}(\mathbf{X},\dot{\mathbf{X}})\dot{\mathbf{X}} - \mathbf{G}(\mathbf{X})) \end{pmatrix} & \text{(2.29a)} \\ \varepsilon\dot{\mathbf{w}} = \mathbf{w_{ref}} - \mathbf{w} & \text{(2.29b)} \end{cases}$$

of the form

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{w}) & \text{(2.30a)} \\ \varepsilon\dot{\mathbf{w}} = \mathbf{g}(\mathbf{x},\mathbf{w},t) & \text{(2.30b)} \end{cases}$$

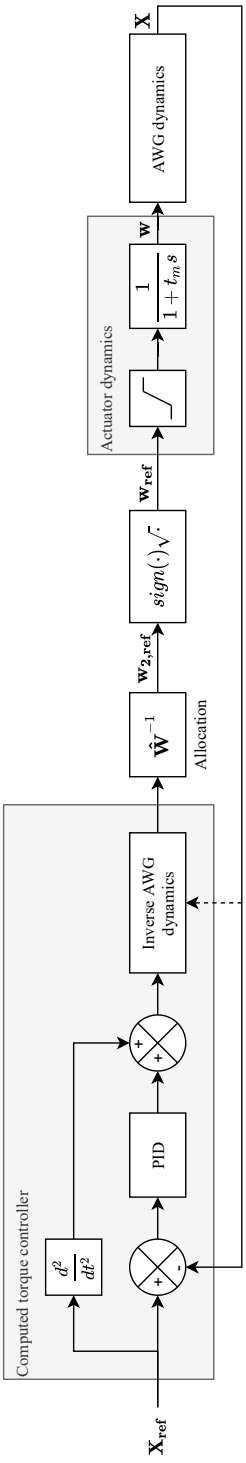


Figure 2.10 – CTC block diagram with actuator saturation.

and $\mathbf{x} = (\mathbf{X}^T\ \dot{\mathbf{X}}^T)^T$ the state vector of the AWG dynamics and $\mathbf{w_{ref}}(\mathbf{X}, t)$ the control output of the computed torque controller.

When ε is small, the state $\mathbf{w}$ converges rapidly to the reference $\mathbf{w_{ref}}$ since its time derivative $\dot{\mathbf{w}} = \frac{1}{\varepsilon}(\mathbf{w_{ref}} - \mathbf{w})$ becomes large. Thereby, for a small enough parameter ε, the system (2.29) exhibits a two-time scale behavior between (i) a slowly varying variable $\mathbf{x}$ and (ii) a fast varying and transient variable $\mathbf{w}$ superimposed on the slow subsystem.

From the singular perturbation theory based on Tikhonov's theorem (Theorem 11.2 from (Khalil, 2002)), the stability of the full system (2.29) can then be inferred from the stability of both its reduced (slow subsystem) model and boundary layer (fast subsystem) model.

The reduced model, or quasi-steady-state model (Spong, 1989), is derived from the singular perturbation model by setting $\varepsilon = 0$. In that case, the $\mathbf{w}$ state converges instantaneously to its unique quasi-steady-state value $\mathbf{w_{ref}}$. Substituting $\mathbf{w}$ with its quasi-steady-state value $\mathbf{w_{ref}}$ in (2.29a) yields the reduced model:

$$\dot{\mathbf{x}} = \begin{pmatrix} \dot{\mathbf{X}} \\ \mathbf{M}(\mathbf{X})^{-1}(\tilde{\mathbf{W}}(\mathbf{X})\mathbf{w_{ref,2}} - \mathbf{C}(\mathbf{X}, \dot{\mathbf{X}})\dot{\mathbf{X}} - \mathbf{G}(\mathbf{X})) \end{pmatrix} \tag{2.31}$$

Applying $\mathbf{w_{ref,2}} = (\cdots\ w_{ref,i}|w_{ref,i}|\ \cdots)^T$, the control output of the computed torque controller defined in (2.20), simplifies the reduced model to:

$$\dot{\mathbf{x}} = \begin{pmatrix} \dot{\mathbf{X}} \\ \ddot{\mathbf{X}}_{\mathbf{ref}} + \mathbf{k_d}(\dot{\mathbf{X}}_{\mathbf{ref}} - \mathbf{X}) + \mathbf{k_p}(\mathbf{X_{ref}} - \mathbf{X}) \end{pmatrix} \tag{2.32}$$

Defining the new state variable $\mathbf{e} = \mathbf{X_{ref}} - \mathbf{X}$, the reduced model finally becomes:

$$\begin{pmatrix} \dot{\mathbf{e}} \\ \ddot{\mathbf{e}} \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{k_p} & -\mathbf{k_d} \end{pmatrix} \begin{pmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{pmatrix} \tag{2.33}$$

This reduced model (2.33) matches the reduced model (2.21) previously introduced. As proven before, it is exponentially stable for $\mathbf{k_p}$ and $\mathbf{k_d}$ definite positive.

The reduced model assumes rapid convergence of $\mathbf{w}$ to $\mathbf{w_{ref}}$. To assess this convergence, stability of the fast dynamics of $\mathbf{w}$ with respect to its quasi-steady-state solution $\mathbf{w_{ref}}$ has to be studied. To study this behavior, a scaled time variable $t_\varepsilon = t/\varepsilon$ is introduced. With this scaled time, the singular perturbation model (2.29) becomes

($\frac{d\mathbf{w}}{dt_\varepsilon} = \frac{1}{\varepsilon}\frac{d\mathbf{w}}{dt}$):

$$\begin{cases} \dfrac{d\mathbf{x}}{dt_\varepsilon} = \varepsilon \mathbf{f}(\mathbf{x}, \mathbf{w}) & \text{(2.34a)} \\ \dfrac{d\mathbf{w}}{dt_\varepsilon} = \mathbf{w} - \mathbf{w_{ref}} & \text{(2.34b)} \end{cases}$$

When ε tends toward zero, the time scale t_ε becomes stretched with respect to time t and the "slow" variable $\mathbf{x}$ appears frozen (2.34a).

Setting $\varepsilon = 0$ in (2.34), the boundary layer model (or the fast subsystem) of the problem is obtained. Applying a change of variable $\mathbf{y} = \mathbf{w} - \mathbf{w_{ref}}$ to shift its equilibrium to the origin, the boundary layer model becomes:

$$\frac{d\mathbf{y}}{dt_\varepsilon} = -\mathbf{y} \tag{2.35}$$

Therefore, the boundary layer model is an exponentially stable first-order linear system.

Since the reduced model (2.33) and the boundary-layer model (2.35) are both exponentially stable at the origin, if $\mathbf{X_{ref}}$ is a feasible trajectory (continuous and has bounded derivative), Tikhonov's theorem (Theorem 11.2 from (Khalil, 2002)) states that the solution of the standard singular perturbation model $\mathbf{x}$ approaches the solution of the reduced model $\bar{\mathbf{x}}$: for $\varepsilon = t_m$ small enough, the singular perturbation problem has a unique solution that verifies:

$$\mathbf{x}(t, \varepsilon) - \bar{\mathbf{x}}(t) = O(\varepsilon) \tag{2.36}$$

Therefore, the computed-torque closed-loop system is stable if the time constant t_m of the propeller dynamics is small enough.

2.4.2 Experimental Results

In the following, experimental results characterizing the AWG performance are presented. Some of the experimental results are displayed on the associated video material: https://youtu.be/Ey88RkaMrNc.

Open-Loop Response

Figure 2.11 shows experimental open-loop responses of the AWG for various disturbances, where δz designates the variation of z with respect to the equilibrium position. In open loop, the AWG has a negligible damping. All DoFs, except the yaw, oscillate freely, at 0.3 Hz for x and y, 0.5 Hz for z, 2 Hz for ψ and θ, and the yaw is diverging.

Controller Tuning

The drag and the inertial effects are neglected for the experiments. Indeed, the maximum drag torque that can be generated with a single propulsion unit is approximately 0.1 N m and the maximum gyroscopic torque when the AWG rotates at 2π rad s^{-1} is approximately 0.4 N m, both significantly smaller than 2.4 N m, the maximum torque that the AWG can generate. Thus, propulsion units are considered pure force generators ($b = 0$ in $\mathbf{W}_b$ expression (2.7)).

The computed torque controller (2.20) of the AWG runs at a sampling frequency of 200 Hz. An integral action in added to the controller to remove steady-state offset. The PID gain matrices are diagonal, leading to a decoupled control of all DoFs:

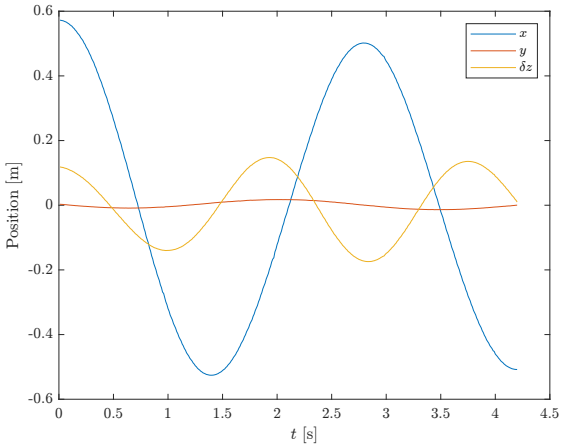
$$\mathbf{k_p} = diag(150, 150, 150, 150, 150, 120)$$
$$\mathbf{k_d} = diag(20, 20, 20, 20, 20, 20)$$
$$\mathbf{k_i} = diag(500, 500, 500, 500, 500, 500)$$

Note that the gain matrices have the same coefficients for all DoFs, except the yaw. Indeed, as shown with the open-loop responses, the yaw has no stable equilibrium, it is more sensitive to modeling errors and consequently to the tuning of the controller.

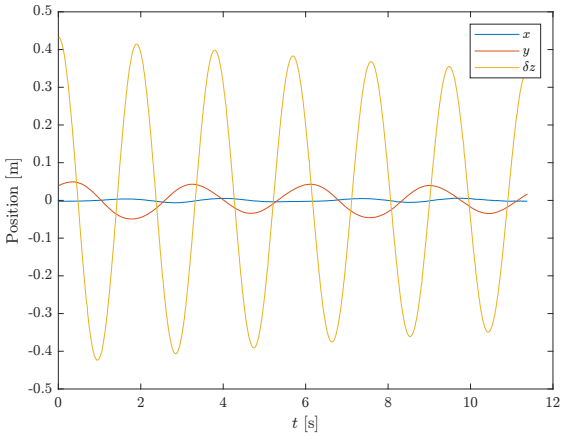
The model parameters are presented in Table 2.1. Experimental and simulated open-loop responses with same initial conditions are compared in Figure 2.12 using these parameters and show little difference. The experimental plot used for thrust coefficient identification is shown in Figure 2.13.

Static Performance

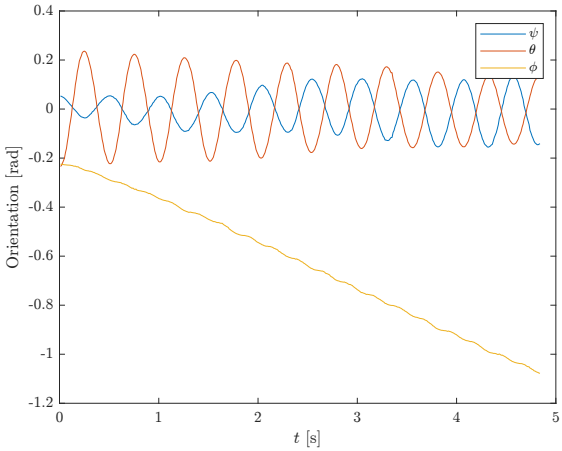
Workspace The workspace of the AMES depends mainly on the robotic carrier. However the motion of the carrier is supposed to be slow, so the fast dynamic workspace generated only by the propellers is restricted around the equilibrium point.



(a) Position disturbance along the $\mathbf{x_f}$ axis.



(b) Position disturbance along the $\mathbf{z_f}$ axis.



(c) Orientation disturbance.

Figure 2.11 – Open-loop responses.

Parameter	Description	Value
m	Total mass	$1.95\,\mathrm{kg}$
J_x	Moments of inertia about $\mathbf{x_b}$	$3.53 \times 10^{-3}\,\mathrm{kg\,m^2}$
J_y	Moments of inertia about $\mathbf{y_b}$	$3.44 \times 10^{-3}\,\mathrm{kg\,m^2}$
J_z	Moments of inertia about $\mathbf{z_b}$	$3.40 \times 10^{-3}\,\mathrm{kg\,m^2}$
k	Spring stiffness	$22\,\mathrm{N\,m^{-1}}$
l_0	Spring free length	$0.78\,\mathrm{m}$
L	Propeller axis to CoM distance	$0.17\,\mathrm{m}$
$\|\mathbf{GA}\|$	Spring to CoM distance	$0.345\,\mathrm{m}$
a	Thrust coefficient	$1.36\,\mu\mathrm{N\,rad^{-2}\,s^2}$
F_{max}	Maximum thrust	$7.2\,\mathrm{N}$

Table 2.1 – Model parameters.

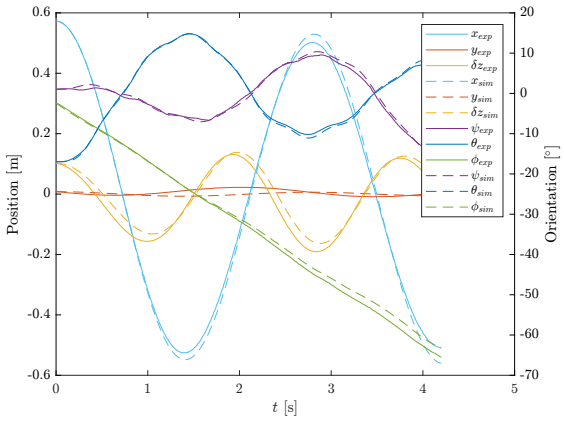


Figure 2.12 – Open-loop comparison of experimental and simulated data.

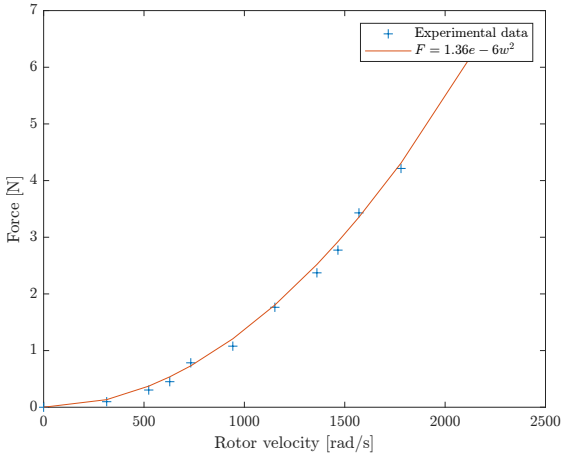


Figure 2.13 – Thrust coefficient identification.

The total orientation workspace (TOW) is the set of reachable positions for any orientation in a given set (Jiang & Gosselin, 2009). Since there is no kinematic constraints on the robot, this workspace is only limited by the spring and motor saturations, and can be found by discretizing the 3D space.

The TOW for a 5° tilt is given in Figure 2.14, in green the theoretical one obtained by a simulation discretizing the 3D space, and in red a cuboid whose vertices are tested experimentally. For the sake of simplicity, the experimental validation is done by testing if the robot can reach a 5° roll and pitch independently. Because of the geometry of the robot, the TOW has two planes of symmetry (Oxz) and (Oyz). The theoretical workspace volume is equivalent to the volume of a cube with a 0.65 m edge while the practical one is equivalent to a cube with only a 0.50 m edge. Uncertainties in the model parameters and especially in the spring characteristics may explain this difference.

To extend the TOW, it is possible to optimize further the design of the AWG. Moving the on-board anchoring point at the CoM, that is $\|\mathbf{GA}\| = 0$, would suppress any moment due to the elastic restoring force. Another possibility consists in increasing the distance between the propeller axis and the CoM, L, to be able to generate a higher torque, but this will deteriorate angular dynamics by increasing the inertia. With both strategies, the theoretical workspace could theoretically have a volume equivalent to a cube with a 1.86 m edge, with the current spring stiffness k.

Pose Characteristics Performance characteristics and test methods for industrial robots are defined by the ISO 9283 standard ("Manipulating industrial robots - Performance criteria and related test methods", 2016).

For each operational coordinate $q \in \mathbf{X}$, the pose accuracy AP_q measures the error between the desired pose q_c and the mean of the attained poses $\bar{q}$ (after repeating the same pose N times):

$$AP_q = (\bar{q} - q_c) \tag{2.37}$$

The positioning accuracy AP_P is the distance between the desired position and the barycenter of the attained positions: $AP_P = \sqrt{AP_x^2 + AP_y^2 + AP_z^2}$.

The pose repeatability RP measures the disparity between attained poses. The positioning repeatability RP_l is the sum of the average position errors $\bar{l} \geq 0$ and 3 times

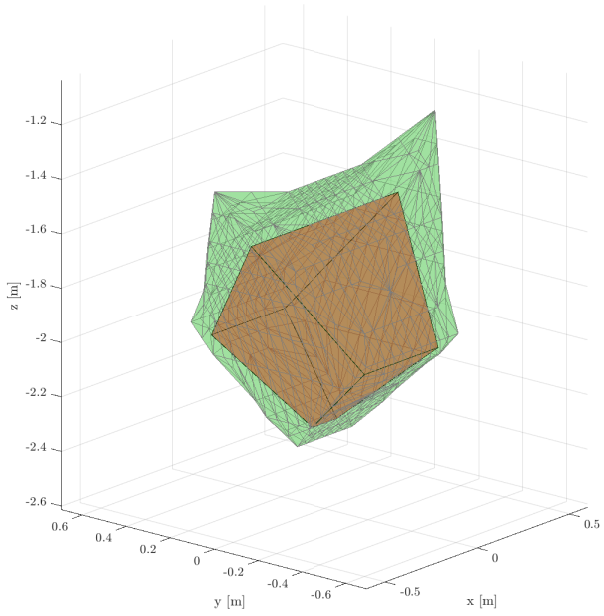


Figure 2.14 – Total orientation workspace of the AWG.

their standard deviation S_l:

$$RP_l = \bar{l} + 3S_l \tag{2.38}$$

For each orientation coordinate $\alpha \in \boldsymbol{\eta}$, the orientation repeatability RP_α is 3 times the standard deviation of the errors on α:

$$RP_\alpha = 3S_\alpha \tag{2.39}$$

The tests are done in a subset of the workspace: a cube centered at the equilibrium point of the robot, with edges parallel to the axes of the inertial frame $\mathscr{R}_f$ and of side length $S = 0.4\,\text{m}$ (see Figure 2.15). The positions P_1 to P_5 are selected according to the ISO 9283 standard. The robot performed $N = 10$ cycles from P_5 to P_1, with a maximum speed of $V_M = 0.63\,\text{m}\,\text{s}^{-1}$ and with a constant null orientation.

The results for the accuracy and repeatability tests are summarized in Table 2.2. For the positioning, the robot has a submillimetric accuracy and repeatability lower than 2 mm. For the orientation, both the accuracy and the repeatability are lower than 1°.

Note that the tool center point considered for assessing translational accuracy and repeatability is the CoM of the robot.

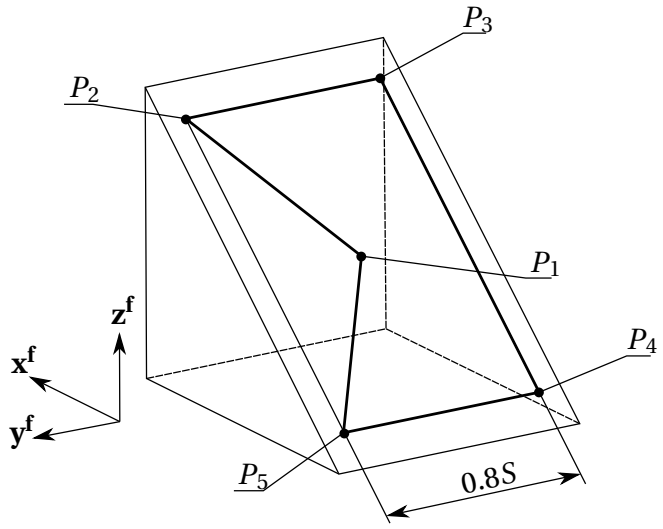


Figure 2.15 – Path (P_5 to P_1).

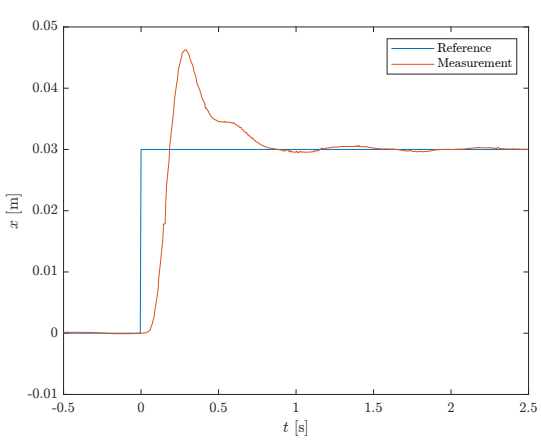
Positions	P_1	P_2	P_3	P_4	P_5
AP_P [mm]	0.14	0.08	0.11	0.10	0.19
RP_l [mm]	0.48	1.49	0.57	1.05	1.73
AP_ψ [°]	0.00	-0.04	-0.01	-0.02	0.01
RP_ψ [°]	0.16	0.61	0.30	0.42	0.39
AP_θ [°]	-0.00	-0.01	0.01	-0.01	0.03
RP_θ [°]	0.13	0.43	0.16	0.29	0.42
AP_ϕ [°]	0.00	-0.02	-0.02	-0.01	-0.01
RP_ϕ [°]	0.27	0.56	0.30	0.31	0.57

Table 2.2 – Accuracy and repeatability.

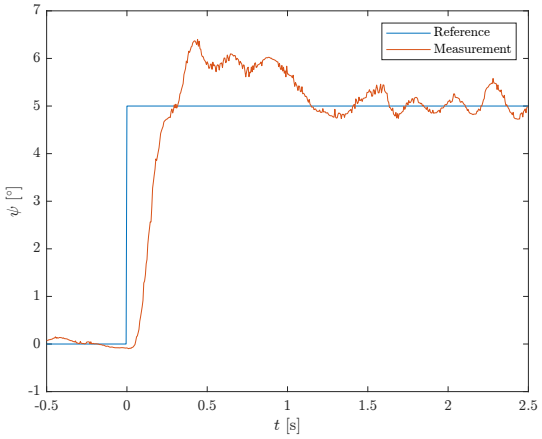
Dynamic Performances

Step Response Table 2.3 summarizes the step responses of the system for each independent DoF. These step responses are shown in Figure 2.16.

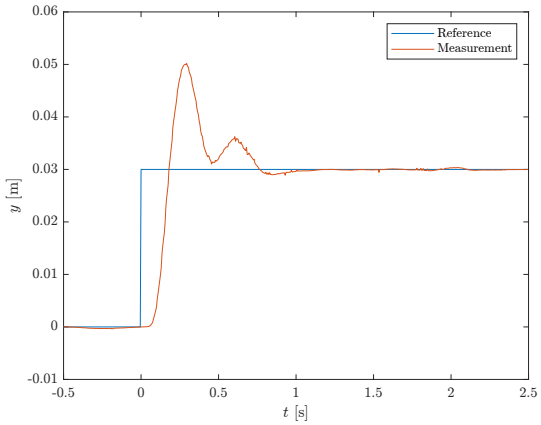
The step responses illustrate the high dynamics of the AWG. The rise time from 10% to 90% is less than 100 ms for a position step and less than 200 ms for an orientation step. The overshoot can be up to 70%, but is acceptable since practical trajectories are smoother and therefore will cause less overshoot. The steady-state error is eliminated, but small orientation oscillations persist due to measurement noise. A slight coupling between DoFs can be observed during transients due to the acceleration torque of the propulsion units.



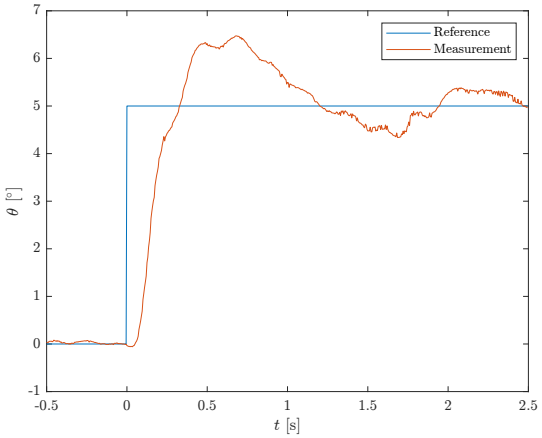
(a) Translation along $\mathbf{x_f}$ axis.



