

RAM

● ROBOTICS
AND
MECHATRONICS

Achieving Stable and Safe Physical Interaction for a Fully Actuated Aerial Robot using Energy Tank-Based Interaction Control

J.D. (Jelle) Zult

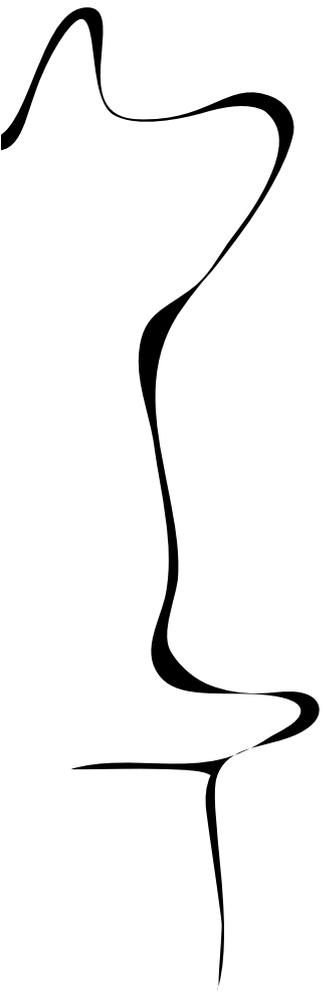
MSC ASSIGNMENT

Committee:

A. Franchi, PhD
R.A.M. Rashad Hashem, MSc
dr. D. Bicego
dr. ir. W.B.J. Hakvoort

May, 2020

013RaM2020
Robotics and Mechatronics
EEMCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands



UNIVERSITY OF TWENTE

Abstract

Faculty of Electrical Engineering, Mathematics and Computer Science
Robotics and Mechatronics

Master of Science

Achieving Stable and Safe Physical Interaction for a Fully Actuated Aerial Robot using Energy Tank-Based Interaction Control

by Jelle ZULT

Recently, there has been a growing focus on utilizing aerial robots for physical interaction tasks. However, controlling this interaction is generally a difficult task, as the physical contact introduces additional dynamics that would require an accurately modeled environment when employing traditional signal-based stability analyses. In practice, such models are hard to obtain at sufficient accuracy, and often not even available up front. Energy-based control techniques, on the other hand, are much better suited for physical interaction: a passive closed loop system enables stable contact with any conceivable passive environment.

Safety is another crucial design criterion for aerial robots interacting with the environment, since passivity alone does not necessarily lead to safe behavior. This becomes apparent in the case of unexpected contact loss during an interaction task, leading to unintended motion and the risk of high impact collisions.

An integrated energy-based solution is presented that tackles both the stability problem and the safety problem for the physical interaction of a fully actuated UAV with the environment. The proposed control system combines impedance control with interaction wrench regulation, while maintaining passivity by using the concept of the energy tank. A port-based approach is used for the controller design and analysis. The augmentation of the energy tank to the rest of the system has been implemented in the port-based framework as well. Unlike signal-based implementations, port-based implementations fit very well with the energy-based paradigm, because the power flows between different subsystems are explicitly modeled. Furthermore, the individual subsystems can be interpreted as physical systems, that can be interconnected to obtain the desired closed loop behavior.

An energy-based approach is used for solving the safety issues that arise when regulating the interaction wrench. A novel concept is introduced, called the *safety extension* of the energy tank. This method detects and responds to safety violations by keeping track of relevant energy levels and power flows associated with the energy tank.

The designed solutions are validated in two different simulation environments. The port-based implementations used for the design are simulated directly in a bond-graph simulation environment, after which passive behavior and a correct functioning of the designed concepts are confirmed. Subsequently, a discrete-time implementation is constructed in C++, such that the validation can be executed in a more practical context and under more realistic conditions of a different simulation environment. The implemented solution remains effective, although the performance slightly decreases.

Contents

Abstract	iii
1 Introduction	1
1.1 Related Work	2
1.1.1 Aerial Physical Interaction	2
1.1.2 Energy Tank Augmentation	4
1.1.3 Safety in Interaction Control	5
1.2 Research Goal	6
1.3 Proposed Method	6
1.4 Thesis Structure	7
2 Background Theory: Energy-Based Modeling and Control	9
2.1 Passivity	9
2.1.1 Control by Interconnection	10
2.2 Port-Based Modeling	10
2.2.1 Bond Graphs	11
2.2.2 Dirac Structure	13
2.3 Twists and Wrenches	14
2.3.1 Twists	15
2.3.2 Wrenches	16
2.3.3 Coordinate Transformations	16
2.3.4 Twists and Wrenches in Bond Graphs	17
2.4 Modeling of the Fully Actuated Hexarotor UAV	17
2.4.1 UAV Actuation Model	17
2.4.2 Dynamic UAV Model	19
3 Energy Tank-Based Wrench/Impedance Control	21
3.1 Impedance Controller	21
3.2 Interaction Wrench Regulation	23
3.2.1 Separation of Concerns: Two Parallel Springs	23
3.2.2 Introducing the J-Frame	24
3.2.3 Stiffness Modulation of the Spatial Spring	25
3.2.4 Spatial Spring Description	25
3.2.5 Interaction Wrench Observer	26
3.3 Passivity Analysis	27
3.4 Energy Tank Augmentation	29
3.4.1 Energy Routing using a Dirac Structure	30
3.4.2 Design of the Energy Tank	32
3.5 Conclusion	33

4	Simulations: Validation of the Energy Tank-Based Controller	35
4.1	Scenario 1: Passivity	36
4.1.1	Description of the Scenario	36
4.1.2	Explanation of the Results	37
4.1.3	Discussion of the Results	37
4.2	Scenario 2: Tank Depletion Behavior	40
4.2.1	Description of the Scenario	40
4.2.2	Explanation and Discussion of the Results	40
4.3	Conclusion	42
5	Towards Safe Interaction: A Safety Extension for Energy Tanks	43
5.1	Existing Solution: Contact Loss Stabilization (CLS)	43
5.1.1	Description of the CLS Algorithm	44
5.1.2	Limitations of the CLS Algorithm	45
5.2	Proposed Solution: A Safety Extension for Energy Tanks	45
5.2.1	Safety Violation Detection	46
5.2.2	Energy Freezing	46
5.2.3	Setpoint Returning	48
5.2.4	Preventing Energy Regeneration After Suspected Contact Loss	49
5.2.5	System Overview	50
5.3	Comparison	50
5.4	Conclusion	53
6	Simulations: Validation of the Safety Extension	55
6.1	Scenario 3: Contact Loss Stabilization	55
6.1.1	Description of the Scenario	55
6.1.2	Explanation of the Results	56
6.1.3	Discussion of the Results	56
6.2	Scenario 4: False Detections - Performance vs Safety	60
6.2.1	Description of the Scenario	60
6.2.2	Explanation and Discussion of the Results	61
6.3	Scenario 5: Rotational Stiction	64
6.3.1	Description of the Scenario	64
6.3.2	Explanation and Discussion of the Results	65
6.4	Conclusion	68
7	Simulations: More Realistic Conditions	69
7.1	Validation of the Energy Tank-Based Controller	70
7.1.1	Explanation and Discussion of the Results	70
7.2	Validation of the Safety Extension	72
7.2.1	Explanation and Discussion of the results	73
7.3	Limitations of the Safety Extension due to Sensor Noise	76
7.4	Conclusion	78
8	Conclusion	79
8.1	Conclusions	79
8.2	Limitations and Recommendations for Future Work	81
A	Table of Simulation Parameters	83

Chapter 1

Introduction

In the last decades, there has been great attention to the wide range of applications of unmanned aerial vehicles (UAVs), which include surveillance, inspection, film-making, mapping, and search and rescue. Due to their agility and mechanical simplicity, conventional multirotor UAVs, like the quadrotor, have shown the potential to offer reliable solutions in each of these applications.

In recent years, there has been a growing focus on utilizing aerial robots for a different set of tasks that require them to physically interact with the environment. This introduces many new challenges that lie on the overlap between aerial robotics and interaction control research. Common objectives include contact stability, safety and adequate interaction performance.

Conventionally, UAVs use signal-based control techniques such as PID motion control, which is effective in free flight, but not suitable for physical interaction. Another example of such a signal-based approach is the use of hybrid wrench/pose control, e.g. in [Park et al. \(2018\)](#), where motion and force control are simultaneously applied to separate directions in the task space. The problem with these techniques is that they require an accurate model of the environment in order to guarantee stability. Such models are often hard to obtain at sufficient accuracy, and moreover, the characteristics of the environment that will be encountered are generally not known up front.

Energy-based control techniques are more suitable for interaction control, as they use passivity of the controlled system as a stability criterion. A passive system can only store or dissipate energy, it cannot create any additional energy. This leads to the possibility of guaranteeing stable interaction any conceivable passive environment ([Stramigioli, 2015](#)).

A well known and widely used example of such a technique is impedance control, which was first introduced in [Hogan \(1985\)](#). The idea of impedance control is to impose the interactive behavior of the robot, rather than independently controlling the position or force. This typically comes down to emulating the behavior of a mass-spring-damper system by introducing virtual spring and damper-like elements, while considering the robot itself as the mass. A major advantage of impedance control is that it causes the robot behave like a passive system, making it suitable for dealing with uncertain environments. Another advantage is that it represents a virtual physical system, such that its interactive behavior can be easily understood by the designer. A disadvantage is its inability to regulate the interaction wrench with the environment. This could be solved by modulating the stiffness of the impedance controller, although that would come at the cost of losing passivity.

The work of this thesis lies in enhancing the traditional impedance control approach, while remaining as much as possible within the energy-based paradigm. By following this approach, stable physical interaction with an unknown environment

can be realized. However, this does not automatically lead to safe interaction. During an interaction task, unexpected contact loss poses a great risk to the UAV, given that it only has limited knowledge about the environment. This work therefore has the additional goal of moving towards safer physical interaction.

The proposed controller combines impedance control with interaction wrench regulation, while maintaining passivity by using the concept of an *energy tank*. This is a tool that can be used for storing a certain limited amount of virtual energy within a control system. This virtual energy can subsequently be *routed* towards subsystems performing control actions that would otherwise require the creation of additional energy. The designed control system will be applied to a fully actuated hexarotor aerial robot. This is a six rotor UAV that can be actuated in each of its six degrees of freedom (DOFs), as opposed to underactuated systems such as quadrotors.

The design of the proposed controller closely follows the energy-based paradigm, by using a port-based implementation for each of its subsystems, including the energy tank. Unlike signal-based implementations, port-based implementations fit very well with the energy-based paradigm, because the power flows between different subsystems are explicitly modeled. The individual subsystems can be interpreted as physical systems, that can be interconnected through *power ports* in order to obtain the desired closed loop behavior. As a result, the analysis and design of the control system becomes more straightforward and intuitive, especially when dealing with energy-based criteria like passivity.

For achieving safe interaction, it is investigated how the proposed control system can be extended for dealing with the safety issues that arise when the interaction wrench with the environment is being regulated. A solution is proposed in the form of a *safety extension* of the energy tank, which detects and responds to safety violations by keeping track of the energy tank state. It is designed to be effective in a multitude of scenarios, without requiring adaptation to the specific environment.

1.1 Related Work

1.1.1 Aerial Physical Interaction

Engaging in physical interaction with the environment using aerial robots is an active field of research that has brought forth many different approaches already, both in terms of the mechanical design of the UAV, as well as the accompanying control system. The following part presents an overview of the state of the art of aerial physical interaction.

Considering the design of the aerial robot for physical interaction, three main approaches can be found in literature. The first approach is to simply attach a rigid (or flexible) tool to a conventional UAV that is under-actuated, for example in [Nguyen and Lee \(2013\)](#); [Pounds and Dollar \(2011\)](#). However, such designs are incapable of exerting a full 6D wrench to the environment, and furthermore suffer from potential stability issues depending on the placement of the tool tip ([Nguyen and Lee, 2013](#)).

In a second widely used approach, a robotic arm is attached to the UAV frame ([Ruggiero et al., 2018](#)), in order to expand its capability in terms of exerting the full desired wrench to the environment. The additional actuation provided by the robotic arm can namely be used to overcome the under-actuation of the UAV itself. However, this also introduces several drawbacks, including a large increase in weight, costs and mechanical complexity. The increased weight subsequently limits the payload, flight time and maximum wrench exertion. Furthermore, achieving satisfactory control over the interaction wrench requires exploitation of the dynamic

coupling between the manipulator and the UAV, which is a complex task that depends on accurate knowledge about both the full dynamic model and the physical states (Ryll et al., 2019).

Recently, an alternative strategy has been gaining attention, which combines a fully actuated UAV with a rigidly attached tool, such that it can be considered as a *flying end effector*, a paradigm proposed by Ryll et al. (2019). Such a design provides the possibility of exerting forces and torques in any direction to the environment, while offering significant improvements regarding the aforementioned drawbacks of the robotic arm approach.

Several approaches have been taken for designing a fully actuated UAV. One main distinction that can be made, is between variable-tilt and fixed-tilt multi-rotors. Variable-tilt multirotors (Ryll et al., 2012, 2016; Kamel et al., 2018) achieve full actuation by controlling tilt angles of different propellers such that the resulting thrust vector can be chosen independent of the orientation of the UAV. However, such systems are considered less suited to respond to instantaneous external disturbances during interaction, due to the additional response time that the tilting of the rotors requires.

Fixed-tilt multirotors (Jiang and Voyles, 2014; Ryll et al., 2017; Rashad et al., 2019) are more suited for interaction control in the sense of being able to respond faster to such external wrenches. Furthermore they offer a mechanically less complex design that can be controlled more straightforwardly. However, compared to variable-tilt multirotors, they are in principle less energetically efficient and generally more limited in terms of the magnitude of the interaction forces they can exert in certain directions.

Different control techniques have been used for physically interacting with the environment. As stated before, both pure motion control (Kamel et al., 2018) and hybrid wrench/pose control techniques (Park et al., 2018) are considered less suited for physical interaction due to their need for an accurate model of the environment in order to guarantee contact stability. Control techniques that are more appropriate for interaction with an unmodeled environment include impedance and admittance control. Admittance control is similar to impedance control in the sense that it also controls the interactive behavior of the robot instead of either its position or force. The difference is that an admittance controller outputs a motion based on a force input, while an impedance controller outputs a force based on a motion input.

In Ryll et al. (2017), in order to achieve the desired interaction behavior of a fully actuated aerial robot, an admittance controller is proposed in combination with a momentum-based interaction wrench observer and an inner loop 6-DOF pose tracking controller. Rashad et al. (2019) proposes the use of an impedance controller, combined with a similar interaction wrench observer, while considering the design and analysis of the system in the energy-based paradigm. For enabling wrench regulation, a wrench tracking control loop is added, augmented by an energy tank to restore the overall passivity of the controlled system. The resulting passivity-based force/impedance controller was based on a control structure designed for a robotic arm (Schindlbeck and Haddadin, 2015), which used a similar energy tank implementation for enabling passive wrench regulation. However, alternative approaches to implementing energy tanks can be found in literature as well, as described in the following subsection.

1.1.2 Energy Tank Augmentation

In the field of interaction control, in most cases the concept of the energy tank is used as a means to passivize an otherwise non-passive control system. In [Schindlbeck and Haddadin \(2015\)](#) an energy tank is used to ensure passivity after a force control loop is introduced alongside an impedance controller. The force control part of the presented force/impedance controller would otherwise violate the system's passivity, through injecting additional energy into the system. In [Ferraguti et al. \(2013, 2015\)](#) variable stiffness impedance control is used to deal with varying tissue stiffness during a surgical needle insertion task by a robotic manipulator. It shows how varying the stiffness would normally cause unstable behavior due to the loss of passivity. Subsequently, energy tank augmentation is shown to successfully restore the passivity of the system, resulting in stable interaction behavior.

Next to passivity, energy tanks have also been considered as a possible way to deal with different higher level control objectives. One important objective could be to guarantee a certain measure of safety. For instance, when explicitly defining a maximum allowed kinetic energy in the system, this limit could be enforced by limiting the maximum tank energy accordingly, as suggested by [Shahriari et al. \(2018\)](#). Another relevant measure for safety, according to [Tadele et al. \(2014\)](#), might be the maximum power that the controller is allowed to inject into the system. This power limit could be enforced using energy tanks as well, for instance by using the valve-based energy tank approach introduced by [Shahriari et al. \(2018\)](#). Another higher level objective that energy tanks might be used for, is the enforcement of task priorities. This can be achieved by limiting the power flow of certain ports connected to the energy tank, as demonstrated in [Shahriari et al. \(2018\)](#).

Signal-Based vs Port-Based Energy Tanks

Within the application of energy tanks for interaction control, there have been several different approaches to their actual implementation. One notable difference is the framework in which the tank has been implemented. Examples of purely signal based implementations on the one hand can be found in [Schindlbeck and Haddadin \(2015\)](#); [Rashad et al. \(2019\)](#). In short, the power flowing into (or out of) the energy tank is computed based on several control signals and velocity feedback signals. Then, based on the energy level in the tank, the control signals that extract energy from the tank can be turned off or on accordingly. This signal-based approach has been shown to work in terms of passifying the system in [Rashad et al. \(2019\)](#). However, it does not fit elegantly with the port-based framework that has been used for the design of the rest of the controller. This prevented the full benefits of the port-based approach to be reaped, such as the otherwise intuitive and straightforward analysis that it offers regarding the energy flows.

Alternatively, the works of [Dietrich et al. \(2017\)](#); [Tadele et al. \(2014\)](#); [Raiola et al. \(2018\)](#) have shown how port-based implementations can be realized. In such an approach, the energy tank is considered as a part of the virtual physical system, being directly connected to the rest of the system through power ports and interconnection structures. This guarantees power continuity, and forces the designer to explicitly consider where the energy flows in the designed system. Furthermore, this approach offers an elegant fit of the energy tank in a port-based framework. As a result, such port-based implementations tend to be more modular and independent from the specific interaction controller that is used. In the works of [Tadele et al. \(2014\)](#); [Raiola et al. \(2018\)](#) the energy tank is modeled as a physical storage element

(e.g. a spring), connected to the rest of the physical system through a modulated transformer. The modulation is computed in a way that causes the desired actuator force to be achieved as a natural consequence of the resulting power connection. In [Dietrich et al. \(2017\)](#) a Dirac structure is used for the interconnection instead. Such a Dirac structure is a power continuous element that can be used to route the energy flow between multiple power ports as desired.

Global vs Local Energy Tanks

Another distinction on the implementation is made in [Dietrich et al. \(2017\)](#), namely between global and local energy tanks. On the one hand, a single global energy tank can be used to keep track of the total energy injected into the system, and enforce passivity on a global level. On the other hand, for instance in the context of different task hierarchy levels, local energy tanks can be used to keep track of the injected energy on each hierarchy level independently. This can be valuable if one wishes to limit the available energy for each task separately. Certain tasks, after all, could be potentially less safe, while other tasks might need more performance and therefore more available energy.

In [Groothuis et al. \(2018\)](#) distributed energy budgets for each actuator have been proposed. This is another example of local energy tanks, although here instead of distributing the energy among tasks, the energy is distributed among actuators. This approach is referred to as *implementation at the actuation level*, which has the advantage of eliminating problems like delays or communication loss, which affect the accuracy of the measured energy flows and therefore potentially violate passivity. It must however be noted that implementing energy tanks at the actuation level would not be suitable for UAV control. This is because, for instance, the action of hovering alone would already require a steady stream of energy flowing to the propeller actuators. Continuously allocating the exact amounts of energy required by each actuator would be practically impossible.

1.1.3 Safety in Interaction Control

In [Schindlbeck and Haddadin \(2015\)](#) it is argued how the application of significant interaction forces to the environment may lead to unsafe behavior, in case of an unexpected loss of contact with this environment. When dealing with impedance control, this can be seen as a rapid release of potential energy from the virtual spring into the system as kinetic energy. Even if an energy tank were implemented, this energy flow would continue until the tank depletes. Therefore, the passivity restoration that is offered by the energy tank does not necessarily prevent unsafe behavior.

Safety-Aware Impedance Control

In order to achieve generally safe behavior of a robotic system in the vicinity of humans, [Tadele et al. \(2014\)](#); [Raiola et al. \(2018\)](#) present an approach that limits both the total energy in the system, as well as the power injected into the system by the controller. Such limits can then directly be related to common safety metrics, such as the Maximum Power Index ([Newman et al., 2000](#)). Subsequently, these energy and power limits are enforced by introducing a safety layer, which observes both the total energy in the system and the power injected by the controller, and limits these by modulating the stiffness and damping matrices of an impedance controller. The overall control framework is augmented with an energy tank in order

to maintain passivity that would otherwise be lost due to the stiffness modulation. However, this means that an empty energy tank would lead to a complete shut-down of the controller. Therefore, implementing such a control framework on a UAV would be unsafe, as it would simply fall down as soon as the energy tank depletes, whereas robotic arms typically remain stationary when the entire controller is shut down. Furthermore, the controller presented in [Raiola et al. \(2018\)](#) only consists of an impedance controller, which lacks the ability to regulate the interaction wrench with the environment. On the contrary, this thesis focuses on the safety issues that arise when this interaction wrench is actually being controlled.

Contact Loss Stabilization

In [Schindlbeck and Haddadin \(2015\)](#) a solution to the safety problem of interaction wrench control is offered in the form of a contact loss stabilization (CLS) algorithm. This solution goes one step further than simply switching on and off the force controller based on whether or not contact is detected at each moment in time. The proposed algorithm is argued to be more robust to sensor noise, which otherwise would lead to rapid switching behavior of the force controller, badly impacting the force tracking performance. However, the proposed algorithm implicitly assumes accurate geometrical knowledge of the environment that the end effector interacts with. Furthermore, it is not completely clear whether the geometrical approach that is taken would be effective and elegant in multi-DOF scenarios with more uncertainty.

1.2 Research Goal

In this thesis, an effort is made towards achieving both stable and safe physical interaction of a fully actuated UAV with an unmodeled environment. Special attention is given to designing a solution that complies completely with the energy-based paradigm. Regarding safe interaction, it is desirable to find a solution that is applicable to a large variety of scenarios, without requiring accurate information about the environment. Furthermore, robust wrench tracking capabilities should be maintained during normal operation. These objectives have led to the formulation of the following research questions:

1. How can stable physical interaction with an unmodeled environment be achieved for a fully actuated aerial robot?
2. How can the safety issues be dealt with that arise when regulating the interaction wrench with the environment, without compromising wrench tracking performance during normal operation?
3. How can the complete design be realized as much as possible within the energy-based paradigm?

1.3 Proposed Method

In this thesis, an integrated energy-based solution is presented that tackles both the stability problem and the safety problem for the physical interaction of a fully actuated UAV with the environment. The complete design is summarized by the diagram in figure [1.1](#).

A passive closed loop system has been realized, such that stable interaction can be guaranteed with any conceivable passive environment. The proposed control system combines impedance control with interaction wrench regulation, while maintaining passivity by using the concept of the energy tank. A port-based implementation is presented for the augmentation of the energy tank to the wrench/impedance controller.

Unlike signal-based implementations, port-based implementations fit very well with the energy-based paradigm, because the power flows between different subsystems are explicitly modeled. The individual subsystems can be interpreted as physical systems, that can be interconnected to obtain the desired closed loop behavior. These properties greatly help during the analysis and design of the control system, especially when dealing with energy-based criteria like passivity. For example, analyzing the passivity of a complex system becomes much easier when one can simply analyze the passivity of each of its subsystems separately.

For achieving safe interaction, a novel concept is introduced called the *safety extension* of the energy tank. This system detects and responds to safety violations by keeping track of relevant energy levels and power flows, and is tightly integrated with the presented port-based implementation of the energy tank. Its operating principles have been derived from generalizing the contact loss stabilization algorithm presented in [Schindlbeck and Haddadin \(2015\)](#), by expressing it in terms of energy. As a result, the algorithm can be applied to many different unsafe scenarios, without requiring any manual adaptation.