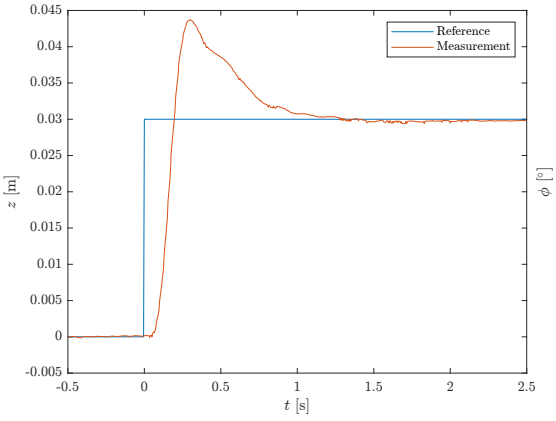
(b) Roll (ψ).



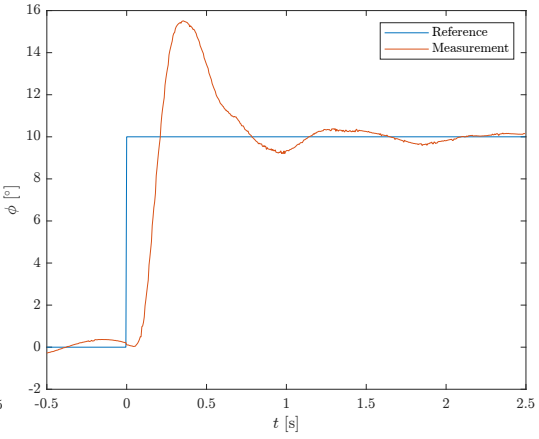
(c) Translation along $\mathbf{y_f}$ axis.



(d) Pitch (θ).



(e) Translation along $\mathbf{z_f}$ axis.



(f) Yaw (ϕ).

Figure 2.16 – Position (left) and orientation (right) step responses.

Table 2.3 – Step responses

	$\mathbf{x_f}$	$\mathbf{y_f}$	$\mathbf{z_f}$	ψ	θ	ϕ
Rise time [s]	0.09	0.08	0.09	0.14	0.18	0.09
Overshoot [%]	54.20	67.23	45.74	28.02	29.45	55.10

Maximum Acceleration To measure the maximum acceleration of the robot, a step reference that saturates the actuators is used. Since the second derivative of the position obtained by the motion capture system is highly noisy and there is no available accelerometer on the robot, the acceleration is obtained using linear regression on a sliding window with 15 acquisitions of the AWG velocity. At the equilibrium position, the maximum acceleration is approximately $7\,\mathrm{m\,s^{-2}}$ in $\mathbf{x_f}$ and $\mathbf{y_f}$ directions and $6\,\mathrm{m\,s^{-2}}$ in $\mathbf{z_f}$ direction.

The AWG can achieve higher accelerations at positions in the workspace where the gravity is not fully compensated by the spring. At 60 cm above the equilibrium position, the maximum downward acceleration is $1.33g$. Note that a standard suspended CDPR cannot achieve a downward acceleration higher than $1g$, which occurs during free fall.

It is possible to reach an even higher acceleration by going beyond the static workspace of the robot and therefore minimizing the gravity compensation from the spring (Xiang et al., 2020).

Disturbance Rejection Carrying a load, for example during a pick and place task, results in a vertical force step disturbance. The rejection of such a disturbance is assessed in a repeatable way by hanging a 0.5 kg weight (25 % of the mass of the AWG without the load) to the bottom of the AWG, where the gripper should be located, and cutting it off. The response is shown in Figure 2.17, the disturbance happens at $t = 0$. The maximum displacement is less than 2 cm, and the disturbance is rejected in approximately 700 ms.

Note that the load could also be included in the dynamic model for pick-and-place applications and not be considered a disturbance. Moreover, since the disturbance rejection performance depends on the type and the tuning of the controller, these results are shown only for comparison purposes with other control laws.

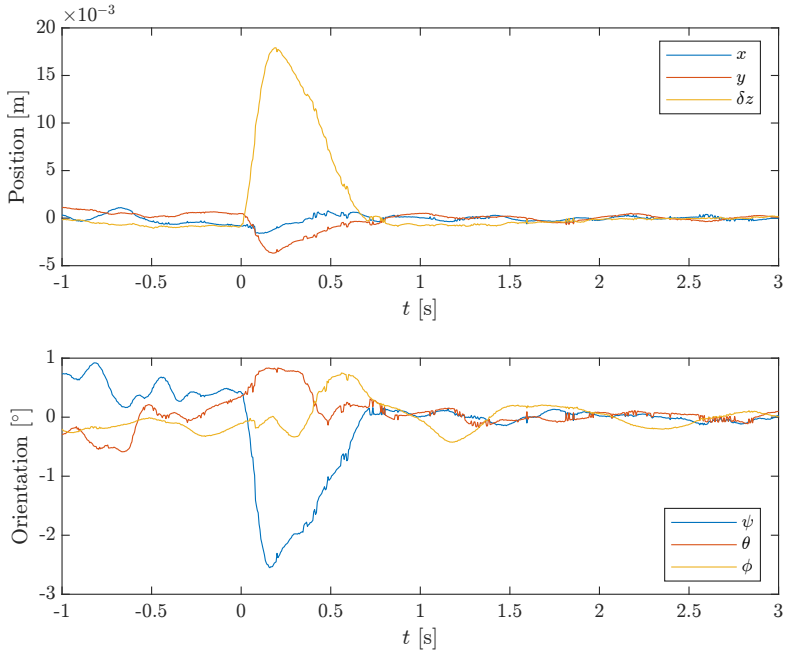


Figure 2.17 – Disturbance rejection.

Energy Performance

The total thrust generated by the AWG is not sufficient to keep the AWG hovering without the spring, therefore it is not possible to do an experimental comparison showing the impact of the spring on the energy consumption. Thus, a model based on the rotational velocity of the propulsion units is used to simulate the energy consumption of the AWG without its suspension spring.

The mechanical power P_m developed by the i-th propeller is the product of the rotational velocity w_i by the drag torque $\tau_i = bw_i^2$. Therefore, it is proportional to the cube of the rotational velocity:

$$P_m = bw_i^3 \tag{2.40}$$

On a propulsion unit, since there is always one propeller rotating at its minimum speed, the total power consumption can be modeled as

$$P = c_1 w^3 + c_2$$

with c_1 and c_2 constants to be identified.

The consumed electric power P_e is the product of the battery voltage U by the battery

current I:

$$P_e = UI \tag{2.41}$$

The voltage U depends on the charge of the battery and on its serial resistance, it is assumed constant. In steady state, that is when the rotational velocities are constant, the mechanical power and the electric power are equal up to a multiplicative constant corresponding to the efficiency. During a smooth trajectory, propeller velocities are considered quasi-constant, so the power consumed for acceleration can be neglected. The telemetry data obtained from ESCs contains the battery voltage and current that is drawn, therefore, it is possible to measure the electrical power consumption of the AWG. In Figure 2.18, electric energy consumption measured on the AWG prototype and simulated consumption are compared for $c_1 = 2.05e-8$ and $c_2 = 8.75$. The reference trajectory is the one used for the accuracy and repeatability tests with 10 cycles. Flat parts of the curve correspond to the equilibrium position, where the consumption is low.

The energy consumption of the AWG without its spring is estimated in simulation. For the same trajectory, the energy consumption without the spring is 49 Wh, 3.8 times higher than with the spring. It justifies the use of a spring to minimize energy consumption of the aerial manipulator without increasing too much the mechanical complexity.

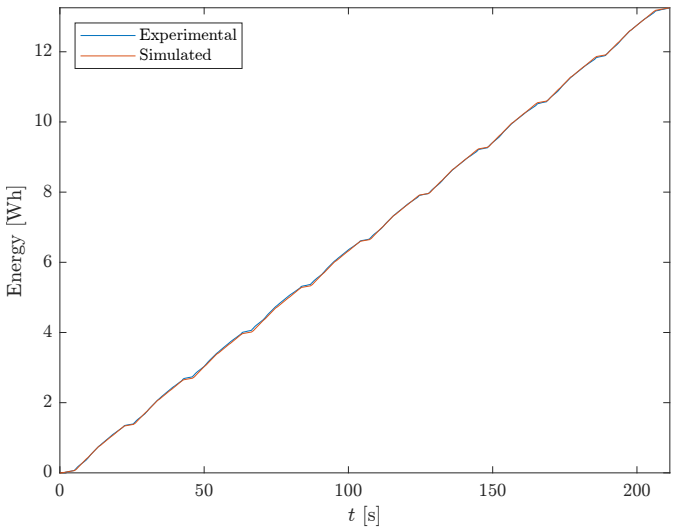


Figure 2.18 – Energy consumption of the AWG.

2.5 Nonlinear Model Predictive Control

After characterization of the AWG performance with a computed torque controller, a nonlinear model predictive control is proposed to improve the system performance.

To control a plant, model predictive control (MPC) uses a model of the plant to predict its most likely future behavior for different inputs. Among these input candidates, the chosen one is the one that minimizes a cost function under constraints on the inputs and states. Unlike the linear-quadratic regulator (also known as LQR), a constrained optimization problem is solved online and periodically, at each time step. Initially intended for linear systems, it is now possible to deal with nonlinear plants thanks to increased computational power and state-of-the-art solvers.

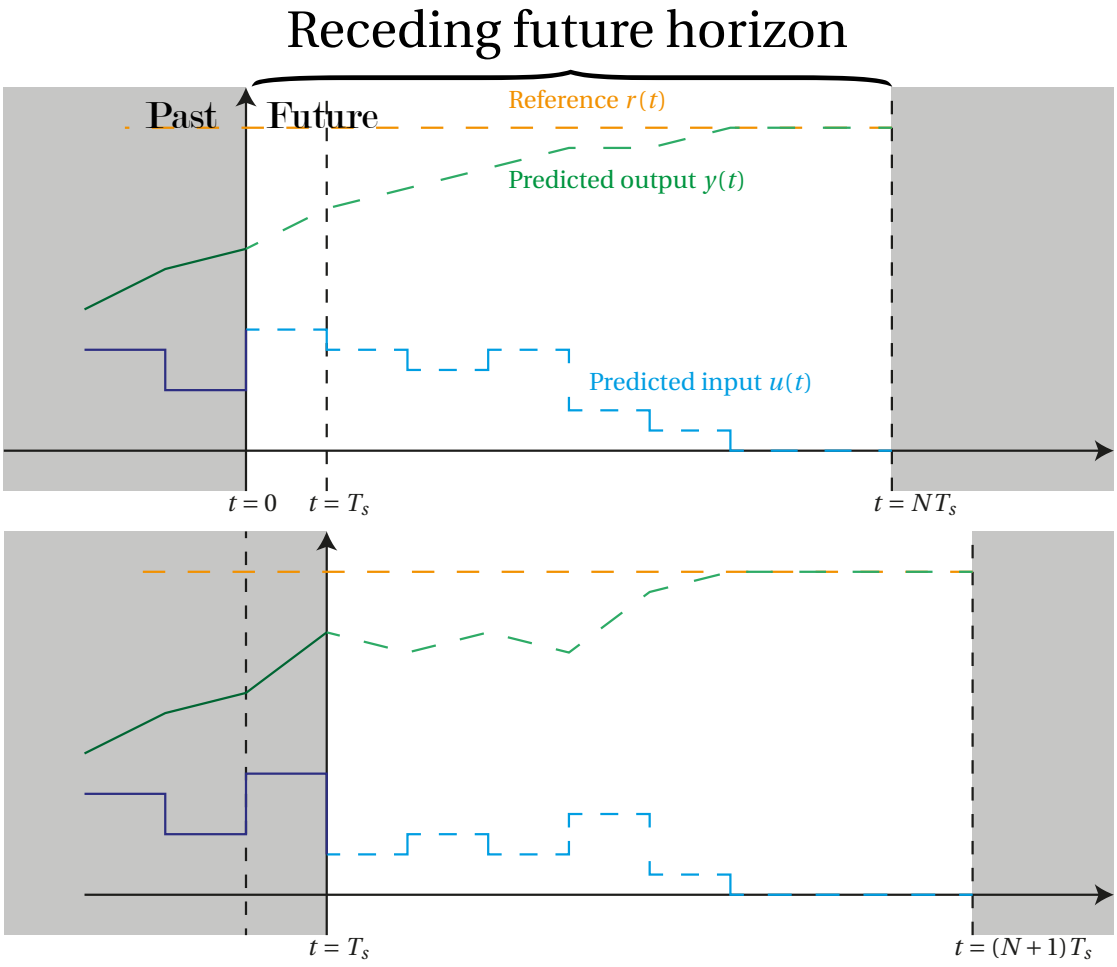


Figure 2.19 – MPC concept, inspired by (Bemporad, 2021).

Figure 2.19 shows two consecutive time steps of MPC, where the output of the system tries to reach a constant reference signal. At each time step, the input and the output

are predicted on a receding horizon (whose length is NT_s), but only the first sample of the optimal input is applied.

There exist many analogies for MPC. Chess engines share some similarities with NMPC. Nowadays, decision-making algorithms are unable to find the best move in a reasonable amount of time. Therefore, they must cope with a suboptimal solution, and hence, the performance of chess engines depends on the efficiency of the optimizer (since there is only a limited amount of time allowed for moves). Similarly, NMPC uses nonconvex models which are not known for their efficiency, with the risk of being trapped in a local minimum. Rather than looking for the "best" solution, different approaches yield suboptimal but "acceptable" behaviors.

An example of MPC for a single-input single-output system is illustrated in Figure 2.20. The quantity J to minimize is a weighted sum of (i) the area between the curves of the reference and the controlled variable and (ii) the area under the curve of the control input:

$$J = \int_0^T \big(|r(t) - y(t)| + \lambda |u(t)|\big)\, dt \tag{2.42}$$

Note that the control input, and consequently the controlled variable, can anticipate the reference signal if the reference signal is known in advance. In this case, for a step reference, the optimal output should be close to a symmetric response before and after the step.

The influence of the parameter λ and of the prediction horizon length T are shown in Figure 2.21. Note that, in this example, the 1-norm is replaced by the square of the 2-norm in the cost function. Two observations can be made: (i) the higher the λ parameter, the higher the penalty on high control input amplitudes, yielding a slower response and (ii) the shorter the prediction horizon T, the more aggressive the response.

Despite ongoing research to solve quadratic problems using analog circuits (Vichik & Borrelli, 2014), most model predictive control implementations use a digital control strategy. Nevertheless, it is possible (and even preferable) to use time-continuous plant models for NMPC with numerical integration. As a matter of fact and in most cases, no explicit or analytical discrete-time model can be derived from a continuous-time nonlinear differential equation.

Let us consider a plant model described by a set of continuous ordinary differential

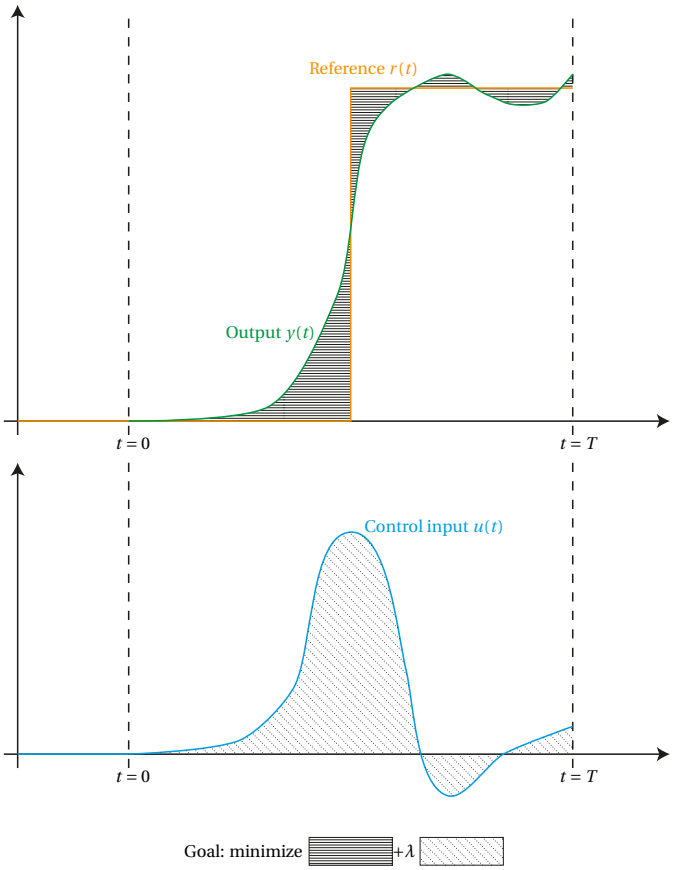


Figure 2.20 – MPC illustration.

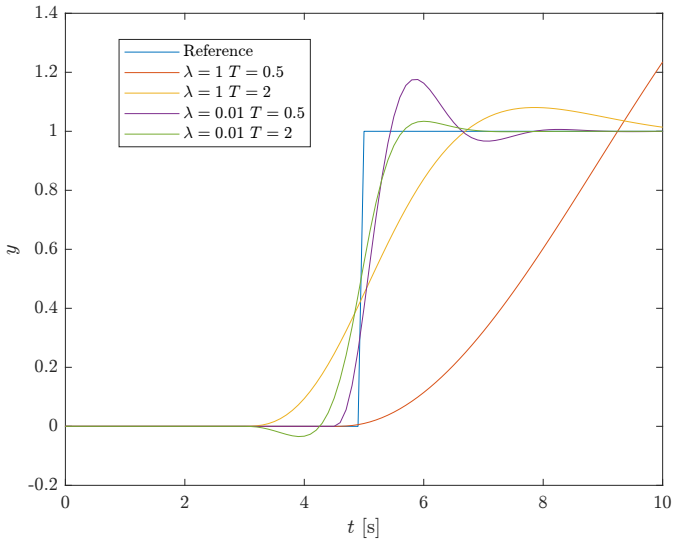


Figure 2.21 – Influence of parameters of the MPC.

equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{2.43}$$

and subject to state and input constraints

$$\mathbf{h}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \tag{2.44}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector and $\mathbf{u} \in \mathbb{R}^m$ the input vector.

Typically, Nonlinear Model Predictive Control aims to solve the following nonlinear optimal control problem (OCP):

$$\begin{aligned} &\min_{\mathbf{u},\mathbf{x}} \left[\int_0^T \left(\left\| \mathbf{y}(t) - \mathbf{y_{ref}}(t) \right\|_{\mathbf{Q}}^2 + \left\| \mathbf{u}(t) - \mathbf{u_{ref}}(t) \right\|_{\mathbf{R}}^2 \right) dt \right] \\ &\text{subject to } \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ &\qquad \mathbf{x}(0) = \mathbf{x_0} \\ &\qquad \mathbf{h}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \end{aligned} \tag{2.45}$$

The cost function is an integral over the control horizon T of the weighted sum of two quadratic terms: the first one measures the deviation between the controlled variables $\mathbf{y} \subset \mathbf{x}$ and their reference $\mathbf{y_{ref}}$, and the second one between the control input $\mathbf{u}$ and its reference $\mathbf{u_{ref}}$. In practice, the reference of the control input $\mathbf{u_{ref}}$ is often omitted to have the second term corresponding to the energy of the control input (in the sense of the energy of a signal). The quadratic error on the controlled variables is weighted by a positive semidefinite (or definite) matrix $\mathbf{Q}$ and the quadratic error on the control input by a positive definite matrix $\mathbf{R}$, the operator $\|\cdot\|_{\mathbf{P}}$ being defined such that $\|\mathbf{v}\|_{\mathbf{P}}^2 = \mathbf{v}^T\mathbf{P}\mathbf{v}$. The weighting matrices are often chosen diagonal. Note that if the $\mathbf{R}$ matrix is diagonal and only semidefinite, i.e. at least one of the eigenvalues is zero, then we can expect the control input terms associated with the zero eigenvalues to behave as a binary signal, switching between the maximum and the minimum value like with a bang-bang controller. The control horizon length T needs to be chosen as short as possible without deteriorating the closed-loop behavior too much. A commonly used rule of thumb states that the control horizon length T could correspond to the closed-loop settling time.

The function $\mathbf{h}$ is often used to impose saturation on the actuators (corresponding to physical limits) or on the states (for example to ensure the state stays in their area of validity). The term $\mathbf{x_0}$ corresponds to the current state of the plant at the instant the

OCP needs to be solved.

2.5.1 Problem Formulation for the AWG

From equations (2.1), (2.2) and (2.14), the nonlinear plant model can be written in the form of

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{w_{ref}}) \tag{2.46}$$

with $\mathbf{x} = (\mathbf{p}^T \, \boldsymbol{\eta}^T \, \dot{\mathbf{p}}^T \, \boldsymbol{\omega}^T \, \mathbf{w}^T)^T$. In order to take into account actuator limitations, we add saturation constraints to propeller velocity reference: $|w_{ref,i}| \leq w_{MAX}$.

Let $\mathbf{y} = (\mathbf{p}^T \, \boldsymbol{\eta}^T \, \dot{\mathbf{p}}^T \, \boldsymbol{\omega}^T)^T$ be the controlled variables, $\mathbf{y_{ref}}$ their reference obtained with a trajectory generator and J the cost function to be minimized:

$$J = \int_0^T \left(\left\| \mathbf{y}(t) - \mathbf{y_{ref}}(t) \right\|_{\mathbf{Q}}^2 + \left\| \mathbf{w_{ref}}(t) \right\|_{\mathbf{R}}^2 \right) dt \tag{2.47}$$

We can identify two issues with this cost function. First, the rate of change of the control input $\mathbf{w_{ref}}$ is not bounded and therefore important chattering may be observed. Second, except at the equilibrium point (i.e. at steady state with $\mathbf{w} = \mathbf{0}$), the control input $\mathbf{w_{ref}}$ is not zero at steady state, so the minimal solution may not correspond to the desired pose for the AWG. In particular, one may note that $J = 0$ is not feasible at steady state except at the equilibrium point, therefore, the minimal solution is a trade-off between the energy of the control input signal and the quadratic error on the controlled variables.

The "delta-input formulation" is one way to deal with these problems. The system is augmented such that the derivative of the physical control input becomes the new control input: $\mathbf{x} = \left(\mathbf{p}^T \, \boldsymbol{\eta}^T \, \dot{\mathbf{p}}^T \, \boldsymbol{\omega}^T \, \mathbf{w}^T \, \mathbf{w_{ref}}^T\right)^T$ and $\mathbf{u} = \dot{\mathbf{w}}_{\mathbf{ref}}$. Since the rotational velocity reference $\mathbf{w_{ref}}$ is constant at steady state, its derivative $\dot{\mathbf{w}}_{\mathbf{ref}}$ is zero for this state. The cost function J is modified consequently:

$$J = \int_0^T \left(\left\| \mathbf{y}(t) - \mathbf{y_{ref}}(t) \right\|_{\mathbf{Q}}^2 + \left\| \dot{\mathbf{w}}_{\mathbf{ref}}(t) \right\|_{\mathbf{R}}^2 \right) dt \tag{2.48}$$

Finally, the continuous OCP to be solved is the following:

$$\begin{aligned} \min_{\mathbf{u},\mathbf{x}} \quad & \left[\int_0^T \left(\left\| \mathbf{y}(t) - \mathbf{y_{ref}}(t) \right\|_{\mathbf{Q}}^2 + \left\| \dot{\mathbf{w}}_{\mathbf{ref}}(t) \right\|_{\mathbf{R}}^2 \right) dt \right] \\ \text{subject to } & \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ & \mathbf{x}(0) = \mathbf{x_0} \\ & \mathbf{h}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \end{aligned} \tag{2.49}$$

2.5.2 Steady-State Error

The delta-input formulation alone does not guarantee a zero steady-state error in closed loop. A counter-example for the linear case can be found in the lecture notes from (Bemporad, 2021). External disturbances, such as wind, and unmodelled or inaccurate dynamics can also cause a steady-state error on the AWG pose. Indeed, there is no guarantee that the optimal control input will enforce a zero steady-state error if the OCP problem is solved based on an inaccurate model whose output does not match the real plant output.

At steady state, i.e. $\dot{\mathbf{x}} = 0$, we assume that the propeller velocity error is zero due to the integral action of its inner velocity control loop. Therefore, the only source of errors comes from model inaccuracies of the AWG mechanism.

To estimate these modeling errors, an extended Kalman filter (EKF) implements the plant model augmented with constant disturbances:

$$\underbrace{\begin{pmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{d}} \end{pmatrix}}_{\dot{\mathbf{x}}_\mathbf{a}} = \underbrace{\begin{pmatrix} \mathbf{f}(\mathbf{x}, \mathbf{u}) + \mathbf{B_d}\mathbf{d} \\ \mathbf{0} \end{pmatrix}}_{\mathbf{f_a}(\mathbf{x_a}, \mathbf{u})} \tag{2.50}$$

where $\mathbf{d}$ is a vector of six constant disturbance variables that takes into account the AWG modeling errors and $\mathbf{B_d} = (\mathbf{0}_{6\times6}\ \mathbf{I}_{6\times6}\ \mathbf{0}_{6\times12})^T$ is a matrix that maps the disturbances to the derivatives of $\dot{\mathbf{p}}$ and $\boldsymbol{\omega}$ (the dimensions of the identity and zero matrices are added to avoid confusion).

These estimated disturbances $\hat{\mathbf{d}}$ are fed back to the NMPC (see Figure 2.22) which will reject them based on the augmented model resulting in a zero steady-state error.

Note that as the number of added disturbances ($\dim(\mathbf{d}) = 6$) is lower than the number of measurements ($\dim(\mathbf{y}) = 12$), the above method does not guarantee offset-free

steady state (Pannocchia et al., 2015) and this point has to be assessed by simulation. However, by limiting the number of additional disturbance states, the size of the augmented system and, thereby, the OCP computation burden are reduced.

A sufficient condition to achieve offset-free steady state will be discussed in Section 2.6.

2.5.3 Controller Implementation

The NMPC is implemented using acados (Verschueren et al., 2019), a recent open-source framework that generates efficient low-level code for optimization-based control. The continuous OCP (2.45) is discretized with a multiple shooting method:

$$\begin{aligned} &\min_{\underline{\mathbf{u}},\underline{\mathbf{x}}} \left[\sum_{k=0}^{N-1} \left(\left\| \mathbf{y_k} - \mathbf{y_{ref,k}} \right\|_{\mathbf{Q}}^2 + \left\| \dot{\mathbf{w}}_{\mathbf{ref,k}} \right\|_{\mathbf{R}}^2 \right) \right] \\ &\text{subject to } \mathbf{x_{a,k+1}} = \mathbf{f_k}(\mathbf{x_{a,k}}, \mathbf{u_k}) \\ &\qquad \mathbf{x_{a,0}} = \tilde{\mathbf{x}}_{\mathbf{0}} \\ &\qquad \mathbf{h}(\mathbf{x_{a,k}}, \mathbf{u_k}) \geq \mathbf{0} \end{aligned} \tag{2.51}$$

where $\mathbf{v_k} = \mathbf{v}(kT_s)$ for a variable $\mathbf{v}$, T_s is the sampling interval, $\mathbf{x_a}$ the state of the augmented model previously defined in (2.50), $\mathbf{f_k}$ the Runge-Kutta method approximation of $\mathbf{f_a}$ at $t = kT_s$, $\tilde{\mathbf{x}}_{\mathbf{0}}$ the state of the augmented model at the time the OCP needs to be solved.

Sequential quadratic programming (SQP) is used to generate quadratic programming (QP) subproblems that solve the OCP using the HPIPM solver (Frison & Diehl, 2020). For real-time compliance, acados performs the real-time iteration scheme that enables a reasonably fast computation of the NMPC output, but at the cost of a suboptimal solution (Houska et al., 2011b).

At the time the experiments were carried out, acados was not compatible with 32-bit architectures. Therefore, the controller runs on a distant computer equipped with an Intel i5-9500 processor instead of the AWG on-board computer. Furthermore, running the controller on a distant computer allows for higher computational power and thereby allows for a better tuning of controller parameters (especially a longer prediction horizon). The distant computer hosts a TCP/IP server. The server receives the input data of the controller (setpoint and estimated state) from the embedded computer, and sends back the control input and the estimated next state.

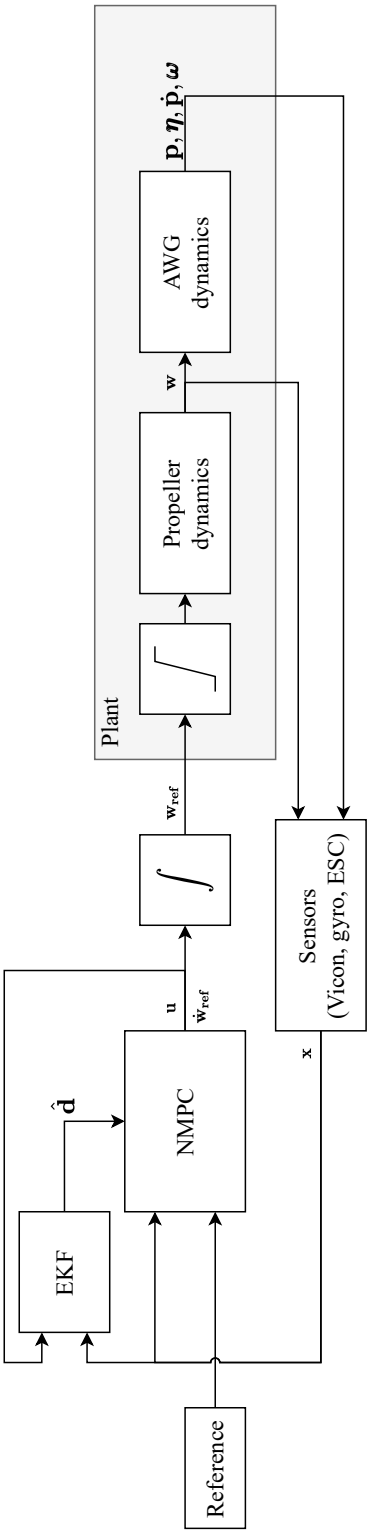


Figure 2.22 – AWG NMPC block diagram.

The electronic architecture of the experimental setup for NMPC experiments is given in Figure 2.23.

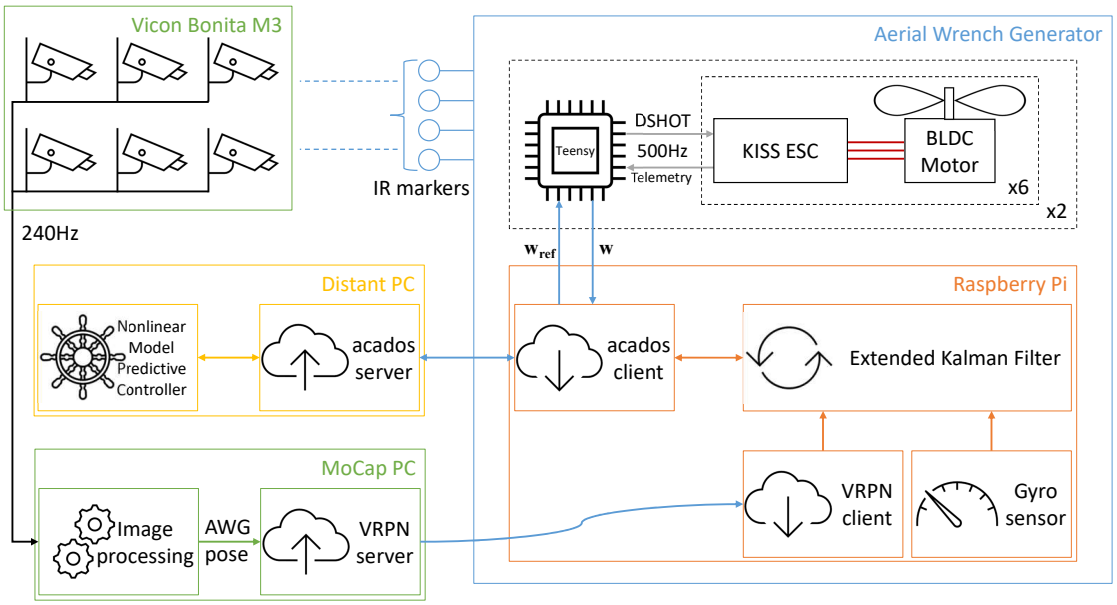


Figure 2.23 – AWG NMPC architecture.

2.5.4 Results

Experimental results using NMPC are presented in this section. The computed torque controller with integral action is used for comparison purpose.

Controller Tuning

As with computed torque control presented previously, the propeller speed control loop is tuned such that, in closed loop, it behaves like a 25 ms time constant first-order system. This model does not hold at high speeds due to current saturations. Hence, propeller speed is limited to 22 000 rpm (instead of the 23 000 rpm maximal speed) by adding a constraint on the corresponding state in the OCP. The rate of change of the desired propeller speed can also be bounded to take into account motor current saturation (Bicego et al., 2020).

The NMPC runs at 100 Hz with a 0.5 s prediction horizon. To track a reference trajectory, we control the pose and the twist of the AWG:

$$\mathbf{y} = \left(\mathbf{p}^T \, \boldsymbol{\eta}^T \, \dot{\mathbf{p}}^T \, \boldsymbol{\omega}^T\right)^T$$

The weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ are diagonal:

$$\mathbf{Q} = diag(\mathbf{Q_p}, \mathbf{Q_\eta}, \mathbf{Q_{\dot{p}}}, \mathbf{Q_\omega})$$

$$\mathbf{R} = R_u \mathbf{I}_{6\times 6}$$

The weight values given in Table 2.4 were tuned on the experimental system. In particular, the weight associated to the control input is significantly smaller by several orders of magnitude than the other weights because of its wide range of variation. Indeed, according to the first-order model assumption for the actuators, the propeller speed can approach its saturation value in less than 100 ms. The control input, which is the rate of change of the propeller speed, can therefore exceed $10\,000\,\mathrm{rad\,s^{-1}}$ (and its square $10^8\,\mathrm{rad^2\,s^{-2}}$).

Parameter	Diagonal entries
$\mathbf{Q_p}$	$150, 150, 150$
$\mathbf{Q_\eta}$	$10, 10, 5$
$\mathbf{Q_v}$	$0.02, 0.02, 0.02$
$\mathbf{Q_\omega}$	$0.1, 0.1, 0.02$
R_u	$1.25e-10$

Table 2.4 – NMPC tuning parameters.

Actuator Dynamics

Propulsion unit dynamics are much faster than closed-loop dynamics of the AWG, and therefore, including them in the NMPC model may not improve significantly the performance of the controller. On the contrary, since it increases the number of states, it limits the length of the prediction horizon and the sampling period.

Simulations have been carried out with and without the propulsion unit dynamics in the NMPC model. During these simulations, no EKF is used and the model is not augmented with constant disturbances. The comparison is shown in Figure 2.24 for a 10 cm step reference along the vertical axis: including propulsion unit dynamics in the NMPC model has a negligible impact on the closed-loop response. Therefore, they can safely be omitted for future experiments.

Furthermore, neglecting the actuator dynamics in the plant model allows for using the signed squared propeller rotational velocities $\mathbf{w_2} = (\cdots\ w_i|w_i|\ \cdots)^T$ as input rather than the rotational velocities $\mathbf{w}$, thus removing some nonlinearities with respect to

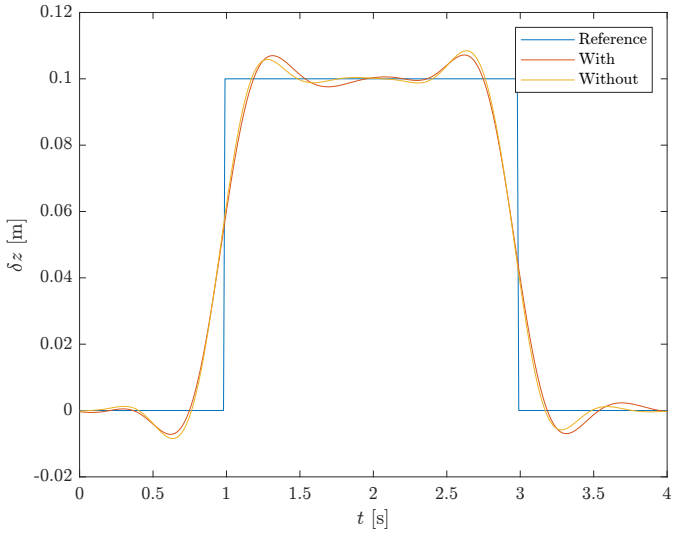


Figure 2.24 – Influence of the propulsion unit dynamics in the NMPC model.

the input. Indeed, at low speed, a small variation of the squared velocity requires an important variation of the velocity itself, but at high speed, it is the opposite: $\frac{dw}{dt} = \frac{dw^2}{dt}\frac{1}{2w}$.

Disturbance Rejection

As discussed in Section 2.5.2, the ability of the NMPC to reject a constant disturbance needs to be verified experimentally. An 0.5 kg load is attached at the bottom of the AWG, similarly to the disturbance rejection experiment with the CTC. The displacement δz along the $\mathbf{z_f}$ axis relatively to the equilibrium position ($\delta z = 0$) with and without the EKF disturbance estimator is shown in Figure 2.25. Without the disturbance estimation, the AWG has a 50 mm steady-state error visible at $t < 0$ on the figure. When the load is cut off at $t = 0$, a 15 mm error persists. Adding the disturbance observer eliminates the steady-state error.

Without the EKF, the transient when the load is cut off lasts less than 0.7 s, similarly to the results obtained with the CTC. With the EKF, however, the transient is much longer. The AWG reaches its steady state after approximately 2 s. Therefore, the disturbance rejection speed is not limited by the NMPC but by the EKF dynamics. The state noise covariance matrix of the EKF can be tuned to speed up the dynamics of the disturbance estimates and thus their rejection.

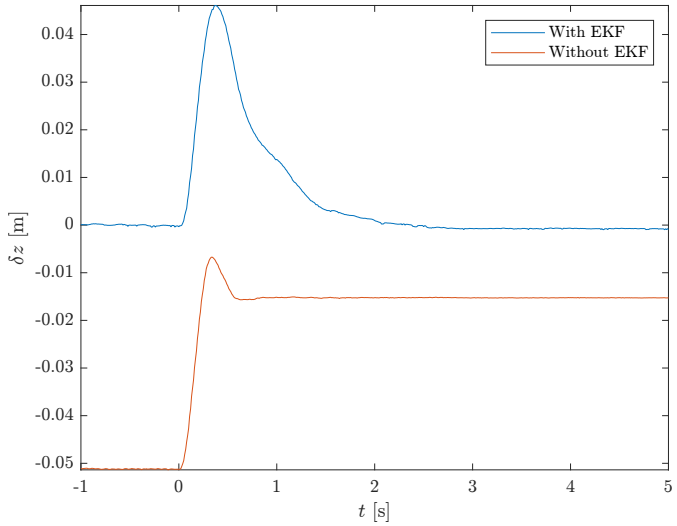


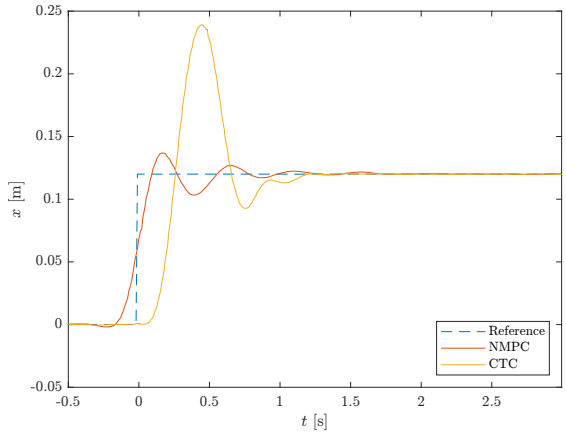
Figure 2.25 – Disturbance rejection with NMPC.

Step Response

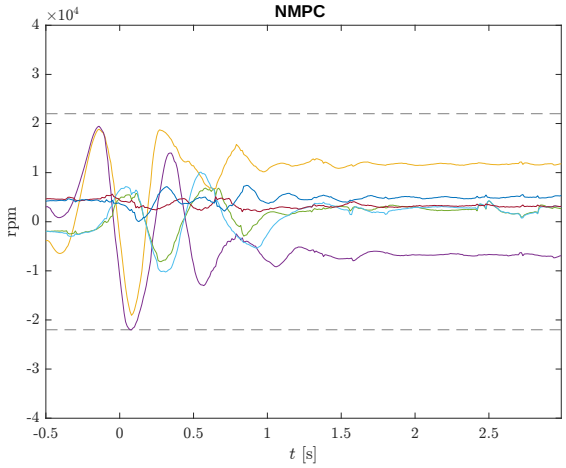
The step responses and the control inputs of the propulsion units in Figure 2.26 compare the NMPC and the CTC for a large step reference along the $\mathbf{x_f}$ axis direction. The NMPC yields dynamics comparable to the CTC, but with significantly less overshoot along all directions (see Table 2.5). Indeed, with actuator saturation, a significant position error is accumulated in the integral term of the CTC during the travel to the setpoint. It results in an integral windup yielding an excessive overshoot while the accumulated error is dissipating. On the other hand, the NMPC anticipates the new reference and actuator saturation, enabling smooth variations of the control inputs. Furthermore, as it can be seen on the lower plots of Figure 2.26, the control input signal of the NMPC exhibits less high frequency noise than the CTC that may cause copper losses in the motor and vibrations, thus adding a beneficial roll-off effect. Therefore, energy consumption should be higher with the computed torque controller. Indeed, during the time slot shown in Figure 2.26, the AWG consumes 10.7 Wh with the computed torque controller while only 3.1 Wh with the NMPC.

Axis	$\mathbf{x_f}$	$\mathbf{y_f}$	$\mathbf{z_f}$
Rise time [s] CTC	0.13	0.13	0.14
Rise time [s] NMPC	0.18	0.18	0.19
Overshoot [%] CTC	95.7	110.6	71.1
Overshoot [%] NMPC	14.0	19.0	9.9

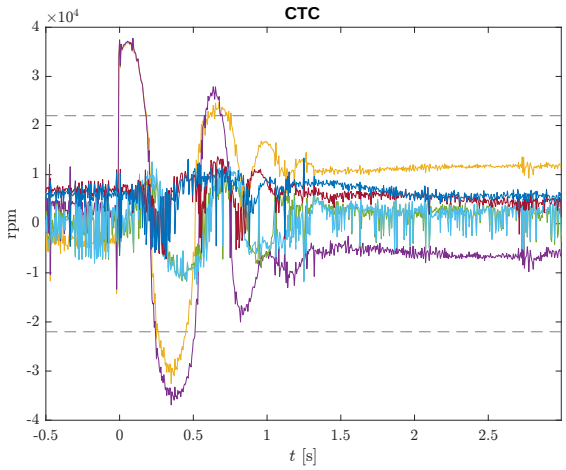
Table 2.5 – Step response comparison.



(a) Step responses.



(b) NMPC control inputs.



(c) CTC control inputs.

Figure 2.26 – Responses for a position step and control inputs (dashed lines are actuator saturation).

Trajectory Tracking

The AWG is following a square-shaped trajectory in Figure 2.27 using a trapezoidal velocity profile with both NMPC and CTC. For the selected trajectory, the root-mean-square error (RMSE) with both controllers are similar (see Table 2.6). However, the trajectory with the CTC is jerkier (see associated video material https://youtu.be/6a4gE4A6bLU) with the trapezoidal velocity profile used to follow the edges of the square. The feedforward term of the CTC is the desired trajectory acceleration and thus is discontinuous. In contrast, the NMPC penalizes the rate of variation of the propeller velocity, thus ensuring smoother response.

The energy consumption of the AWG during this trajectory with both controllers is shown in Figure 2.28. Here again, the NMPC reduces the energy consumption by 29 %.

Axis	$\mathbf{x_f}$	$\mathbf{y_f}$	$\mathbf{z_f}$
CTC [mm]	3.8	3.8	1.2
NMPC [mm]	2.4	3.2	2.7

Table 2.6 – RMSE during trajectory tracking with NMPC and CTC.

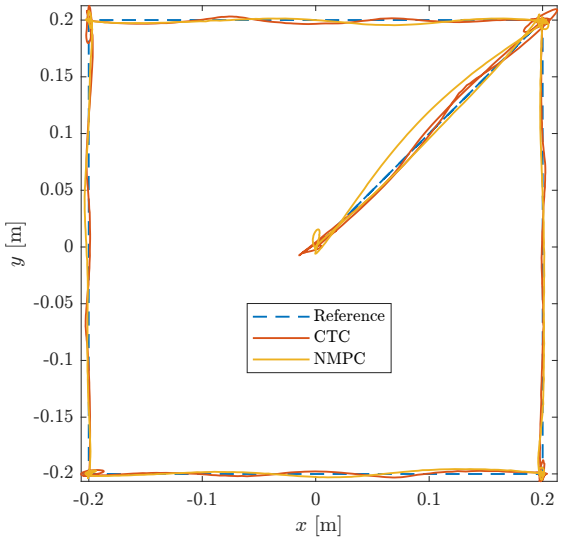
2.6 Offset-Free NMPC

2.6.1 Introduction

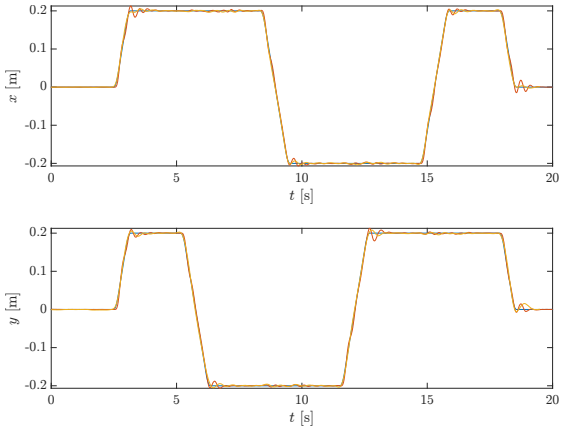
This section presents a sufficient condition for offset-free NMPC.

Various proofs have been proposed in the literature for constant disturbance models. The linear case has been studied by Maeder et al., and it has been asserted that the delta-input formulation is an alternative to the constant disturbance model to achieve offset-free response (Maeder et al., 2009). Morari and Maeder extended these results for nonlinear plants and NMPC (Morari & Maeder, 2012). However, this approach requires a discrete-time model of the nonlinear plant, which can not be written explicitly in most cases.

In the following, the proof from Morari and Maeder is adapted for continuous plants augmented with the delta-input formulation rather than a constant disturbance model. It is then applied to the AWG control, saving 6 states (resulting in a total of 18 states) since there is no need to add constant disturbance states when using the delta-input formulation. The validity of the continuous approach is tested in simulation, since the discretization is not taken into account in the proof.



(a) 2D trajectory.



(b) Trajectory tracking.

Figure 2.27 – 2D trajectory tracking by the AWG with NMPC and CTC.

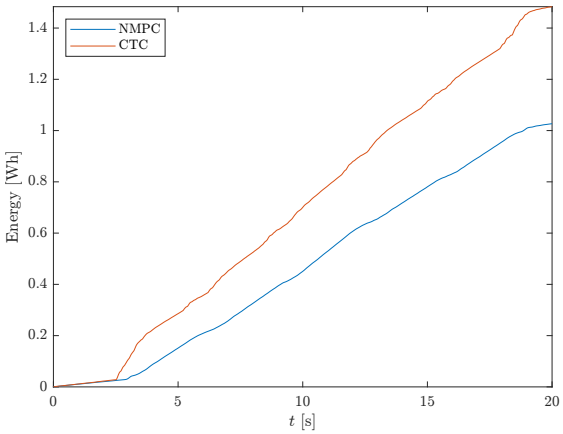


Figure 2.28 – Energy consumption of the AWG with NMPC and CTC.

2.6.2 Continuous Model and Delta-Input Formulation

The nonlinear state-space representation of the real system to be controlled is not exactly known, but it can be written generically as:

$$\begin{aligned}\dot{\mathbf{x}}_\mathbf{p} &= \mathbf{f}_\mathbf{p}(\mathbf{x}_\mathbf{p}, \mathbf{d}_\mathbf{p}, \mathbf{u}) \\ \mathbf{y}_\mathbf{p} &= \mathbf{g}_\mathbf{p}(\mathbf{x}_\mathbf{p}, \mathbf{d}_\mathbf{p})\end{aligned} \tag{2.52}$$

with $\mathbf{d}_\mathbf{p}$ unknown disturbances, $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ the input of the system and $\mathbf{y}_\mathbf{p}(t) \in \mathbb{R}^{n_y}$ the measured outputs.

Let

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x})\end{aligned} \tag{2.53}$$

be the nominal model and

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \dot{\mathbf{u}} &= \mathbf{u}_\Delta \\ \mathbf{y} &= \mathbf{g}(\mathbf{x})\end{aligned} \tag{2.54}$$

the augmented model with delta-input formulation, with $\mathbf{u}_\Delta$ the input of the augmented system.