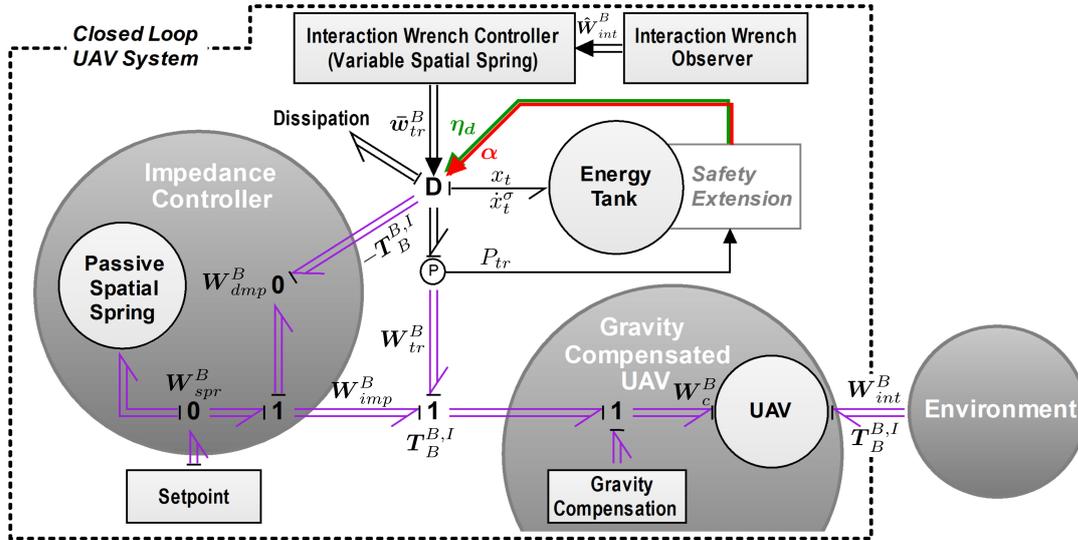


FIGURE 1.1: Overview of the proposed control system, represented using the bond graph notation (see §2.2)

1.4 Thesis Structure

The content of this thesis is divided into six main chapters: Chapter 2 provides a structured overview of the most relevant concepts from the energy-based modeling and control paradigm, such that later discussions and choices can be better understood. Chapter 3 presents the first part of the proposed solution, which aims at achieving stable physical interaction for a fully actuated UAV with an unmodeled environment. The resulting design is validated in chapter 4 through simulations.

Chapter 5 describes the second part of the proposed solution, which aims to improve safety during physical interaction by introducing a novel concept called the *safety extension* of the energy tank. Its effectiveness and versatility are evaluated in simulation in chapter 6. Finally, chapter 7 describes the results of validation simulations performed under more realistic conditions, which have been used as a replacement to the originally planned experiments on the real hexarotor UAV. These experiments had to be cancelled due to a malfunctioning of the UAV, followed by the recent measures regarding the coronavirus outbreak.

Chapter 2

Background Theory: Energy-Based Modeling and Control

In many applications, robots are required to physically interact with their environment. However, controlling this interaction is often a difficult task, as the dynamics of the combined robot-environment system determines whether it is stable or not. Proving stability using a traditional signal-based approach, requires an accurate model of both the robot and the environment. Physical contact models that are sufficiently accurate are typically very difficult to obtain. Moreover, the characteristics of the encountered environment are often not even known a priori. Energy-based control techniques, on the other hand, are more suitable for interaction control with unknown or uncertain environments.

Therefore, within this thesis, special attention is given to realizing a control system that complies as much as possible with the energy-based paradigm. As a result, several concepts from this paradigm have been used throughout this work.

In this chapter, the goal is to give a structured overview of the most relevant concepts, such that later discussions and choices made can be better understood. In §2.1, the importance of passivity is explained in the context of interaction control with an unknown environment. Also, the approach of *control by interconnection* has been outlined. In §2.2, an explanation is given on the merits of the port-based approach for modeling and control. Additionally, as the *bond graph* representation is used throughout this thesis, an introductory explanation is given on the used notation. Another concept used within the port-based modeling paradigm, called the *Dirac Structure*, is introduced here as well. Furthermore, in this thesis rigid body kinematics and dynamics have been described by means of twists and wrenches, about which an overview is presented in §2.3, as well as a way to use them in bond graphs. Finally, the dynamic model of the fully actuated hexarotor that is assumed in this work is described in §2.4, including its bond graph representation.

2.1 Passivity

In the context of physical interaction, passivity-based control is an effective means of establishing stable physical contact with unknown environments.

Passivity is a property that naturally holds for physical dynamic systems. Intuitively speaking, a passive system can be defined as a system that can only store or dissipate energy, and thus it cannot produce any additional energy. In other words, the energy within a passive system can only increase when this is externally supplied by the environment that it interacts with. Therefore, when assuming a passive environment, the total energy in the combined system of a passive robot interacting

with this environment can never increase. Thus, passivity allows for stable interaction with any conceivable passive environment. In fact, it is shown in [Stramigioli \(2015\)](#) to be even a necessary condition in that case: for any non-passive controller, one can always conceive a passive environment that results in unbounded behavior after interconnecting the two.

2.1.1 Control by Interconnection

There are multiple ways to design a controller such that the closed loop system becomes passive. One strategy follows from recognizing that two passive systems connected through a power port will result in a new passive system. The physical system to be controlled is already passive, and the controller could be designed to behave as a second physical system that would result in the desired behavior. This approach is referred to as *control by interconnection*.

Impedance control, first introduced in [Hogan \(1985\)](#), is a widely used technique that follows this principle. The goal in impedance control is to make the controlled system appear as a physical system to the environment, exhibiting a certain desired impedance. This means that the interactive behavior of the closed loop system is imposed, instead of imposing either its motion or exerted force. The imposed behavior is typically that of a (multi-DOF) mass-spring-damper system, which could be implemented on a robot by connecting a virtual spring and damper, while keeping the felt inertia unaltered.

Because it imposes the interactive behavior, impedance control never leads to an ill-posed control problem, as opposed to force control or position control, where this depends on the environment ([Folkertsma and Stramigioli, 2017](#)). Furthermore, if it is implemented following the *control by interconnection* approach, the impedance controller can be regarded as a physical system, allowing the designer to easily understand its interactive behavior. On the other hand, interaction tasks often require the robot to exert a certain wrench on the environment. On its own, impedance control does not offer a precise regulation of this interaction wrench. To achieve the desired wrench, the stiffness of the impedance controller could for instance be modulated, but this would compromise the passivity guarantee that made impedance control so interesting in the first place.

2.2 Port-Based Modeling

Physical systems can be modeled effectively and elegantly within the port-based framework. In this approach, interactions between (sub-)systems are modeled by means of power ports, which explicitly define the energetic relation between each of the systems. For example, the interaction between a robotic end effector and the environment could be modeled in terms of energy exchange, by means of a (multi-dimensional) connection between two power ports: the end effector port and the environment port. In the same way, when the choice is made to design the controller as a physical system, the interaction between the controller and the robot through the actuators could be modeled by power ports as well.

The port-based approach is used throughout this thesis, allowing for a straightforward and intuitive design and analysis of the proposed control system. Specifically, the bond graph notation has been used, of which the essentials needed for understanding the presented work have been explained below.

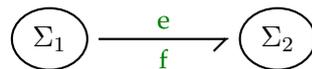
2.2.1 Bond Graphs

A suitable way for graphically representing a physical system in the port-based framework is the bond graph notation. A bond graph (Paynter, 1960) is a graphical description of a physical dynamic system. Furthermore, it is domain independent and highly modular, such that physical models across domains (e.g. electrical, mechanical, thermal, hydraulic) can be integrated seamlessly.

In this thesis, bond graphs have been used to represent the different dynamic systems that have been designed using the port-based approach. Here, only the aspects will be discussed, that are needed for understanding the presented work. To get a more solid understanding of the bond graph notation and its merits, the reader is referred to Gawthrop (1991) for a brief introduction, or to Breedveld (2008) for a complete and in-depth overview.

Bonds/Ports

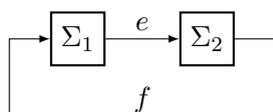
The most fundamental element that a bond graph consists of, is the *bond*, represented by the half-arrow. A bond represents the energy connections between different (sub)systems, where the direction of the arrow indicates the assumed positive direction of the power flow.



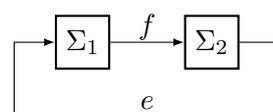
Each bond carries the dual variables referred to as the *effort* (e) and the *flow* (f). These are the domain-independent generalization of e.g. a force and a velocity (mechanical domain) or a voltage and a current (electrical domain). The dual product of the effort and flow always results in the power that flows through the associated bond.

Causality

A perpendicular stroke is placed either at the base or at the tip of the bond, to indicate *causality*. Causality indicates which variable (effort or flow) 'causes' the other within a system or element. The location of the stroke defines the causal direction of the effort, which simultaneously defines the flow causality in the opposite direction. In case, for instance, the system is a moving mass on which a force is applied, the velocity (flow) of the moving mass follows from the integration of the applied force (effort) over time. Therefore, such a system would have an effort-in causality.



(A) Effort-in causality for system 2 (Σ_2), with the equivalent block diagram below.



(B) Effort-in causality for system 1 (Σ_1), with the equivalent block diagram below.

Single-port elements

A part of the elementary physical behaviors can be modeled using single-port elements, which include *sources* (**Se** and **Sf**), *storage* elements (**C** and **I**), and the dissipative *resistor* element (**R**). These elements define a *constitutive relation* between the effort and flow of the bond that it is connected to.

A source of effort (**Se**) imposes a certain effort (e.g. a voltage source), while a source of flow (**Sf**) imposes a certain flow (e.g. a current source). Therefore, these elements have a fixed causality associated to them:

$$\text{Se} \longrightarrow \nearrow \qquad \text{Sf} \longleftarrow \nearrow$$

A **C**-type storage element (e.g. a capacitor or a spring) requires a flow-in causality, while an **I**-type storage element (e.g. an inductor or a mass) requires an effort-in causality. In the linear case, these have the following constitutive relations:

$$e(t) = \frac{1}{C} \int f(t) dt \qquad f(t) = \frac{1}{I} \int e(t) dt$$

$$\longrightarrow \nearrow \text{I} \qquad \longleftarrow \nearrow \text{C}$$

A resistor element (e.g. viscous friction) instantaneously relates an effort to a flow, such that both causalities are allowed. In the linear case, it has the following constitutive relation:

$$e(t) = R \cdot f(t) \qquad f(t) = \frac{1}{R} e(t)$$

$$\longrightarrow \nearrow \text{R} \qquad \longleftarrow \nearrow \text{R}$$

0- and 1-junctions

In order to define how the different elements are interconnected, the so-called **0**- and **1**-junctions are used. These junctions are power-continuous multi-port elements, meaning that the sum of incoming powers is zero, and any number of bonds can be connected to them. A **1**-junction sets all the flows of the connected bonds equal, while a **0**-junction sets all the efforts of the connected bonds equal. To maintain power-continuity, in case of a **1**-junction, the signed sum of all efforts must equal zero. For a **0**-junction, the signed sum of all flows must equal zero.

$$\begin{array}{ccc} \begin{array}{c} e_2 \uparrow \\ \text{---} \text{1} \text{---} \\ f_1 \leftarrow \quad \rightarrow f_3 \end{array} & & \begin{array}{c} e_2 \uparrow \\ \text{---} \text{0} \text{---} \\ f_1 \leftarrow \quad \rightarrow f_3 \end{array} \\ f_1 = f_2 = f_3 & & e_1 = e_2 = e_3 \\ e_1 - e_2 - e_3 = 0 & & f_1 - f_2 - f_3 = 0 \end{array}$$

Note that the causality is not defined up front, but follows from the specific combination of elements and/or subsystems that will be connected to the junction. After causality is assigned, a one junction must have exactly one flow input (defining all the other flows), and a zero junction must have exactly one effort input (defining all the other efforts).

Two-port elements

The last two fundamental elements used in bond graphs are the transformer **TF** and the gyrator **GY** (labeled **MTF** and **MGY** if they are modulated over time). These are power-continuous two-port elements, that establish a relation between the efforts and flows of the two connected bonds.

An example of a transformer in the mechanical domain is an ideal lever, which relates the force and velocity of one side with the force and velocity of the other side, through the lever arm length. An example of a gyrator, is an ideal electric motor, which relates the incoming current to an applied torque, and angular velocity to a voltage. In terms of efforts and flows, this yields the following relations:

$$\begin{array}{ccc}
 \begin{array}{c} e_1 \\ \hline f_1 \end{array} \nearrow \text{TF} \begin{array}{c} e_2 \\ \hline f_2 \end{array} \\
 \quad \quad \quad r
 \end{array}
 &
 &
 \begin{array}{c} e_1 \\ \hline f_1 \end{array} \nearrow \text{GY} \begin{array}{c} e_2 \\ \hline f_2 \end{array} \\
 \quad \quad \quad r
 \end{array}$$

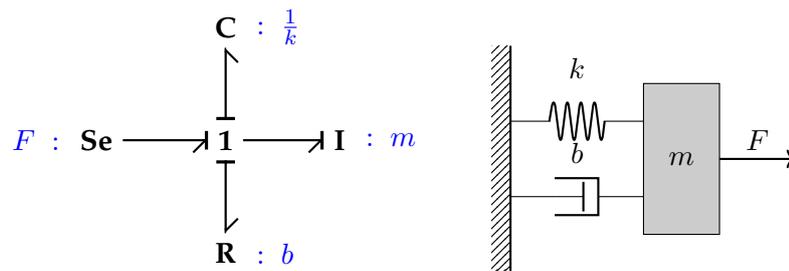
$$\begin{array}{ccc}
 f_2 = r \cdot f_1 & & e_1 = r \cdot f_2 \\
 e_1 = r \cdot e_2 & & e_2 = r \cdot f_1
 \end{array}$$

where r is the transformation ratio, which may be time dependent. Furthermore, for a transformer the causality remains in the same direction, while for a gyrator the causality 'flips'. The positive power direction will remain the same in both cases.



Example: Mass-spring-damper system

At this point, a simple physical system could be constructed by combining the various elements, and interconnecting them through junctions. The following example shows the bond graph representation of a simple mass-spring-damper system:



Computationally speaking, the spring force, damper force, and external force are summed in the 1-junction, and then applied to the inertial element, which integrates the incoming force to arrive at its internal state (momentum). The inertial element then returns the associated velocity (flow), which is 'distributed' to the other elements through the 1-junction, based on which their respective forces can be computed.

2.2.2 Dirac Structure

A Dirac structure is a concept used within the port-based modeling paradigm, to define the interconnection structure between different ports in a generalized way. It is a power-continuous multi-port element, that can be seen as a generalization of the

transformer, the gyrator, and the 0- and 1-junction. While the transformer relates flows to flows (and efforts to efforts), and the gyrator flows to efforts, the Dirac structure can relate different combinations of flows and efforts at the same time.

As a result, it enables routing the energy flow between the different connected ports in any conceivable way, as long as no energy is stored or dissipated. Mathematically, this can be represented as a matrix multiplication, where the different corresponding efforts and flows are ordered from top to bottom. The Dirac structure can represent a transformer as follows, using an inwards positive power direction:

$$\begin{pmatrix} e_1 \\ -f_2 \end{pmatrix} = \begin{bmatrix} 0 & r \\ -r & 0 \end{bmatrix} \begin{pmatrix} f_1 \\ e_2 \end{pmatrix}$$

$$\begin{array}{c} \xrightarrow[e_1]{f_1} \text{TF} \xrightarrow[f_2]{e_2} \\ \quad \quad \quad r \end{array} \qquad \begin{array}{c} \xrightarrow[e_1]{f_1} \text{D} \xleftarrow[-f_2]{e_2} \text{0} \xrightarrow[f_2]{e_2} \\ \quad \quad \quad r \end{array}$$

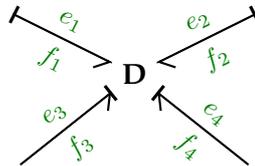
In the same way, the gyrator can be represented by a Dirac structure as well:

$$\begin{pmatrix} e_1 \\ -e_2 \end{pmatrix} = \begin{bmatrix} 0 & r \\ -r & 0 \end{bmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

$$\begin{array}{c} \xrightarrow[e_1]{f_1} \text{GY} \xrightarrow[f_2]{e_2} \\ \quad \quad \quad r \end{array} \qquad \begin{array}{c} \xrightarrow[e_1]{f_1} \text{D} \xleftarrow[-e_2]{f_2} \text{1} \xrightarrow[f_2]{e_2} \\ \quad \quad \quad r \end{array}$$

In both cases, the matrix turns out to be skew-symmetric. In fact, when the positive power direction of all bonds is inwards (or all outwards), having a skew-symmetric matrix yields power continuity. Therefore, as long as the skew-symmetry holds, any conceivable routing of the energy can be realized by manipulating the elements of this matrix accordingly. The following example shows a possible implementation of a four-port Dirac structure, where the connected ports may be of different dimensions as well.

$$\begin{pmatrix} e_1 \\ e_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{bmatrix} 0 & -C & 0 & A^T \\ C^T & 0 & I & B(t) \\ 0 & -I & 0 & 0 \\ -A & -B^T(t) & 0 & 0 \end{bmatrix} \begin{pmatrix} f_1 \\ f_2 \\ e_3 \\ e_4 \end{pmatrix}$$



2.3 Twists and Wrenches

Screw theory uses twists and wrenches for describing the motion and dynamics of rigid bodies. There, a twist is the six-dimensional generalization of the velocity of a rigid body, and a wrench is the six-dimensional generalization of the forces applied to it. The following section lays out the basic rules for working with twists and wrenches. For a complete explanation on screw theory, the reader is referred to [Stramigioli and Bruyninckx \(2001\)](#).

2.3.1 Twists

The configuration of a rigid body can be described by the homogeneous matrix, which belongs to the special Euclidean group $SE(3)$.

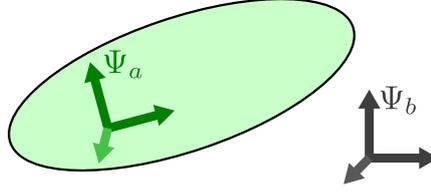


FIGURE 2.2: A rigid body with frame Ψ_a attached, and a separate second frame Ψ_b .

In the example in figure 2.2, the configuration of the frame attached to the rigid body (Ψ_a) expressed in Ψ_b is then defined as follows:

$$\mathbf{H}_a^b = \begin{pmatrix} \mathbf{R}_a^b & \mathbf{p}_a^b \\ \mathbf{0} & 1 \end{pmatrix} \quad (2.1)$$

where $\mathbf{R}_a^b \in SO(3)$ is the rotation matrix describing the orientation of Ψ_a expressed in Ψ_b , and $\mathbf{p}_a^b \in \mathbb{R}^3$ is the position of the origin of Ψ_a expressed in Ψ_b .

The six-dimensional generalized velocity of this rigid body can then be described by a twist, $\tilde{\mathbf{T}} \in se(3)$, which can be found in the following two ways:

$$\begin{aligned} \tilde{\mathbf{T}}_a^{a,b} &= \mathbf{H}_b^a \dot{\mathbf{H}}_a^b \\ \tilde{\mathbf{T}}_a^{b,b} &= \dot{\mathbf{H}}_a^b \mathbf{H}_b^a \end{aligned} \quad (2.2)$$

where $\tilde{\mathbf{T}}_a^{a,b}$ is the twist of Ψ_a w.r.t. Ψ_b expressed in Ψ_a , and $\tilde{\mathbf{T}}_a^{b,b}$ is the same twist expressed in Ψ_b . The resulting twist always has the following form:

$$\tilde{\mathbf{T}} = \begin{pmatrix} \tilde{\omega} & \mathbf{v} \\ \mathbf{0} & 0 \end{pmatrix} \quad (2.3)$$

where \mathbf{v} represents a linear velocity (different from $\dot{\mathbf{p}}$) and $\tilde{\omega}$ is the angular velocity of the rigid body in the *tilde form*, which is defined as follows:

$$\tilde{\omega} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \quad (2.4)$$

The twist can subsequently also be represented using the following column vector:

$$\mathbf{T} = \begin{pmatrix} \omega \\ \mathbf{v} \end{pmatrix} \quad (2.5)$$

Furthermore, one can show that the twist indeed describes a generalized velocity of a rigid body. When introducing an additional frame Ψ_c that is also attached to the rigid body (see figure 2.3), then the twists of both of these frames w.r.t. Ψ_b are identical.

$$\dot{\mathbf{H}}_c^a = 0 \quad \rightarrow \quad \mathbf{T}_a^{a,b} = \mathbf{T}_c^{a,b} \quad (2.6)$$

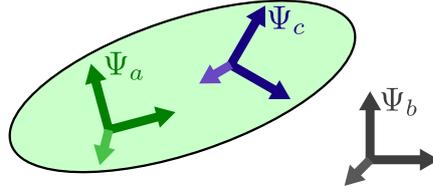


FIGURE 2.3: A rigid body with coordinate frames Ψ_a and Ψ_c attached, and a separate third coordinate frame Ψ_b .

2.3.2 Wrenches

Wrenches are a six-dimensional generalization of the forces and torques applied to a rigid body. They are defined as the dual of the twist, such that their dual product yields power:

$$P = (\mathbf{W}^a)^T \mathbf{T}_c^{a,b} \quad (2.7)$$

Note that the wrench and twist must be expressed in the same coordinate system for this to hold. Furthermore, because the wrench is a co-vector (being the dual of the vector describing the twist), it is conventionally represented by a row vector. However, in this thesis, it turned out much more practical to represent the wrench by a column vector, so this convention has been used instead. Its six-dimensional representation then turns out to be the following:

$$\mathbf{W} = \begin{pmatrix} \boldsymbol{\tau} \\ \mathbf{f} \end{pmatrix} \quad (2.8)$$

where $\boldsymbol{\tau}$ represents a torque and \mathbf{f} a force.

2.3.3 Coordinate Transformations

The way to change the coordinate frame in which the twist is expressed can be found by combining the expressions in eq. 2.2 and making use of $(\mathbf{H}_a^b)^{-1} = \mathbf{H}_b^a$:

$$\dot{\mathbf{H}}_a^b = \mathbf{H}_a^b \tilde{\mathbf{T}}_a^{a,b} = \tilde{\mathbf{T}}_a^{b,b} \mathbf{H}_a^b \quad (2.9)$$

$$\tilde{\mathbf{T}}_a^{a,b} = \mathbf{H}_b^a \tilde{\mathbf{T}}_a^{b,b} \mathbf{H}_a^b \quad (2.10)$$

Performing the same coordinate transformation on the vector representation of the twist can be done by using the *adjoint matrix*, which is formulated as follows:

$$\text{Ad}_{\mathbf{H}_b^a} = \begin{bmatrix} \mathbf{R}_b^a & \mathbf{0} \\ \tilde{\mathbf{p}}_b^a \mathbf{R}_b^a & \mathbf{R}_b^a \end{bmatrix} \quad (2.11)$$

Then the coordinate transformation for the vector representation of the twist can be done in the following way:

$$\mathbf{T}_a^{a,b} = \text{Ad}_{\mathbf{H}_b^a} \mathbf{T}_a^{b,b} \quad (2.12)$$

The way to apply the same coordinate transformation to the wrench can be found by recognizing that the power should not depend on the chosen coordinate frame:

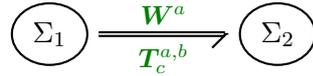
$$P = (\mathbf{W}^a)^T \mathbf{T}_a^{a,b} = (\mathbf{W}^b)^T \mathbf{T}_a^{b,b} \quad (2.13)$$

By combining equations 2.12 and 2.13, the following expression can be obtained for changing coordinates of the wrench:

$$\mathbf{W}^a = \text{Ad}_{\mathbf{H}_a}^T \mathbf{W}^b \quad (2.14)$$

2.3.4 Twists and Wrenches in Bond Graphs

Within bond graphs, the representations of twists (\mathbf{T}) and wrenches (\mathbf{W}) in \mathbb{R}^6 can be considered flows and efforts respectively, as their dual product yields power. So-called *multi-bonds*, represented by a double arrow, are used to indicate that the dual variables are multi-dimensional.



Coordinate transformations can be implemented by using the six-dimensional MTF_6 element, which uses the appropriate adjoint matrix for performing this transformation:



$$\mathbf{T}_c^{b,b} = \text{Ad}_{\mathbf{H}_a}^b \mathbf{T}_c^{a,b}, \quad \mathbf{W}^a = \text{Ad}_{\mathbf{H}_a}^T \mathbf{W}^b$$

2.4 Modeling of the Fully Actuated Hexarotor UAV

The fully actuated hexarotor that is considered in this thesis is shown in figure 2.4, including the different coordinate frames that are used for its mathematical description below. It consists of six propellers located at the vertices of a planar hexagon, which have been tilted by a fixed angle in order to obtain full actuation. The body-fixed frame Ψ_B is attached to the UAV's center of mass, with the x and y axes lying in the plane, and the z axis aligned with the local upward direction. The frame Ψ_E is attached to the end effector, and has the same orientation as Ψ_B . Furthermore, Ψ_I denotes an inertial frame. Lastly, to each of the six propellers a coordinate frame Ψ_{p_i} has been associated, where i refers to the i -th propeller.