Let us introduce the following observer of the augmented system:

$$\begin{pmatrix}\dot{\hat{\mathbf{x}}} \\ \dot{\hat{\mathbf{u}}}\end{pmatrix} = \begin{pmatrix}\mathbf{f}(\hat{\mathbf{x}}, \hat{\mathbf{u}}) + \mathbf{l}_\mathbf{x}(\mathbf{y}_\mathbf{p} - \mathbf{g}(\hat{\mathbf{x}})) \\ \mathbf{u}_\Delta + \mathbf{l}_\mathbf{u}(\mathbf{y}_\mathbf{p} - \mathbf{g}(\hat{\mathbf{x}}))\end{pmatrix} \tag{2.55}$$

with $\mathbf{l}_\mathbf{x}$ and $\mathbf{l}_\mathbf{u}$, function to be designed that ensure stability the observer. The steady-state target $(\overline{\mathbf{x}}, \overline{\mathbf{u}})$ to be tracked by the NMPC controller is a static equilibrium of the nominal plant model for a desired reference $\mathbf{r}$:

$$\begin{aligned}\mathbf{0} &= \mathbf{f}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) \\ \mathbf{r} &= \mathbf{g}(\overline{\mathbf{x}})\end{aligned}$$

The NMPC problem is then formulated as follows:

$$\begin{aligned} &\min_{\mathbf{u},\mathbf{x}} \left[M(\mathbf{x}(T) - \overline{\mathbf{x}}) + \int_0^T l(\mathbf{x}(t) - \overline{\mathbf{x}}, \mathbf{u}_\Delta(t))\, dt \right] \\ &\text{subject to } \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ &\qquad \mathbf{x}(0) = \hat{\mathbf{x}}(0) \\ &\qquad \mathbf{u}(0) = \hat{\mathbf{u}}(0) \\ &\qquad \mathbf{h}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \end{aligned} \tag{2.56}$$

with l the running cost (often a quadratic form) and M the terminal cost that could be chosen to ensure the stability of the controller. Note that, in this work, the stability of the control is assessed experimentally. Grimm et al. show that a long enough finite horizon size is sufficient to ensure the closed-loop stability using the NMPC (Grimm et al., 2005). The NMPC problem does not depend on the effective input $\mathbf{u}$ of the nominal plant, but on its estimate $\hat{\mathbf{u}}(0)$ in which all model errors are lumped.

The closed-loop system is illustrated with a block diagram in Figure 2.29.

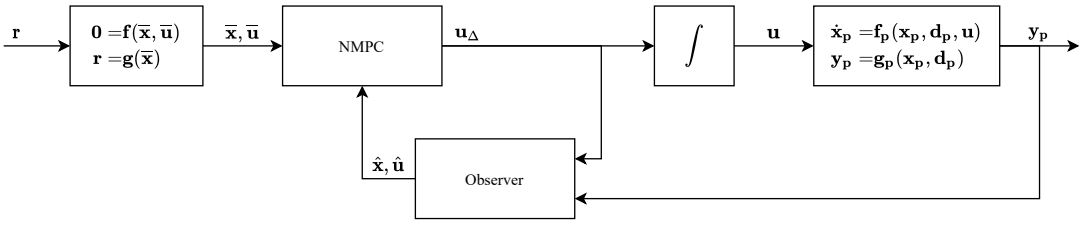


Figure 2.29 – NMPC block diagram for offset-free control.

2.6.3 Sufficient Condition for Offset-Free Control

Let us introduce the theorem providing a sufficient condition for offset-free control in case of a delta-input formulation of the problem.

Assuming the stability of the closed-loop system, the following assumption is made:

Assumption 1 *For an asymptotically constant feasible reference* $\mathbf{r}(t \to \infty) = \mathbf{r}_\infty$ *and disturbance* $\mathbf{d_p}(t \to \infty) = \mathbf{d_{p,\infty}}$*, all the states converge. Therefore, the measured outputs* $\mathbf{y_p}$ *and the input* $\mathbf{u}$ *reach steady-state values, respectively* $\mathbf{y_{p,\infty}}$ *and* $\mathbf{u}_\infty$*.*

In the linear case, the observability and the controllability of the augmented system

are required for offset-free control. In the nonlinear case, both conditions are replaced as follows:

Assumption 2 *For all* $\mathbf{y} = \mathbf{r}$*, there exists a unique* $(\mathbf{x}_\infty, \mathbf{u}_\infty)$ *such that*

$$\begin{aligned} \mathbf{0} &= \mathbf{f}(\mathbf{x}_\infty, \mathbf{u}_\infty) \\ \mathbf{y} &= \mathbf{g}(\mathbf{x}_\infty) \end{aligned} \tag{2.57}$$

Using the implicit function theorem, it is possible to define a condition on the linearized model to test Assumption 2 locally in the neighborhood of the steady state. Let us introduce the following matrices:

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \mathbf{A}$$

$$\frac{\partial}{\partial \mathbf{u}} \mathbf{f}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \mathbf{B}$$

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}_\infty) = \mathbf{C}$$

Proposition 1 *Assumption 2 holds locally if* $n_y = n_u$ *and*

$$rank \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{pmatrix} = n_x + n_u \tag{2.58}$$

Proof 1 *Let us introduce the function*

$$\mathscr{I}_{\mathscr{O}}(\mathbf{y}, \mathbf{z}) = \begin{pmatrix} \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{g}(\mathbf{x}) - \mathbf{y} \end{pmatrix} \tag{2.59}$$

where $\mathbf{z} = (\mathbf{x}, \mathbf{u})$ *and such that* $\mathscr{I}_{\mathscr{O}}(\mathbf{y}_\infty, \mathbf{z}_\infty) = \mathbf{0}$ (2.57)*. Its Jacobian is defined by:*

$$\frac{\partial}{\partial \mathbf{z}} \mathscr{I}_{\mathscr{O}}(\mathbf{y}_\infty, \mathbf{z}_\infty) = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{pmatrix} \tag{2.60}$$

According to the implicit function theorem, if the Jacobian $\frac{\partial}{\partial \mathbf{z}} \mathscr{I}_{\mathscr{O}}(\mathbf{y}_\infty, \mathbf{z}_\infty)$ *is invertible, then there exists a unique solution* $\mathbf{z}_\infty = (\mathbf{x}_\infty, \mathbf{u}_\infty)$ *to* (2.57)*. Assuming* $n_y = n_u$*, the Jacobian is a square matrix of dimension* $n_x + n_u$*. As the Jacobian is invertible if and only if the matrix rank is* $n_x + n_u$*, it ends the proof of Proposition 1.*

In the case of a linear and stable observer, $n_u \geq n_y$ guarantees that, at steady state, the observer is nominally error-free (the function $\mathbf{l_u}$ (2.55) is replaced by a full rank $n_u \times n_y$ matrix). This may not be the case with a nonlinear observer, hence the following assumption:

Assumption 3 *At steady state, the observer* (2.55) *is nominally error-free:*

$$\mathbf{l_u}(\mathbf{y_{p,\infty}} - \mathbf{g}(\hat{\mathbf{x}}_\infty)) = \mathbf{0} \implies \mathbf{y_{p,\infty}} - \mathbf{g}(\hat{\mathbf{x}}_\infty) = \mathbf{0} \tag{2.61}$$

Let the control law be defined by the function $\mathbf{c_0}$: $\mathbf{u_{\Delta 0}}(t) = \mathbf{c_0}(\hat{\mathbf{x}}(t), \hat{\mathbf{u}}(t), \mathbf{r}(t))$ where $\mathbf{u_{\Delta 0}}$ is the solution of the OCP (2.56).

Assumption 4 *The NMPC is designed to be nominally error-free at steady state, i.e. for all feasible reference* $\mathbf{r}_\infty$*,*

$$\mathbf{f}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \mathbf{0} \quad \textit{and} \quad \mathbf{c_0}(\mathbf{x}_\infty, \mathbf{u}_\infty, \mathbf{r}_\infty) = \mathbf{0} \implies \mathbf{g}(\mathbf{x}_\infty) = \mathbf{r}_\infty \tag{2.62}$$

holds for all $\mathbf{x}_\infty$ *and for all* $\mathbf{u}_\infty$*.*

Finally, the theorem giving a sufficient condition for offset-free control is the following:

Theorem 1 *If Assumptions 1-4 hold and* $n_u = n_y$*, then* $\mathbf{y_p}(t \to \infty) = \mathbf{r}(t \to \infty)$*.*

Proof 2 *Assumption 1 enforces asymptotic closed-loop stability. Therefore, only* $\mathbf{y_{p,\infty}} = \mathbf{r}_\infty$ *needs to be proven.*

Combining Assumptions 2 and 3, at steady state, the observer satisfies

$$\mathbf{0} = \mathbf{f}(\hat{\mathbf{x}}_\infty, \hat{\mathbf{u}}_\infty) \tag{2.63}$$

and

$$\mathbf{y_{p,\infty}} = \mathbf{g}(\hat{\mathbf{x}}_\infty) \tag{2.64}$$

From (2.63) *and Assumption 4, it follows that*

$$\mathbf{g}(\hat{\mathbf{x}}_\infty) = \mathbf{r}_\infty \tag{2.65}$$

since at steady state $\mathbf{c_0}(\hat{\mathbf{x}}_\infty, \hat{\mathbf{u}}_\infty, \mathbf{r}_\infty) = \mathbf{0}$.

Finally, combining these two last equations:

$$\mathbf{y_{p,\infty}} = \mathbf{r}_\infty \tag{2.66}$$

2.6.4 Application to the AWG

The delta-input formulation of the NMPC AWG dynamic model allows for a smooth control input, thus reducing chattering. Moreover, the previous theorem shows that there is no need to augment the system with constant disturbances if the delta-input formulation is used and some requirements are met.

The nonlinear dynamics of the AWG are linearized in the neighborhood of the steady state $(\mathbf{X}_\infty, \mathbf{w_{2,\infty}})$:

$$\mathbf{M}(\mathbf{X}_\infty)\delta\ddot{\mathbf{X}} + \mathbf{K_X}(\mathbf{X}_\infty)\delta\mathbf{X} = \tilde{\mathbf{W}}(\mathbf{X}_\infty)\delta\mathbf{w_{2,\infty}} \tag{2.67}$$

with $\mathbf{w_{2,\infty}}$ the signed squared propeller rotational velocities at steady state.

The linearization is detailed in Appendix A. Note that this model ignores actuator dynamics because of the results from Section 2.5.4.

For Theorem 1 to hold, only 6 state variables can be measured (since $n_u = 6$). Therefore, only the pose measurement is used: $\mathbf{g}(\mathbf{x}) = \mathbf{X}$.

Let us define the following matrices to test Assumption 2:

$$\mathbf{A}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \begin{pmatrix} \mathbf{0} & -\mathbf{M}^{-1}\mathbf{K_X} \\ \mathbf{I} & \mathbf{0} \end{pmatrix} \tag{2.68}$$

$$\mathbf{B}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \begin{pmatrix} \mathbf{M}^{-1}\tilde{\mathbf{W}} \\ \mathbf{0} \end{pmatrix} \tag{2.69}$$

$$\mathbf{C}(\mathbf{x}_\infty, \mathbf{u}_\infty) = \begin{pmatrix} \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{2.70}$$

To simplify the notations, the dependence on the fixed point of the matrices is dropped.

Let us introduce the Jacobian

$$\mathbf{J}_\infty = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{pmatrix} \tag{2.71}$$

From Proposition 1, Assumption 2 is valid if $\mathbf{J}_\infty$ is of full rank. It can be easily verified from the previous expressions of the matrices that $\det(\mathbf{J}_\infty) = -\det(\mathbf{M}^{-1}\tilde{\mathbf{W}}) \neq 0$. Therefore, Assumption 2 holds.

The observer is designed using an EKF, therefore Assumption 3 also holds since $n_u = n_y$.

Let the cost function be

$$J = \int_0^T \left(\left\| \mathbf{X}(t) - \overline{\mathbf{X}}(t) \right\|_{\mathbf{Q}}^2 + \|\mathbf{u}_\Delta(t)\|_{\mathbf{R}}^2 \right) dt \tag{2.72}$$

Since $\dot{\mathbf{X}} = \mathbf{0}$ at steady state, the target is obtained directly from the reference: $\overline{\mathbf{X}} = \mathbf{r}$. Hence, the cost function becomes:

$$J = \int_0^T \left(\|\mathbf{X}(t) - \mathbf{r}(t)\|_{\mathbf{Q}}^2 + \|\mathbf{u}_\Delta(t)\|_{\mathbf{R}}^2 \right) dt \tag{2.73}$$

In conclusion, offset-free control can be reached under the Assumptions 1 and 4.

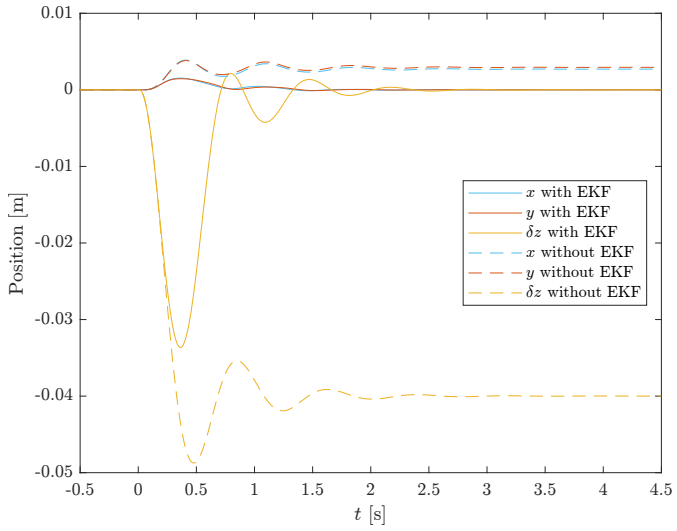
2.6.5 Validation in Simulation

The proposed approach is validated in simulation. A disturbance emulating a 0.25 kg load placed at a 25 cm offset along both $\mathbf{x_b}$ and $\mathbf{y_b}$ directions is added to the simulation model. This disturbance behaves as an external wrench $(\mathbf{F_d}^T \ \mathbf{N_d}^T)^T$ on the CoM G.

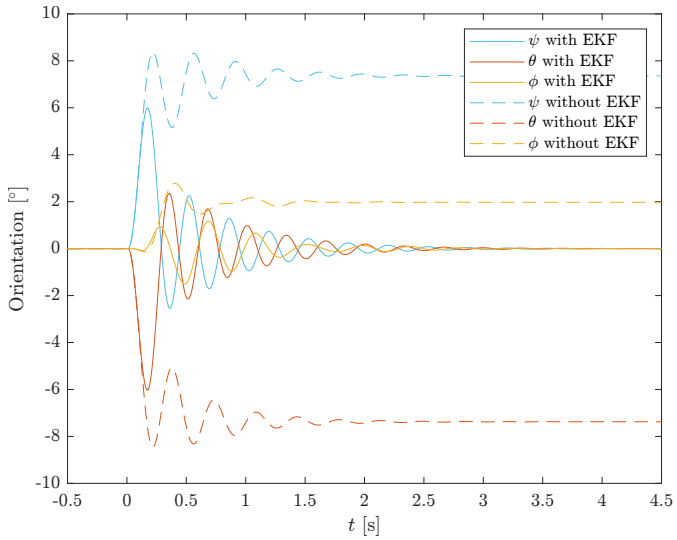
Figure 2.30 compares the closed-loop response when the disturbance is added to the nominal model at $t = 0$ with and without the EKF observer. The observer allows for eliminating the steady-state offset that appears on all DoFs. The mismatch $\hat{\mathbf{w}}_\mathbf{2} - \mathbf{w_2}$ between the observed and simulated signed squared propeller rotational velocities is shown in Figure 2.31. At steady state, the disturbance is lumped in the estimated

signed squared propeller rotational velocities $\hat{\mathbf{w}}_2$:

$$\tilde{\mathbf{W}}(\hat{\mathbf{w}}_2 - \mathbf{w}_2) = \begin{pmatrix} \mathbf{F_d} \\ \mathbf{N_d} \end{pmatrix} \tag{2.74}$$



(a) Position response.



(b) Orientation response.

Figure 2.30 – Response to a disturbance step with and without EKF.

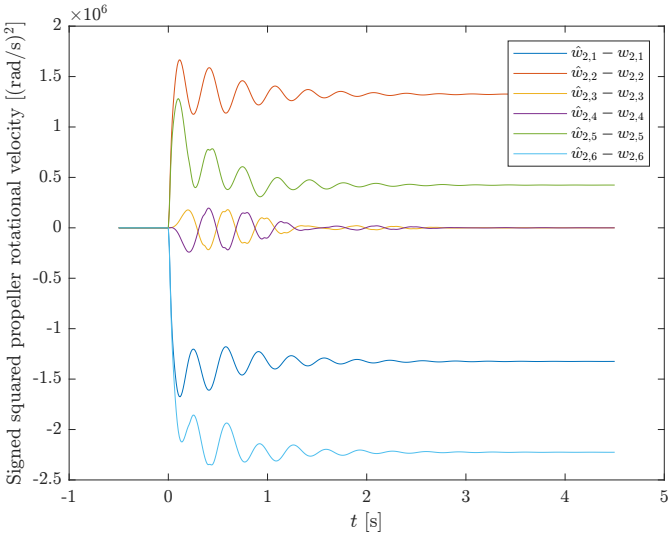


Figure 2.31 – Error between observed and simulated signed squared propeller rotational velocities: $\hat{\mathbf{w}}_2 - \mathbf{w}_2$.

2.7 Conclusion

This chapter introduces the concept of AWG. Its dynamic model is derived and validated experimentally using a computed torque control law. The stability of the control law is proven using singular perturbation theory. The performance of the AWG is assessed experimentally and highlights the added value of the spring with respect to the energy efficiency.

A nonlinear model predictive controller is then proposed to overcome some drawbacks of the computed torque controller. The improvements on the closed-loop dynamics are validated experimentally. A sufficient condition is proposed for offset-free tracking based on Morari and Maeder's approach, and is tested in simulation.

3 Aerial Manipulator Suspended from a CDPR

3.1 Introduction

The suspension of the aerial manipulator reduces drastically the energy consumption thanks to gravity compensation as shown in Chapter 2. Nevertheless, the workspace of the aerial manipulator is restricted around its equilibrium point due to the spring restoring force. Furthermore, this restoring force yields additional energy consumption for stationary trajectories away from the equilibrium point. However, if the equilibrium point could move along with the aerial manipulator, even at a slower pace, energy consumption could be significantly reduced and the workspace of the AMES could match the workspace of its carrier. Indeed, the motion of the equilibrium point would not hinder the dynamics of the AMES thanks to the decoupling action of the spring. This concept is somehow similar to the macro/micro manipulator system (Sharon et al., 1993), where the big (macro) robotic arm provides a large workspace and high speed to the high-precision but smaller (micro) robot at its tip.

In this chapter, the AWG is carried by a four-cable suspended CDPR. With its carrier, the AMES is overactuated (10 actuators for 6 DoFs). There exist different ways in the literature to handle redundancy, known as redundancy resolution or control allocation. For systems with heterogeneous actuators, specific control allocation algorithms have been developed. Frequency-apportioned control allocation uses a weighted pseudo-inverse of the Jacobian to consider both the steady-state performance and frequency response of the actuators (Lallman et al., 2001). Model predictive control allocation takes into account actuator constraints, namely their dynamics and saturation, and solves an optimization problem in real time to distribute control inputs (Yu Luo et al., 2004). More recently, we evaluated this last strategy on the PiSaRo4 robot, a planar suspended CDPR with fast on-board unidirectional force generators (Khayour et al.,

2020).

During the thesis, two approaches were considered. In the first approach, a partitioned control strategy using two independent controllers to track reference trajectories of CDPR and AWG end effectors was implemented. Therefore the redundancy is handled during the planification and the generation of the trajectories. This simple control law allows for validating the added value of the carrier with respect to energy efficiency. In the second approach, a NMPC implements a unified control strategy that solves the redundancy by minimizing the energy consumption of the whole AMES.

3.2 Dynamic Modeling

3.2.1 Parametrization

The geometric parameters of the system are defined in Figure 3.1 and Figure 3.2.

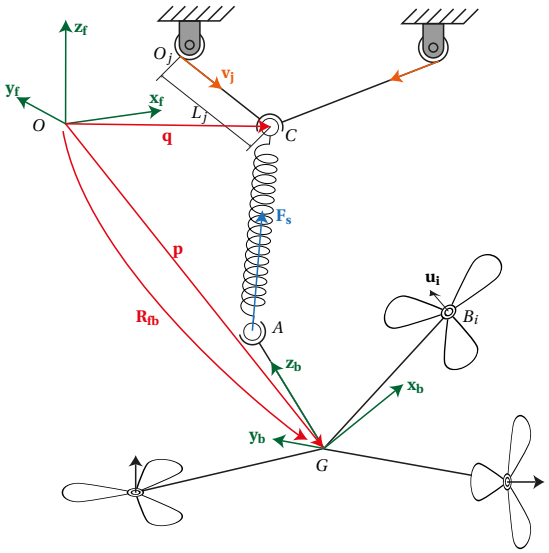


Figure 3.1 – AMES parameters.

Let $\mathscr{R}_f = (O, \mathbf{x_f}, \mathbf{y_f}, \mathbf{z_f})$ be an inertial frame with O its origin and $\mathscr{R}_b = (G, \mathbf{x_b}, \mathbf{y_b}, \mathbf{z_b})$ a moving frame positioned at the CoM G of the AWG with its $\mathbf{z_b}$ axis pointing toward the on-board anchoring point of the spring A. The rotation matrix $\mathbf{R_{fb}} \in SO(3)$ describes the orientation of $\mathscr{R}_b$ with respect to $\mathscr{R}_f$. The AWG has six propulsion units. The position of the center of the i-th propulsion unit is B_i, and $\mathbf{u_i}$ is the unit thrust direction vector. The CDPR has four cables which intersection is the second anchoring point of the spring C. The position vectors of the CoM G and of the cable intersection C are

respectively **p** and **q**. The cable output position O_j on the winch pulley is considered constant, with $\mathbf{v_j}$ the unit cable direction vector such that $\mathbf{O_jC} = L_j\mathbf{v_j}$ where $L_j \geq 0$ is the length of the j-th cable. The force applied by the spring on the AWG is noted $\mathbf{F_s}$. The spring restoring force is the only coupling between the AWG and the CDPR dynamics and it can be expressed as:

$$\mathbf{F_s} = k\left(\|\mathbf{AC}\| - l_0\right)\frac{\mathbf{AC}}{\|\mathbf{AC}\|} \tag{3.1}$$

with l_0 the free length and k the stiffness of the spring.

As shown in Figure 3.2, each cable is equipped with a balancing spring to maintain a minimal tension and also to create a passive equilibrium point, reducing the energy consumption of the CDPR. Each motor drives two winches with helical grooves, one on the spring side to wind the cable attached to the balancing spring and the other one on the end-effector side to wind the end-effector cable. The angular position of the j-th winch motor is α_j. The motor applies a torque τ_j on the winch. The radii of the winches are r and R respectively on the balancing-spring side and the end-effector side.

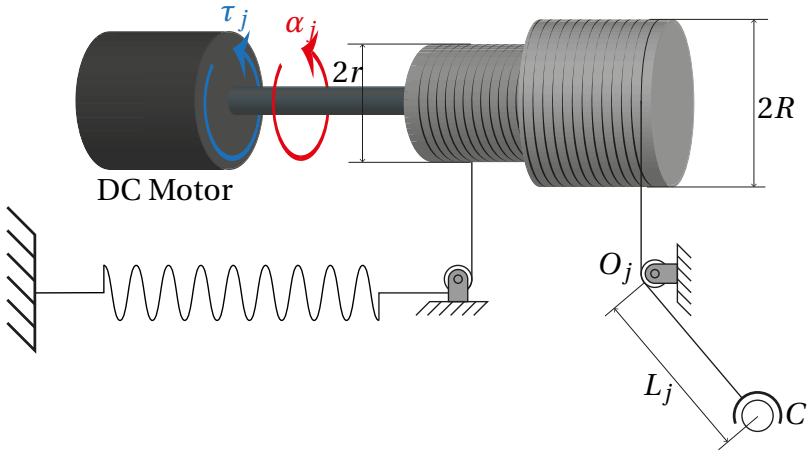


Figure 3.2 – CDPR actuated winch parameters.

3.2.2 CDPR Dynamics

The CDPR platform (i.e. at the intersection of the four cables) is assumed to be a massless point particle. Indeed, only translational motion is needed for the CDPR, thus the platform may be limited to a small rigid body used to attach at one point the four cable ends. The cables, made of braided stainless steel, are considered straight lines and their elasticity is neglected compared to the elasticity of the balancing

springs.

With these assumptions, it is possible to derive the dynamic model of the CDPR in its canonical formulation (Chellal et al., 2017; Khosravi & Taghirad, 2011):

$$\mathbf{M_c}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C_c}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G_c}(\mathbf{q},\mathbf{X}) = \mathbf{J}^T\boldsymbol{\tau} \tag{3.2}$$

with $\mathbf{X}$ the pose of the AWG and the matrices $\mathbf{M_c}$, $\mathbf{C_c}$, $\mathbf{G_c}$, $\mathbf{J}$ defined after the derivations below.