2.4.1 UAV Actuation Model

Following the model used in Rashad et al. (2019), the applied control wrench $\bar{\mathbf{W}}_c^B$ depends on the thrust generated by each of the propellers. Each propeller frame Ψ_{p_i} is defined by the position of its origin $\xi_{p_i}^B$ and the rotation matrix $\mathbf{R}_{p_i}^B$. The origin locations are defined as followed:

$$\xi_{p_i}^B := \mathbf{R}_z(\psi_i) [L \ 0 \ 0]^T \quad (2.15)$$

The rotation matrix for each propeller has been set as follows:

$$\mathbf{R}_{p_i}^B := \mathbf{R}_z(\psi_i) \mathbf{R}_x(\alpha_i) \quad (2.16)$$

where α_i determines the fixed tilt of the i -th propeller. This tilt is alternated between a positive and negative value, as defined by the following expression:

$$\alpha_i = (-1)^{i+1} \alpha^* \quad (2.17)$$

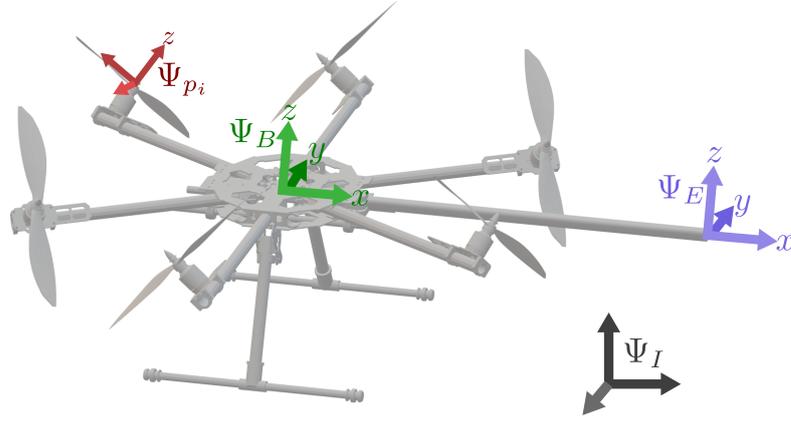


FIGURE 2.4: Schematic overview of the fully actuated hexarotor and the different defined coordinate frames.

where α^* denotes the chosen fixed tilt angle, which is set at $\alpha^* = 47^\circ$.

Each propeller generates a thrust, of which the magnitude is denoted by $\lambda_i \in \mathbb{R}^+$. Next to the thrust itself, a drag torque can also be associated to each propeller:

$$\tau_{d,i} = \gamma \sigma_i \lambda_i \quad (2.18)$$

where γ denotes the drag-to-thrust ratio that depends on the specific propeller that is used, and σ_i describes the rotation direction of the i -th propeller.

The combined effect of the thrust vectors and drag torques that each propeller generates, results in the following actual control wrench:

$$\bar{\mathbf{W}}_c^B = \begin{pmatrix} \tau_c^B \\ \mathbf{f}_c^B \end{pmatrix} = \sum_i \lambda_i \begin{pmatrix} \boldsymbol{\xi}_{p_i}^B \wedge \mathbf{u}_i + \gamma \sigma_i \mathbf{u}_i \\ \mathbf{u}_i \end{pmatrix} =: \mathbf{M} \boldsymbol{\lambda} \quad (2.19)$$

where $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_6]^T$ and \mathbf{u}_i stands for the direction of the thrust generated by the i -th propeller, which is computed as:

$$\mathbf{u}_i := \mathbf{R}_{p_i}^B [0 \ 0 \ 1]^T \quad (2.20)$$

The matrix \mathbf{M} in eq. 2.19 will be referred to as the control allocation matrix. Furthermore, the thrust magnitudes $\boldsymbol{\lambda}$ can saturate as it is upper and lower bounded:

$$\lambda_i = \begin{cases} 0 & \text{if } \lambda_{des,i} < 0 \\ \lambda_{max} & \text{if } \lambda_{des,i} > \lambda_{max} \\ \lambda_{des,i} & \text{else} \end{cases} \quad (2.21)$$

where λ_{max} is the maximum propeller thrust and $\lambda_{des,i}$ is the desired thrust for the i -th propeller. Also note that it is now assumed that the desired thrust can be generated instantly, so additional (e.g. motor) dynamics are neglected. The desired thrusts are computed by using the control allocation matrix:

$$\boldsymbol{\lambda}_{des} = \mathbf{M}^{-1} \mathbf{W}_c^B \quad (2.22)$$

where \mathbf{W}_c^B denotes the controller output wrench (as opposed to the *actual* control

wrench $\bar{\mathbf{W}}_c^B$). Therefore, when considering this actuation model, the controller output wrench equals the actual control wrench if the actuation limits are not exceeded:

$$0 \leq \lambda_{des,i} \leq \lambda_{max} \quad \forall i \in [1, \dots, 6] \quad \rightarrow \quad \bar{\mathbf{W}}_c^B = \mathbf{M}\mathbf{M}^{-1}\mathbf{W}_c^B = \mathbf{W}_c^B \quad (2.23)$$

Also note that this simplification assumes exact knowledge of the control allocation matrix.

2.4.2 Dynamic UAV Model

The UAV itself is modeled as a rigid body in SE(3) with mass m and constant mass moment of inertia matrix $\mathbf{J} \in \mathbb{R}^{3 \times 3}$ expressed in Ψ_B . The resulting dynamic model can be represented by the following equations of motion (Rashad et al., 2019):

$$\begin{aligned} \dot{\mathbf{H}}_B^I &= \mathbf{H}_B^I \tilde{\mathbf{T}}_B^{B,I} \\ \mathcal{I} \dot{\mathbf{T}}_B^{B,I} &= \tilde{\mathbf{P}}^B \mathbf{T}_B^{B,I} + \mathbf{W}_g^B + \bar{\mathbf{W}}_c^B + \mathbf{W}_{int}^B \end{aligned} \quad (2.24)$$

where \mathbf{W}_g^B denotes the wrench applied by gravity, $\bar{\mathbf{W}}_c^B$ is the aforementioned actual control wrench, and \mathbf{W}_{int}^B is the interaction wrench applied by the environment. The generalized momentum \mathbf{P}^B depends on the twist and the generalized inertia tensor \mathcal{I} :

$$\mathbf{P}^B := \mathcal{I} \mathbf{T}_B^{B,I}, \quad \text{where} \quad \mathcal{I} = \begin{pmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{0} & m\mathbf{I}_3 \end{pmatrix} \quad (2.25)$$

Furthermore, $\tilde{\mathbf{P}}^B$ is the 6x6 skew-symmetric matrix representation of the generalized momentum, which is defined by

$$\tilde{\mathbf{P}} := \begin{pmatrix} \tilde{\mathbf{P}}_\omega & \tilde{\mathbf{P}}_v \\ \tilde{\mathbf{P}}_v & \mathbf{0} \end{pmatrix}, \quad \text{where} \quad \mathbf{P} = \begin{pmatrix} \mathbf{P}_\omega \\ \mathbf{P}_v \end{pmatrix} \quad (2.26)$$

where the tilde map $\tilde{(\cdot)}$ operating on $\mathbf{P}_\omega \in \mathbb{R}^3$ and $\mathbf{P}_v \in \mathbb{R}^3$ is as defined in eq. 2.4.

The dynamic model given by eq. 2.24 can subsequently be constructed as a bond graph model, as shown in figure 2.5. Note that here the 'actuation' block implements the actuation model described in §2.4.1. The 'sensing' block has no effect in this model, meaning that perfect sensing of the rigid body twist $\mathbf{T}_B^{B,I}$ has been assumed.

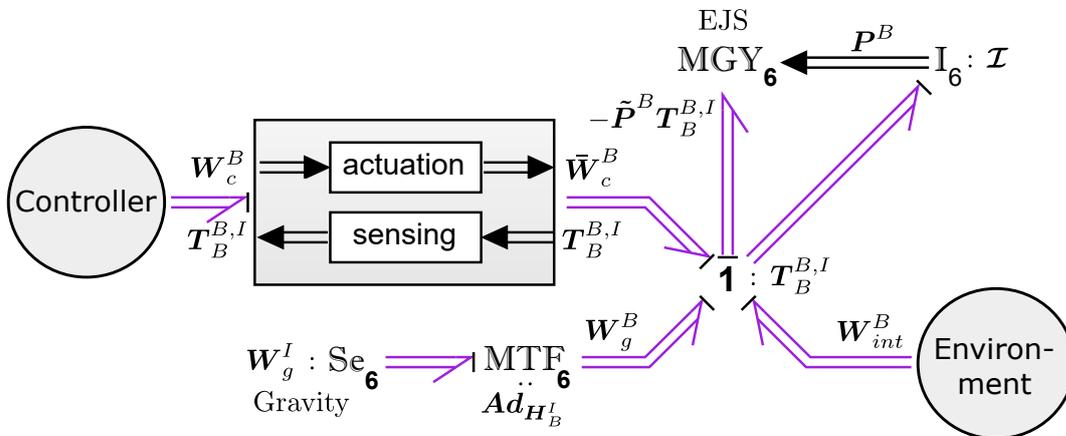


FIGURE 2.5: Bond graph representation of the dynamic rigid body UAV model.

Chapter 3

Energy Tank-Based Wrench/Impedance Control

One of the research goals of this thesis is to achieve stable physical interaction of a fully actuated UAV with an unknown environment. An added objective is to make sure that the designed solution fits well in the port-based methodology in its entirety, rather than having a mixed signal-based and port-based solution as is the case in [Rashad et al. \(2019\)](#).

This chapter presents the proposed approach for satisfying the above objectives. Stable interaction with any arbitrary environment can be achieved by making sure that the controlled system behaves passively (as previously explained in §2.1). This has been accomplished through the use of an impedance controller, which is intrinsically passive. A description of the used impedance controller can be found in §3.1.

Regulation of the interaction wrench (as described in §3.2) is subsequently implemented by using a momentum-based observer for estimating the interaction wrench, and modulating the stiffness of the impedance controller spring accordingly, while keeping its setpoint constant. It must be noted here that the actual implementation consists of two parallel virtual springs, one of which is modulated while the other is kept at constant stiffness.

Also note that, due to the stiffness modulation, the passivity of the controlled UAV system is lost, as made clear in the passivity analysis in §3.3. The passivity is subsequently restored by introducing an energy tank, of which a modular port-based implementation is proposed in §3.4.

3.1 Impedance Controller

The design of the intrinsically passive impedance controller is based on the work of [Rashad et al. \(2019\)](#), which exploits the nonlinear geometric structure of rigid body dynamics (as summarized in §2.4.2) by proposing the use of a 6D spatial spring. On one end, this spring attaches to the UAV's end effector frame Ψ_E , while the other end attaches to the desired frame Ψ_D (see figure 3.1). Just like a simple 1D spring, such a spatial spring stores energy, the amount of which depends on the relative configuration of the frames attached to each end of the spring, as well as on its stiffness matrix. The spring can be designed by choosing its energy function $\mathcal{H}_{spr}(\mathbf{R}, \boldsymbol{\xi})$, after which its behavior is completely defined. In this case the energy function has been chosen to take the following form, based on [Stramigioli and Duindam \(2001\)](#):

$$\mathcal{H}_{spr}(\mathbf{R}, \boldsymbol{\xi}) = \frac{1}{4} \boldsymbol{\xi}^T \mathbf{K}_t \boldsymbol{\xi} + \frac{1}{4} \boldsymbol{\xi}^T \mathbf{R} \mathbf{K}_t \mathbf{R}^T \boldsymbol{\xi} - \text{tr}(\mathbf{G}_o(\mathbf{R} - \mathbf{I}_3)) \quad (3.1)$$

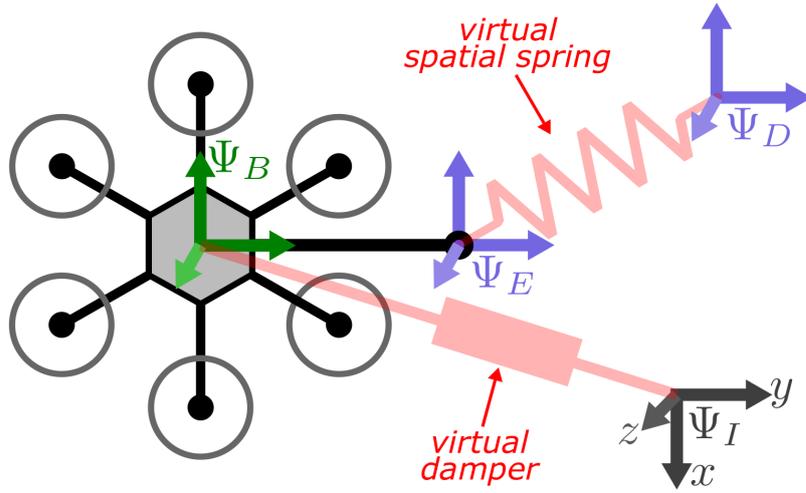


FIGURE 3.1: Schematic representation of the impedance controller, which consists of a virtual spatial spring attached between the end effector (Ψ_E) and desired frame (Ψ_D), as well as a virtual damper attached to the UAV's center of mass (Ψ_B) and the inertial frame Ψ_I .

where \mathbf{R} represents the relative orientation and $\boldsymbol{\xi}$ the relative displacement between either end of the spring. Furthermore, $\mathbf{K}_t \in \mathbb{R}^{3 \times 3}$ denotes the translational stiffness matrix, $\mathbf{G}_o \in \mathbb{R}^{3 \times 3}$ denotes the orientational co-stiffness (definition in eq. 3.3) and $\text{tr}(\cdot)$ denotes the matrix trace.

The torques $\boldsymbol{\tau}$ and forces \mathbf{f} that this spring would exert on the UAV, expressed in Ψ_E , have been derived to the following expressions (Rashad et al., 2019):

$$\begin{aligned}\tilde{\boldsymbol{\tau}}_{spr}^E(\mathbf{R}, \boldsymbol{\xi}) &= -2 \text{as}(\mathbf{G}_o \mathbf{R}) - \text{as}(\mathbf{G}_t \mathbf{R}^T \tilde{\boldsymbol{\xi}}^2 \mathbf{R}) \\ \tilde{\mathbf{f}}_{spr}^E(\mathbf{R}, \boldsymbol{\xi}) &= -\mathbf{R}^T \text{as}(\mathbf{G}_t \tilde{\boldsymbol{\xi}}) \mathbf{R} - \text{as}(\mathbf{G}_t \mathbf{R}^T \tilde{\boldsymbol{\xi}} \mathbf{R})\end{aligned}\quad (3.2)$$

where $\text{as}(\cdot)$ denotes the anti-symmetric part of a matrix. The co-stiffnesses \mathbf{G}_t , \mathbf{G}_o and \mathbf{G}_c are defined as followed:

$$\mathbf{G}_{(\cdot)} = \frac{1}{2} \text{tr}(\mathbf{K}_{(\cdot)}) \mathbf{I}_3 - \mathbf{K}_{(\cdot)} \quad (3.3)$$

The expressions in eq. 3.2 together define the wrench exerted on the UAV, expressed in Ψ_E . The representation of this wrench in \mathbb{R}^6 is as follows:

$$\mathbf{W}_{spr}^E = \begin{pmatrix} \boldsymbol{\tau}_{spr}^E \\ \mathbf{f}_{spr}^E \end{pmatrix} \quad (3.4)$$

The impedance controller also includes a spatial damper, attached on one end to the UAV's center of mass (i.e. at Ψ_B), and on the other end to an inertial frame (Ψ_I). Instead of defining this damper in a geometrically consistent way, as was done for the spatial spring, it is simply defined as a linear damper, defined by the 6x6 symmetric positive definite damping matrix \mathbf{K}_d . It is argued in Fasse and Broenink (1997) that this will be sufficient for quasi-static tasks. The wrench exerted by the damper, expressed in Ψ_B , equals:

$$\mathbf{W}_{dmp}^B = \mathbf{K}_d \mathbf{T}_B^{B,I} \quad (3.5)$$

Note that, as a slight difference to Rashad et al. (2019), the damping wrench has been defined without a minus sign. This choice will be beneficial at a later stage, when fitting this equation into the Dirac structure described in section 3.4. As a result, the impedance control wrench is now written as follows:

$$\mathbf{W}_{imp}^B = \mathbf{Ad}_{\mathbf{H}_B^E}^T \mathbf{W}_{spr}^E - \mathbf{W}_{dmp}^B \quad (3.6)$$

where $\mathbf{Ad}_{\mathbf{H}_B^E}^T$ denotes the transpose of an adjoint map, which in this case causes the wrench \mathbf{W}_{spr}^E to change coordinates from Ψ_E to Ψ_B .

By adding the gravity compensation feedforward term $-\hat{\mathbf{W}}_g^B$ to the impedance controller output, the wrench that the complete controller exerts on the UAV becomes the following:

$$\mathbf{W}_c^B = \mathbf{W}_{imp}^B - \hat{\mathbf{W}}_g^B \quad (3.7)$$

where $\hat{\mathbf{W}}_g^B$ is an estimate of \mathbf{W}_g^B , the wrench that gravity applies to the UAV.

3.2 Interaction Wrench Regulation

As noted before, a impedance controller by itself is not capable of regulating the interaction wrench with the environment. However, in many interaction scenarios, this is needed for achieving the desired interactive behavior. In figure 3.2 such a scenario has been illustrated. The goal for the UAV is to apply a user-defined interaction wrench (in this case a normal force) to a flat surface.

In order to add wrench tracking capabilities to the impedance controller, several approaches could be taken. In Rashad et al. (2019) this is achieved by introducing a PID controller that steers the difference between the observed wrench and desired wrench to zero. Here, the observed wrench is estimated by means of an interaction wrench observer, as described in §3.2.5. The output of this controller (\mathbf{W}_{tr}^B), referred to as the *wrench tracking controller*, is then added as an extra term to the control wrench:

$$\mathbf{W}_c^B = \mathbf{W}_{imp}^B - \hat{\mathbf{W}}_g^B + \mathbf{W}_{tr}^B \quad (3.8)$$

However, as the computed tracking wrench \mathbf{W}_{tr}^B lacks a physical meaning in this case, the choice is made to use a different approach. Instead, a virtual variable stiffness spring is implemented, substituting the PID controller. This variable stiffness implementation can be explained physically as a modulation of the stiffness matrix of the virtual spatial spring used by the impedance controller described above (§3.1). Subsequently, this stiffness modulation could be controlled to steer the difference between the observed and desired interaction wrench to zero, as described in §3.2.3.

However, instead of applying the stiffness modulation to this spring directly, the stiffness modulation is applied to a second parallel spring that is introduced. The first spring, which will be referred to as the *passive spring*, can then be kept at constant stiffness. The reason for this construction with both a passive spring and a modulated spring is explained below.

3.2.1 Separation of Concerns: Two Parallel Springs

The choice for adding a parallel spring, instead of simply modulating the impedance controller spring itself, has been made deliberately. This gives the conceptual advantage of *separation of concerns*.

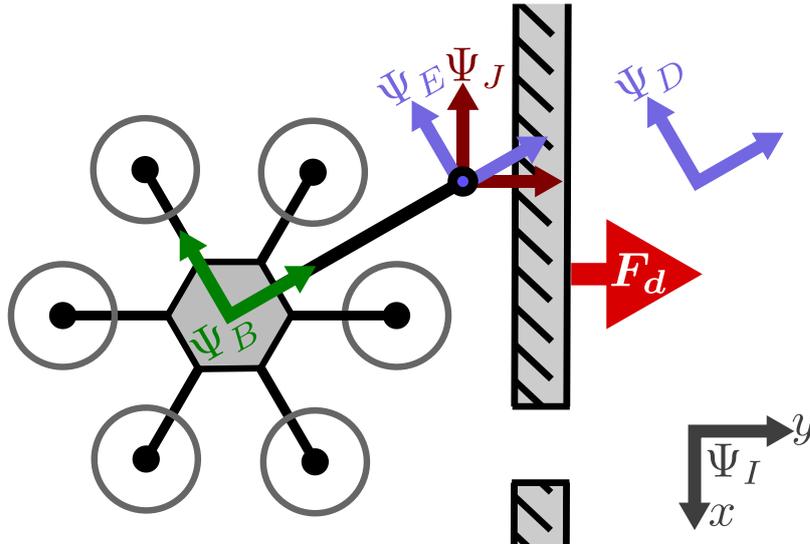


FIGURE 3.2: Schematic drawing of a typical interaction scenario: the goal is to apply a certain desired force F_d normal to the surface while making the end-effector take on the configuration defined by desired frame Ψ_D . The different coordinate frames used in the modeling phase have been visualized.

The fixed impedance spring will be responsible for motion control, and thus maintaining the desired configuration. As this part of the controller is inherently passive, it does not need any passifying by means of e.g. energy tank augmentation (see §3.4). Therefore, it can always be enabled, which is necessary for the UAV to prevent it from drifting off due to disturbances.

The variable impedance spring will be responsible for wrench tracking. This is the passivity-violating part of the controller, and has therefore been connected to an energy tank. The variable impedance spring can always be disabled if necessary, as this will not lead to a loss in control over the motion of the UAV.

3.2.2 Introducing the J-Frame

When regulating the interaction wrench, a certain coordinate frame has to be chosen in which the controller operates. One possible choice would be to regulate the interaction wrench in the end effector frame Ψ_E . However, this is strictly speaking not the frame in which the interaction task will be defined. For example, in the case that the defined task is to apply a certain normal force on a surface (as in figure 3.2), this desired force is actually expressed in a frame that does not rotate with respect to this surface. At the same time, however, one does know that the interaction will always take place at the end effector location (p_E^I).

Therefore a new coordinate frame is introduced, called the *J-frame* (as shown in figure 3.2), which has its origin at the end effector, and its orientation (R_J^I) aligned with the interaction task (aligned to the surface of the wall in this case). The configuration of the J-frame w.r.t. the inertial frame can be described by the following homogeneous matrix:

$$H_J^I = \begin{bmatrix} R_J^I & p_E^I \\ 0 & 1 \end{bmatrix} \quad (3.9)$$

Regulating the interaction wrench in the J-frame makes sure that the work is always performed in the desired direction, independent of the orientation of the UAV with respect to the surface.

One must note, however, that it cannot be assumed that the geometry of the environment is always known accurately, such that defining \mathbf{R}_J^I might become a problem. On the other hand, some estimate of the geometric features of the environment will always be necessary, considering that the interaction task must also be defined in one way or the other. Therefore, the accuracy of defining \mathbf{R}_J^I directly depends on the accuracy with which the task can be defined.

3.2.3 Stiffness Modulation of the Spatial Spring

As the stiffness matrix of a 6-DoF spatial spring is modulated in order to achieve the desired wrench, a controller must be defined that performs this modulation. A diagonal stiffness matrix $\mathbf{\Lambda}_v$ is introduced, of which the elements are independently controlled. This diagonal stiffness matrix will be used as the stiffness matrix of the spatial spring, expressed in Ψ_J .

$$\mathbf{K}_v^J = \mathbf{\Lambda}_v \quad (3.10)$$

Based on the relative configuration of Ψ_D and Ψ_E , as well as on the stiffness matrix, one can compute the actual wrench that the spatial spring exerts on the end-effector. As later equations require the stiffness to be expressed in the end-effector frame, the following coordinate transformation is applied to compute the stiffness matrix expressed in Ψ_E :

$$\mathbf{K}_v^E = (\mathbf{Ad}_{\mathbf{H}_E^J})^T \mathbf{\Lambda}_v \mathbf{Ad}_{\mathbf{H}_E^J} \quad (3.11)$$

The next step is to modulate the elements of $\mathbf{\Lambda}_v$ such that the wrench tracking error is steered towards zero. This can be achieved by using a proportional control law, that sets the time derivative of $\mathbf{\Lambda}_v$:

$$\dot{\mathbf{\Lambda}}_v = \mathbf{K}_p (\hat{\mathbf{W}}_{int}^J - \mathbf{W}_d^J) \quad (3.12)$$

where \mathbf{K}_p is a positive diagonal matrix of proportional gains, $\hat{\mathbf{W}}_{int}^J$ is the observed interaction wrench and \mathbf{W}_d^J the desired interaction wrench, both expressed in the J-frame. The used interaction wrench observer has been described in §3.2.5.