The CDPR is composed of 4 actuated winches with moment of inertia I_c and 4 rigid massless cables. Euler's second law of motion applied to the j-th winch yields:

$$I_c\ddot{\alpha}_j = \tau_j - rF_j + RT_j \tag{3.3}$$

with $F_j = kr\alpha_i$ the restoring force of the j-th balancing spring and $T_j \geq 0$ the tension of the j-th cable. Equation (3.3) can also be written in matrix form:

$$\mathbf{I_c}\ddot{\boldsymbol{\alpha}} = \boldsymbol{\tau} - kr^2\boldsymbol{\alpha} + R\mathbf{T} \tag{3.4}$$

with $\mathbf{I_c} = I_c\mathbf{I}$, $\boldsymbol{\alpha} = (\alpha_1\ \alpha_2\ \alpha_3\ \alpha_4)^T$, $\boldsymbol{\tau} = (\tau_1\ \tau_2\ \tau_3\ \tau_4)^T$ and $\mathbf{T} = (T_1\ T_2\ T_3\ T_4)^T$.

In order to attain the canonical formulation (3.2), two steps are required. First, a wrench analysis at the CDPR end effector allows for removing the cable tensions $\mathbf{T}$ by expressing them using the spring restoring force $\mathbf{F_s} = \mathbf{F_s}(\mathbf{q},\mathbf{X})$. Then, the first- and second-order kinematic models linking the winch angles $\boldsymbol{\alpha}$ (and derivatives) to the CDPR end-effector position $\mathbf{q}$ (and derivatives) are derived.

Wrench Analysis

Let $\mathbf{F_{Tj}}$ be the force applied to the CDPR end effector by the j-th cable

$$\mathbf{F_{Tj}} = -T_j\mathbf{v_j} \tag{3.5}$$

and $\mathbf{J}(\mathbf{q}) = {}^f\mathbf{J}(\mathbf{q})$ the Jacobian matrix expressed in the inertial frame $\mathscr{R}_f$

$$\mathbf{J}(\mathbf{q}) = \left({}^f\mathbf{v_1}\ {}^f\mathbf{v_2}\ {}^f\mathbf{v_3}\ {}^f\mathbf{v_4}\right)^T \tag{3.6}$$

Since the CDPR platform is assumed to be massless, forces acting on the end effector

are balanced:

$$\sum {}^{f}\mathbf{F_{Tj}} = {}^{f}\mathbf{F_s} \tag{3.7}$$

Therefore, we can map the cable tensions $\mathbf{T}$ to the force applied to the CDPR end effector:

$${}^{f}\mathbf{F_s} = -\mathbf{J}^T\mathbf{T} \tag{3.8}$$

Kinematics

The cables being considered rigid, the cable lengths $\mathbf{L} = (L_1\ L_2\ L_3\ L_4)^T$ and the winch angles $\boldsymbol{\alpha}$ are linked by a holonomic constraint:

$$\mathbf{L} - \mathbf{L_0} = R\boldsymbol{\alpha} \tag{3.9}$$

with $\mathbf{L_0}$ the cable lengths at the equilibrium point of CDPR.

Another holonomic constraint relates the cable lengths $\mathbf{L}$ to the position of the end effector $\mathbf{q}$:

$$L_j = \left\|\mathbf{q} - \mathbf{OO_j}\right\| \tag{3.10}$$

The time derivative of equation (3.10) yields:

$$\frac{x - O_{jx}}{\left\|\mathbf{q} - \mathbf{OO_j}\right\|}\dot{x} + \frac{y - O_{jy}}{\left\|\mathbf{q} - \mathbf{OO_j}\right\|}\dot{y} + \frac{z - O_{jz}}{\left\|\mathbf{q} - \mathbf{OO_j}\right\|}\dot{z} = R\dot{\alpha}_j \tag{3.11}$$

which results in the first-order kinematic model:

$$\mathbf{J}\dot{\mathbf{q}} = R\dot{\boldsymbol{\alpha}} \tag{3.12}$$

The second-order kinematic model is obtained by taking the time derivative of the first-order kinematic model (3.12):

$$\mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} = R\ddot{\boldsymbol{\alpha}} \tag{3.13}$$

with

$$\dot{\mathbf{J}} = \sum_i \frac{\partial \mathbf{J}}{\partial q_i} \dot{q}_i \tag{3.14}$$

Canonical Formulation

Using the first- and second-order kinematic models (3.12) and (3.13), combined with equation (3.9), we can replace the joint coordinates $\boldsymbol{\alpha}$ by the operational coordinates $\mathbf{q}$ in (3.4):

$$\frac{1}{R}\mathbf{I_c}\underbrace{(\mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}})}_{R\ddot{\boldsymbol{\alpha}}} = \boldsymbol{\tau} - k\frac{r^2}{R}(\mathbf{L}(\mathbf{q}) - \mathbf{L_0}) + R\mathbf{T} \tag{3.15}$$

After left-multiplying the equation (3.15) by $\mathbf{J}^T$ and combining it with (3.8), we obtain the dynamic model of the CDPR in its canonical formulation (3.2) with:

$$\begin{cases} \mathbf{M_c}(\mathbf{q}) &= \frac{1}{R}\mathbf{J}^T\mathbf{I_c}\mathbf{J} \\ \mathbf{C_c}(\mathbf{q},\dot{\mathbf{q}}) &= \frac{1}{R}\mathbf{J}^T\mathbf{I_c}\dot{\mathbf{J}} \\ \mathbf{G_c}(\mathbf{q},\mathbf{X}) &= k_c\frac{r^2}{R}\mathbf{J}^T(\mathbf{L}-\mathbf{L_0}) + R^f\mathbf{F_s} \end{cases} \tag{3.16}$$

3.2.3 AWG Dynamics

Let us recall the AWG dynamics (2.8):

$$\underbrace{\begin{pmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{S}^T\mathbf{J}_b\mathbf{S} \end{pmatrix}}_{\mathbf{M_a}(\mathbf{X})}\begin{pmatrix} \ddot{\mathbf{p}} \\ \ddot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{S}^T(\mathbf{J}_b\dot{\mathbf{S}} + [\mathbf{S}\dot{\boldsymbol{\eta}}]_\times\mathbf{J}_b\mathbf{S}) \end{pmatrix}}_{\mathbf{C_a}(\mathbf{X},\dot{\mathbf{X}})}\begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} -m^f\mathbf{g} - {}^f\mathbf{F_s} \\ -\mathbf{S}^{Tb}\mathbf{N_s} \end{pmatrix}}_{\mathbf{G_a}(\mathbf{X},\mathbf{q})} = \underbrace{\begin{pmatrix} \mathbf{R_{fb}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{S}^T \end{pmatrix}\mathbf{W}_b\mathbf{w_2}}_{\tilde{\mathbf{W}}(\mathbf{X})} \tag{3.17}$$

The only change is the spring restoring force that here also depends on $\mathbf{q}$: $\mathbf{F_s} = \mathbf{F_s}(\mathbf{X},\mathbf{q})$. Therefore, the AWG dynamics has a dependency on $\mathbf{q}$:

$$\mathbf{M_a}(\mathbf{X})\ddot{\mathbf{X}} + \mathbf{C_a}(\mathbf{X},\dot{\mathbf{X}})\dot{\mathbf{X}} + \mathbf{G_a}(\mathbf{X},\mathbf{q}) = \tilde{\mathbf{W}}(\mathbf{X})\mathbf{w_2} \tag{3.18}$$

Note that a subscript **a** is added to the matrices of the AWG in the canonical formulation to avoid confusion with the CDPR dynamic model.

The closed-loop propeller velocity dynamics (2.14) are also unchanged:

$$\dot{\mathbf{w}} = \frac{1}{t_m}(\mathbf{w_{ref}} - \mathbf{w}) \tag{3.19}$$

3.3 Experimental Setup

The experimental setup is illustrated in Figure 3.3. The CDPR carrier is a commercial robot provided by Haption called Inca^{TM}, [https://www.haption.com/en/products-en/inca-en.html]. In this experimental setup, the anchoring point of the spring is driven by four cables. The CDPR is actuated by four Maxon RE40 (48V) DC motors with encoders. A digital PI (proportional-integral) controller running on an FPGA regulates the current of each motor at 25 kHz. A higher-level controller runs on the on-board CPU of the AWG and handles the digital regulation of the angular velocity of cable winches at 400 Hz, the position control of the anchoring point (translations) and the pose control of the AWG (translations and rotations) at 100 Hz. These control loops are using respectively feedbacks from the winch motor encoders, the 3D measurements of a Vicon Bonita motion-capture system (infrared markers on the anchoring point and on the AWG) and the 3D inertial measurements on the AWG with an on-board gyroscope sensor (MPU-9150).

The electronic architecture of the AMES for the experiments using the partitioned controller is illustrated in Figure 3.4. An industrial PC/104 rack drives the CDPR motors. It is equipped with a Pentium M 1.6 GHz CPU, power supplies and FPGAs. A Linux operating system with a Xenomai real-time core extension runs on the CPU. A TCP/IP socket server sends the current reference periodically to the FPGAs, at 500 Hz. The current reference is updated asynchronously at a slower refresh rate (400 Hz) by the on-board CPU of the AWG. At each reference update request, the TCP/IP client receives back the measured currents, the position and velocity of the winches from the incremental encoder and the I^2t estimations to prevent overheating of the motors. The initial position of the winches can be obtained by kinetostatic analysis, i.e. by solving equation (3.2) with $\mathbf{q} = \dot{\mathbf{q}} = \mathbf{0}$. If the socket server does not receive a request for 500 ms, a software watchdog disables the motors for security purpose.

3.4 Partitioned Controller

A partitioned control scheme (see Figure 3.6) is proposed to impose a desired trajectory $\mathbf{X_{ref}} = (\mathbf{p_{ref}}^T \, \boldsymbol{\eta}_{\mathbf{ref}}{}^T)^T$ on the end effector, or equivalently on the AWG CoM. Accordingly,

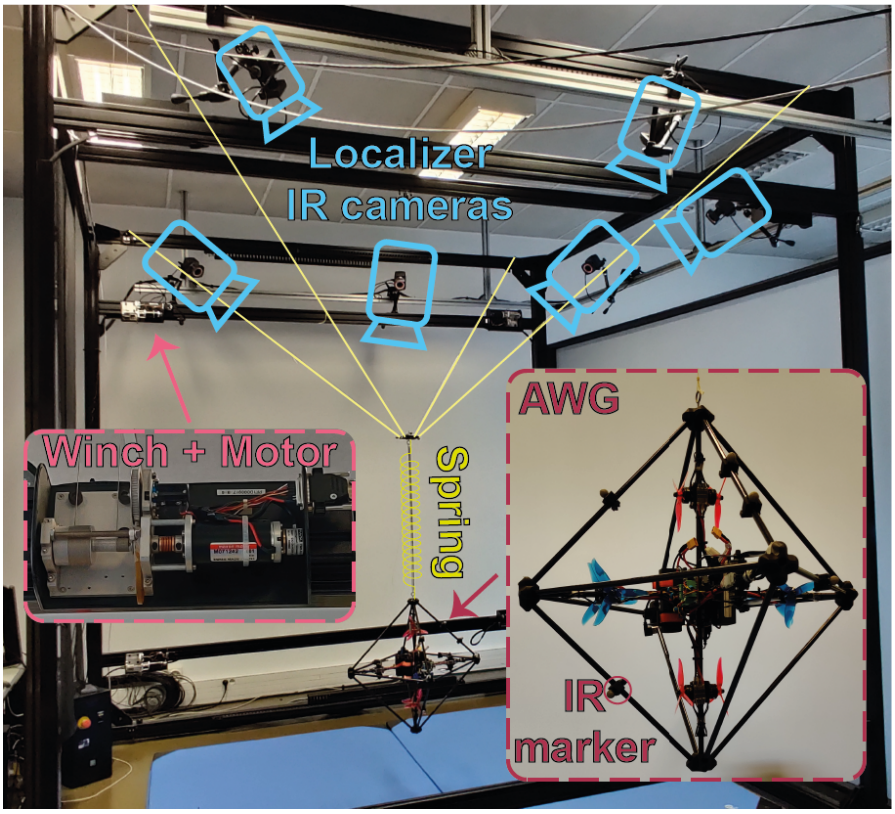


Figure 3.3 – Experimental setup.

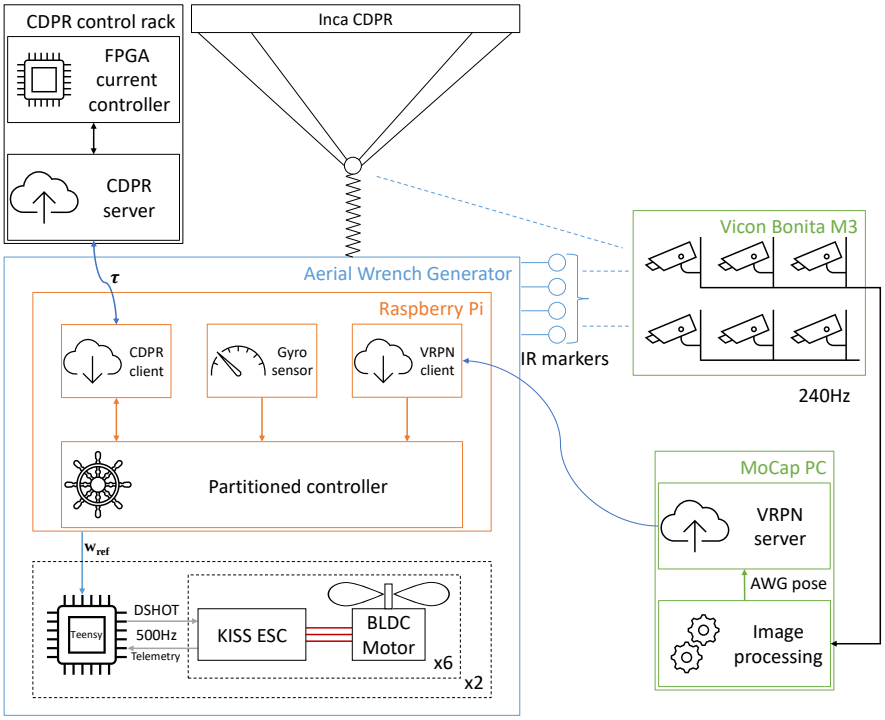


Figure 3.4 – Electronic architecture of the AMES.

the CDPR and AWG end effectors have their own independent controllers and the redundancy is handled by the planification and the generation of the trajectories. This choice is relevant thanks to the spring decoupling the dynamics of both subsystems: the AWG and the CDPR dynamics are only coupled by the relative positions of the spring ends A and C. Furthermore, independently from this decoupling, the CDPR is supposed to be much slower than the AWG (15 times slower settling time with our experimental setup). So, the loop controlling the pose of the AWG is tuned to track high-acceleration displacements, while the CDPR position loop handles large-amplitude and slow-acceleration motion tracking.

3.4.1 Redundancy Resolution

The energy consumption of the AWG increases significantly with the torque $\mathbf{N_s}$ generated by the spring restoring force on the AWG as seen in Section 2.4.2. A simple but suboptimal solution to deal with the actuator redundancy while reducing the energy consumption is to choose the anchoring point reference position $\mathbf{q_{ref}}$ such that the AWG CoM G and the spring ends A and C stay aligned with constant relative distances:

$$\mathbf{q_{ref}}(\mathbf{X_{ref}}) = \mathbf{p_{ref}} + \mathbf{R_{fb}}(\boldsymbol{\eta}_{\mathbf{ref}}) \begin{pmatrix} 0 \\ 0 \\ \delta z_{eq} \end{pmatrix} \tag{3.20}$$

with $\mathbf{R_{fb}}(\boldsymbol{\eta}_{\mathbf{ref}})$ the rotation matrix $\mathbf{R_{fb}}$ at orientation $\boldsymbol{\eta}_{\mathbf{ref}}$ and δz_{eq} the vertical distance between G and C at the equilibrium of the AWG. This strategy is illustrated in Figure 3.5.

Notably, with this strategy, if the reference trajectory is at constant orientation, the anchoring point just follows the movement of the AWG with an offset. Furthermore, if the reference orientation corresponds to vector $\mathbf{GA}$ being vertical, the control scheme will tend to maintain the spring anchoring point C on the vertical line going through G and maintain a constant distance $\|\mathbf{CG}\|$ (see Figure 3.5).

Note that finding the optimal $\mathbf{q_{ref}}$ that minimizes the AWG energy consumption could be done by minimizing $\|\mathbf{w_2}\|_{\frac{3}{2}}$, with $\|\cdot\|_{\frac{3}{2}}$ the $\frac{3}{2}$-norm, since the power consumption of a propulsion unit is affine with respect to the cubic rotational velocity (see Section 2.4.2). At steady state, since $\tilde{\mathbf{W}}(\mathbf{X})^{-1}\mathbf{G_a}(\mathbf{X},\mathbf{q}) = \mathbf{w_2}$, this can be written as a nonlinear

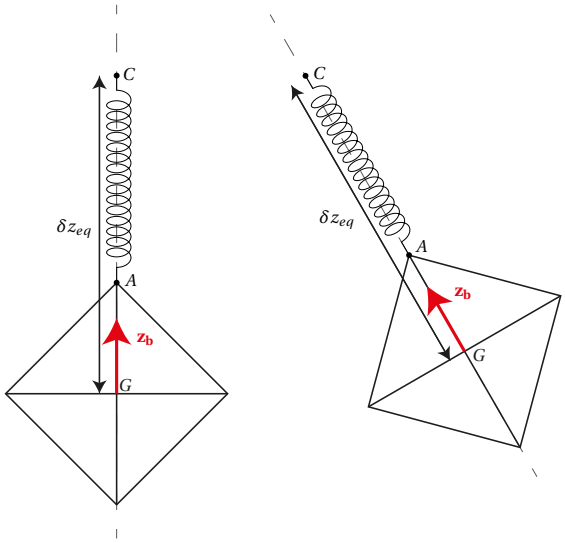


Figure 3.5 – Redundancy resolution strategy.

optimization problem:

$$\mathbf{q_{ref}} = \min_{\mathbf{q}} \left\| \tilde{\mathbf{W}}(\mathbf{X})^{-1} \mathbf{G_a}(\mathbf{X}, \mathbf{q}) \right\|_{\frac{3}{2}} \tag{3.21}$$

3.4.2 AWG Control

The AWG follows a desired trajectory $\mathbf{X_{ref}}$ using the CTC, from Section 2.4.1, running at 100 Hz.

Since the carrier is much slower than the AWG, the previous proof of stability can be considered still valid. The stability could also be proven again using singular perturbation theory applied on a three-time scale system: high-bandwidth propellers, AWG dynamics and slow carrier dynamics (Esteban et al., 2013).

3.4.3 CDPR Control

To reject significant nonlinearities due to dry friction on pulleys, a cascade control scheme is implemented to control the CDPR pose (see Figure 3.6). The controller runs at 400 Hz. The inner loop consists in a PI controller regulating the rotational velocity $\dot{\boldsymbol{\alpha}}$ of the winches.

The outer loop that tracks the translational position reference $\mathbf{q_{ref}}$ of the anchoring

point C implements a proportional controller with a feedforward term (P+ff):

$$\dot{\boldsymbol{\alpha}}_{\mathbf{ref}} = \frac{1}{R}\mathbf{J}\left(k_{ff}\dot{\mathbf{q}}_{\mathbf{ref}} + k_q(\mathbf{q}_{\mathbf{ref}} - \mathbf{q})\right) \tag{3.22}$$

with k_{ff} and k_q positive gains and $\dot{\boldsymbol{\alpha}}_{\mathbf{ref}}$ the reference for the winch velocities of the inner loop.

The exponential convergence toward $\mathbf{q}_{\mathbf{ref}}$ of this control scheme using joint velocity inner loops has been first proven in (Kelly & Moreno, 2005) for serial robots. Since then, this two-loop cascade control scheme has been successfully applied to redundant robots (Soto & Campa, 2013), parallel robots (Campa et al., 2016) and CDPRs (Chellal et al., 2015).

To guarantee positive tension of the cables, a tension distribution algorithm can be included if required (Chellal et al., 2015). However, with the present suspended configuration and the balancing springs that guarantee a minimal tension, all the cable tensions remained positive during the experiments.

3.4.4 Experimental Results

The performance of the AMES is assessed experimentally. This section presents results on the dynamic and energy performances of the robot. The reference paths used for the tests are fifth-order polynomial trajectories. The experiments are available on the video [https://youtu.be/NxJjCoystsA].

Dynamic Maneuver Handling

A fast trajectory reference is used to highlight the heterogeneous dynamics of the AMES as shown in Figure 3.7, where $\delta\mathbf{p}$ and $\delta\mathbf{q}$ are the relative positions of the AWG and the carrier anchoring point with respect to their initial position. As expected, the AWG follows with millimetric precision the reference trajectory while the CDPR anchoring point follows the AWG lagging behind during transients. The CDPR platform behaves in closed loop as a first-order system with 0.5 s time constant after experimental controller tuning. It can also be noted that the steady-state error tends toward zero for both responses.

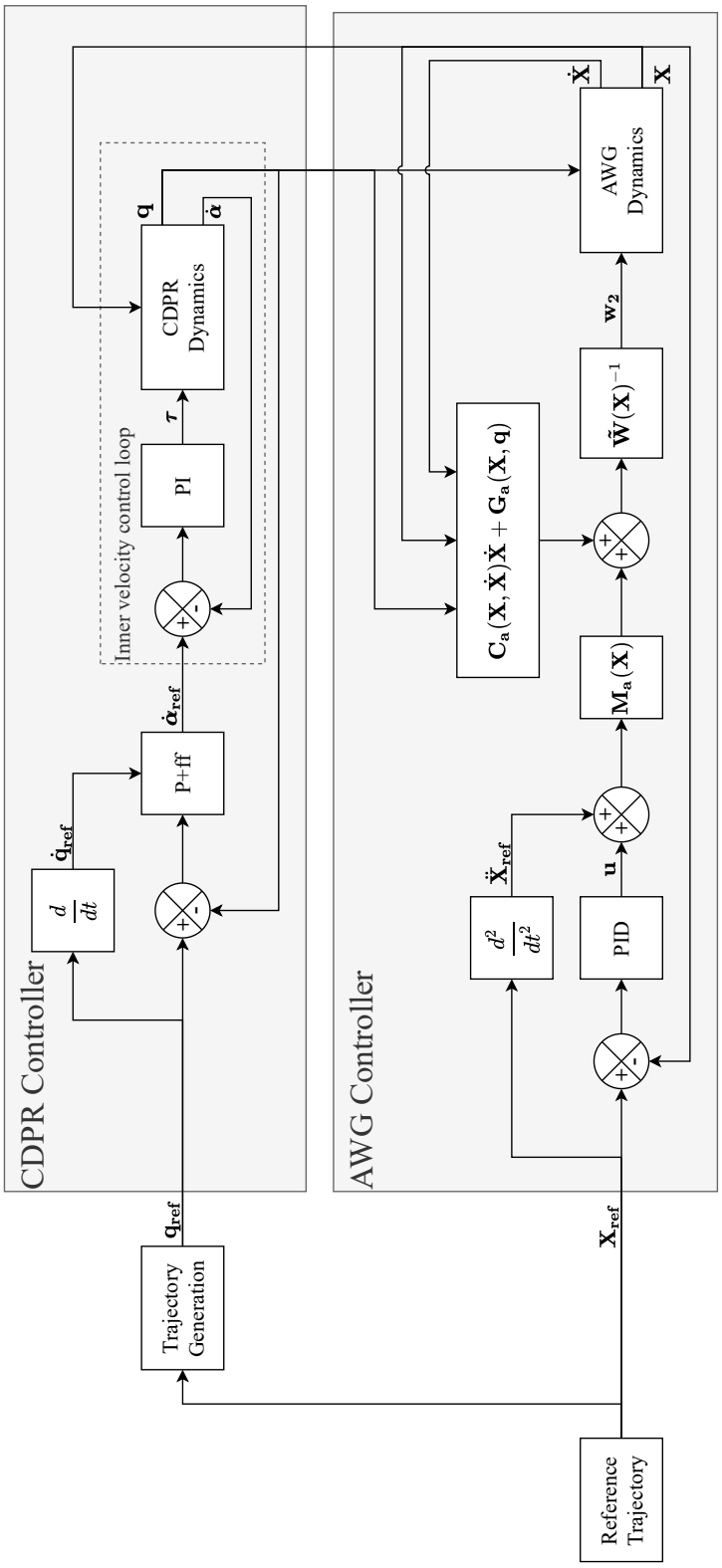


Figure 3.6 – AMES partitioned controller.

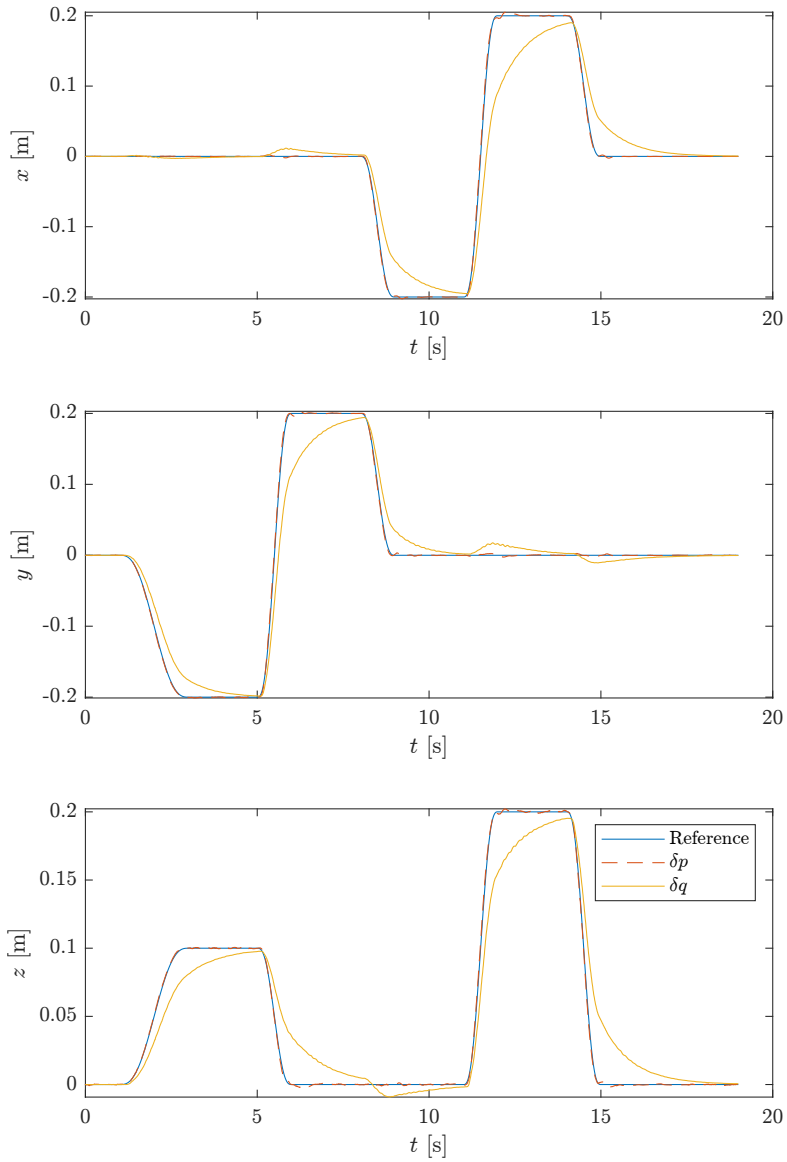


Figure 3.7 – Dynamic maneuver handling.

Energy Consumption

Two experiments are carried out to assess the added value of the CDPR carrier with respect to energy consumption. First, the same AWG trajectory $\mathbf{X_{ref}}$ with a vertical $\boldsymbol{\eta}_{\mathbf{ref}}$ is tracked with and without motion of the CDPR.

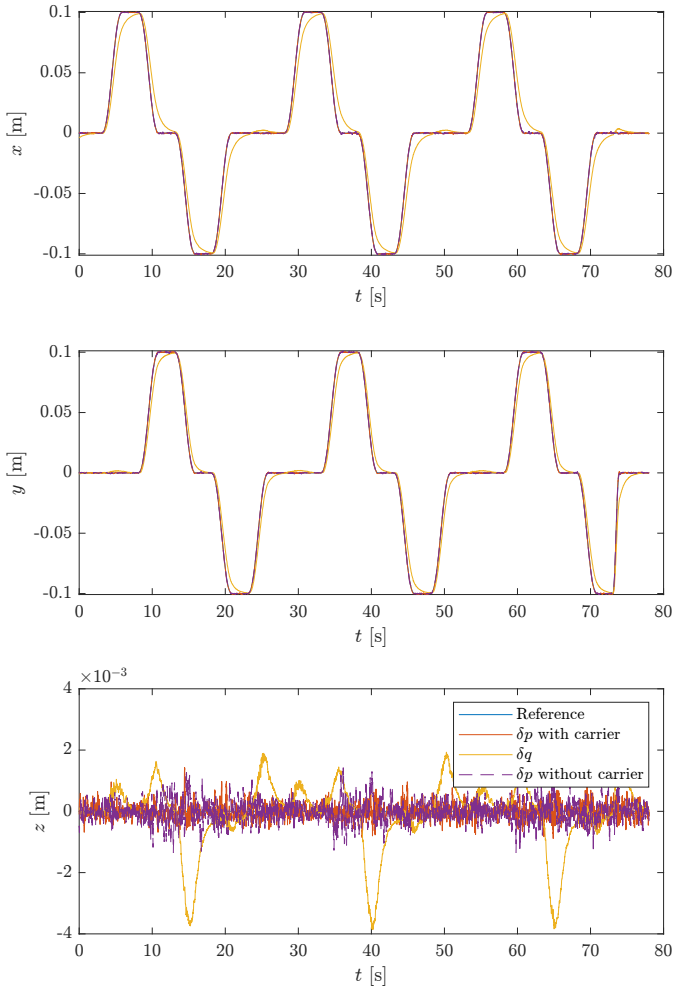
The power consumption of the AWG is measured as in Section 2.4.2 using the battery voltage and the currents provided by the ESCs. The power consumption of the CDPR can be estimated by adding the mechanical power output and the resistive losses, neglecting core losses and switching losses in the winch drive MOSFETs. The mechanical power output of a motor is the product of the torque τ_j by the rotational velocity $\dot{\alpha}_j$. Since the torque is proportional to the current i_c on a DC motor, the mechanical power becomes: $P_m = k_{em} i_c \dot{\alpha}$, with k_{em} the torque constant of the motor. Knowing the resistance of the windings R_m, the resistive losses are: $P_c = R_m i_c^2$. Hence, the estimated electrical power consumption of a DC motor is given by: $P_{motor} = k_{em} i_c \dot{\alpha} + R_m i_c^2$.

The relative positions of the AWG and of the CDPR carrier, as well as the power consumption of the propulsion units, are given in Figure 3.8. It is worth noticing that the carrier does not have a significant impact on the root-mean-square error (RMSE) of the AWG position (see Table 3.1): the RSME stays submillimetric as highlighted in Section 2.4.2. Nonetheless, the power consumption of the propulsion units is clearly lower when using a carrier, especially at steady state. This can be heard in the audio track of the associated video: thanks to the carrier motion, the propeller noise tends to decrease at steady state. Without the carrier, the AWG consumes 3.27 Wh during the trajectory. Adding the carrier lowers the consumption to 1.52 Wh (−52 %). The global energy consumption is also significantly reduced since the carrier, with its balancing springs, only consumes 0.05 Wh.

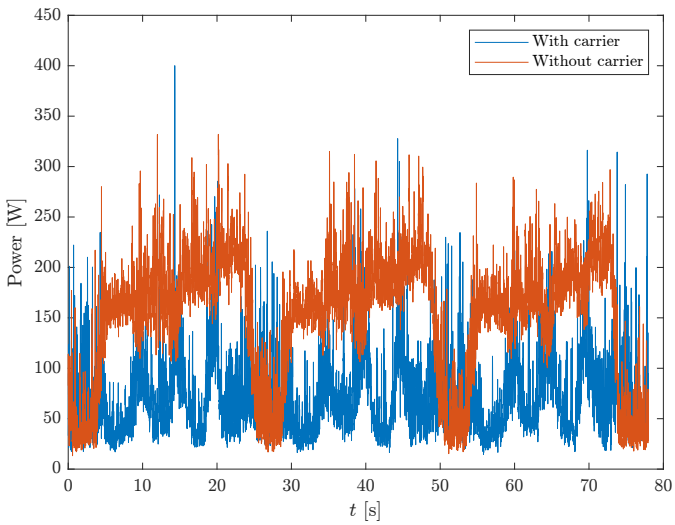
The carrier also improves energy efficiency for stationary orientations of the AWG platform. For a 5 deg roll angle, the average power consumption with the carrier is 92.5 W, −51 % less than without the carrier (188.6 W).

Axis	$\mathbf{x_f}$	$\mathbf{y_f}$	$\mathbf{z_f}$
With carrier	0.56	0.61	0.34
Without carrier	0.45	0.39	0.42

Table 3.1 – RMSE per axis in [mm].



(a) Trajectories.



(b) Power consumption.

Figure 3.8 – AWG power consumption comparison with and without the CDPR carrier.

3.5 Nonlinear Model Predictive Controller

In Section 3.4, a partitioned controller allowed for simultaneous control of the AWG and its CDPR carrier. The heterogeneous dynamics of both systems were not taken into account. The carrier was always lagging behind the AWG, leaving room for improvements.

Nonlinear model predictive control predicts the likely future behavior of the plant using a dynamic model and therefore can compensate for the lag between the AWG and the CDPR. This improves the strategy (3.21) since the NMPC uses a dynamic model rather than a kinematic one.

In the following, after briefly recalling the problem formulation for NMPC, the resolution of the control allocation is discussed. Then, the sufficient condition for offset-free control from Chapter 2 is tested.

3.5.1 Prediction Model

Because of important dry friction on pulleys, the rotational velocity $\dot{\boldsymbol{\alpha}}$ of the winches is still regulated with the PI controller proposed in Section 3.4.3. The winches are assumed to have first-order dynamics with time constant $t_w = 0.5\,\mathrm{s}$:

$$\ddot{\boldsymbol{\alpha}} = \frac{1}{t_w}(\dot{\boldsymbol{\alpha}}_{\mathbf{ref}} - \dot{\boldsymbol{\alpha}}) \tag{3.23}$$

Since the movement of the carrier is supposed to be slow in its large workspace, the Jacobian matrix $\mathbf{J}(\mathbf{q})$ (3.12) evolves slowly. So, the time-derivative of the Jacobian matrix in the second-order kinematic model (3.13) can be neglected:

$$\mathbf{J}\ddot{\mathbf{q}} = R\ddot{\boldsymbol{\alpha}} \tag{3.24}$$

Using (3.12) and (3.24) in (3.23) yields the movement of the anchoring point C modeled as a first-order system with the same time constant t_w:

$$\ddot{\mathbf{q}} = \frac{1}{t_w}(\dot{\mathbf{q}}_{\mathbf{ref}} - \dot{\mathbf{q}}) \tag{3.25}$$

The anchoring point dynamics (3.25) is used rather than the winch dynamics (3.23) since it can model any generic carrier and it alleviates the model complexity. By

choosing the carrier dynamics, the inverse Jacobian matrix does not appear in the dynamic model described by the differential equation (3.25). Instead, the Jacobian matrix is used as a gain on the plant input to map the NMPC control signal $\dot{\mathbf{q}}_{\mathbf{ref}}$ to the reference signal $\dot{\boldsymbol{\alpha}}_{\mathbf{ref}}$ of the winch velocity loop as illustrated in Figure 3.9.

The plant model of the AMES is obtained by combining equations (3.17) and (3.25):

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}_{\mathbf{nom}}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} &= \mathbf{g}_{\mathbf{nom}}(\mathbf{x}) \end{aligned} \tag{3.26}$$

with $\mathbf{u} = (\dot{\mathbf{q}}_{\mathbf{ref}}^T \, \mathbf{w_2}^T)^T$.

The plant model is augmented with integrators before the AWG input $\mathbf{w_2}$ such that the plant model with input $(\dot{\mathbf{q}}_{\mathbf{ref}}^T \, \mathbf{w_2}^T)^T$ is transformed into a model more suitable for NMPC systems with input $\mathbf{u_{aug}} = (\dot{\mathbf{q}}_{\mathbf{ref}}^T \, \dot{\mathbf{w}}_{\mathbf{2}}^T)^T$:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}_{\mathbf{aug}}(\mathbf{x}_{\mathbf{aug}}, \mathbf{u}_{\mathbf{aug}}) \\ \mathbf{y} &= \mathbf{g}_{\mathbf{aug}}(\mathbf{x}_{\mathbf{aug}}) \end{aligned} \tag{3.27}$$

This synthesis model is in the delta-input form as introduced in Section 2.5. At the expense of an increased dimension of the state vector, this formulation enforces an optimal control input $\mathbf{u_{aug}} = \mathbf{0}$ for any desired steady state. Furthermore, this formulation allows for penalizing the derivative of the AWG nominal control input $\mathbf{w_2}$ in the cost function, which results in a smoother control signal and consequently a smoother variation of the current in the brushless DC motors.

3.5.2 Cost Function and Control Allocation

Control allocation aims at distributing a desired total control effort over the redundant winch and propeller actuators. With MPC, the control allocation strategy is designed by the choice of the cost function while enforcing the constraints on actuator saturation.

In equation (2.73) from Section 2.6, the cost function consisted in a weighted sum of the tracking error and the energy of the control input signal. For the AMES, this corresponds to the following cost function J:

$$J = \int_0^T \left(\|\mathbf{X}(t) - \mathbf{X}_{\mathbf{ref}}(t)\|_{\mathbf{Q}}^2 + \|\dot{\mathbf{w}}_{\mathbf{2}}(t)\|_{\mathbf{R_{dw}}}^2 + \left\|\dot{\mathbf{q}}_{\mathbf{ref}}(t)\right\|_{\mathbf{R_{dq}}}^2 \right) dt \tag{3.28}$$

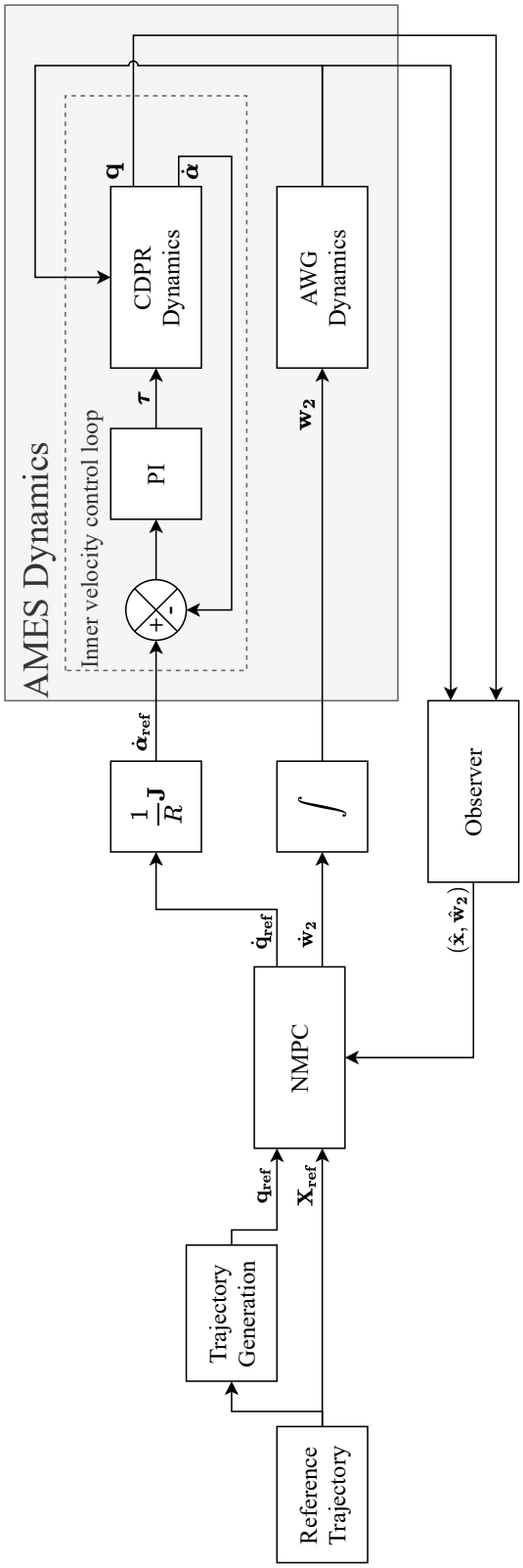


Figure 3.9 – NMPC block diagram for the control of the AMES.

However, neither the signed squared propeller rotational velocities $\mathbf{w_2}$ nor the carrier position $\mathbf{q_{ref}}$ are minimized is this cost function. Thereby, no particular solution $(\mathbf{w_{2,\infty}}^T\ \mathbf{q_{ref,\infty}}^T)^T$ is preferred among the multiple solutions for the steady state $(\mathbf{X}_\infty\ \dot{\mathbf{w}}_{\mathbf{2},\infty}^T\ \dot{\mathbf{q}}_{\mathbf{ref},\infty}^T)^T = (\mathbf{X_{ref}}\ \mathbf{0}^T\ \mathbf{0}^T)^T$. To obtain the solution that minimizes the power consumption of the AWG, a penalty on the propeller velocity should be added in the cost function, yielding:

$$J = \int_0^T \left(\|\mathbf{X}(t) - \mathbf{X_{ref}}(t)\|^2_{\mathbf{Q}} + \|\mathbf{w_2}(t)\|^2_{\mathbf{R_w}} + \|\dot{\mathbf{w}}_\mathbf{2}(t)\|^2_{\mathbf{R_{dw}}} + \left\|\dot{\mathbf{q}}_\mathbf{ref}(t)\right\|^2_{\mathbf{R_{dq}}} \right) dt \tag{3.29}$$

However, the propeller velocities are not zero at steady state ($\mathbf{w_{2,\infty}} \neq \mathbf{0}$) if the desired orientation in $\mathbf{X_{ref}}$ does not correspond to a vertical $\mathbf{z_b}$ axis. Consequently, the previous cost function may not be minimal at the system steady state ($\mathbf{u} \neq \mathbf{0}$). The solution minimizing (3.29) is then a trade-off between the quadratic errors on the controlled variables and the energy of the AWG control input signal $\mathbf{w_2}$. As a consequence, an offset-free steady state cannot be reached.

To overcome this issue, an alternative is to replace the penalty on the propeller velocities by a penalty on a desired position of the carrier $\mathbf{q_{ref}}$. The desired carrier position $\mathbf{q_{ref}}$ is chosen to minimize the power consumption of the AWG at steady state by solving the optimization problem (3.21), yielding finally the following cost function:

$$J = \int_0^T \left(\|\mathbf{X}(t) - \mathbf{X_{ref}}(t)\|^2_{\mathbf{Q_X}} + \left\|\mathbf{q}(t) - \mathbf{q_{ref}}(t)\right\|^2_{\mathbf{Q_q}} + \|\dot{\mathbf{w}}_\mathbf{2}(t)\|^2_{\mathbf{R_{dw}}} + \left\|\dot{\mathbf{q}}_\mathbf{ref}(t)\right\|^2_{\mathbf{R_{dq}}} \right) dt \tag{3.30}$$

3.5.3 Offset-Free NMPC

In this section, the delta-input formulation is used accordingly to the development in Section 2.6 to design and prove the offset-free nature of the control. As stated in the previous section, the AWG dynamic model is augmented with integrators on the propeller inputs to achieve the delta-input formulation, resulting in controlling the variation of the propeller velocities. On the other hand, the carrier dynamics can be already considered a delta-input formulation without introducing additional integrators. As illustrated in Figure 3.10, moving the integrator on the outputs of the carrier model (at the top) to the inputs yields an equivalent model (at the bottom) with the same input-output behavior. This equivalent model is used as a delta-input formulation of the carrier dynamics where the control signal is a variation of the anchoring point position, i.e. a velocity reference.

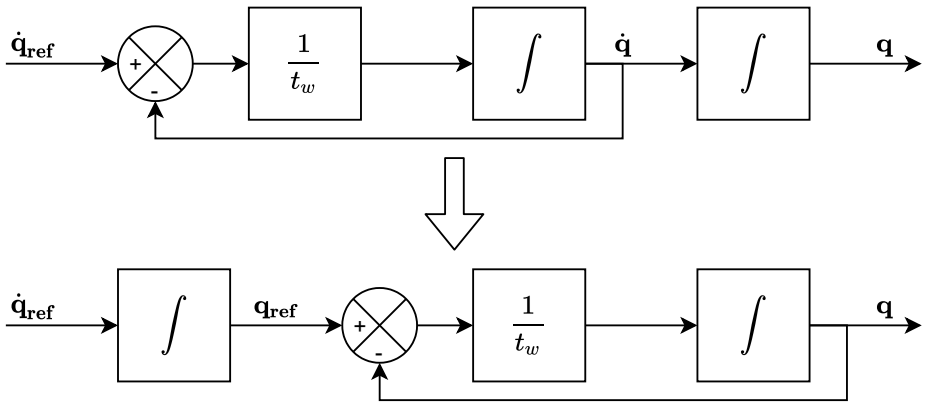


Figure 3.10 – Equivalent delta-input formulation of the carrier dynamics.

With the whole AMES (carrier and AWG) model in the delta-input formulation, offset-free property of the NMPC can be inferred from Theorem 1 of Section 2.6.3.

Assuming that only the pose of the AWG $\mathbf{X}$ ($\dim(\mathbf{X}) = 6$) and the position of the carrier $\mathbf{q}$ ($\dim(\mathbf{q}) = 3$) are measured, the number of measurements is $n_y = 9$. From the previous modeling, the AMES model (3.27) has $n_u = \dim(\dot{\mathbf{w}}_\mathbf{2}) + \dim(\dot{\mathbf{q}}_\mathbf{ref}) = 9$ delta inputs. Thereby, $n_u = n_y$ and the controller is offset-free if Assumptions 1-4 can be verified.

To test Proposition 1 (in order to verify Assumption 2), the following matrices of the AMES state-space model are introduced based on the linearized models from Appendix A:

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & -\mathbf{M}^{-1}\mathbf{K}_\mathbf{X} & -\mathbf{M}^{-1}\mathbf{K}_\mathbf{q} & \mathbf{M}^{-1}\tilde{\mathbf{W}} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\frac{1}{t_w}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{3.31}$$

$$\mathbf{B} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \frac{1}{t_w}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{3.32}$$

$$\mathbf{C} = \begin{pmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{pmatrix} \tag{3.33}$$

Let us introduce the Jacobian

$$\mathbf{J}_\infty = \begin{pmatrix} \mathbf{0} & -\mathbf{M}^{-1}\mathbf{K}_\mathbf{X} & -\mathbf{M}^{-1}\mathbf{K}_\mathbf{q} & \mathbf{M}^{-1}\tilde{\mathbf{W}} & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\frac{1}{t_w}\mathbf{I} & \mathbf{0} & \frac{1}{t_w}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{3.34}$$

The Jacobian is of full rank since $\det(\mathbf{J}_\infty) = -\frac{1}{t_w}\det(\mathbf{M}^{-1}\tilde{\mathbf{W}}) \neq 0$. Therefore, from Proposition 1, Assumption 2 holds.

The observer is designed using an EKF, therefore Assumption 3 also holds since there are as many measured variables as inputs.

In conclusion, offset-free control can be reached under the Assumptions 1 and 4.

3.5.4 Simulation Results

The NMPC scheme of Figure 3.9 is implemented to control the AMES in simulation.

Controller Tuning

The measurement of the AWG velocity $\dot{\mathbf{X}}$ is considered in the observer along with the measurement of the AWG pose $\mathbf{X}$ and the carrier position $\mathbf{q}$ to improve the estimates with a nonlinear plant model. As highlighted by Morari and Maeder, the condition $n_u = n_y$ can be relaxed if there is a linear dependency between the measurements, i.e. if the measurements are degenerate (Morari & Maeder, 2012). Indeed, since at steady state $\dot{\mathbf{X}} = \mathbf{0}$, we admit that Assumptions 2-3 are still valid, and so is Theorem 1.

The NMPC runs at 100 Hz. In order to have a 1 s prediction horizon, the discretization sampling period must be chosen such that the OCP stays solvable in strictly less than 10 ms. Indeed, increasing the sampling period for the discretization reduces the number of discretization steps, which is proportional to the computational complexity. Here, the prediction model is discretized with a 0.02 s sampling period, reducing the computational burden as proposed in (Bicego et al., 2020). The forces generated by propellers are limited to 7.2 N.

The following first simulations are carried out for a constant orientation of the AWG,

where its $\mathbf{z_b}$ axis is vertical. Therefore, the carrier position reference $\mathbf{q_{ref}}$, that minimizes the power consumption of the AWG for a steady state $\mathbf{X_{ref}}$, has a simple analytic form:

$$\mathbf{q_{ref}}(\mathbf{X_{ref}}) = \mathbf{p_{ref}} + \begin{pmatrix} 0 \\ 0 \\ \delta z_{eq} \end{pmatrix} \tag{3.35}$$

with δz_{eq} the vertical distance between the CoM of the AWG G and the anchoring point C at the equilibrium of the AWG.