3.2.4 Spatial Spring Description

The used spatial spring model is similar to the model used in Rashad et al. (2019), described in section §3.1. However, that model only uses the translational and orientational part of the stiffness matrix, not the coupling part. And as the used stiffness matrix in this case is not diagonal, due to the coordinate transformation described in §3.2.2, it is desirable to model the effect of the coupling part of the stiffness matrix as well. This coupling part appears in the model described in Stramigioli and Duijdam (2001), which compared to the other model yields an extra term for both the computed torque and computed force. For the sake of clarity, the matrices \mathbf{R}_E^D , $\tilde{\xi}_E^D$ are abbreviated by \mathbf{R} , $\tilde{\xi}$ respectively:

$$\begin{aligned} \tilde{\tau}_{spr}^E(\mathbf{R}, \xi) &= -2 \text{as}(\mathbf{G}_o \mathbf{R}) - \text{as}(\mathbf{G}_t \mathbf{R}^T \tilde{\xi}^2 \mathbf{R}) - 2 \text{as}(\mathbf{G}_c \tilde{\xi} \mathbf{R}) \\ \tilde{\mathbf{f}}_{spr}^E(\mathbf{R}, \xi) &= -\mathbf{R}^T \text{as}(\mathbf{G}_t \tilde{\xi}) \mathbf{R} - \text{as}(\mathbf{G}_t \mathbf{R}^T \tilde{\xi} \mathbf{R}) - 2 \text{as}(\mathbf{G}_c \mathbf{R}) \end{aligned} \quad (3.13)$$

In conclusion, the wrench exerted by the spatial spring depends on the relative configuration between the desired frame and the end effector frame, \mathbf{H}_E^D . This wrench

consists of the force and torque components described in eq. 3.13:

$$\mathbf{W}_{spr}^E(\mathbf{H}_E^D) = \begin{pmatrix} \boldsymbol{\tau}_{spr}^E \\ \mathbf{f}_{spr}^E \end{pmatrix} \quad (3.14)$$

In case of a variable stiffness spring, obviously the equations become dependent on the variable stiffness matrix as well:

$$\mathbf{W}_{spr,v}^E = \mathbf{f}(\mathbf{H}_E^D, \mathbf{K}_v^E) \quad (3.15)$$

Lastly, the wrench computed by the interaction wrench controller must be expressed in frame Ψ_B , such that it can be added to the other control wrenches. The resulting output will be denoted by $\bar{\mathbf{w}}_{tr}$:

$$\bar{\mathbf{w}}_{tr} = \mathbf{W}_{spr,v}^B = \mathbf{Ad}_{\mathbf{H}_B^E}^T \mathbf{W}_{spr,v}^E \quad (3.16)$$

3.2.5 Interaction Wrench Observer

A momentum-based wrench observer is used to get an estimate for the interaction wrench that the environment exerts on the UAV end effector, $\hat{\mathbf{W}}_{int}^B$. The same observer is used as designed in Rashad et al. (2019). In short, this observer is constructed by first considering the following momentum dynamics of the UAV (as introduced in §2.4.2):

$$\dot{\mathbf{P}}^B = \underbrace{\tilde{\mathbf{P}}^B \boldsymbol{\mathcal{I}}^{-1} \mathbf{P}^B + \mathbf{W}_g^B + \bar{\mathbf{W}}_c^B}_{\mathbf{f}(\mathbf{P}^B, \bar{\mathbf{W}}_c^B)} + \mathbf{W}_{int}^B \quad (3.17)$$

Next, the following constant wrench observer is designed:

$$\dot{\hat{\mathbf{W}}}_{int}^B = \mathbf{K}_{obs}(\mathbf{P}^B - \hat{\mathbf{P}}), \quad (3.18)$$

$$\dot{\hat{\mathbf{P}}} = \mathbf{f}(\mathbf{P}^B, \bar{\mathbf{W}}_c^B) + \mathbf{K}_{obs}(\mathbf{P}^B - \hat{\mathbf{P}}) \quad (3.19)$$

where $\mathbf{K}_{obs} \in \mathbb{R}^{6 \times 6}$ is a positive diagonal matrix containing the observer gains, and $\hat{\mathbf{P}} \in \mathbb{R}^6$ is an estimate of the rigid body momentum.

By taking the time derivative of eq. 3.18 and making substitutions using eq. 3.17 and 3.19, the following expression can be derived, which specifies how the estimated interaction wrench relates to the actual interaction wrench (given that $\mathbf{f}(\mathbf{P}^B, \bar{\mathbf{W}}_c^B)$ is exactly known).

$$\dot{\hat{\mathbf{W}}}_{int}^B + \mathbf{K}_{obs} \hat{\mathbf{W}}_{int}^B = \mathbf{K}_{obs} \mathbf{W}_{int}^B \quad (3.20)$$

It shows the behavior of six independent first-order low-pass filters, where the tracking speed for each component of the interaction wrench depends on the corresponding observer gain. The filter for the i -th component, corresponding to $\mathbf{K}_{obs,i}$, takes the following form in the Laplace domain.

$$\frac{1}{\frac{s}{\mathbf{K}_{obs,i}} + 1} \quad (3.21)$$

Finally, note that as the control law is implemented in Ψ_J , the observed wrench must be expressed in Ψ_J as well:

$$\hat{\mathbf{W}}_{int}^J = (\mathbf{Ad}_{\mathbf{H}_J^B})^T \hat{\mathbf{W}}_{int}^B \quad (3.22)$$

3.3 Passivity Analysis

The control system that has been constructed up to now, has been illustrated in figure 3.3. The interconnection between the different subsystems has been represented using the *bond graph* notation (as explained in §2.1), such that it is explicitly visible where energy exchange takes place. In this way, analyzing the passivity of the closed loop system becomes a straightforward and easy task, as it can be done in a graphical manner. In contrast to the known inequality-based passivity analysis used in e.g. Rashad et al. (2019); Ferraguti et al. (2015); Shahriari et al. (2017), the graphical passivity analysis demonstrated in this section offers an immediate oversight of the passivity in the system, as well as a clear understanding of how the passivity is affected when changing the design. Also note that the possibility of performing such a graphical analysis is one of the advantages of using the port-based framework for both the dynamic model and the control system design.

Now let us consider the passivity of the closed loop UAV system, indicated by the dashed line in figure 3.3. Note that in this case, each of the bonds consists of a certain wrench \mathbf{W} and a certain twist \mathbf{T} , both represented in \mathbb{R}^6 . Together these define the power flowing through the corresponding bond (as explained in §2.3.4).

$$P = (\mathbf{W}^a)^T \mathbf{T}_{\bullet}^{a,\bullet} \quad (3.23)$$

where the wrench and twist have both been defined in the same coordinate frame (Ψ_a in this example), and the bullets (\bullet) can represent any pair of coordinate frames.

The different depicted subsystems include the *impedance controller*, the *gravity compensated UAV*, the *environment*, and the *exert wrench* subsystem. Note that both the impedance controller and the gravity compensated UAV consist of smaller interconnected subsystems. The interconnections, consisting of bonds and junctions, are power continuous by definition. This means that energy is neither created nor dissipated here. Therefore, a system is automatically passive if all of its interconnected subsystems are passive. Because the goal is to make sure that the closed loop UAV system is passive as a whole, all of its subsystems are evaluated for passivity below.

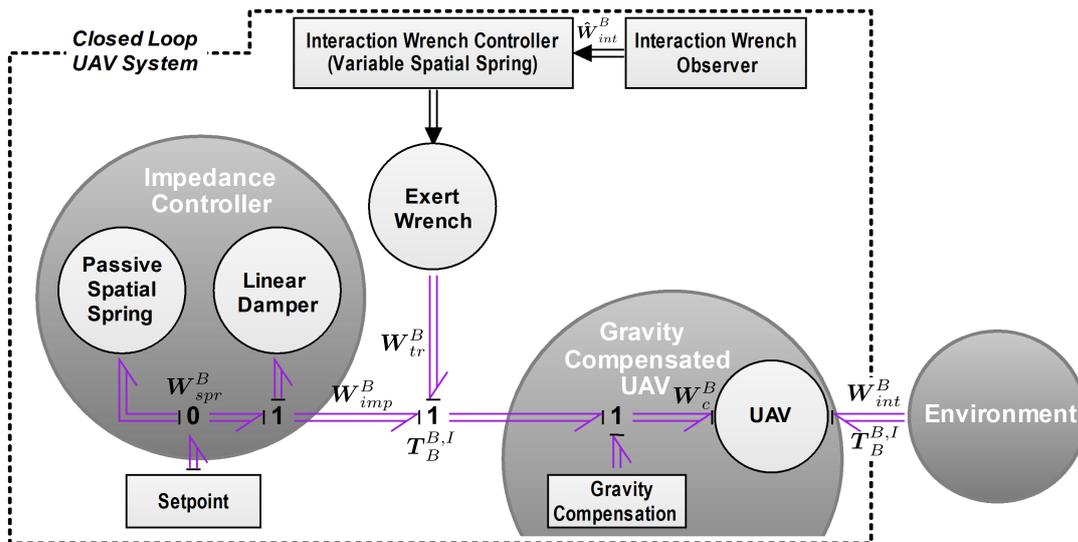


FIGURE 3.3: Version 1 of the control system: in this form, passivity is violated by the power injected by the "Exert Wrench" subsystem.

The impedance controller is passive by design. It consists of a passive spatial

spring, which stores energy, and a damper, which dissipates energy. Therefore no energy is being created in this subsystem. Note, however, that changing the setpoint (defined by Ψ_D), which is connected to one of the ends of the spring, could in fact inject newly created energy into the spring. This is clearly a non-passive action, which is considered to be performed by a separate subsystem named *setpoint*. By not allowing the setpoint to change, this subsystem will not inject (or extract) any energy into the impedance controller subsystem. Therefore, for the remainder of this analysis, the setpoint is assumed constant:

$$\dot{\mathbf{H}}_D^I = \mathbf{0} \quad (3.24)$$

Note, however, that in the scenarios simulated in the rest of this work, the setpoint is moved in a piecewise manner in order to achieve the desired behavior of the UAV. Therefore, one could say that the passivity analysis does not hold for those exact instances where the setpoint is moved. This introduces the requirement of carefully moving the setpoint, to avoid introducing excessive amounts of energy into the system.

Next, consider the gravity compensated UAV subsystem, which consists of the UAV and the gravity compensation subsystems. The UAV is modeled as a 3D rigid body with six rotors located at different locations and orientations, that together directly apply the control wrench \mathbf{W}_c^B at the UAV control input. At the same time, gravity acts on the UAV's center of mass, which can be expressed as the gravity wrench \mathbf{W}_g^B . This wrench is compensated by computing an estimate of this wrench ($\hat{\mathbf{W}}_g^B$) and subtracting it as a feedforward term, as implemented by the *gravity compensation* subsystem. The gravity compensation is in principle a non-passive action, but the gravity compensated UAV system as a whole will still be passive if the estimated gravity wrench $\hat{\mathbf{W}}_g^B$ is exactly equal to the actual gravity wrench \mathbf{W}_g^B . There is of course always a certain error involved in the wrench estimate:

$$\mathbf{W}_{g,err}^B = \mathbf{W}_g^B - \hat{\mathbf{W}}_g^B \quad (3.25)$$

The power injected due to this gravity compensation error would equal:

$$P_{g,err} = (\mathbf{W}_{g,err}^B)^T \mathbf{T}_B^{B,I} \quad (3.26)$$

However, this injected power will be assumed small enough such that it is negligible compared to the energy extracted by e.g. the virtual damper or due to external dissipation. This must in fact also be assumed for all the other potential passivity-violating factors, such as modeling errors or delays.

Lastly, the *exert wrench* subsystem must be considered. This subsystem directly applies the output wrench \mathbf{W}_{tr}^B from the interaction wrench controller, which injects the following wrench tracking power P_{tr} into the system:

$$P_{tr} = (\mathbf{W}_{tr}^B)^T \mathbf{T}_B^{B,I} \quad (3.27)$$

This injection of power prevents the closed loop UAV system from being passive. This subsequently compromises guaranteed contact stability with any passive environment. In other words, the UAV could encounter an environment that destabilizes it upon interaction.

3.4 Energy Tank Augmentation

A way to restore passivity of the closed loop system, is through the augmentation of an energy tank, which recently found its way into the field of aerial interaction control (Rashad et al., 2019), motivated by previous results in its application to interaction control for a ground manipulator (Schindlbeck and Haddadin, 2015). The concept of the energy tank can be sub-categorized under a more general approach, called *energy routing* (Duindam and Stramigioli, 2004). When applying energy routing, the total energy content of the system can remain unaffected, while the controller is able to perform otherwise non-passive actions, by intentionally directing the energy flows within the system.

This is exactly what happens when introducing the energy tank. The tank can be used for storing a certain limited amount of virtual energy within a control system. This energy may for instance originate from a virtual damper, which would otherwise directly dissipate this energy. Alternatively, the energy in the tank could be initialized to a certain amount. Subsequently, this virtual energy can be routed towards subsystems performing control actions that would otherwise require the creation of additional energy. In other words, all of the energy that is injected into the system by the controller, must be drawn from the energy tank, such that the total energy within the system does not increase. Consequently, when this virtual tank is depleted of energy, no more energy can be added to the system by the controller. On the other hand, all energy that is dissipated from the system, for example due to virtual damping, could be fed back into the energy tank. In the most extreme case, all dissipated energy is routed towards the energy tank, resulting in a lossless system. As a lossless system is on the border of being passive, one could decide to take a safety margin by only feeding a part of the dissipated energy back into the energy tank.

In case of interaction control, as soon as the energy tank depletes, the robot loses its ability to perform its current interaction task. However, the exact effect of a depleted energy tank will depend on the way that it is interconnected with the control system. For example, in the case of using impedance control, the stiffness of the impedance controller spring can be modulated in order to regulate the interaction wrench. The act of modulating this stiffness violates passivity, as the energy within the spring increases when the stiffness increases. Then, as one possible option, an energy tank could be augmented to supply the energy that it takes to modulate the spring stiffness. A depleted energy tank should, in that case, prevent any further increase in stiffness. As a second option, the energy tank could be connected in such a way, that it supplies the energy injected into the system by the spring itself. In that case, a depleted energy tank should result in disabling the spring as a whole. The specific application determines what kind of behavior would be the most desirable.

In Rashad et al. (2019) an energy tank is augmented in such a way that the power injected by the wrench regulation is drawn from a limited supply of energy. Upon depletion of this energy, the wrench tracking term W_{tr}^B gets deactivated, after which wrench regulation is no longer possible until the tank gets filled with energy again. This approach effectively restores the passivity of the closed loop UAV system when one considers the energy tank as a part of this system. However, it uses a signal-based implementation, which does not fit well with the port-based framework in which the rest of the control system is designed. The aim in this thesis is instead to derive a pure port-based solution. Therefore an implementation of the energy tank is proposed that fits seamlessly into the port-based framework, as presented below.

3.4.1 Energy Routing using a Dirac Structure

The concept of energy routing, as explained above, means that energy flows from certain parts of the closed loop system are intentionally directed to other parts, such that otherwise non-passive actions can be performed without increasing the total energy content in the system.

The proposed port-based implementation of the energy tank augmentation should route the energy flows in such a way that the power used for wrench tracking (P_{tr}) is drawn directly from the energy tank. Simultaneously, the power dissipated by the virtual damper (P_{dmp}) should be directed towards the energy tank with a certain defined efficiency $0.0 \leq \eta_d \leq 1.0$. The power into the energy tank should then equal

$$P_t = -P_{tr} + \eta_d P_{dmp} \quad (3.28)$$

where it should also be noted that $P_{dmp} = (\mathbf{T}_B^{B,I})^T \mathbf{K}_d \mathbf{T}_B^{B,I}$ is always positive, while $P_{tr} = (\mathbf{W}_{tr}^B)^T \mathbf{T}_B^{B,I}$ can be both positive and negative (the power can flow in two directions). The relevant energy flows and their intended routing have been illustrated in figure 3.4, where the remaining power from the virtual damper that is actually dissipated has been denoted by P_{diss} .

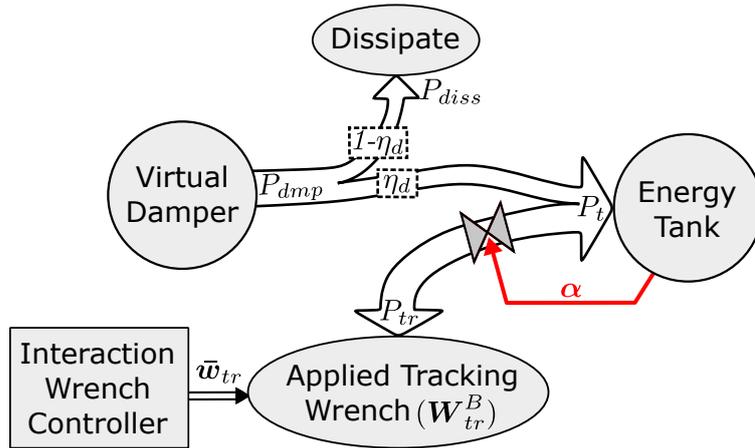


FIGURE 3.4: Intended routing of the energy flows.

The actually applied tracking wrench \mathbf{W}_{tr}^B , during normal operation, equals the computed wrench controller output \bar{w}_{tr} .

$$\mathbf{W}_{tr}^B = \alpha \bar{w}_{tr} \quad (3.29)$$

However, if the energy tank is empty, \mathbf{W}_{tr}^B should be equal to zero, to prevent any further power to be drawn. Therefore the wrench controller output is multiplied by a scalar α , referred to as the *valve gain*, which ranges from 0 to 1, as regulated by the energy tank (described in §3.4.2).

In this way the valve gain prevents any further tracking wrench to be exerted when the energy tank is empty, and thus no additional power will be drawn. However, this would also cause the stiffness modulation controller (eq. 3.12) to start continuously increasing the stiffness of the variable virtual spring (a sort of integral windup). To prevent this effect, the stiffness modulation control law in eq. 3.12 is updated to the following:

$$\dot{\Lambda}_v = \mathbf{K}_p (\hat{\mathbf{W}}_{int}^J - \alpha \mathbf{W}_d^J) \quad (3.30)$$

The desired energy routing described above can be implemented effectively within the port-based framework by making use of a Dirac structure. A Dirac structure is a power-continuous multi-port element that enables routing the energy flow between the different ports in any conceivable way, as long as no energy is stored or dissipated. Mathematically it can be represented as a matrix equation (as introduced before in §2.2.2), with on the left side the causal output variables and on the right side the causal input variables. The matrix itself, in order to yield a power-continuous system, must be skew-symmetric.

The proposed Dirac structure implementation has four ports, as can be seen in the new control system in figure 3.5. However, for a better understanding, it will be build up by first only considering the power continuous connection between the energy tank and the applied tracking wrench. This can be realized by having a Dirac structure with two power ports, of which the first is a one-dimensional port that connects to the energy tank. The power conjugate variables of this port can be defined with the energy tank state x_t as the effort and \dot{x}_t^σ as the associated flow, such that the power towards the energy tank is defined as

$$P_t = \dot{x}_t^\sigma x_t \quad (3.31)$$

The second port is a six-dimensional port defined by effort \mathbf{W}_{tr}^B and flow $\mathbf{T}_B^{B,I}$. The power of this port should be the negative of the wrench tracking power, as these power ports have been defined in opposite direction:

$$\dot{x}_t^\sigma x_t = -(\mathbf{W}_{tr}^B)^T \mathbf{T}_B^{B,I} \quad (3.32)$$

The relations described by eq. 3.29 and 3.32 could then be represented by a Dirac structure as follows:

$$\begin{pmatrix} \mathbf{W}_{tr}^B \\ \dot{x}_t^\sigma \end{pmatrix} = \begin{bmatrix} 0 & \frac{\alpha}{x_t} \bar{\mathbf{w}}_{tr} \\ -\frac{\alpha}{x_t} \bar{\mathbf{w}}_{tr}^T & 0 \end{bmatrix} \begin{pmatrix} \mathbf{T}_B^{B,I} \\ x_t \end{pmatrix} \quad (3.33)$$

Note that the interaction wrench controller output signal ($\bar{\mathbf{w}}_{tr}$) is now used to modulate the power flowing through the Dirac structure, which offers a power continuous way of applying any signal-based controller output.

This implementation can then be extended to include the power P_{dmp} that should be regenerated from the virtual damper. This requires the addition of a third power port, which has been defined by effort \mathbf{W}_{dmp}^B and flow $-\mathbf{T}_B^{B,I}$. The constitutive relation between this effort and flow must follow that of the previously defined virtual damper:

$$\mathbf{W}_{dmp}^B = \mathbf{K}_d \mathbf{T}_B^{B,I} \quad (3.34)$$

This leads to the following damping power being routed towards the energy tank:

$$P_{dmp} = (\mathbf{T}_B^{B,I})^T \mathbf{K}_d \mathbf{T}_B^{B,I} \quad (3.35)$$

However, in order to implement the intended energy routing described by eq. 3.28, the damping regeneration efficiency η_d must be included as well. Writing eq. 3.28 out in terms of the port variables, gives the following relation:

$$P_t = \dot{x}_t^\sigma x_t = -(\mathbf{W}_{tr}^B)^T \mathbf{T}_B^{B,I} + \eta_d (\mathbf{T}_B^{B,I})^T \mathbf{K}_d \mathbf{T}_B^{B,I} \quad (3.36)$$

To be able to implement this relation with a Dirac structure, a fourth port is needed, namely the *dissipation port*. The reason for this is that the Dirac structure itself does

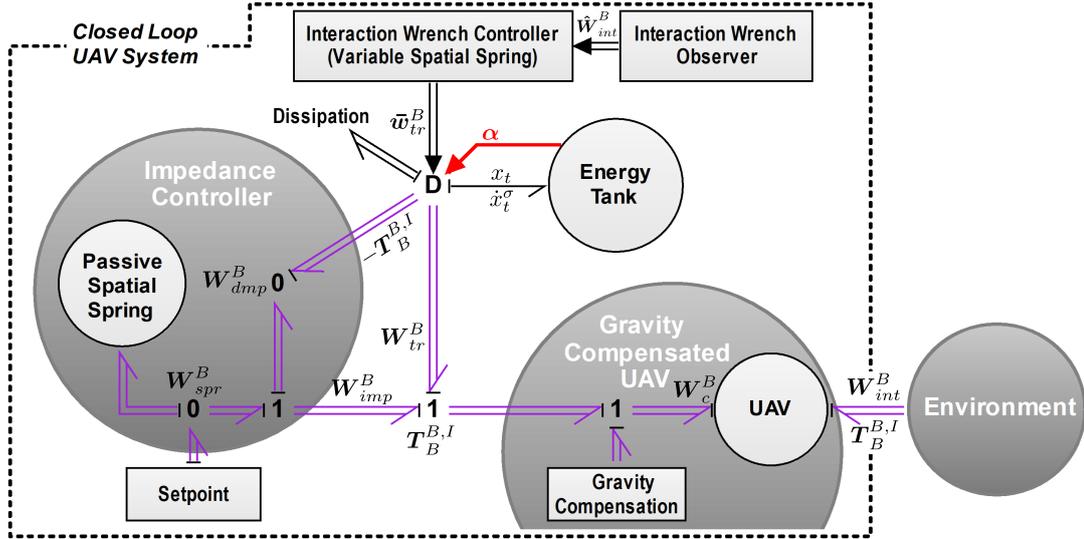


FIGURE 3.5: Version 2 of the control system: A port-based augmentation of an energy tank is proposed for restoring passivity.

not allow for dissipating energy, so the energy dissipated by having $0 \leq \eta_d < 1$ must be routed somewhere else. The power towards this dissipation port becomes:

$$P_{diss} = (\mathbf{W}_{diss})^T \mathbf{T}_{diss} = (1 - \eta_d) (\mathbf{T}_B^{B,I})^T \mathbf{K}_d \mathbf{T}_B^{B,I} \quad (3.37)$$

where \mathbf{W}_{diss} is the six-dimensional effort and \mathbf{T}_{diss} the six-dimensional flow associated with the dissipation port.

The following mathematical expression for the proposed four-port Dirac structure has been derived for connecting the four ports in a way that satisfies all of the above relations:

$$\begin{pmatrix} \mathbf{W}_{tr}^B \\ \mathbf{W}_{dmp}^B \\ \mathbf{T}_{diss} \\ \dot{x}_t^\sigma \end{pmatrix} = \begin{bmatrix} 0 & 0 & 0 & \frac{\alpha}{x_t} \bar{w}_{tr}^B \\ 0 & 0 & I & \frac{\eta_d}{x_t} \mathbf{K}_d \mathbf{T}_B^{B,I} \\ 0 & -I & 0 & 0 \\ -\frac{\alpha}{x_t} \bar{w}_{tr}^T & -\frac{\eta_d}{x_t} (\mathbf{T}_B^{B,I})^T \mathbf{K}_d & 0 & 0 \end{bmatrix} \begin{pmatrix} \mathbf{T}_B^{B,I} \\ -\mathbf{T}_B^{B,I} \\ \mathbf{W}_{diss} \\ x_t \end{pmatrix} \quad (3.38)$$

where

$$\mathbf{W}_{diss} = (1 - \eta_d) \mathbf{K}_d \mathbf{T}_{diss} \quad (3.39)$$

has been implemented by the 'Dissipation' element (see figure 3.5).

The resulting Dirac structure successfully implements the desired energy routing behavior in the port-based framework. The implementation of the energy tank itself has also been carried out in the port-based framework, and is described in the next subsection.

3.4.2 Design of the Energy Tank

In its most simple form, the energy tank can be modeled by a single storage element, represented by a \mathbf{C} in figure 3.6. However, on top of that it must adhere to two additional behaviors.

Firstly, if the energy tank almost depletes, it must be prevented from depleting any further, such that the tank energy never drops below the lower energy limit (E_t^-). This is enforced by reducing the valve gain α towards zero when the tank gets

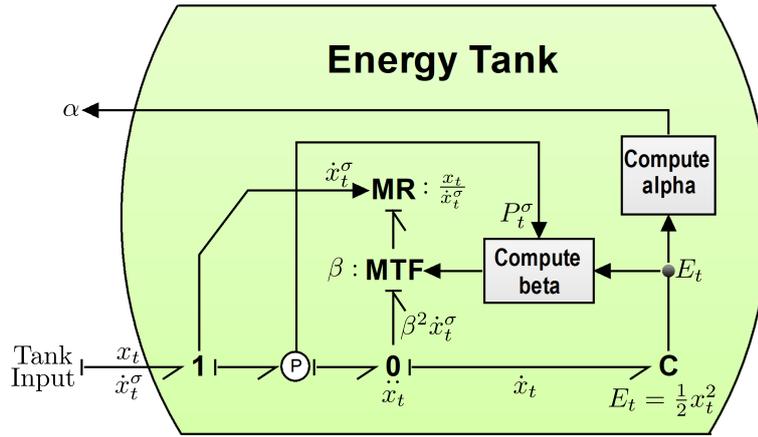


FIGURE 3.6: The proposed port-based implementation of the energy tank itself.

close to depletion. This has been implemented using a smooth transition from 1 to 0, as suggested by [Shahriari et al. \(2018\)](#):

$$\alpha = \begin{cases} 0 & \text{if } E_t \leq E_t^- \\ \frac{1}{2} \left[1 - \cos \left(\frac{E_t - E_t^-}{E_{window}} \pi \right) \right] & \text{if } E_t^- < E_t < E_t^- + E_{window} \\ 1 & \text{else} \end{cases} \quad (3.40)$$

A second behavior that the energy tank must show, is that it stops gaining energy as soon as its maximum limit E_t^+ has been reached. This can be enforced by introducing the scalar β , which can be considered a switch that adheres to the following law:

$$\beta = \begin{cases} 1 & \text{if } E_t \geq E_t^+ \wedge P_t^\sigma > 0 \\ 0 & \text{else} \end{cases} \quad (3.41)$$

where P_t^σ is the power at the input of the tank. The scalar β modulates a transformer, represented by the **MTF** symbol in the bond graph in figure 3.6. In this way, if $\beta = 1$, all of the power flowing into the tank will be dissipated by the modulated resistor, represented by the **MR** symbol. This bond-graph compatible approach to enforcing the maximum limit of the tank has been suggested by [Dietrich et al. \(2017\)](#). The actual power flowing into the tank (P_t) then depends both on the power at the input of the tank (P_t^σ) and on the scalar β :

$$P_t = (1 - \beta^2) P_t^\sigma \quad (3.42)$$

3.5 Conclusion

This chapter presented the proposed approach for achieving stable physical interaction with the environment for a fully actuated UAV. Additional objectives were to make sure that the designed solution does not require accurate information about the environment and that it fits well in the port-based framework.

An impedance controller has been used as the basis of the passivity-based control system, as it is intrinsically passive and deals well with uncertain environments. Next to having stable contact, the exact interaction wrench should be regulated as

well, for which a wrench tracking controller has been added to the system. This controller has been implemented as a modulated spatial spring, such that the applied wrench gains a physical interpretation.

In order to restore the passivity that was lost due to the addition of wrench regulation, an energy tank has been augmented to the control system. Both the energy tank itself and its interconnection to the rest of the control system fit seamlessly into the port-based framework, benefiting further design and analysis. Moreover, this approach has led to a modular solution, where a notable advantage is the fact that the implementation of the energy tank is independent from the specific choice of wrench controller and observer.

Chapter 4

Simulations: Validation of the Energy Tank-Based Controller

In this chapter, the designed energy-tank based wrench/impedance controller will be validated in a simulation environment constructed using the software package *20-sim*. In this environment, the bond graph model described in §2.4.2 can be directly simulated. The complete designed control system described in chapter 3 has been implemented as well. The setup of the simulation has been illustrated in figure 4.1, where the depicted wall has been modeled using a hybrid contact model: in the normal direction it acts as a stiff spring-damper system, and in the lateral direction it exerts a linear viscous friction on the end effector. A simulation scenario has been constructed where the UAV approaches the flat surface, establishes contact, and starts regulating the interaction wrench. The used simulation and controller parameters have been summarized in a table in appendix A.

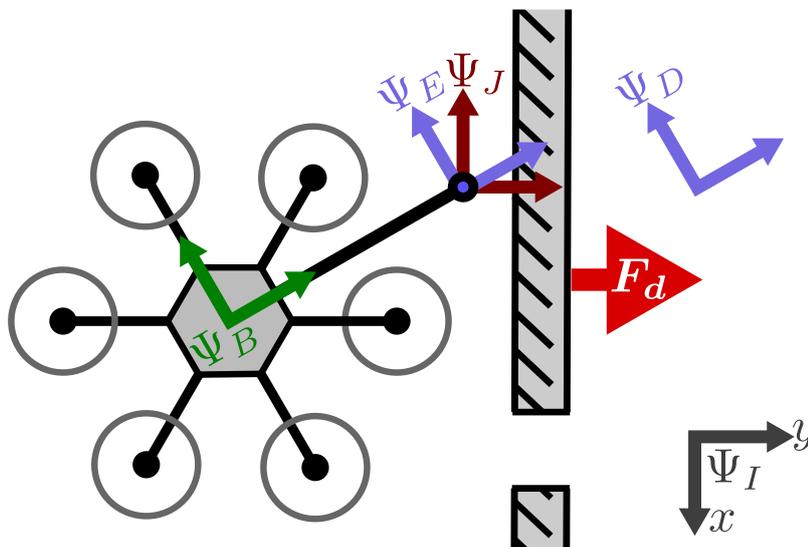


FIGURE 4.1: Schematic drawing of a typical interaction scenario: the goal is to apply a certain desired force F_d normal to the surface while making the end-effector take on the configuration defined by desired frame Ψ_D . The different used coordinate frames have been visualized.

The goal of the following simulations is to verify a correct working of the designed controller. Firstly, in §4.1 the passivity of the closed loop system is checked by keeping track of the different energy levels in the system. This can be done by keeping track of the total energy within the closed loop UAV system, which is then only allowed to increase due to externally injected energy from the environment.

Secondly, in §4.2 it is checked whether the energy tank behavior and its interference with the controller are as intended when it gets close to depletion.

4.1 Scenario 1: Passivity

In the scenario used in the following simulation, the UAV approaches a flat surface, establishes contact and starts regulating the interaction wrench. The different phases of this scenario are described in more detail below.

4.1.1 Description of the Scenario

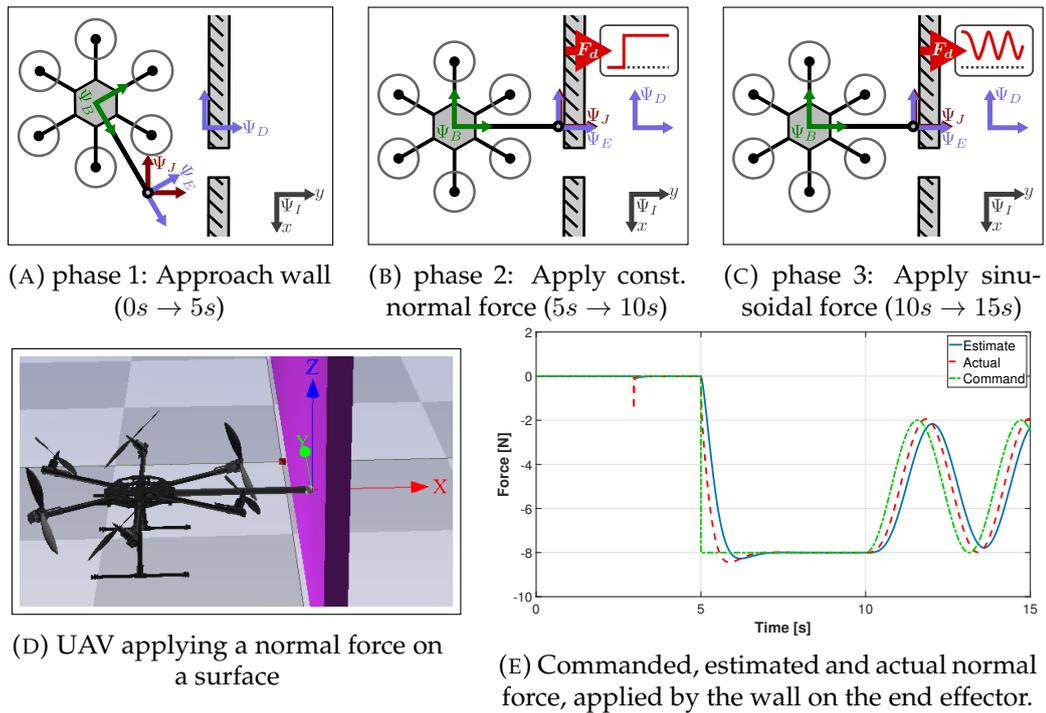


FIGURE 4.2: Description of scenario 1.

In phase 1, illustrated by figure 4.2a, the UAV approaches the wall. This is done by setting the setpoint of the impedance controller at a certain location on the surface of the wall, with an orientation such that the end-effector will be directed normal to the surface of the wall. During phase 1, the wrench controller is kept disabled. In the plot of the normal force (figure 4.2e) one can observe a peak at around 3.0s due to a contact bounce.

In phase 2, illustrated by figure 4.2b, the setpoint is displaced behind the surface of the wall, such that the impedance controller spring becomes extended. At the same time the wrench controller is enabled, which starts increasing the stiffness of the parallel spring in order to steer the applied normal force towards the commanded normal force F_d .

In phase 3, illustrated by figure 4.2c, the same conditions apply as in phase 2, but now the commanded normal force is varied over time by a sinusoidal function.

4.1.2 Explanation of the Results

First and foremost, it is important to verify that the designed solution indeed still guarantees passivity. This is done by monitoring the total energy within the closed loop UAV system (E_{total}), and subtracting the external energy that is injected by the environment ($E_{external}$). The resulting amount of energy should not increase at any moment in time, if the system is indeed passive.

The applied normal force over time has been shown in figure 4.3, as well as the resulting energy levels for the closed loop system, with and without energy tank. These energy levels include the "potential" energy, "kinetic" energy, "wrench tracking" energy, and "total - external" energy.

The plotted *potential* energy is a sum of the energy in the impedance controller spring, the gravitational energy of the UAV, and the energy injected by gravity compensation. In the ideal case of perfect sensors and actuators, like is the case in this simulation, the energy injected by gravity compensation exactly cancels out the gravitational energy, such that the shown potential energy effectively only consists of the energy stored within the impedance controller spring.

The plotted *kinetic* energy is simply the kinetic energy of the rigid body that models the UAV. The *wrench tracking* energy represents all the energy that is injected into the UAV by the wrench tracking controller, i.e. the integration of P_{tr} (in eq. 3.27).

The *total* energy represents the sum of the kinetic and potential energy of the controlled UAV. The *external* energy represents the energy physically put into the UAV by the environment. Therefore, the energy resulting from subtracting the external energy from the total energy, is the amount that should not increase if the controlled UAV system is passive.

4.1.3 Discussion of the Results

First of all, from figure 4.3a it is clear that the tracking of the commanded interaction force has successfully been performed, as well as the estimation of this force. Note that at around 3s, a bump shows up in the actual force. This is caused by the slight position overshoot that the UAV experiences, as from the start the setpoint is placed to barely touch the wall.

Furthermore, a certain tracking delay can be seen between the actual, commanded and estimated force. The delay between the actual and estimated force arises because the designed wrench observer acts as a first order low-pass filter on each component of the interaction wrench (as shown in eq. 3.20). The delay that can be observed between the commanded and actual force can be attributed to the proportional control law (eq. 3.12) that controls the time derivative of the stiffness gains corresponding to the modulated virtual spring. The tracking performance could be improved by increasing both the stiffness modulation gains (K_p) and the observer gains (K_{obs}), although this would come at the cost of an increased sensitivity to sensor noise and could destabilize these controllers.

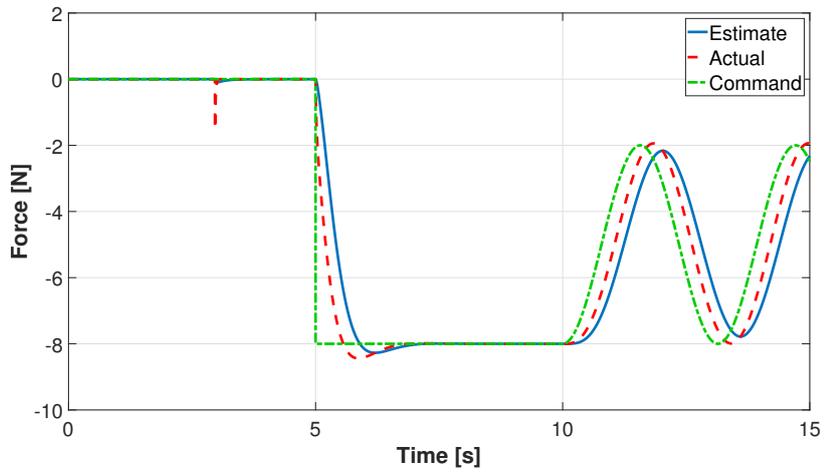
The energy levels resulting from the system without energy tank (figure 4.3b) show passive behavior only in phase 1. During this phase, one can only observe a decrease in the *total - external* energy curve. Passivity during this phase makes sense, as wrench control is not yet enabled, effectively reducing the controller to the intrinsically passive impedance controller.

At the instant of the start of phase 2 (i.e. at 5s) a sudden increase in energy can be seen, which is due to the change of the setpoint of the impedance controller. Afterwards, in phase 2 and phase 3, multiple regions of increasing energy are observed,

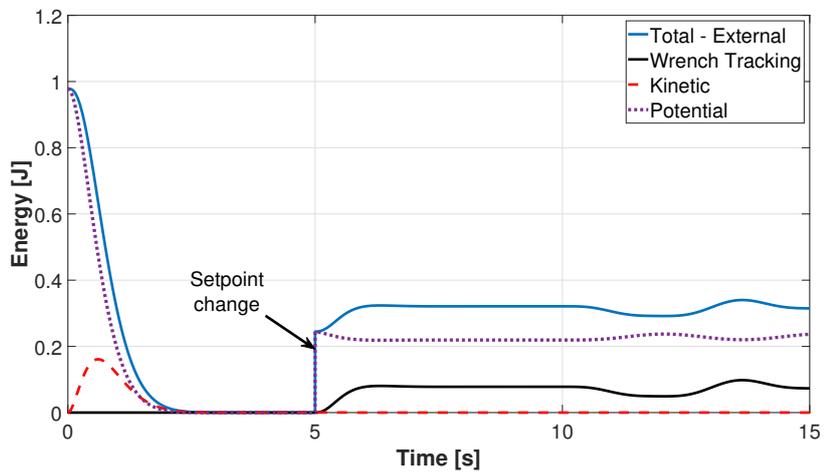
meaning that the closed loop system is not passive here. These regions of increasing energy are due to the wrench controller introducing additional energy into the system, resulting in a loss of passivity.

The results after adding the energy tank can be found in figure 4.3c. This time, the system behaves passively during all of the phases: The *total - external* energy level decreases during phase 1, and remains near constant (decreases slightly) during the interaction with the wall in phases 2 and 3. Thus, the closed loop energy never increases. The only exception is at the instant of the start of phase 2. However, this can be attributed to the energy added to the system by displacing the setpoint of the impedance controller. This is clearly a passivity-violating action, that has not been connected to the energy tank. Whether or not this action should be connected to the energy tank is a design choice. It might depend on the setup of the higher level controller that controls this setpoint.

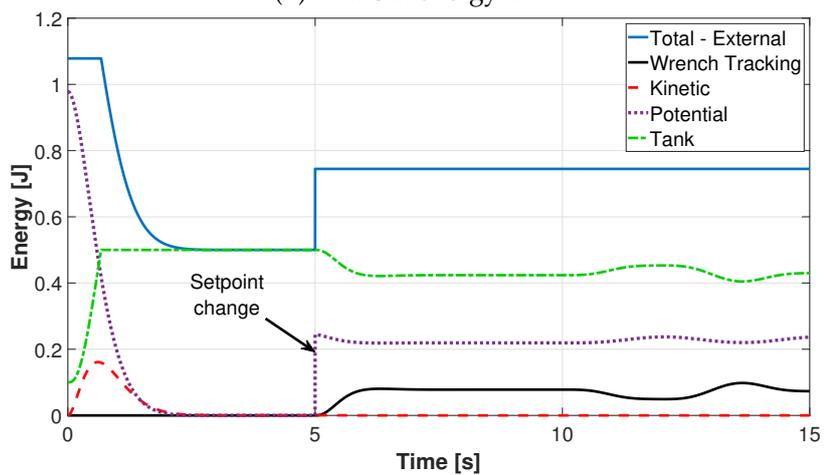
In the above case, 100% of the energy dissipated by the virtual damper has been redirected towards the energy tank. Furthermore, also 100% of the energy regained by the wrench controller from the environment has been stored in the energy tank. This behavior follows from setting both the parameters η_d and η_t to their max value of 1.0. As a result, the controlled UAV becomes a lossless system. This can be seen in figure 4.3c, where the *total - external* energy curve remains constant (until the energy tank saturates). However, in a practical controller, having $\eta_d = \eta_t = 1.0$ would not be a wise decision, because a lossless system is only marginally passive. A practical implementation would therefore require some passivity margin acting as a buffer to imperfections in the controlled system, such as modeling errors, disturbances or sensor noise. Such a passivity margin could be achieved by setting these parameters to lower values: $0.0 < \eta_d < 1.0$ and $0.0 < \eta_t < 1.0$.



(A) Commanded, estimated and actual normal force, applied by the wall on the UAV's end-effector.



(B) Without energy tank



(C) Including energy tank ($\eta_d = 1.0$)

FIGURE 4.3: Applied normal force, and energy levels of the closed loop system (scenario 1).

4.2 Scenario 2: Tank Depletion Behavior

The energy tank has been designed to not interfere with the controller as long it is not empty. However, at the moment that it is almost depleted, the valve gain α starts to decrease as well. This will restrict the power flowing out of the tank, which automatically reduces the power available for wrench tracking. Therefore, while the energy tank depletes, the wrench tracking controller will gradually shut off.

It is important to validate that, when the energy tank depletes, the way it interferes with the controller is as intended. In the following simulation this will be investigated.

4.2.1 Description of the Scenario

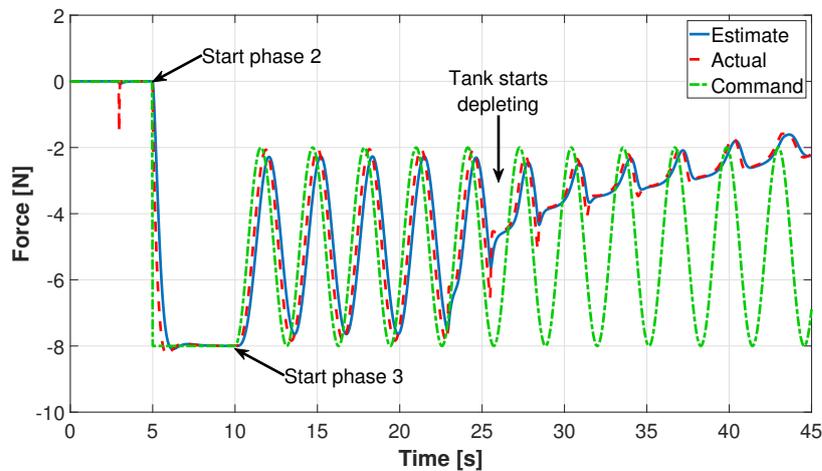


FIGURE 4.4: Commanded, estimated and actual normal force for scenario 2. It is identical to scenario 1, except that phase 3 continues for much longer now.

Scenario 2 is actually identical to scenario 1, with the exception that phase 3 this time takes 35s instead of 5s. Also, the stiffness of the wall has been decreased for this scenario (see the table in appendix A), such that the UAV loses more energy to the wall during interaction. In this way, the tank starts to deplete at around 25s.

4.2.2 Explanation and Discussion of the Results

The energy tank behavior in the event of tank depletion has been shown in figure 4.5, including the resultant applied force by the wrench controller, as well as the different energy levels in the closed loop UAV system.

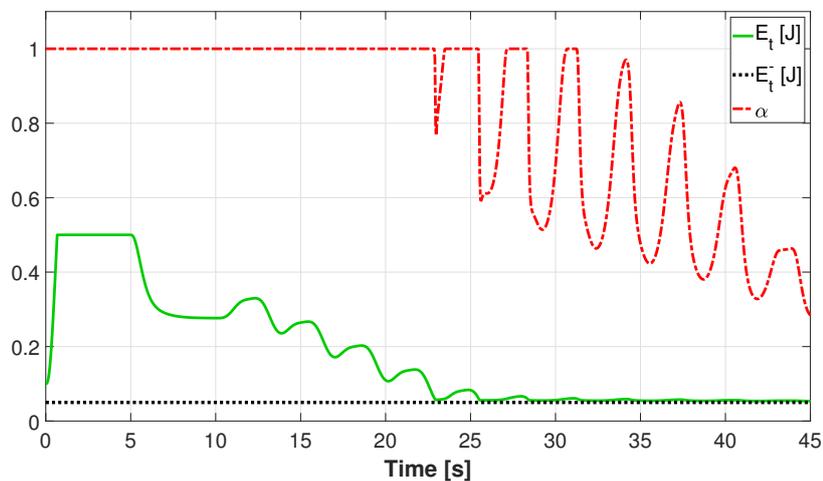
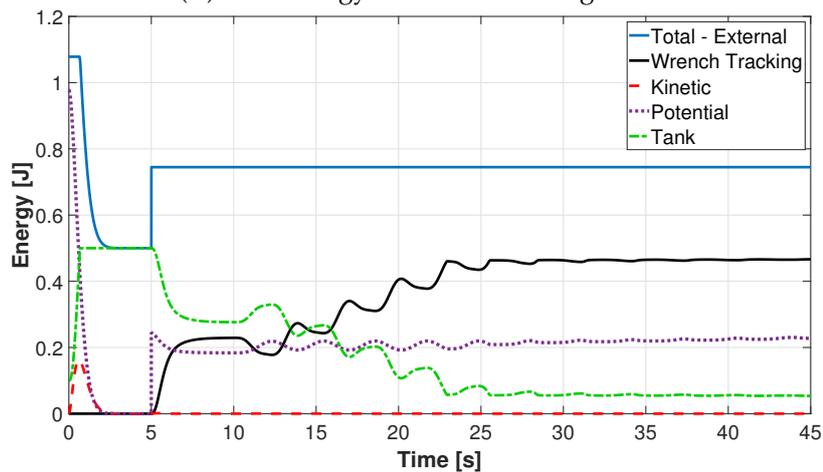
During phase 1, the approach phase, the tank energy quickly grows towards its maximum, after which it remains constant. Then the tank energy starts decreasing during phase 2, until it settles at a constant value. This makes sense, as the wrench controller only uses energy when the UAV is in motion, which is when elastic energy is being stored into the wall. When the wrench controller reaches the commanded wrench, which is constant in phase 2, it will naturally stop adding energy to the wall, and so stop using energy from the energy tank.