The weighting matrices are selected diagonal (see Table 3.2 for numerical values):

$\mathbf{Q_X} = diag(Q_p\mathbf{I}, \mathbf{Q_\eta})$

$\mathbf{Q_q} = Q_q\mathbf{I}$

$\mathbf{R_{dw}} = R_{dw}\mathbf{I}$

$\mathbf{R_{dq}} = R_{dq}\mathbf{I}$

Parameter	Diagonal entries
Q_p	25
$\mathbf{Q_\eta}$	$4.9e0, 4.9e0, 1.3e-3$
Q_q	2.5
R_{dw}	$1.1e-17$
R_{dq}	$4.05e-2$

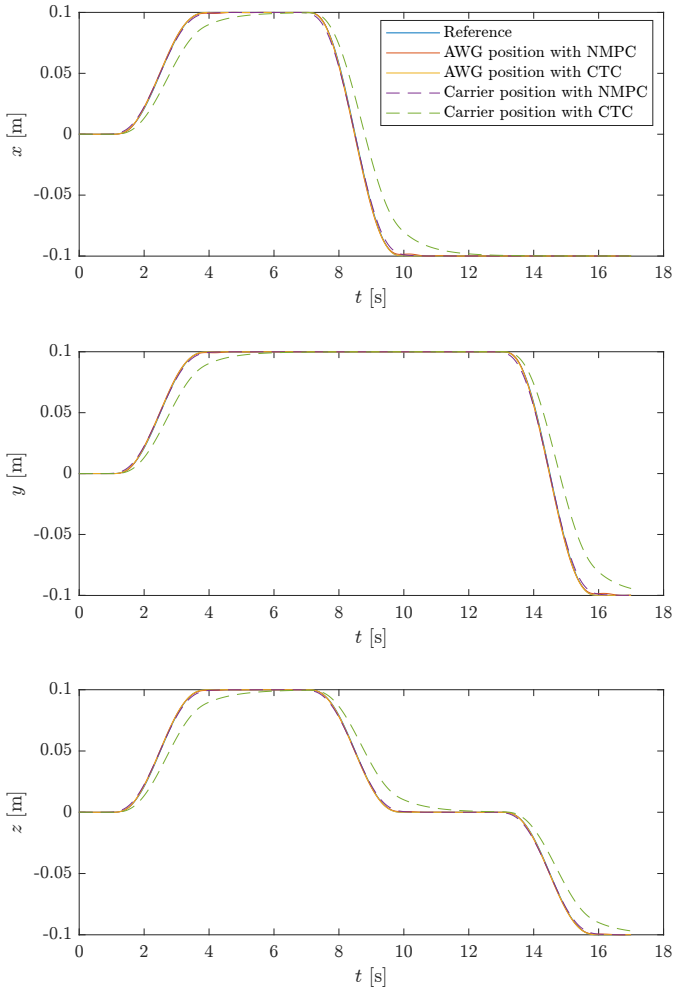
Table 3.2 – NMPC tuning parameters for the AMES.

Trajectory Tracking

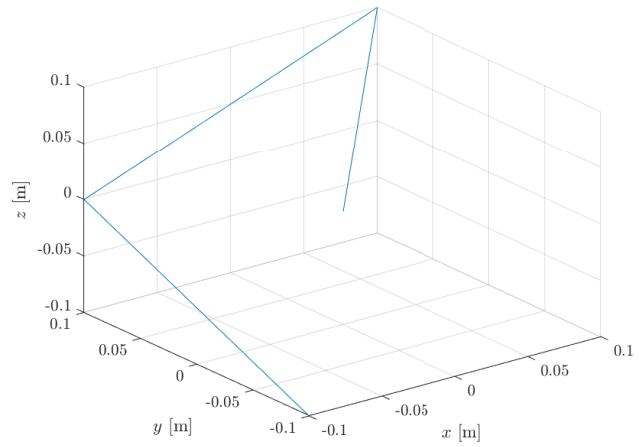
Figure 3.11 shows the positions of the AWG and the carrier during the tracking of a 3D trajectory reference with both NMPC and partitioned controller (referred as CTC in figure legends). The NMPC anticipates the slow dynamics of the carrier, suppressing the lag visible with the partitioned controller.

Disturbance Rejection

A disturbance step is added at $t = 0$ to the simulation model by emulating a 0.25 kg load. The closed-loop responses with and without inclusion of the EKF observer in



(a) Trajectory tracking.



(b) 3D reference trajectory.

Figure 3.11 – Tracking of a 3D trajectory using NMPC and partitioned controller.

the control scheme of Figure 3.9 are compared in Figure 3.12. The steady-state error vanishes when the EKF estimates the augmented plant states. The load disturbance is lumped in the estimates of the states $\hat{\mathbf{w}}_2$ and $\hat{\mathbf{q}}$ such that the NMPC can take the disturbance into account, while it remains unaccounted for if measured $\mathbf{w}_2$ states are used.

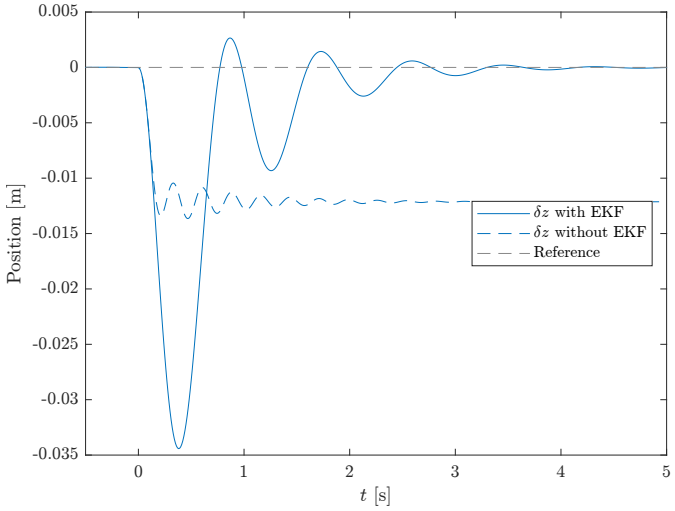


Figure 3.12 – Response of the NMPC to a step disturbance.

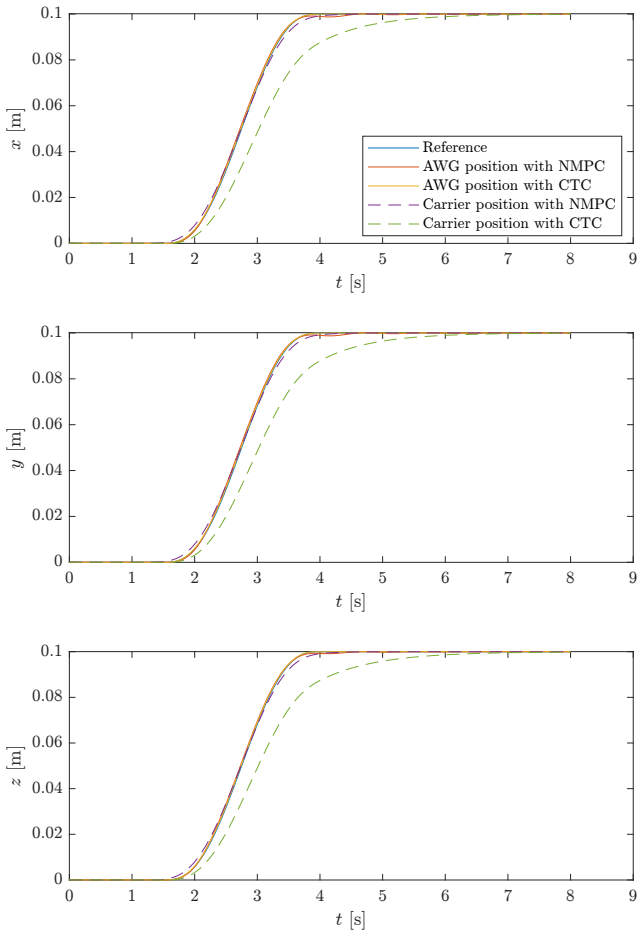
Energy Efficiency

Finally, the energy consumption of the AWG is compared using the NMPC and the partitioned controller in Figure 3.13. The NMPC reduces further the power consumption compared to the partitioned controller. This is explained by the carrier lagging behind the AWG obtained with the partitioned controller that can be observed in Figure 3.13a. Since the NMPC anticipates the slow dynamics of the carrier, it almost perfectly cancels the moment of the spring restoring force, which is the main source of power consumption.

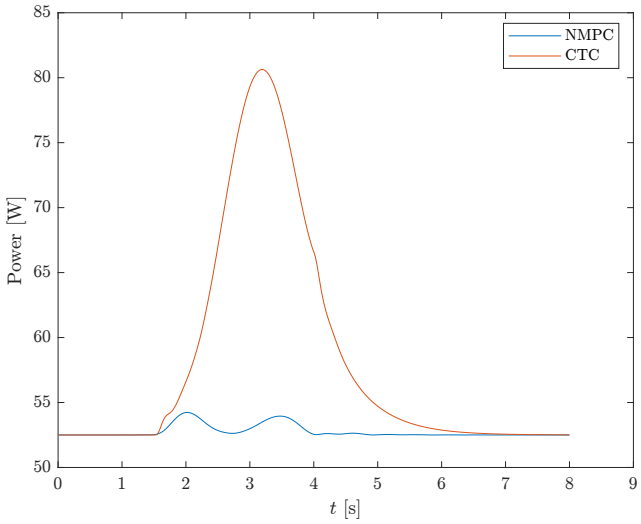
3.6 Conclusion

In this chapter, a CDPR carrier is added to the AWG to move the anchoring point of the spring. Two solutions are proposed to resolve the redundancy.

A partitioned controller is implemented to control independently the carrier and the AWG. The controller is tested experimentally and, as a result, the energy consumption is reduced by approximately 50%. However, there is still room for improvements since



(a) Trajectory.



(b) Power consumption.

Figure 3.13 – AWG power consumption comparison with NMPC and partitioned controller.

the carrier lags behind the AWG.

A nonlinear model predictive controller overcomes some drawbacks of the previous strategy anticipating the slow carrier dynamics. The controller is tested in simulation to validate the disturbance rejection capabilities and its energy efficiency.

4 Conclusion

This thesis introduces the concept of Aerial Manipulator with Elastic Suspension, a new type of robot, hybrid between a serial manipulator and an aerial manipulator with the spring acting as a flexible linkage. A prototype has been built to validate experimentally the proof of concept. The main results are summarized in the following as well as some perspectives.

4.1 Contributions

This thesis started as a continuation of the DexterWide project, where additional thrusters have been embedded on the end effector of CDPRs to improve disturbance rejection and trajectory tracking dynamics. The resulting robots are overactuated, the end effectors being fully actuated even without the cables. The AMES is proposed as a simpler alternative to CDPRs, where the end effector, an omnidirectional aerial manipulator, is suspended from a robotic carrier by a spring for gravity compensation. A prototype has been built, initially tested without a robotic carrier.

A computed torque controller is developed to assess experimentally the performance characteristics of the prototype. The stability of the control law is proven using singular perturbation theory. Experiments validated the added value of the spring with respect to energy efficiency. The closed-loop bandwidth of the carrierless AMES is improved using a nonlinear model predictive controller, allowing for eliminating the steady-state error without the windup of the integral term that happens with the computed torque controller. The resulting control input is also smoother, reducing copper and switching losses, and thus adding a beneficial roll-off effect. A sufficient condition is proposed to guarantee offset-free tracking and is validated in simulation.

A CDPR carrier is then added to control the position of the spring anchoring point. As a consequence, the AMES becomes overactuated. To resolve the redundancy, two solutions are proposed. First, a partitioned controller is implemented to control independently the carrier and the aerial manipulator. The reference position of the carrier is chosen to cancel the moment of the spring restoring force at the center of mass. Experiments show significant improvements of the energy efficiency, the consumption being reduced by approximately 50%. However, the carrier is lagging behind the aerial manipulator, leaving room for further improvements. A nonlinear model predictive controller is then implemented. It combines the control of both subsystems and anticipates their dynamics. The controller is tested in simulation and improves the energy efficiency.

4.2 Perspectives

4.2.1 Improvements

The experimental performance assessment of the AMES shows that the main limiting factor for the workspace and for the energy efficiency is the moment at the center of mass due to the spring restoring force. An optimal design strategy is under development in collaboration with another PhD student, in which the spring is anchored at the center of mass of a new aerial manipulator. On this new prototype, unidirectional thrusters are preferred rather than bidirectional propulsion units for a better power-to-mass ratio, since the number of motors and propellers is reduced while preserving the maximum wrench that can be obtained. Indeed, with the actual prototype, only half of the motors are used simultaneously, the remaining ones are idling.

Experiments and simulations using the NMPC result in slower disturbance rejection dynamics compared to the computed torque controller, due to the slow convergence of the EKF-based observer. More advanced nonlinear observers, such as the moving horizon estimator (MHE), may accelerate the state estimation and, therefore, improve the closed-loop performance. Also, the sufficient condition proposed for offset-free NMPC requires an experimental validation.

The control schemes implemented during this thesis are suited for contactless tasks. They can be modified by adding a wrench observer in order to estimate external forces and torques to accomplish tasks requiring physical interaction, similarly to the disturbance observer introduced for the NMPC (Bodie et al., 2021; Ryll et al., 2017). Combining physical interaction capable control schemes with embedded cameras

replacing the motion capture system widens the range of applications of the AMES.

4.2.2 Applications

Two future projects in the ICube lab will use the contributions of this thesis.

The STRAD project (STReet Art Drone) aims at drawing graffiti on large vertical surfaces with fast and accurate motions of an AMES (see Figure 4.1). The AMES embeds a camera for visual servoing with respect to a painting in progress.

Figure 4.1 – STRAD project illustration (from https://www.dextair.com/).

The TIR4sTREEt project (thermal infrared for street trees) uses a tethered balloon to carry the aerial manipulator (see Figure 4.2). The aerial manipulator allows for a fast

and accurate positioning of sensors for urban climatology.

Figure 4.2 – TIR4sTREEt project illustration (from https://www.dextair.com/).

A Linearization of the AWG Dynamics

A.1 Nonlinear AWG Dynamics

Let us recall AWG dynamics in the canonical form from equation (2.8):

$$\underbrace{\begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T\mathbf{J}_b\mathbf{S} \end{pmatrix}}_{\mathbf{M(X)}} \begin{pmatrix} \ddot{\mathbf{p}} \\ \ddot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T(\mathbf{J}_b\dot{\mathbf{S}} + [\mathbf{S}\dot{\boldsymbol{\eta}}]_\times \mathbf{J}_b\mathbf{S}) \end{pmatrix}}_{\mathbf{C(X,\dot{X})}} \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} -m{}^f\mathbf{g} - {}^f\mathbf{F_s} \\ -\mathbf{S}^T{}^b\mathbf{N_s} \end{pmatrix}}_{\mathbf{G(X)}} = \underbrace{\begin{pmatrix} \mathbf{R_{fb}} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T \end{pmatrix} \mathbf{W}_b \mathbf{w_2}}_{\tilde{\mathbf{W}}\mathbf{(X)}} \quad \text{(A.1)}$$

The spring restoring force $\mathbf{F_s}$ and the associated moment $\mathbf{N_s}$ at CoM are:

$${}^f\mathbf{F_s} = k\,(\|\mathbf{AC}\| - l_0)\,\frac{{}^f\mathbf{AC}}{\|\mathbf{AC}\|} \quad \text{(A.2)}$$

$${}^b\mathbf{N_s} = \|\mathbf{GA}\|\,{}^b\mathbf{z_b} \times \mathbf{R}_{\mathbf{fb}}^T\,{}^f\mathbf{F_s} \quad \text{(A.3)}$$

Let l and $\mathbf{u}$ be defined such that: $l = \|\mathbf{CA}\|$ and $\mathbf{u} = \frac{\mathbf{AC}}{\|\mathbf{AC}\|}$.

The rate of change of $\|\mathbf{AC}\|$ is the relative velocity of C with respect to A projected along $\mathbf{u}$. Therefore, $\dot{l} = \mathbf{u}^T(\dot{\mathbf{q}} - \mathbf{v_A})$.

Since

$$\mathbf{v_A} = \dot{\mathbf{p}} + \mathbf{AG} \times \boldsymbol{\omega} \tag{A.4}$$

we have:

$$\dot{l} = -\underbrace{\begin{pmatrix} {}^f\mathbf{u}^T & ({}^b\mathbf{u} \times {}^b\mathbf{AG})^T \end{pmatrix}}_{\mathbf{J}} \begin{pmatrix} \dot{\mathbf{p}} \\ \boldsymbol{\omega} \end{pmatrix} + {}^f\mathbf{u}^T \dot{\mathbf{q}} \tag{A.5}$$

The wrench generated by the spring on the AWG can be written:

$$\begin{pmatrix} {}^f\mathbf{F_s} \\ {}^b\mathbf{N_s} \end{pmatrix} = k(l - l_0) \underbrace{\begin{pmatrix} {}^f\mathbf{u} \\ \|GA\| \, {}^b\mathbf{z_b} \times {}^b\mathbf{u} \end{pmatrix}}_{\mathbf{J}^T} \tag{A.6}$$

From equations (2.2) and (A.5):

$$\dot{l} = \underbrace{-\mathbf{J} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}(\boldsymbol{\eta}) \end{pmatrix}}_{\tilde{\mathbf{J}}} \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\eta}} \end{pmatrix} + {}^f\mathbf{u}^T \dot{\mathbf{q}} \tag{A.7}$$

Finally, equations (A.1), (A.6) and (A.7) yield:

$$\underbrace{\begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T \mathbf{J}_b \mathbf{S} \end{pmatrix}}_{\mathbf{M(X)}} \begin{pmatrix} \ddot{\mathbf{p}} \\ \ddot{\boldsymbol{\eta}} \end{pmatrix} + \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T (\mathbf{J}_b \dot{\mathbf{S}} + [\mathbf{S}\dot{\boldsymbol{\eta}}]_\times \mathbf{J}_b \mathbf{S}) \end{pmatrix}}_{\mathbf{C(X,\dot{X})}} \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\eta}} \end{pmatrix}$$
$$+ \underbrace{\begin{pmatrix} -m^f\mathbf{g} \\ \mathbf{0} \end{pmatrix} - k(l(\mathbf{X},\mathbf{q}) - l_0)\mathbf{J}(\mathbf{X},\mathbf{q})^T}_{\mathbf{G(X,q)}} = \underbrace{\begin{pmatrix} \mathbf{R_{fb}} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^T \end{pmatrix} \mathbf{W}_b \mathbf{w_2}}_{\tilde{\mathbf{W}}\mathbf{(X)}} \tag{A.8}$$

A.2 Linearized AWG Dynamics

$$\mathbf{f_{AWG}}(\mathbf{X}, \dot{\mathbf{X}}, \ddot{\mathbf{X}}, \mathbf{q}, \mathbf{f}) = \mathbf{M(X)}\ddot{\mathbf{X}} + \mathbf{C(X,\dot{X})}\dot{\mathbf{X}} + \mathbf{G(X,q)} - \tilde{\mathbf{W}}\mathbf{(X)}\mathbf{w_2} \tag{A.9}$$

Let $\boldsymbol{\gamma} = (\mathbf{X}_\infty, \dot{\mathbf{X}}_\infty, \ddot{\mathbf{X}}_\infty, \mathbf{q}_\infty, \mathbf{w_{2,\infty}})$ be a steady state, therefore $\dot{\mathbf{X}}_\infty = \ddot{\mathbf{X}}_\infty = \mathbf{0}$.

$$\frac{\partial \mathbf{f_{AWG}}}{\partial \mathbf{w_2}}(\boldsymbol{\gamma}) = -\overline{\mathbf{W}}(\mathbf{X}_\infty) \tag{A.10}$$

$$\frac{\partial \mathbf{f_{AWG}}}{\partial \ddot{\mathbf{X}}}(\boldsymbol{\gamma}) = \mathbf{M}(\mathbf{X}_\infty) \tag{A.11}$$

$$\frac{\partial \mathbf{f_{AWG}}}{\partial \dot{\mathbf{X}}}(\boldsymbol{\gamma}) = \mathbf{0} \tag{A.12}$$

$$\frac{\partial \mathbf{f_{AWG}}}{\partial \mathbf{X}}(\boldsymbol{\gamma}) = -k\left(\mathbf{J}(\mathbf{X}_\infty, \mathbf{q}_\infty)^T \tilde{\mathbf{J}}(\mathbf{X}_\infty, \mathbf{q}_\infty) + (l(\mathbf{X}_\infty, \mathbf{q}_\infty) - l_0)\frac{\partial \tilde{\mathbf{J}}(\mathbf{X}, \mathbf{q})^T}{\partial \mathbf{X}}(\boldsymbol{\gamma})\right) - \frac{\partial \tilde{\mathbf{W}}(\mathbf{X})}{\partial \mathbf{X}}(\boldsymbol{\gamma})\mathbf{w_{2,\infty}} \tag{A.13}$$

$$\frac{\partial \mathbf{f_{AWG}}}{\partial \mathbf{q}}(\boldsymbol{\gamma}) = -k\left(\mathbf{J}(\mathbf{X}_\infty, \mathbf{q}_\infty)^T \frac{\partial l(\mathbf{X}, \mathbf{q})}{\partial \mathbf{q}}(\boldsymbol{\gamma})\right) \tag{A.14}$$

Thereby, the linearized model is:

$$\mathbf{M}(\mathbf{X}_\infty)\delta\ddot{\mathbf{X}} + \mathbf{K_X}(\mathbf{X}_\infty, \mathbf{q}_\infty)\delta\mathbf{X} + \mathbf{K_q}(\mathbf{X}_\infty, \mathbf{q}_\infty)\delta\mathbf{q} = \tilde{\mathbf{W}}(\mathbf{X}_\infty)\delta\mathbf{w_{2,\infty}} \tag{A.15}$$

A.3 Linearized Carrier Dynamics

The model of the carrier, with its velocity inner-loop is:

$$\ddot{\mathbf{q}} = \frac{1}{t_w}(\dot{\mathbf{q}}_{\mathbf{ref}} - \dot{\mathbf{q}}) \tag{A.16}$$

Taking the antiderivative, the dynamic model of the anchoring point becomes:

$$\dot{\mathbf{q}} = \frac{1}{t_w}(\mathbf{q_{ref}} - \mathbf{q}) \tag{A.17}$$

B Teensyshot Firmware

B.1 Introduction

Electronic speed controllers (ESCs) allow for using synchronous motors as if they were standard DC motors (hence the name “brushless DC motor”): they are powered with a DC power source (e.g. a Lipo battery) and the rotation of the rotor follows the rotating magnetic field with an optimal phase shift controlled using an estimator based on back electromotive force. However, most of the ESCs used for UAVs do not regulate the rotating magnetic field velocity. Indeed, for a constant input signal (called usually "throttle"), the rotor velocity depends mainly on the input voltage and on the mechanical load. Therefore, an actuator-level velocity loop is mandatory to control in a robust way the rotor velocity.

Initially, ESCs used analog signals (PWM) as input, which are sensitive to electromagnetic noise. Some recent ESCs propose alternative digital communications. Most recently, KISS ESCs introduced the DShot protocol. The DShot protocol has a 2000-step full resolution, a CRC checksum for error detection and can be used to send specific commands to the ESC. It also allows for requesting telemetry data, which contains among others the velocity of the rotating magnetic field. The telemetry data are sent through a dedicated 115 200 bps serial link.

The open-source Teensyshot firmware provides speed control for brushless DC motors without additional sensors. Teensyshot implements a fast anti-windup PID speed regulation loop running at 500 Hz on a Teensy microcontroller unit (both 3.5 and 4.0 models). It can handle up to 6 ESCs simultaneously. The presentation video features two contra-rotating propellers: https://youtu.be/n8SLrK5eN3c. An interference shadow can be seen in this video, showing that both propellers are well synchronized.

B.2 Main Tasks

Teensyshot implements two main asynchronous tasks: (i) communication with the host and (ii) communication with ESCs.

B.2.1 Communication with Host

The host and the Teensy board communicate according to a master/slave architecture through a 115200 bps serial link. The host sends the velocity reference as well as the PID coefficients (enabling online tuning of the controller) for all ESCs. In return, it receives telemetry data (containing magnetic field velocity, ESC temperature, battery voltage, current drawn by ESC) and an error code for debugging purposes for each ESC. A security watchdog stops the motors if there is no communication with the host for 40ms.

A Simulink S-Function is developed to communicate with Teensyshot through a Simulink diagram (see Figure B.1). This block is integrated into the open-source project RPIt and is used for the experiments. RPIt handles multiple Teensy boards thanks to the unique serial number of each board.