During phase 3, when the commanded force becomes a sinusoidal function of time, one can observe this sinusoid in the tank energy as well. This is caused by the energy exchange between the UAV and the wall due to the varying applied force.

When the UAV exerts a larger force to the wall than the wall exerts back, it will move further into the wall, requiring the wrench controller to use power from the energy tank. Conversely, when the wall exerts more force than the UAV, the UAV is pushed away from the wall, causing the wrench controller to use negative power.

However, what can also be observed in phase 3, is that on average the tank energy is gradually decreasing. Therefore, during the interaction with the wall, more energy is lost than gained on average. This can be attributed in this simulation to the physical friction in the wall. In a real experiment, however, many more effects would likely affect how much energy would be dissipated in such a scenario.

After 25s, the energy tank starts to deplete. When the tank energy gets close to zero, the valve gain indeed starts decreasing. This affects the applied normal force, shown in figure 4.4. The wrench controller is no longer able to apply the commanded normal force, as it does not have the energy available to do so. Finally, in figure 4.5b one can observe how the wrench tracking energy (the energy injected by the wrench controller) no longer increases after the energy tank depletes. In conclusion, the energy tank functions as intended when it depletes.

(A) Tank energy + resultant valve gain α 

(B) The different energy levels

FIGURE 4.5: Behavior of the energy tank and the energy levels, in case depletion of the tank occurs (scenario 2).

4.3 Conclusion

In this chapter, simulations have been performed in order to validate the operation of the energy tank-based wrench/impedance controller designed in chapter 3.

The proposed port-based implementation of the energy tank augmentation has been shown to successfully passify the closed loop UAV system. When disabling the energy tank, increases in the *"total - external"* energy could be observed during certain intervals. After enabling the energy tank, these increases were mitigated.

In an additional simulation scenario, the behavior of the energy tank near depletion has been examined. Due to the internal friction of the wall, a continuous energy exchange between the UAV and the environment led to a net loss of energy over time. This effect became visible when monitoring the tank energy, highlighting the added benefit of the energy tank in providing some kind of energy-based awareness of such physical processes. This could form an interesting topic for future research.

Apart from this short sidetrack, a correct functioning of the energy tank, when it approaches depletion, has been demonstrated. A gradual shut down of the wrench regulation can indeed be seen in that case, as was intended. This, however, raises the question of what action should be taken when the energy tank actually depletes during the execution of a certain interaction task. Energy should always be expected to be lost, as in practical situations, there is always some form of external dissipation. Therefore, it would not be a bad idea to, for instance, gradually supply additional energy to the tank over time, or re-initialize the tank after depletion has occurred. Although this would strictly not be a passive action, the injection of this additional energy could now be done in a controlled and intentional manner, rather than allowing a controller to inject energy 'as it pleases'.

In conclusion, the intended behavior of the designed control system has been demonstrated in simulation, with emphasis on the correct functioning of the augmented energy tank. It should, however, be noted that the simulation in which the validation is performed, is rather ideal. It makes use of the single rigid body hexarotor model described in §2.4.2, which is exactly the model assumed in the design of the control system. A more thorough simulation (or experiment) would be needed in order to evaluate robustness to practical phenomena, such as rotor dynamics, sensor noise, limited bandwidth, external disturbances and modeling errors.

Chapter 5

Towards Safe Interaction: A Safety Extension for Energy Tanks

Next to achieving *stable* contact with the environment for a fully actuated UAV, another research goal of this thesis is to find a way to ensure a certain level of safety during the interaction. In the context of this goal, two main safety concerns have been focused on, both of which arise when the end effector of the UAV loses contact while trying to apply a certain interaction wrench. The first concern is the high amount of *kinetic energy* that the UAV will accumulate when it attempts to apply a wrench while there is nothing to apply this wrench to, which could subsequently lead to high impact collisions. The second safety concern is the *unintended displacement* that the UAV would experience as a result of contact loss, because even low kinetic energy collisions with obstacles could be unsafe, for instance when the rotors are hit. An additional objective in the design process, is to find a solution that is effective in a multitude of scenarios, without having to adapt it to the specific scenario.

This chapter presents how an energy tank can be extended in order to deal with the above-mentioned safety issues. In §5.1 an existing solution ([Schindlbeck and Haddadin, 2015](#)) is summarized and evaluated on suitability in this context, after which an argument is made why it is too limited in dealing with different scenarios and uncertain environments. An alternative approach is described in §5.2, introducing a safety extension for the energy tank, which imposes certain energy-based safety limitations onto the system. In §5.3, the newly introduced safety extension is compared to the existing solution from literature. It is shown how certain conditions lead to equivalent behavior of the two algorithms, after which the benefits and limitations of the new solution are discussed.

5.1 Existing Solution: Contact Loss Stabilization (CLS)

As mentioned before, contact loss while attempting to apply a wrench leads to issues such as rapidly gaining too much kinetic energy, or unintended motion that could put the UAV in an unsafe situation. A solution is offered by [Schindlbeck and Haddadin \(2015\)](#) in the form of *contact loss stabilization*. It is argued that the solution is not as trivial as simply disabling wrench control at points in time when no contact is detected. Such a solution would be too sensitive to sensor noise, leading to the wrench controller rapidly switching on and off (chattering behavior).

A more robust algorithm has been proposed, that aims at solving such chattering behavior. It consists of a rotational and a translational part, but to get the idea, only the translational part will be considered here. The rotational part follows the same principle. Note that in the following explanation, a couple of blanks have been

filled in based on an interpretation of the original explanation in [Schindlbeck and Haddadin \(2015\)](#). As a matter of fact, in the original explanation, not all variables were explicitly defined.

5.1.1 Description of the CLS Algorithm

The idea of the translational part of the algorithm is illustrated in figure 5.1. An end effector is in contact with a surface at point \mathbf{p}_E . An impedance control spring tries to steer the end effector position towards desired position \mathbf{p}_D . At the same time a force controller is used to achieved the desired interaction force denoted by \mathbf{F}_d . The vector from \mathbf{p}_E to \mathbf{p}_D is denoted by $\Delta\mathbf{p}$.

$$\Delta\mathbf{p} = \mathbf{p}_D - \mathbf{p}_E \quad (5.1)$$

For the algorithm, a new variable $\psi(t)$ is introduced, which is interpreted to be mathematically defined as follows:

$$\psi(t) = \|\Delta\mathbf{p}(0) - \Delta\mathbf{p}(t)\| = \|[\mathbf{p}_D(0) - \mathbf{p}_E(0)] - [\mathbf{p}_D(t) - \mathbf{p}_E(t)]\| \quad (5.2)$$

The variable $\psi(t)$ is interpreted as the displacement of $\Delta\mathbf{p}(t)$ w.r.t. its position at time $t = 0$. This displacement basically causes the force controller to be gradually deactivated, as defined by the curve shown in figure 5.1b. This curve shows how a newly introduced variable, ρ_t , gradually decreases from 1 to 0 as $\psi(t)$ increases from $\|\Delta\mathbf{p}(0)\|$ to $\|\Delta\mathbf{p}(0)\| + d_{max}$. The resulting ρ_t is used at a later stage to scale down the force control action accordingly. The parameter d_{max} defines the distance over which this gradual decrease takes place, such that chattering behavior is prevented.

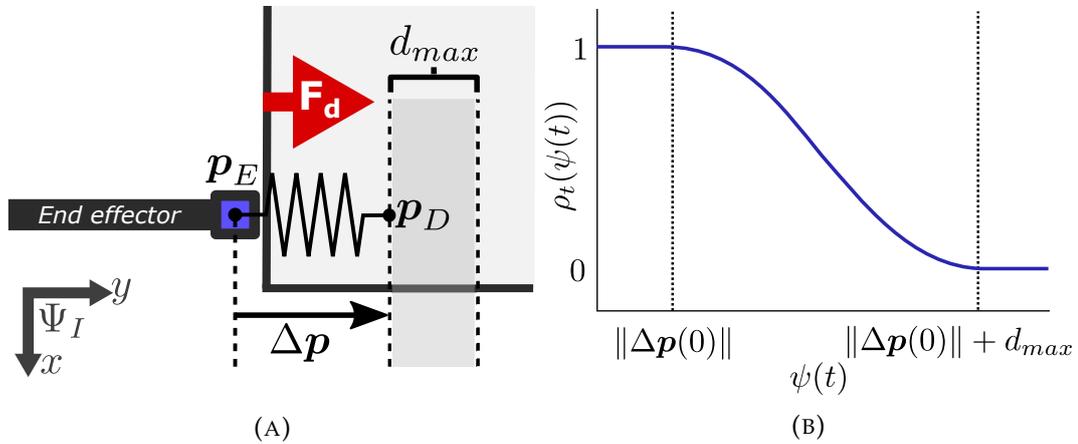


FIGURE 5.1: Illustration of the working principle of the *contact loss stabilization* algorithm introduced by [Schindlbeck and Haddadin \(2015\)](#).

The eventual algorithm has been defined by the following equations, which not only consider the displacement variable $\psi(t)$, but perform a different check as well:

$$\rho_t(\psi(t)) = \begin{cases} 1 & \text{if } \mathbf{f}_d^T \Delta\mathbf{p}(t) \geq 0 \\ \frac{1}{2} \left[1 + \cos\left(\frac{\psi(t) - \|\Delta\mathbf{p}(0)\|}{d_{max}} \pi\right) \right] & \text{if } \mathbf{f}_d^T \Delta\mathbf{p}(t) < 0 \wedge \psi(t) \in \\ & [\|\Delta\mathbf{p}(0)\|, \|\Delta\mathbf{p}(0)\| + d_{max}] \\ 0 & \text{else} \end{cases} \quad (5.3)$$

This additional check is performed because the displacement $\psi(t)$ is not necessarily the displacement in the relevant direction (which in this case is the y -direction). The check makes sure that ρ_t does not start decreasing unless $\Delta p(t)$ is in opposite direction to the desired force, i.e. $f_d^T \Delta p(t) < 0$.

5.1.2 Limitations of the CLS Algorithm

One problem that has not been addressed in the introduction of this contact loss stabilization algorithm, is the question of what point in time should be defined as $t = 0$, which determines how $\Delta p(0)$ should be defined. If it were defined e.g. at the point in time when contact is established, or when the force controller is turned on, one would need quite accurate knowledge of the geometry of the contact surface, in order to avoid problems.

To illustrate this point, consider the situation in figure 5.1, but this time including some geometrical variation in the wall surface, as illustrated by figure 5.2. Let us now consider the vector $\Delta p(0)$ to be defined as the Δp shown in this illustration, which is the starting point of this scenario. Then, as the desired position p_D is gradually moved downwards along the x -axis, the length of $\Delta p(t)$ will grow smaller and smaller. At some point, it will start pointing in the opposite direction to the desired force F_d , and thus the algorithm will start scaling down the force controller. To prevent this from happening, the position of p_D would need to be adjusted based on the shape of the wall, thus requiring accurate knowledge about the geometry of the wall surface. This knowledge would have to be especially accurate since it is desirable, for achieving safety, to set $\Delta p(0)$ as small as possible. In other words, the smaller that one sets $\Delta p(0)$, the larger the significance of geometrical variations in the wall surface becomes.

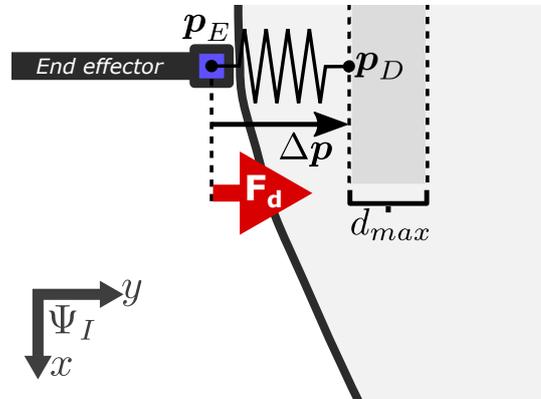


FIGURE 5.2: Example of a scenario where the CLS algorithm introduced by Schindlbeck and Haddadin (2015) would fail.

To prevent the need for accurate knowledge of the wall geometry, an alternative approach is introduced in section 5.2. In addition, the attempt is made to arrive at a more generally applicable contact loss stabilization algorithm. The algorithm should not merely work for predefined scenarios, whereby the geometrical features are accurately known, but also for scenarios that are not strictly accounted for beforehand.

5.2 Proposed Solution: A Safety Extension for Energy Tanks

In this thesis, a different method has been developed for dealing with potential contact loss when regulating the interaction wrench. In short, the proposed approach is

to detect contact loss by monitoring the energy that is drawn from the energy tank, and subsequently responding by altering the behavior of the energy tank.

This approach offers several potential benefits compared to the geometrical approach presented in §5.1. Compared to this previous method, using an energy-based approach for detecting contact loss is expected to yield a more simple and more generally applicable algorithm. The latter is to say that the algorithm can be applied to a wider variety of tasks and environments, without requiring specific adaptations or accurate knowledge about the environment. The reason for this expectation is that the detection becomes limited to the monitoring of just one single state: the energy in the energy tank, as opposed to a full 3D position and orientation. Therefore, considering energy allows safety restrictions to be defined in a more abstract way, while still maintaining a physical meaning.

Furthermore, as the energy tank already limits the energy that the wrench controller can inject, it can be easily extended to enforce stricter limitations aimed at improving safety. Therefore the proposed method is based on restricting the available energy in the tank, such that the wrench controller will only be able to inject a certain predefined safe amount of energy into the system after contact loss. However, it is not desirable in a real implementation to always limit the tank energy to such a relatively low amount of energy, because many tasks probably require a much larger total amount of energy to be executed successfully. In order to overcome this problem, one could think of ways to distinguish between energy used for executing a task, and energy lost due to contact loss.

5.2.1 Safety Violation Detection

One strategy to do this would be to keep track of the amount of power that is drawn due to wrench tracking (P_{tr}). In case of contact loss, a high amount of energy will be drawn from the tank within a short amount of time, which can subsequently be detected as a *safety violation*. A basic method for performing such a detection is to simply put a threshold on P_{tr} , where the threshold will be referred to as P_{unsafe} . A more sophisticated algorithm may be chosen to replace this in the future.

$$Safety\ Violation\ Detected = \begin{cases} True & \text{if } P_{tr} > P_{unsafe} \\ False & \text{else} \end{cases} \quad (5.4)$$

Note, however, that the simple strategy of immediately shutting off the force control action based upon such a power threshold, would be highly sensitive to sensor noise if the threshold is placed sufficiently low. Instead, this detection is considered as a starting point, after which the amount of available energy in the tank is limited to a safe amount for a certain duration. The implementation of this approach is described below, and has been conceptualized with the term *energy freezing*.

5.2.2 Energy Freezing

As stated above, to maintain enough energy for executing certain tasks while limiting the available energy after a detected safety violation, the concept of *energy freezing* is introduced. The strategy here, as illustrated in figure 5.3, is to freeze a large part of the energy in the tank as soon as a safety violation is detected. This frozen part (E_{frozen}) will not be usable, only the liquid energy (E_{liquid}) left behind can be used. After a while, the frozen energy will *melt*, such that it becomes usable again. In

this way, the actual energy content in the tank is not altered, such that the passivity of the system is maintained.

The amount of energy that is frozen will be such that the liquid energy left behind will be lesser or equal to a certain predefined safe amount E_{safe} :

$$\begin{aligned} \text{if Safety Violation Detected} &\rightarrow \text{set } E_{frozen} \text{ such that } E_{liquid} \leq E_{safe} \\ &\text{where } E_{liquid} = E_t - E_{frozen} \end{aligned} \quad (5.5)$$

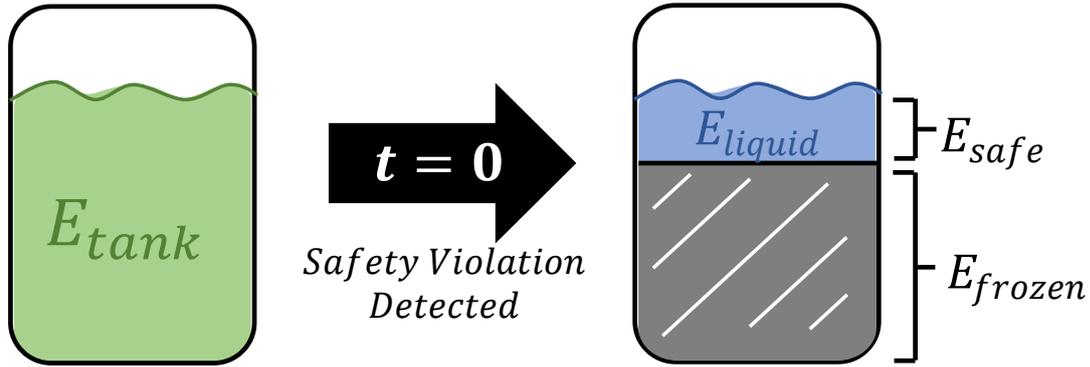


FIGURE 5.3: Illustration of the *energy freezing* algorithm.

From now on, the valve gain α will be a function of E_{liquid} instead of E_t . Furthermore, as E_{frozen} is always kept above E_t^- , one can use 0 as the lower level for E_{liquid} , instead of E_t^- . Equation 3.40 has therefore been updated to the following:

$$\alpha = \begin{cases} 0 & \text{if } E_{liquid} \leq 0 \\ \frac{1}{2} \left[1 - \cos \left(\frac{E_{liquid}}{E_{window}} \pi \right) \right] & \text{if } 0 < E_{liquid} < E_{window} \\ 1 & \text{else} \end{cases} \quad (5.6)$$

Energy Melting

Once a portion of the energy has been frozen, in order to return to the initial state, there should be a mechanism that *melts* this energy again. One case in which this becomes clear, is the false detection scenario. In this scenario, a false safety violation detection shows up, while the wrench tracking controller is being used for executing an interaction task. Such a false detection could be caused by a spike in sensor noise showing up in the velocity estimate. This spike then also shows up in the injected power P_{tr} estimate, which could exceed the threshold P_{unsafe} , causing a safety violation detection. As a consequence, a certain amount of energy freezes, leaving only a small amount of energy E_{safe} for executing the task. This energy would likely deplete shortly after, as it is continued to be used to fulfill the task. The result is that the task execution stops while the situation is not actually unsafe. If this happens too often, it becomes impossible to actually execute any tasks.

To counter this, a policy must be put in place that melts the frozen energy after it is clear that the detected safety violation was actually a false detection. The policy proposed here is to gradually melt the energy over time, the speed of which is determined by the *melting rate* parameter. To prevent that energy melting occurs during actual contact loss, the melting process is stopped if the safety extension actually

starts responding, which is when $\alpha < 1.0$.

$$\frac{d}{dt}E_{frozen} = \begin{cases} 0 & \text{if } E_{frozen} \leq E_t^- \\ 0 & \text{if } \alpha < 1.0 \\ -\text{melting rate} & \text{else} \end{cases} \quad (5.7)$$

Note that it should also be made sure that E_{frozen} does not drop below the minimum tank energy E_t^- .

The above rules for the energy freezing and melting policy have been implemented using the following algorithm:

Algorithm 1: Energy Freezing & Melting Algorithm

Inputs: E_t , Safety Violation Detected
Outputs: E_{liquid}
Parameters: E_{safe} , E_t^- , melting rate, E_{window}
Variables: E_{frozen}
Initialize: $E_{frozen} = E_t^-$;
 // Freezing:
if Safety Violation Detected **and** $E_t - E_{safe} > E_{frozen}$ **then**
 | $E_{frozen} = E_t - E_{safe}$;
end
 // Melting:
if $E_{frozen} > E_t^-$ **and** $(E_t - E_{frozen}) > E_{window}$ **then**
 | $E_{frozen} = \max(E_{frozen} - \text{melting rate} \cdot \Delta t, E_t^-)$
end
 // Output:
 $E_{liquid} = E_t - E_{frozen}$

5.2.3 Setpoint Returning

The only action that the safety extension is able to perform in its current form is to turn off the interaction wrench controller by reducing α to zero. However, as this prevents the problem of a large increase in kinetic energy, it does not necessarily solve the *unintended motion* problem.

The problem of unintended motion persists when the setpoint of the virtual spring is too far beyond the point of contact with the environment. While the modulated spring would be deactivated after contact loss by the safety extension, the passive spring would continue to steer the end effector towards the setpoint, although in a controlled way. Moving towards this configuration defined by the setpoint, could result in unsafe situations, such as having the rotors collide with an obstacle.

If this 'desired' pose should actually not be reached, some additional action has to be taken by the safety extension when responding to the contact loss. The chosen response is to move the setpoint back to the last known safe pose. The following algorithm describes how exactly this has been implemented:

Algorithm 2: Setpoint Returning Algorithm

Inputs: $\alpha, E_{liquid}, \mathbf{H}_E^i, \mathbf{H}_D^i$
Outputs: $\mathbf{H}_{D,new}^i$
Parameters: E_{safe}
Variables: $lastSafePose$
Initialize: $lastSafePose = \mathbf{H}_E^i$
if $E_{liquid} > E_{safe}$ **then**
 | $lastSafePose = \mathbf{H}_E^i$
end
if $\alpha < 1.0$ **then**
 | $\mathbf{H}_{D,new}^i = lastSafePose$
else
 | $\mathbf{H}_{D,new}^i = \mathbf{H}_D^i$
end

An alternative solution could be to already prevent the setpoint of the passive spring from moving beyond the contact point in the first place. However, the modulated spring would still require a certain minimal extension to be able to robustly apply the desired wrench. This problem could be solved by not requiring both springs to share the same setpoint. Each spring would have their own extension state, and as a result the separation of concerns would be expanded. The passive spring setpoint could be placed near the point of contact with the environment. The modulated spring setpoint could then be placed beyond this contact point, possibly at a fixed distance or rotation w.r.t. the first setpoint

While this alternative solution might increase the complexity of the user-defined input, it would remove the need for a *setpoint returning* algorithm. Not having to implement setpoint returning prevents a potential point of failure, especially in unexpected scenarios that have not been considered beforehand. However, in this work setpoint returning has been implemented nevertheless, such that the conceptual advantage of having a single spatial spring can be preserved. Whether this is actually worth it, might be an interesting consideration for future work.

5.2.4 Preventing Energy Regeneration After Suspected Contact Loss

The energy dissipated by the virtual damper, during normal operation, is routed towards the energy tank with an efficiency defined by the parameter η_d , where $0.0 \leq \eta_d \leq 1.0$.

However, during suspected contact loss, it is better to actually dissipate energy from the system, instead of allowing the wrench controller to inject it back into the system again. Dissipating all of the energy from the virtual damper is therefore desired during such unsafe situations, which can be accomplished by setting η_d to zero when the situation is potentially unsafe.

To implement this behavior, one has to define what is considered a potentially unsafe situation. It has here been defined by considering a certain duration after energy got frozen as potentially unsafe. This is implemented by checking whether the amount of liquid energy remains within a certain margin from E_{safe} . This is because, when energy is frozen, the amount of liquid energy will be lower than or equal to E_{safe} .

$$\eta_d = \begin{cases} 0 & \text{if } E_{liquid} \leq 1.1 \cdot E_{safe} \\ \eta_{d,0} & \text{else} \end{cases} \quad (5.8)$$

where $\eta_{d,0}$ is the default value for η_d , that should be used during normal operation. For a similar reason, the efficiency of the combined power routed into the energy tank, η_t , will also be set to zero in case of potential unsafety:

$$\eta_t = \begin{cases} 0 & \text{if } E_{liquid} \leq 1.1 \cdot E_{safe} \\ \eta_{t,0} & \text{else} \end{cases} \quad (5.9)$$

This efficiency has been implemented by integrating it into the law that defines the behavior for β , which causes eq 3.41 to be replaced by the following:

$$\beta = \begin{cases} 1 & \text{if } E_t \geq E_t^+ \wedge P_t^\sigma > 0 \\ 1 - \eta_t & \text{if } E_t < E_t^+ \wedge P_t^\sigma > 0 \\ 0 & \text{else} \end{cases} \quad (5.10)$$

5.2.5 System Overview

In figure 5.4, an overview can be found of the resulting design of the proposed safety extension and how it interfaces with the energy tank. Figure 5.5 subsequently shows how this structure is fitted into the overall control system.

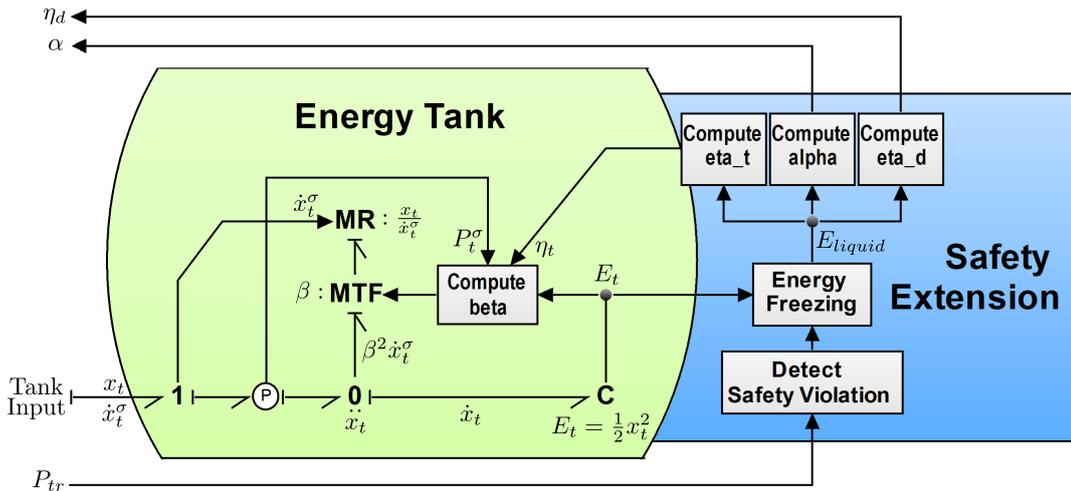


FIGURE 5.4: Overview of energy tank implementation after adding the safety extension.

5.3 Comparison

The geometry-based CLS algorithm (§5.1) introduced by Schindlbeck and Haddadin (2015) has been the main inspiration for the proposed energy-based solution (§5.2). In fact, it can be shown that both algorithms show equivalent behavior under the conditions implicitly and explicitly assumed in Schindlbeck and Haddadin (2015), suggesting that the newly proposed algorithm is indeed more general.

The equivalence follows from the realization that a force applied over a distance can be related to energy. In the current context, this can be shown by considering the power injected by the wrench tracking controller.

$$P_{tr} = (\mathbf{W}_{tr}^B)^T \mathbf{T}_B^{B,I} \quad (5.11)$$

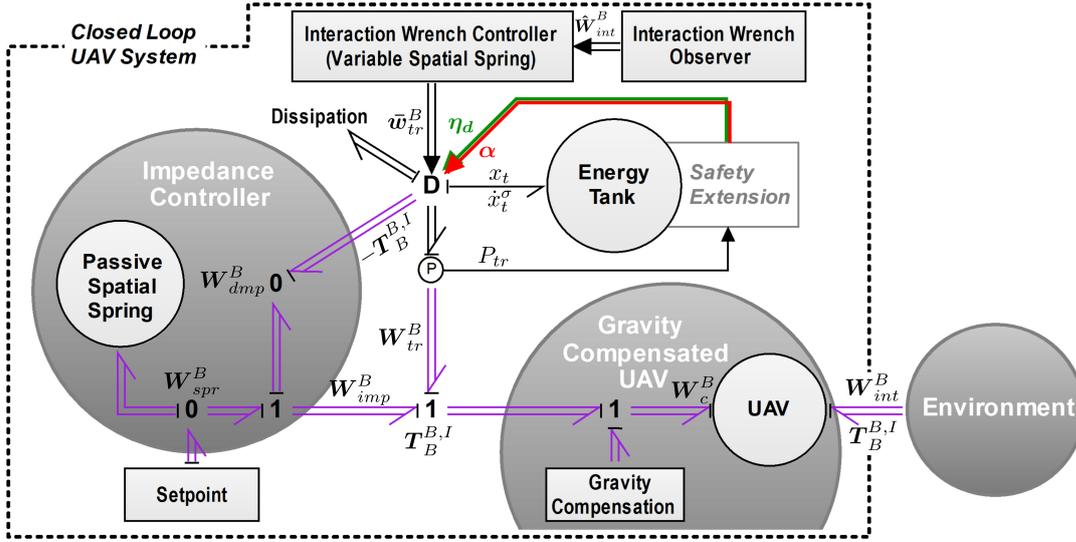


FIGURE 5.5: Overview of how the energy tank with safety extension fits in the overall control structure.

Expressing the twist and wrench in the inertial frame will not change the power. Also, note that the twist is equal for any point on a rigid body, such that $\mathbf{T}_B^{I,I} = \mathbf{T}_E^{I,I}$. Therefore the wrench tracking power can be expressed as followed:

$$P_{tr} = (\mathbf{W}_{tr}^I)^T \mathbf{T}_E^{I,I} \quad (5.12)$$

The twist and wrench can be separated into their translational and rotational components.

$$P_{tr} = (\mathbf{f}_{tr}^I)^T \mathbf{v}_E^{I,I} + (\boldsymbol{\tau}_{tr}^I)^T \boldsymbol{\omega}_E^{I,I} \quad (5.13)$$

Working out the \mathbf{v} component of the twist, by substituting $\mathbf{v} = \dot{\mathbf{p}} + (\mathbf{p} \wedge \boldsymbol{\omega})$, yields:

$$P_{tr} = \underbrace{(\mathbf{f}_{tr}^I)^T \dot{\mathbf{p}}_E^{I,I}}_{P_{tr,trans}} + (\mathbf{f}_{tr}^I)^T (\mathbf{p}_E^I \wedge \boldsymbol{\omega}_E^{I,I}) + (\boldsymbol{\tau}_{tr}^I)^T \boldsymbol{\omega}_E^{I,I} \quad (5.14)$$

In case of a pure translation, only the first term remains. The power that is injected in that case, is referred to as $P_{tr,trans}$. The energy that the wrench tracking controller injects in case of a pure translation, starting from $t = 0$, becomes:

$$E_{tr,trans} = \int_0^t (\mathbf{f}_{tr}^I)^T \dot{\mathbf{p}}_E^{I,I} dt \quad (5.15)$$

Then, to further draw the relation to the equations of the geometry-based algorithm, the steady state case will be considered where the applied control force equals the user-defined desired force \mathbf{f}_d (as was assumed in the design of that algorithm). This also requires assuming the user-defined desired force to be constant.

$$\begin{aligned} \mathbf{f}_{tr}^I &= \mathbf{f}_d \\ \dot{\mathbf{f}}_d &= 0 \end{aligned} \quad (5.16)$$

In that case, the translational part of the injected energy (eq. 5.15) can be rewritten to:

$$E_{tr,trans}(t) = \mathbf{f}_d^T \int_0^t \dot{\mathbf{p}}_E^{I,I}(t) dt = \mathbf{f}_d^T (\mathbf{p}_E(t) - \mathbf{p}_E(0)) \quad (5.17)$$

From previous analysis (eq. 5.2) it is known that $\mathbf{p}_E(t) - \mathbf{p}_E(0) = \Delta\mathbf{p}(0) - \Delta\mathbf{p}(t)$ if the desired position is constant, which has indeed been assumed. Consequently, the injected energy can be rewritten as:

$$E_{tr,trans}(t) = -\mathbf{f}_d^T \Delta\mathbf{p}(t) + \mathbf{f}_d^T \Delta\mathbf{p}(0) \quad (5.18)$$

Using this expression of the injected energy, the quantity $\mathbf{f}_d^T \Delta\mathbf{p}(t)$ in the original CLS algorithm (eq. 5.3) can now be substituted in order to express this policy in terms of the energy injected by the wrench tracking controller.

$$\rho_t(t) = \begin{cases} 1 & \text{if } E_{tr,trans}(t) \leq \mathbf{f}_d^T \Delta\mathbf{p}(0) \\ \frac{1}{2} \left[1 + \cos\left(\frac{E_{tr,trans}(t) - \mathbf{f}_d^T \Delta\mathbf{p}(0)}{\|\mathbf{f}_d\| d_{max}} \pi\right) \right] & \text{if } \mathbf{f}_d^T \Delta\mathbf{p}(0) < E_{tr,trans}(t) < \mathbf{f}_d^T \Delta\mathbf{p}(0) + \|\mathbf{f}_d\| d_{max} \\ 0 & \text{else} \end{cases} \quad (5.19)$$

This policy is exactly implemented by the proposed energy-based algorithm, when realizing that ρ_t and α are both used to scale down the wrench controller output. The value of α depends on E_{liquid} , as defined by eq. 5.6, which is repeated once more below (slightly rewritten to see the equivalence more easily):

$$\alpha(t) = \begin{cases} 1 & \text{if } E_{liquid}(t) \geq E_{window} \\ \frac{1}{2} \left[1 + \cos\left(\frac{E_{liquid}(t) - E_{window}}{E_{window}} \pi\right) \right] & \text{if } 0 < E_{liquid}(t) < E_{window} \\ 0 & \text{else} \end{cases} \quad (5.20)$$

Where the liquid energy in the tank after a safety violation detection at $t = 0$, would be described by the following expression, assuming no energy to be regenerated (which is assured after a detection due to the policy in §5.2.4).

$$E_{liquid}(t) = E_{safe} - E_{tr,trans}(t) \quad (5.21)$$

The equivalence would be complete when setting the associated parameters accordingly as well:

$$\begin{aligned} E_{safe} &= \mathbf{f}_d^T \Delta\mathbf{p}(0) + \|\mathbf{f}_d\| d_{max} \\ E_{window} &= \|\mathbf{f}_d\| d_{max} \end{aligned} \quad (5.22)$$

To summarize, when applying the assumptions made for the original CLS algorithm on the newly proposed energy-based algorithm, it will show equivalent behavior when setting the corresponding parameters as described above.

This equivalence, however, also assumes that an appropriate point in time is taken as $t = 0$, a problem not addressed in Schindlbeck and Haddadin (2015), leading to e.g. the limitation described in §5.1.2. For the newly proposed algorithm this has been addressed in a general fashion by means of the safety violation detection. Another difference is that the previous algorithm only has a meaning for pure translations and pure rotations, while the energy-based solution has a meaning for everything in between. Furthermore, the assumption of a steady state applied wrench (eq. 5.16) is not necessary anymore.

A disadvantage, on the other hand, could be that moving away from the geometrical approach compromises the ability to put explicit restrictions on displacement. There are certain scenarios conceivable in which the limitation of injected energy by wrench regulation will not prevent potentially dangerous displacements.

Broadly speaking, more generality and simplicity has been achieved when dealing with contact loss using the proposed safety extension of the energy tank, while the loss of explicit restriction on unintended displacements might turn out to form a problem in certain scenarios.

5.4 Conclusion

This chapter examined the question of how the physical interaction of a UAV with its environment can be made more safe. In this context, two specific safety concerns have been considered, both of which arise as a consequence of contact loss while the interaction wrench is being regulated. The resulting unintended displacement could lead to the rotors colliding with obstacles, while the accumulating kinetic energy could eventually lead to high impact collisions.

In this chapter, a *safety extension* of the energy tank has been introduced as a solution to tackle the above safety concerns. An existing solution conceived by [Schindlbeck and Haddadin \(2015\)](#), called contact loss stabilization (CLS), has been summarized and evaluated on suitability for solving the above problems. However, it was found to be too limited in dealing with different scenarios and uncertain environments.

Instead, a new contact loss solution has been introduced in the form of a safety extension, compatible with the energy tank implementation described in chapter 3. This solution consists of multiple algorithms, aimed at fulfilling all of the above objectives. The *energy freezing* algorithm deals with the accumulating kinetic energy problem, while the *setpoint returning* algorithm aims at reducing the unintended displacement after contact loss. It has been noted that this *setpoint returning* algorithm would not be necessary if the passive spring and the modulated spring setpoints could be moved independently from each other. Removing this shared setpoint constraint could be considered in future work.

Subsequently, it has been shown how the safety extension, when subjected to a number of assumptions, will show equivalent behavior to the original CLS algorithm, suggesting that the new solution is indeed more general.

In conclusion, a safety extension has been presented as a solution to the mentioned safety issues, that fits elegantly with the energy tank implementation designed in chapter 3. Moreover, it has been designed to be generally applicable to multiple scenarios, without requiring accurate knowledge about the environment. Compared to the previous CLS algorithm, it seemingly offers more generality, simplicity and effectiveness in uncertain environments, although it has lost the ability to put explicit limitations on unintended displacements.

Chapter 6

Simulations: Validation of the Safety Extension

In this chapter, the goal is to determine whether the safety extension designed in chapter 5 operates as intended and to what extent it satisfies the formulated objectives. This is done by performing several simulations within a simulation environment constructed using the software package *20-sim*. This environment is the same as described in chapter 4, of which the simulation and control parameters are given in appendix A.

Firstly, in order to test whether the safety extension operates as intended, a translational contact loss scenario has been simulated in §6.1, which is equivalent to the scenario on which the designed solution was based. Then, in §6.2, it is examined how wrench tracking performance is affected by false detections of safety violations, caused by sensor noise. The simulated scenario remains the same, but a significant disturbance is introduced, in the form a sharp peak in the velocity estimate. Finally, the general applicability of the safety extension is tested in §6.3, by transferring it to an alternative unsafe scenario, without changing any parameters. A scenario is simulated in which the UAV applies a torque on an object that resists motion due to rotational stiction. As soon as this stiction is overcome, the resistance drops significantly, resulting in the UAV rapidly gaining rotational velocity.

6.1 Scenario 3: Contact Loss Stabilization

As the design of the safety extension was based on the context of translational contact loss, this is the context in which its behavior will first be verified. A translational contact loss event has been simulated in this scenario, referred to as *scenario 3*. Here, the UAV approaches a wall and applies a normal force, after which contact with the surface is lost due to a lateral motion induced by a discontinuous setpoint change.

6.1.1 Description of the Scenario

Scenario 3 has been described graphically in figure 6.1. The first two phases of this scenario are similar to those of scenario 1 and 2 (see §4.1 and §4.2), although they are twice as short in duration, to help better visualize the results. In phase 1, the UAV approaches the wall with its end-effector, in an orientation normal to the surface of the wall. At the start of phase 2, the wrench controller is enabled, which starts applying a constant normal force. This requires the setpoint to be placed behind the wall. Before phase 2 ends, at 4s, the setpoint is moved in a direction tangential to the wall. This is done in such a way that the end-effector loses contact with the wall at around 5s, marking the start of phase 3.

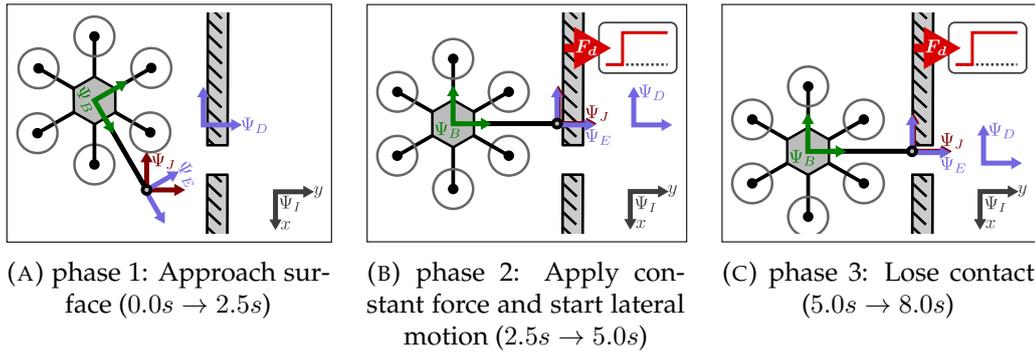


FIGURE 6.1: Description of scenario 3.

6.1.2 Explanation of the Results

In figure 6.2a the plot of the normal force can be found, from which it is indeed clear that after 5s the *actual force* instantly becomes zero due to contact loss. The results in figure 6.2 show how the safety algorithm responds to the contact loss event. Shortly after the instance of contact loss, the safety extension responds by freezing a large part of the tank energy, leaving only 0.05J of usable energy behind. Shortly after this, this liquid energy depletes completely, and thus the valve gain (α) decreases to zero.

At this point, the effect of $\alpha < 1.0$ depends on which options have been enabled for the safety extension. Three different cases have been considered:

1. No safety extension at all.
2. The basic safety extension (i.e. *energy freezing* is enabled).
3. Both *energy freezing* and *setpoint returning* are enabled.

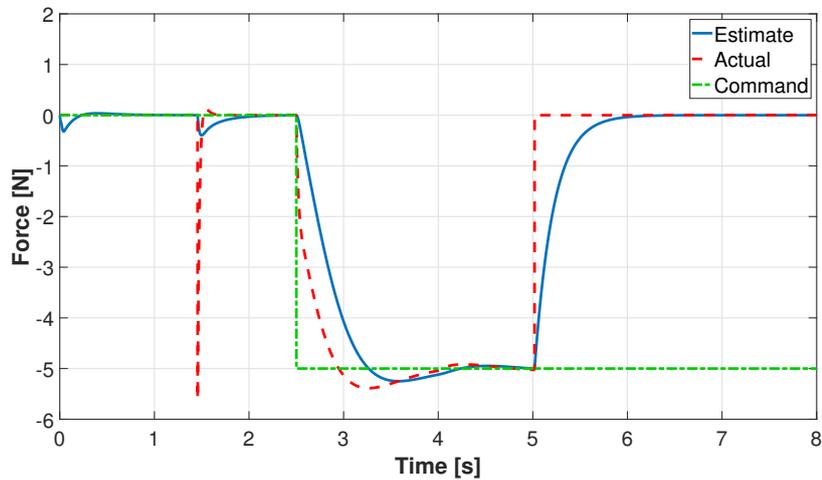
6.1.3 Discussion of the Results

The plots figure 6.4a compares the displacement that the UAV undergoes after contact loss, between the different safety options mentioned above. Clearly, the first option of having no safety extension at all, is not desirable when looking at these results. It shows the largest unintended displacement, as well as potentially unstable behavior. In figure 6.4b, the stiffness in the relevant direction is shown, which has been modulated to reach the desired normal force. For the *No Safety Extension* option, this stiffness continues to rise after contact loss, until it reaches a certain predefined maximum. In a real experiment, however, such a high stiffness could be expected to destabilize the system. In the energy levels shown in figure 6.5, the *No Safety Extension* option leads to the highest kinetic energy as well.

On the other hand, the results for the two options where the safety extension has been enabled, show that much safer behavior is indeed achieved. In terms of limiting kinetic energy, both options show a large improvement, and show similar performance to each other (see figure 6.5). In preventing unintended displacement, the option that includes *setpoint returning* performs the best. This is quite obvious, as the energy tank only disabled the modulated spring, not the passive spring. Therefore, the UAV would still attempt to move towards its setpoint, despite having detected the contact loss.

At this point we might ask the question whether it would not be better to already have the passive spring setpoint in a safe position from the start (before contact is

actually lost). To do this, the passive spring could be given a setpoint that is different from the modulated spring setpoint, instead of imposing identical setpoints. Hence, an alternative solution to *setpoint returning* could be to set only the modulated spring setpoint behind the wall, and leave the passive spring setpoint at the wall surface.



(A) Commanded, estimated and actual force

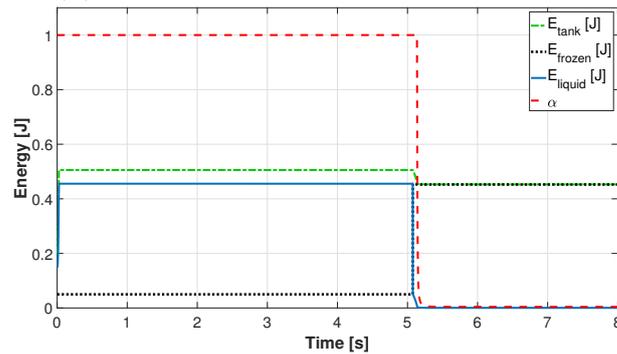
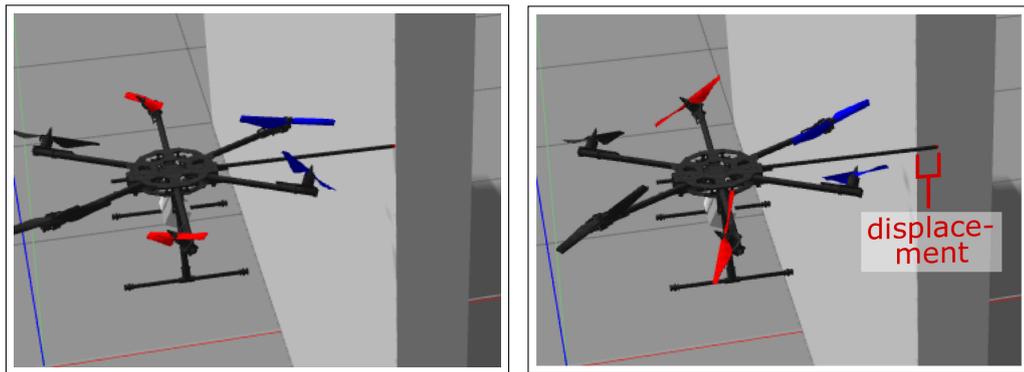
(B) Tank levels + resultant valve gain α

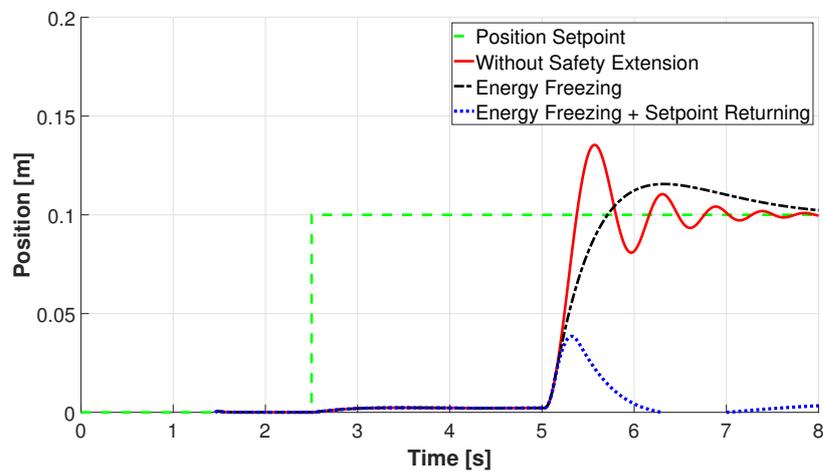
FIGURE 6.2: Behavior of the applied normal force and the energy tank safety extension in the event of contact loss (scenario 3). *Energy freezing* enabled, *setpoint returning* disabled.



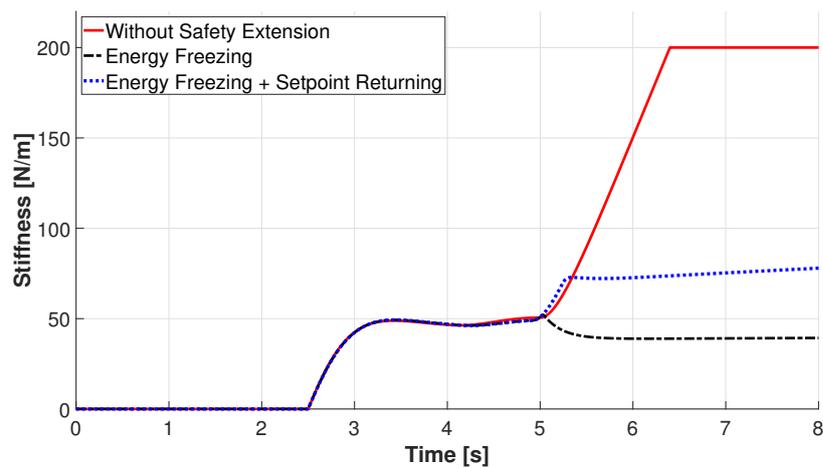
(A) Before contact loss.

(B) After contact loss.

FIGURE 6.3: Screenshot of the simulation (scenario 3), right before and right after contact loss.