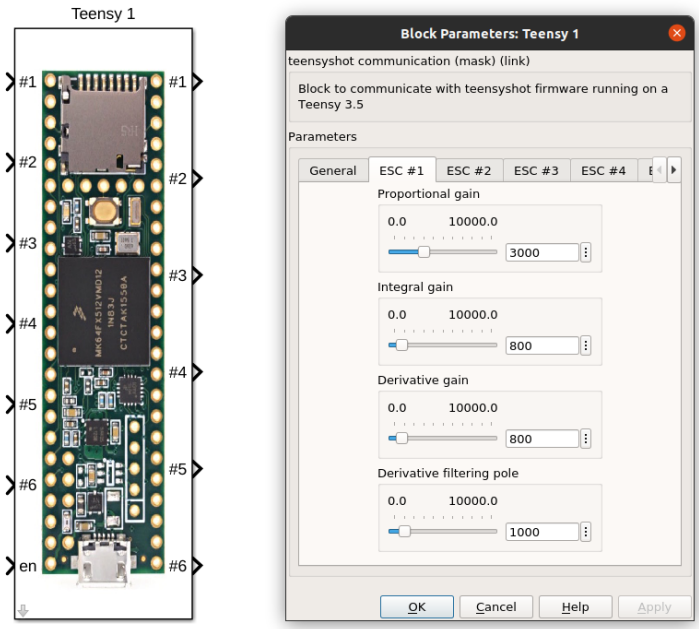


Figure B.1 – Teensyshot Simulink block.

B.2.2 Communication with ESCs

The DShot signal containing the throttle value is sent at 500 Hz. Since DShot is a new and non-standard protocol, there is no dedicated hardware module that provides a DShot interface. The version of DShot used for Teensyshot is DShot600. DShot600 communicates at 600 kbaud and therefore may need all the CPU resources during communication. To save precious CPU time, the waveform corresponding to the DShot signal is generated using Direct Memory Access (DMA) that updates output pins of the Teensy board asynchronously with hardware interrupts.

Independently from the DShot signal, the Teensy board listens for telemetry data from ESCs. Ideally, if there is no telemetry error, each DShot signal is answered by a telemetry package. After reception of the telemetry data, the variable containing the throttle value is updated by the PID controller. In case of an important rate of lost or corrupted telemetry packages, the host is warned through an error code.

B.3 Anti-Windup PID

The digital PID implemented in Teensyshot is illustrated in Figure B.2. Note that the integral term is calculated using the forward Euler approximation and the filtering coefficient f is a number between 0 and 1.

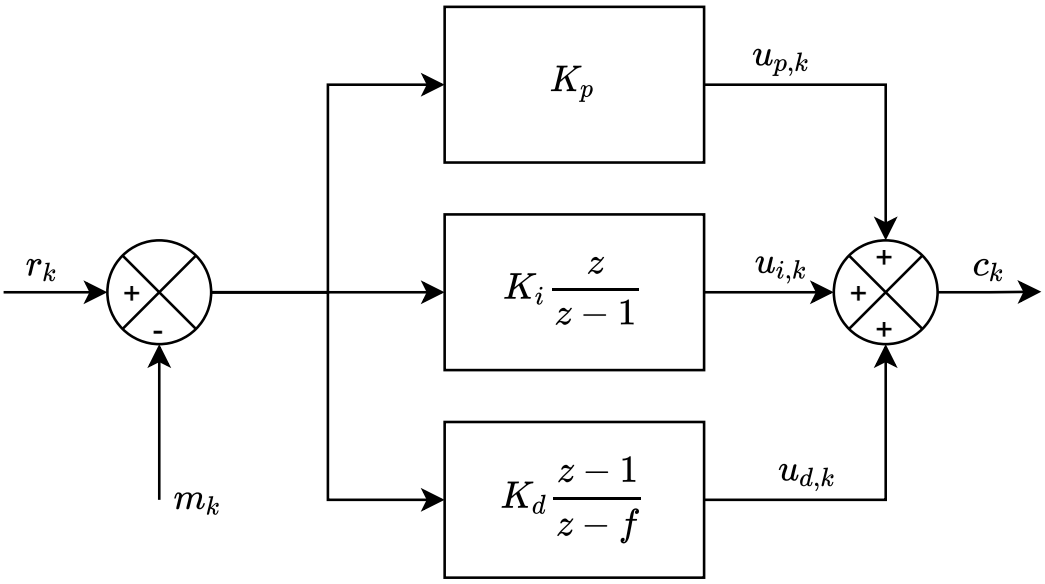


Figure B.2 – Digital PID.

To avoid integral windup, when the output u of the controller reaches saturation, the

value of the integral term is modified such that the output u stays at the saturation value. The implementation of the anti-windup is explained in Algorithm 1. The experiments presented in this manuscript use the following tuning coefficients: $K_p = 0.3$, $K_i = 0.024$, $K_d = 0.08$, $f = 0.1$.

Algorithm 1: Anti-windup PID.

input : Reference signal r_k, measurement signal m_k.
output : Control input c_k.
parameter : Tuning coefficients K_p, K_d, K_i, f and saturation values MIN and MAX.

```
// Computation of the error
```
$e_{k-1} \leftarrow e_k$;
$e_k \leftarrow r_k - m_k$;
```
// Computation of the derivative term
```
$u_{d,k-1} \leftarrow u_{d,k}$;
$u_{d,k} \leftarrow f u_{d,k-1} - K_d(e_k - e_{k-1})$;
```
// Computation of the integral term
```
$u_{i,k-1} \leftarrow u_{i,k}$;
```
// Output before anti-windup
```
$u_k \leftarrow K_p e_k + u_{d,k} + u_{i,k-1} + K_i e_k$;
```
// Anti-windup:  control signal equals exactly the saturation
```
if $u_k \leq MAX$ *and* $u_k \geq MIN$ **then**
    ```
    // If no saturation
    ```
    $u_{i,k} \leftarrow u_{i,k-1} + K_i e_k$;
else if $u_k > MAX$ **then**
    $u_{i,k} \leftarrow MAX - (K_p e_k + u_{d,k})$;
else
    $u_{i,k} \leftarrow MIN - (K_p e_k + u_{d,k})$;
end
```
// The control signal
```
$c_k \leftarrow K_p e_k + u_{d,k} + u_{i,k}$;

Publications

The articles published during this thesis and software developments are listed below.

Articles

In Conference Proceedings

Yiğit, A., Grappe, G., Cuvillon, L., Durand, S., & Gangloff, J. (2020). Preliminary Study of an Aerial Manipulator with Elastic Suspension. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 4287–4293. https://doi.org/10.1109/ICRA40945.2020.9196942

Yiğit, A., Arpa Perozo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021a). Improving Dynamics of an Aerial Manipulator with Elastic Suspension Using Nonlinear Model Predictive Control. *2021 IEEE International Conference on Robotics and Au-tomation (ICRA)*

Yiğit, A., Arpa Perozo, M., Ouafo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021). Aerial Manipulator Suspended From a Cable-Driven Parallel Robot: Preliminary Experimental Results. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*

Khayour, I., Cuvillon, L., Butin, C., Yigit, A., Durand, S., & Gangloff, J. (2020). Improving Disturbance Rejection and Dynamics of Cable Driven Parallel Robots with On-board Propellers. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6564–6569. https://doi.org/10.1109/IROS45743.2020.9341033

Bertrand, J., Yiğit, A., & Durand, S. (2020). Embedded Event-based Visual Odometry. *2020 6th International Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP)*, 1–8. https://doi.org/10.1109/EBCCSP51266.2020.9291346

Journal Papers

Yiğit, A., Arpa Perozo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021b). Novel Omnidirectional Aerial Manipulator With Elastic Suspension: Dynamic Control and Experimental Performance Assessment. *IEEE Robotics and Automation Letters*, *6*(2), 612–619. https://doi.org/10.1109/LRA.2020.3048880

Software Developments

Main developer of Teensyshot with my supervisor Jacques Gangloff: https://github.com/jacqu/teensyshot

Contributor to the Raspberry Pi Simulink Coder toolkit (RPIt) developed by Jacques Gangloff https://github.com/jacqu/rpit

Contributor to acados https://github.com/acados/acados

Bibliography

Begey, J., Cuvillon, L., Lesellier, M., Gouttefarde, M., & Gangloff, J. (2019). Dynamic Control of Parallel Robots Driven by Flexible Cables and Actuated by Position-Controlled Winches. *IEEE Transactions on Robotics*, *35*(1), 286–293. https://doi.org/10.1109/TRO.2018.2875415

Bemporad, A. (2021). Model Predictive Control. http://cse.lab.imtlucca.it/~bemporad/teaching/mpc/imt/1-linear_mpc.pdf

Bertrand, J., Yiğit, A., & Durand, S. (2020). Embedded Event-based Visual Odometry. *2020 6th International Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP)*, 1–8. https://doi.org/10.1109/EBCCSP51266.2020.9291346

Bicego, D., Mazzetto, J., Carli, R., Farina, M., & Franchi, A. (2020). Nonlinear Model Predictive Control with Enhanced Actuator Model for Multi-Rotor Aerial Vehicles with Generic Designs. *Journal of Intelligent & Robotic Systems*. https://doi.org/10.1007/s10846-020-01250-9

Bodie, K., Brunner, M., Pantic, M., Walser, S., Pfandler, P., Angst, U., Siegwart, R., & Nieto, J. (2021). Active Interaction Force Control for Contact-Based Inspection With a Fully Actuated Aerial Vehicle. *IEEE Transactions on Robotics*, *37*(3), 709–722. https://doi.org/10.1109/TRO.2020.3036623

Brescianini, D., & D'Andrea, R. (2018). An omni-directional multirotor vehicle. *Mechatronics*, *55*, 76–93. https://doi.org/10.1016/j.mechatronics.2018.08.005

Brunner, M., Bodie, K., Kamel, M., Pantic, M., Zhang, W., Nieto, J., & Siegwart, R. (2020). Trajectory Tracking Nonlinear Model Predictive Control for an Overactuated MAV. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 5342–5348. https://doi.org/10.1109/ICRA40945.2020.9197005

Campa, R., Bernal, J., & Soto, I. (2016). Kinematic modeling and control of the Hexapod parallel robot. *2016 American Control Conference (ACC)*, 1203–1208. https://doi.org/10.1109/ACC.2016.7525081

Chellal, R., Cuvillon, L., & Laroche, E. (2017). Model identification and vision-based H_∞ position control of 6-DoF cable-driven parallel robots. *International Journal of Control*, *90*(4), 684–701. https://doi.org/10.1080/00207179.2016.1220623

Chellal, R., Cuvillon, L., & Laroche, E. (2015). A kinematic vision-based position control of a 6-DoF cable-driven parallel robot. *Mechanisms and machine science* (pp. 213–225). Springer, Cham. https://doi.org/10.1007/978-3-319-09489-2{_}15

Chen, Y., Bruschetta, M., Picotti, E., & Beghi, A. (2019). MATMPC - A MATLAB Based Toolbox for Real-time Nonlinear Model Predictive Control. *2019 18th European Control Conference (ECC)*, 3365–3370. https://doi.org/10.23919/ECC.2019.8795788

Choi, S. Y., Choi, B. H., Jeong, S. Y., Gu, B. W., Yoo, S. J., & Rim, C. T. (2014). Tethered aerial robots using contactless power systems for extended mission time and range. *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*, 912–916. https://doi.org/10.1109/ECCE.2014.6953495

Danko, T. W., & Oh, P. Y. (2013). A hyper-redundant manipulator for Mobile Manipulating Unmanned Aerial Vehicles. *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, 974–981. https://doi.org/10.1109/ICUAS.2013.6564784

El-Ghazaly, G., Gouttefarde, M., & Creuze, V. (2015). Hybrid cable-thruster actuated underwater vehicle-manipulator systems: A study on force capabilities. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1672–1678. https://doi.org/10.1109/IROS.2015.7353592

Erskine, J., Chriette, A., & Caro, S. (2019). Control and Configuration Planning of an Aerial Cable Towed System. *2019 International Conference on Robotics and Automation (ICRA)*, 6440–6446. https://doi.org/10.1109/ICRA.2019.8794396

Esteban, S., Gordillo, F., & Aracil, J. (2013). Three-time scale singular perturbation control and stability analysis for an autonomous helicopter on a platform. *International Journal of Robust and Nonlinear Control*, *23*(12), 1360–1392. https://doi.org/10.1002/rnc.2823

Franchi, A., & Mallet, A. (2017). Adaptive closed-loop speed control of BLDC motors with applications to multi-rotor aerial vehicles. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 5203–5208. https://doi.org/10.1109/ICRA.2017.7989610

Frison, G., & Diehl, M. (2020). HPIPM: a high-performance quadratic programming framework for model predictive control. http://arxiv.org/abs/2003.02547

Gangloff, J., Yiğit, A., & Lesellier, M. (2020). RPIt. https://github.com/jacqu/RPIt/

Gouttefarde, M., Collard, J.-F., Riehl, N., & Baradat, C. (2015). Geometry Selection of a Redundantly Actuated Cable-Suspended Parallel Robot. *IEEE Transactions on Robotics*, *31*(2), 501–510. https://doi.org/10.1109/TRO.2015.2400253

Grimm, G., Messina, M., Tuna, S., & Teel, A. (2005). Model predictive control: for want of a local control Lyapunov function, all is not lost. *IEEE Transactions on Automatic Control*, *50*(5), 546–558. https://doi.org/10.1109/TAC.2005.847055

Hamandi, M., Usai, F., Sablé, Q., Staub, N., Tognon, M., & Franchi, A. (2021). Design of multirotor aerial vehicles: A taxonomy based on input allocation. *The International Journal of Robotics Research*, *40*(8-9), 1015–1044. https://doi.org/10.1177/02783649211025998

Heredia, G., Jimenez-Cano, A., Sanchez, I., Llorente, D., Vega, V., Braga, J., Acosta, J., & Ollero, A. (2014). Control of a multirotor outdoor aerial manipulator. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3417–3422. https://doi.org/10.1109/IROS.2014.6943038

Houska, B., Ferreau, H. J., & Diehl, M. (2011a). ACADO toolkit-An open-source framework for automatic control and dynamic optimization. *Optimal Control Applications and Methods*, *32*(3), 298–312. https://doi.org/10.1002/oca.939

Houska, B., Ferreau, H. J., & Diehl, M. (2011b). An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range. *Automatica*, *47*(10), 2279–2285. https://doi.org/10.1016/j.automatica.2011.08.020

Huber, F., Kondak, K., Krieger, K., Sommer, D., Schwarzbach, M., Laiacker, M., Kossyk, I., Parusel, S., Haddadin, S., & Albu-Schaffer, A. (2013). First analysis and experiments in aerial manipulation using fully actuated redundant robot arm. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3452–3457. https://doi.org/10.1109/IROS.2013.6696848

Jacquet, M., & Franchi, A. (2021). Motor and Perception Constrained NMPC for Torque-Controlled Generic Aerial Vehicles. *IEEE Robotics and Automation Letters*, *6*(2), 518–525. https://doi.org/10.1109/LRA.2020.3045654

Jiang, Q., & Gosselin, C. M. (2009). Maximal Singularity-Free Total Orientation Workspace of the Gough–Stewart Platform. *Journal of Mechanisms and Robotics*, *1*(3). https://doi.org/10.1115/1.3147200

Kawasaki, K., Motegi, Y., Zhao, M., Okada, K., & Inaba, M. (2015). Dual connected Bi-Copter with new wall trace locomotion feasibility that can fly at arbitrary tilt angle. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 524–531. https://doi.org/10.1109/IROS.2015.7353422

Kelly, R., & Moreno, J. (2005). Manipulator motion control in operational space using joint velocity inner loops. *Automatica*, *41*(8), 1423–1432. https://doi.org/10.1016/j.automatica.2005.03.008

Khalil, H. K. (2002). *Nonlinear Systems* (3rd). Prentice Hall.

Khan, W., & Nahon, M. (2013). Toward an Accurate Physics-Based UAV Thruster Model. *IEEE/ASME Transactions on Mechatronics*, *18*(4), 1269–1279. https://doi.org/10.1109/TMECH.2013.2264105

Khayour, I., Cuvillon, L., Butin, C., Yigit, A., Durand, S., & Gangloff, J. (2020). Improving Disturbance Rejection and Dynamics of Cable Driven Parallel Robots with On-board Propellers. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6564–6569. https://doi.org/10.1109/IROS45743.2020.9341033

Khosravi, M. A., & Taghirad, H. D. (2011). Dynamic Analysis and Control of Cable Driven Robots with Elastic Cables. *Transactions of the Canadian Society for Mechanical Engineering*, *35*(4), 543–557. https://doi.org/10.1139/tcsme-2011-0033

Lallman, F., Davidson, J., & Bundick, W. (2001). Integrated reconfigurable control allocation. *AIAA Guidance, Navigation, and Control Conference and Exhibit*. https://doi.org/10.2514/6.2001-4083

Maeder, U., Borrelli, F., & Morari, M. (2009). Linear offset-free Model Predictive Control. *Automatica*, *45*(10), 2214–2222. https://doi.org/10.1016/j.automatica.2009.06.005

Manipulating industrial robots - Performance criteria and related test methods. (2016).

Miermeister, P., Lachele, M., Boss, R., Masone, C., Schenk, C., Tesch, J., Kerger, M., Teufel, H., Pott, A., & Bulthoff, H. H. (2016). The CableRobot simulator large scale motion platform based on cable robot technology. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3024–3029. https://doi.org/10.1109/IROS.2016.7759468

Morari, M., & Maeder, U. (2012). Nonlinear offset-free model predictive control. *Automatica*, *48*(9), 2059–2067. https://doi.org/10.1016/j.automatica.2012.06.038

Moura, S. J., Siegel, J. B., Siegel, D. J., Fathy, H. K., & Stefanopoulou, A. G. (2010). Education on vehicle electrification: Battery Systems, Fuel Cells, and Hydrogen. *2010 IEEE Vehicle Power and Propulsion Conference*, 1–6. https://doi.org/10.1109/VPPC.2010.5729150

Murray, R. M., Sastry, S. S., & Zexiang, L. (1994). *A Mathematical Introduction to Robotic Manipulation* (1st). CRC Press, Inc.

Pannocchia, G., Gabiccini, M., & Artoni, A. (2015). Offset-free MPC explained: novelties, subtleties, and applications. *IFAC-PapersOnLine*, *48*(23), 342–351. https://doi.org/10.1016/j.ifacol.2015.11.304

Park, S., Lee, J., Ahn, J., Kim, M., Her, J., Yang, G.-H., & Lee, D. (2018). ODAR: Aerial Manipulation Platform Enabling Omnidirectional Wrench Generation. *IEEE/ASME*

Transactions on Mechatronics, *23*(4), 1907–1918. https://doi.org/10.1109/TMECH.2018.2848255

Pounds, P. E. I., Bersak, D. R., & Dollar, A. M. (2011). Grasping from the air: Hovering capture and load stability. *2011 IEEE International Conference on Robotics and Automation*, 2491–2498. https://doi.org/10.1109/ICRA.2011.5980314

Ruggiero, F., Lippiello, V., & Ollero, A. (2018). Aerial Manipulation: A Literature Review. *IEEE Robotics and Automation Letters*, *3*(3), 1957–1964. https://doi.org/10.1109/LRA.2018.2808541

Ryll, M., Bicego, D., & Franchi, A. (2016). Modeling and control of FAST-Hex: A fully-actuated by synchronized-tilting hexarotor. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1689–1694. https://doi.org/10.1109/IROS.2016.7759271

Ryll, M., Muscio, G., Pierri, F., Cataldi, E., Antonelli, G., Caccavale, F., & Franchi, A. (2017). 6D physical interaction with a fully actuated aerial robot. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 5190–5195. https://doi.org/10.1109/ICRA.2017.7989608

Sarkisov, Y. S., Kim, M. J., Bicego, D., Tsetserukou, D., Ott, C., Franchi, A., & Kondak, K. (2019). Development of SAM: cable-Suspended Aerial Manipulator *. *2019 International Conference on Robotics and Automation (ICRA)*, 5323–5329. https://doi.org/10.1109/ICRA.2019.8793592

Sharon, A., Hogan, N., & Hardt, D. E. (1993). The macro/micro manipulator: An improved architecture for robot control. *Robotics and Computer-Integrated Manufacturing*, *10*(3), 209–222. https://doi.org/10.1016/0736-5845(93)90056-P

Six, D., Briot, S., Chriette, A., & Martinet, P. (2018). The Kinematics, Dynamics and Control of a Flying Parallel Robot With Three Quadrotors. *IEEE Robotics and Automation Letters*, *3*(1), 559–566. https://doi.org/10.1109/LRA.2017.2774920

SkyCam. (n.d.). https://www.live-production.tv/mobile-production/specialist-cameras/skycam.html

Soto, I., & Campa, R. (2013). Two-Loop Control of Redundant Manipulators: Analysis and Experiments on a 3-DOF Planar Arm. *International Journal of Advanced Robotic Systems*, *10*(1), 85. https://doi.org/10.5772/53515

Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2005). *Robot Modeling and Control*. Wiley. https://books.google.fr/books?id=muCMAAAACAAJ

Spong, M. W. (1989). Adaptive control of flexible joint manipulators. *Systems & Control Letters*, *13*(1), 15–21. https://doi.org/10.1016/0167-6911(89)90016-9

Tognon, M., & Franchi, A. (2017). Landing and Take-off on/from Sloped and Non-planar Surfaces with more than 50 Degrees of Inclination. *9th International Micro Air Vehicles Conference*, 97–102. https://hal.laas.fr/hal-01673466

Tognon, M., & Franchi, A. (2018). Omnidirectional Aerial Vehicles With Unidirectional Thrusters: Theory, Optimal Design, and Control. *IEEE Robotics and Automation Letters*, *3*(3), 2277–2282. https://doi.org/10.1109/LRA.2018.2802544

Tognon, M., & Franchi, A. (2021). *Theory and Applications for Control of Aerial Robots in Physical Interaction Through Tethers* (Vol. 140). Springer International Publishing. https://doi.org/10.1007/978-3-030-48659-4

Tognon, M., Gabellieri, C., Pallottino, L., & Franchi, A. (2018). Aerial Co-Manipulation With Cables: The Role of Internal Force for Equilibria, Stability, and Passivity. *IEEE Robotics and Automation Letters*, *3*(3), 2577–2583. https://doi.org/10.1109/LRA.2018.2803811

Verschueren, R., Frison, G., Kouzoupis, D., van Duijkeren, N., Zanelli, A., Novoselnik, B., Frey, J., Albin, T., Quirynen, R., & Diehl, M. (2019). acados: a modular open-source framework for fast embedded optimal control. http://arxiv.org/abs/1910.13753

Vichik, S., & Borrelli, F. (2014). Solving linear and quadratic programs with an analog circuit. *Computers & Chemical Engineering*, *70*, 160–171. https://doi.org/10.1016/j.compchemeng.2014.01.011

Xiang, S., Gao, H., Liu, Z., & Gosselin, C. (2020). Dynamic transition trajectory planning of three-DOF cable-suspended parallel robots via linear time-varying MPC. *Mechanism and Machine Theory*, *146*, 103715. https://doi.org/10.1016/j.mechmachtheory.2019.103715

Yiğit, A., Arpa Perozo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021a). Improving Dynamics of an Aerial Manipulator with Elastic Suspension Using Nonlinear Model Predictive Control. *2021 IEEE International Conference on Robotics and Au-tomation (ICRA)*.

Yiğit, A., Arpa Perozo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021b). Novel Omnidirectional Aerial Manipulator With Elastic Suspension: Dynamic Control and Experimental Performance Assessment. *IEEE Robotics and Automation Letters*, *6*(2), 612–619. https://doi.org/10.1109/LRA.2020.3048880

Yiğit, A., Arpa Perozo, M., Ouafo, M., Cuvillon, L., Durand, S., & Gangloff, J. (2021). Aerial Manipulator Suspended From a Cable-Driven Parallel Robot: Preliminary Experimental Results. *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Yiğit, A., Grappe, G., Cuvillon, L., Durand, S., & Gangloff, J. (2020). Preliminary Study of an Aerial Manipulator with Elastic Suspension. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 4287–4293. https://doi.org/10.1109/ICRA40945.2020.9196942

Yu Luo, Serrani, A., Yurkovich, S., Doman, D., & Oppenheimer, M. (2004). Model predictive dynamic control allocation with actuator dynamics. *Proceedings of the 2004 American Control Conference*, 1695–1700. https://doi.org/10.23919/ACC.2004.1386823

Zhang, K., Chermprayong, P., Alhinai, T. M., Siddall, R., & Kovac, M. (2017). SpiderMAV: Perching and stabilizing micro aerial vehicles with bio-inspired tensile anchoring systems. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6849–6854. https://doi.org/10.1109/IROS.2017.8206606