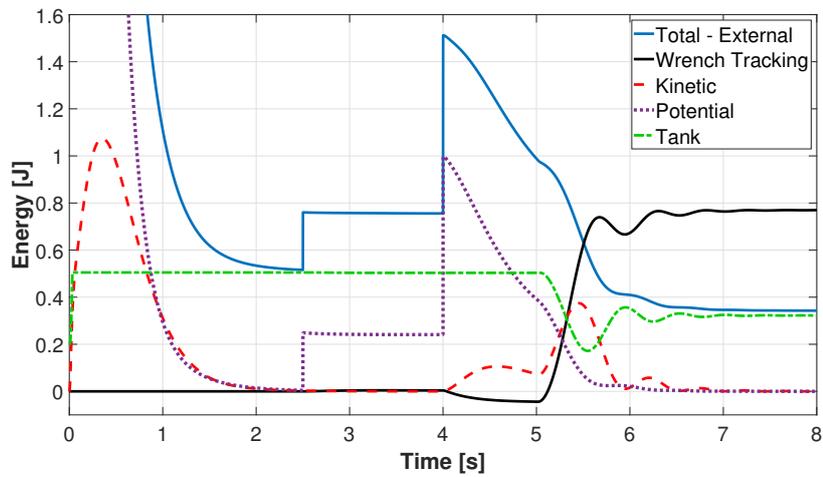


(A) The displacement of the UAV end-effector in the direction normal to the surface of the wall.



(B) The stiffness of the modulated spring in the direction corresponding to the applied normal force.

FIGURE 6.4: Comparing the behavior after contact loss between three options for the safety extension.



(A) Without safety extension

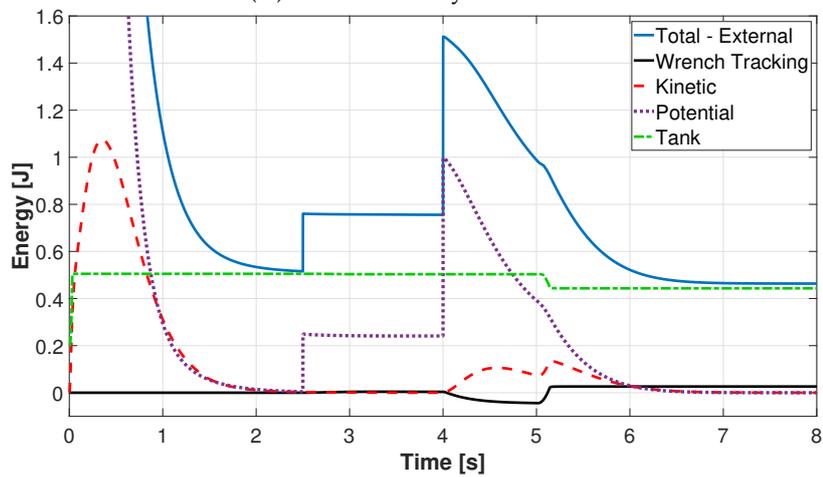
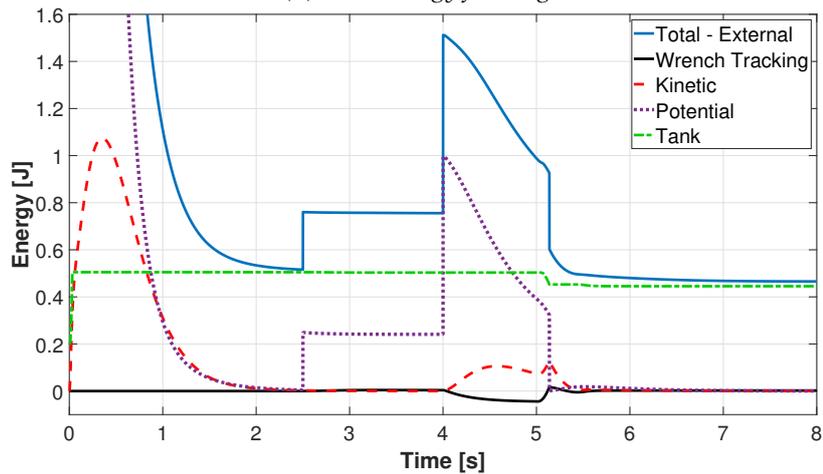
(B) With *energy freezing*(C) With *energy freezing* and *setpoint returning*

FIGURE 6.5: A comparison of the energy levels between three different options for the safety extension, in the event of contact loss (scenario 3).

6.2 Scenario 4: False Detections - Performance vs Safety

Although the safety extension has been shown to respond effectively to contact loss, it has not been investigated whether it might be oversensitive in practical scenarios. An oversensitive safety extension would turn off wrench tracking in too many cases, even when there is no contact loss or any other unsafe event occurring. This would negatively affect the UAV's performance in the case of normal task execution.

During the design phase, sensor noise had been identified as a potential cause for such false detections: if there is a spike in the velocity estimate, this will show up in the power estimate P_{tr} as well, triggering a detection. Therefore, the safety extension had been designed to essentially consist of two layers for detecting unsafe events:

1. Thresholding of the power used by the wrench controller.
2. Consequentially limiting the amount of usable (liquid) energy in the tank.

These steps require the definition of two parameters. For step 1, the power threshold parameter P_{unsafe} must be chosen. Step 2 requires defining the amount of energy that is left unfrozen (E_{safe}). The following simulations are aimed at showing how the choice of these parameters affects how the safety extension responds to peaks in the velocity estimate, in contrast to its response to real contact loss.

6.2.1 Description of the Scenario

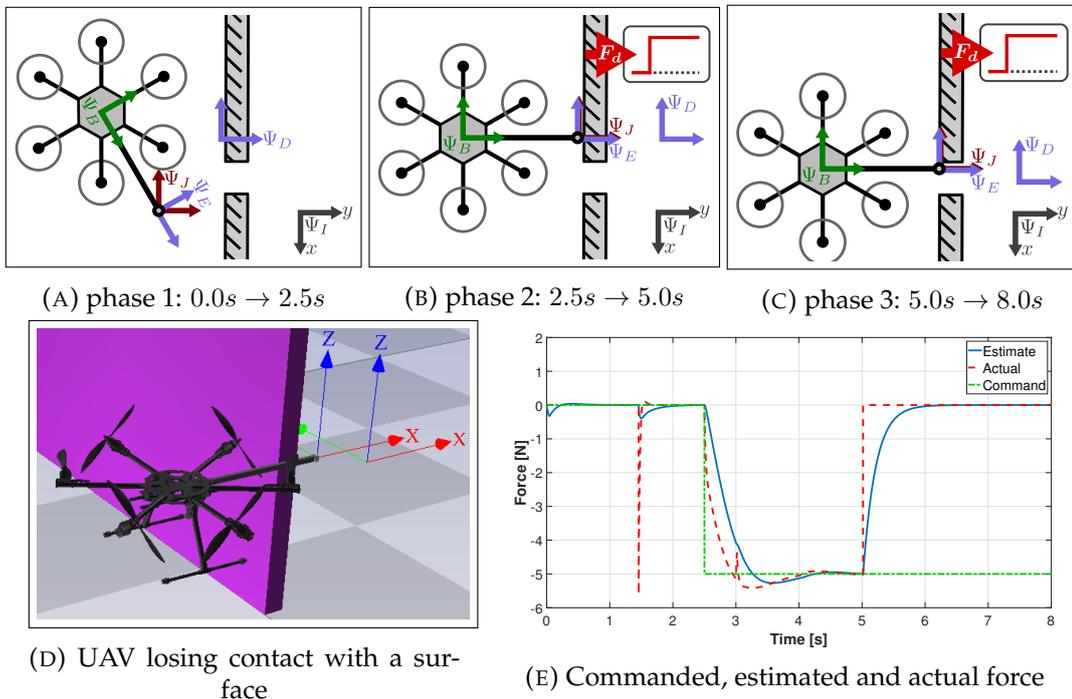


FIGURE 6.6: Description of scenario 4. It is identical to scenario 3, except that during phase 2 a significant peak is added as noise to the velocity estimate.

The simulated scenario is identical to scenario 3 (§6.1), except for the peak that is now added to the estimated velocity during phase 2. This peak adds 0.2 m/s to the velocity estimate, for $\frac{1}{60} \text{ s}$, and starts at 3.0 s . These values are chosen slightly

arbitrarily, but the point here is to illustrate merely what kind of effect such a disturbance could have on the algorithm. Two simulations have been run, to show the effect of changing the parameter E_{safe} on the response to such a sensor disturbance (figure 6.8 and 6.9. Afterwards, more values for E_{safe} have been tested, to illustrate its approximate effect on safety (figure 6.7).

6.2.2 Explanation and Discussion of the Results

The results of the first simulation, where $E_{safe} = 0.050J$, are shown in figure 6.8. At 3s, when the peak occurs, one can indeed see a response in the energy tank. A large part of the energy in the tank gets frozen, as a result of the power P_{tr} exceeding the threshold. However, as the energy left unfrozen ($= 0.050J$) is large enough, this energy does not deplete right away. Afterwards, the frozen energy starts melting gradually over time, leaving more and more usable energy for the wrench controller. Finally, when the real contact loss occurs (at 5s), the safety extension still performs as intended.

Also note that the exerted normal force is mostly unaffected by the false detection, except for the peak that shows up at 3s. This peak is mostly due to the immediate response of the virtual damper to the peak in the velocity estimate.

The results of the second simulation, where $E_{safe} = 0.015J$, are shown in figure 6.9. This time, the exerted force is clearly affected by the false detection. After the initial detection of a high power used by the wrench controller, not enough energy is left unfrozen, such that the liquid energy quickly depletes. Obviously, such a strong response to a false detection is undesirable, as it might regularly interrupt the UAV from fulfilling its task. Therefore, a too low value of E_{safe} will have a negative impact on performance in this sense.

On the other side, choosing E_{safe} too high will have an impact on safety, as more energy will be left in potentially unsafe situations, leading to a slower response to unsafe events. This can be seen in the results shown in figure 6.7, where the displacement of the UAV has been plotted over time, in the direction normal to the surface of the wall. This curve has been plotted for multiple values of E_{safe} . It can be seen how the lowest value for E_{safe} indeed results in the lowest displacement after contact loss.

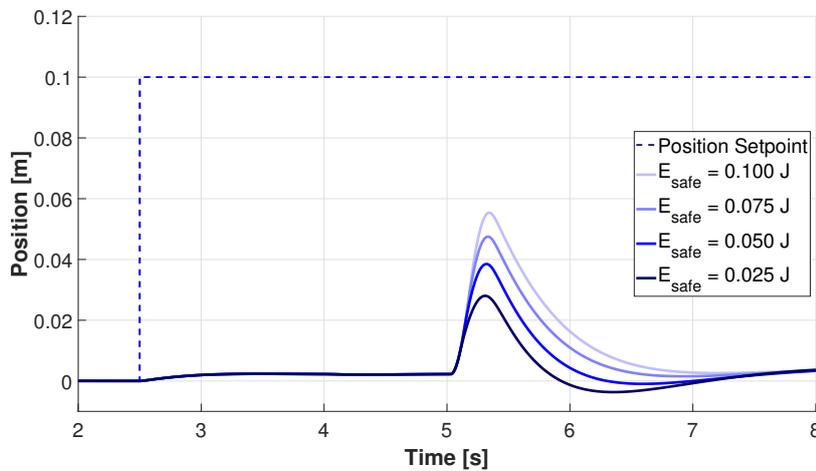
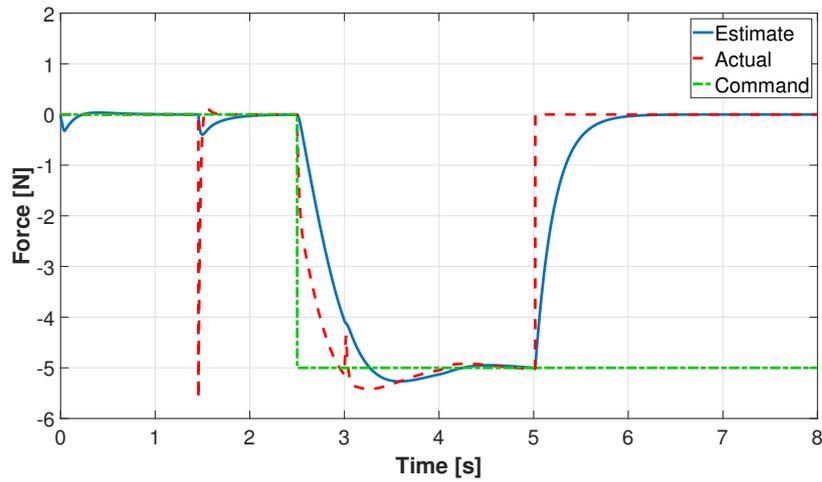


FIGURE 6.7: The effect of parameter E_{safe} on the displacement of the UAV after contact loss, in the direction normal to the surface of the wall.



(A) Commanded, estimated and actual force

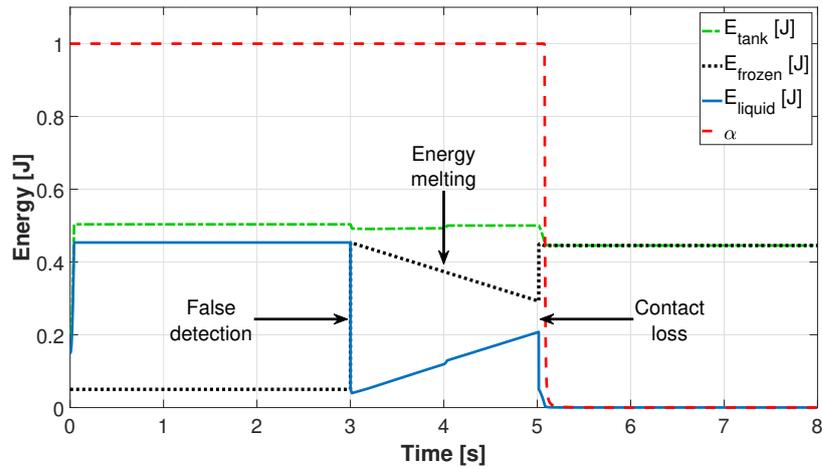
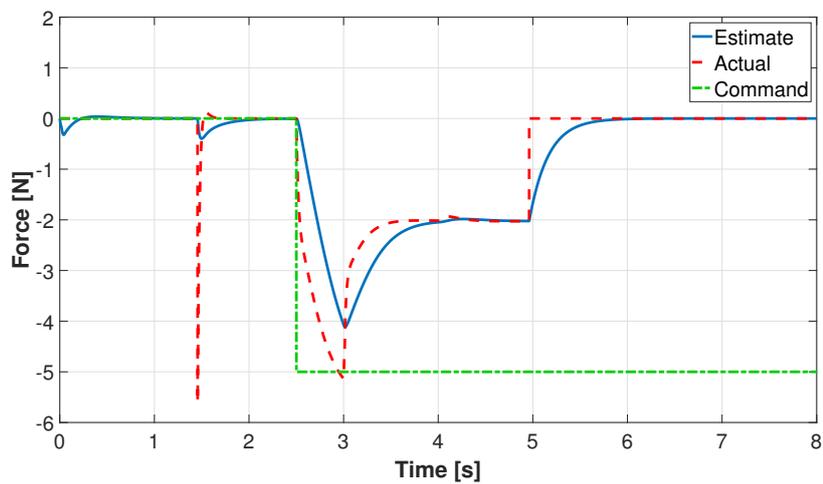
(B) Tank levels + resultant valve gain α

FIGURE 6.8: Desirable response of the designed safety extension to a disturbance that causes a false detection (scenario 4). The response to this false detection (at 3s) is such that it has no effect on the wrench tracking performance, while the response to actual contact loss (at 5s) is still effective. $E_{\text{safe}} = 0.050J$



(A) Commanded, estimated and actual force

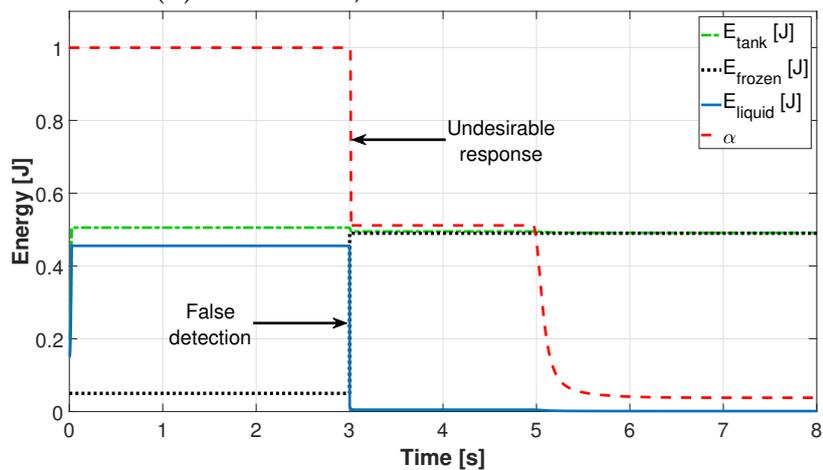
(B) Tank levels + resultant valve gain α

FIGURE 6.9: Undesirable response of the designed safety extension to a disturbance that causes a false detection (scenario 4). This time the response to the false detection (at 3s) is to partially shut down the wrench controller, illustrating the downside to choosing E_{safe} too low. $E_{\text{safe}} = 0.015J$

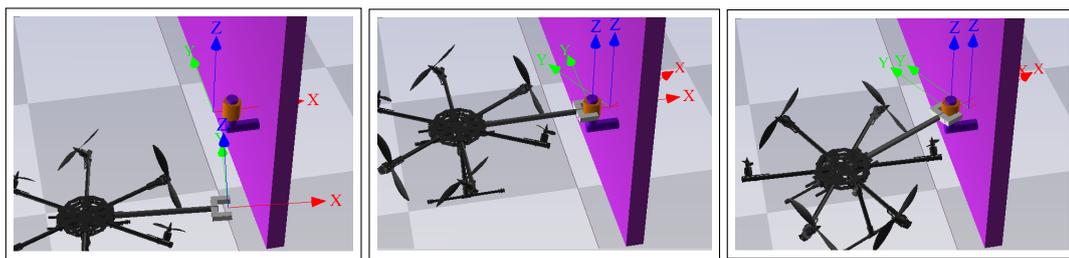
6.3 Scenario 5: Rotational Stiction

The safety extension has been demonstrated to operate as intended and deal well with the translational contact loss scenario, on which its design was based. However, as it is deemed important to find a generally applicable solution to such safety issues, it has been tested on an alternative unsafe scenario as well.

Therefore, in scenario 5, a rotational contact loss event has been simulated, in the form of a valve-opening task that involves significant stiction. In this simulation, the UAV approaches the valve handle, establishes contact with its end effector, and starts applying a torque around the z -axis in order to rotate the valve handle by overcoming the rotational stiction that resists this motion (see figure 6.10). As soon as the stiction has been overcome, all of the rotational resistance drops to a relatively low value, leading to a contact-loss type of situation. The remaining friction after overcoming the stiction has been neglected in this simulation, as it has no significant effect on the overall contact-loss behavior.

The goal of the following simulations is to demonstrate how the designed safety extension is applicable to different types of scenarios in different unmodeled environments, without having to adapt the algorithm itself.

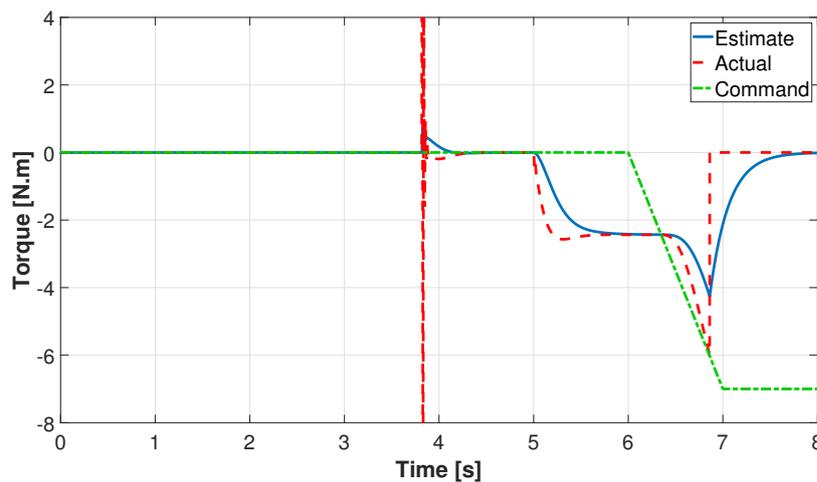
6.3.1 Description of the Scenario



(A) phase 1: 0.0s \rightarrow 5.0s

(B) phase 2: 5.0s \rightarrow 6.8s

(C) phase 3: 6.8s \rightarrow 8.0s



(D) Commanded, estimated and actual torque

FIGURE 6.10: Description of scenario 5.

Scenario 5 can be separated into three main phases, which have been visualized in figure 6.10. In phase 1, ranging from 0.0s to 5.0s, the UAV approaches the object to be manipulated. Note that a large contact peak can be seen in the interaction

torque with the environment (figure 6.10d). This is due to the contact model, which consists of highly stiff springs and dampers in the x and y direction, as well as a stiff rotational spring-damper in the rotation around the z -axis, which keep the gripper on the end effector in place. As soon as the gripper comes in contact with the object, its orientation and position are 'forced' towards the equilibrium position. However, as phase 1 does not provide any valuable information for testing of the effectiveness of the safety extension, it has been omitted in the results, making the behavior of interest better visible.

In phase 2, which ranges from 5.0s to 6.8s, the desired frame is placed at a 0.5 rad rotation around the z -axis with respect to the end effector frame. This already causes an increase in the interaction torque. Next, the interaction wrench controller starts increasing the rotational stiffness of this spring, such the interaction torque gradually increases. At the end of this phase, this torque will be high enough (i.e. 6 N·m) to overcome the rotational stiction, after which the object's resistance to rotational motion disappears.

In phase 3, which ranges from 6.8s to 8.0s, the aftermath of this type of 'contact loss' can be seen. The interaction torque with the environment becomes zero, and based on the options selected for the safety extension, the UAV gains a certain amount of kinetic energy, and thus rotational displacement as well.

6.3.2 Explanation and Discussion of the Results

As noted before, almost all of phase 1 (the approach phase) has not been included in the results, as it distracts from the information that is actually interesting for examining the effectiveness of the safety extension.

The behavior of the energy tank, in the event of rotational contact loss, has been shown in figure 6.11. Here, *energy freezing* has been enabled, and *setpoint returning* disabled. The behavior is as expected: it is similar to the translational contact loss scenario. However, just before contact loss, already a minor amount of energy is lost, as the applied torque is being increased, probably due to the compliance of the environment. In the translational contact loss simulations (figure 6.2) this effect was not visible, likely due to the additional energy regenerated from the virtual damper (because the UAV was translating sideways at that point in time).

The rotational contact loss response of the rotation around the z -axis, θ_z , can be found in figure 6.12. Three different safety extension settings have been compared here, which are: *no safety*, *energy freezing*, and *energy freezing + setpoint returning*. In the case where the safety extension is disabled, θ_z reaches the setpoint much faster, after which oscillations can be seen. During this process, the stiffness of the variable impedance spring keeps being increased by the wrench tracking controller, which in practical situation would soon lead to instability. In the case where just *energy freezing* is enabled, θ_z still reaches the setpoint, but much slower and in a more controlled way. When adding *setpoint returning*, obviously most of this rotational displacement is prevented.

The energy levels in the system for the three different safety extension settings are shown in figure 6.13. First of all, it can be seen that the system remains passive for each setting, as can be expected due to the energy tank itself still being enabled. The passivity can be observed because there is no increase in the "total - external" energy level, when ignoring the increase due to the setpoint change at 5s.

Secondly, it is clear that the kinetic energy becomes much higher when the safety extension is disabled, compared to the other two cases. A relatively large amount of energy can be drawn from the energy tank and converted to kinetic energy. In

both cases where the safety extension is enabled, the kinetic energy is limited to roughly equal amounts. However, in case of no *setpoint returning*, this kinetic energy is sustained much longer. This can be expected, as the passive virtual spring can continue to supply energy to the system, as shown by the *potential* energy curve.

In conclusion, the safety extension, despite being designed in the context of translational contact loss, transfers well to a rotational contact loss scenario. In future work, it would be interesting to test its effectiveness in many other types of environments, and while performing more complex tasks.

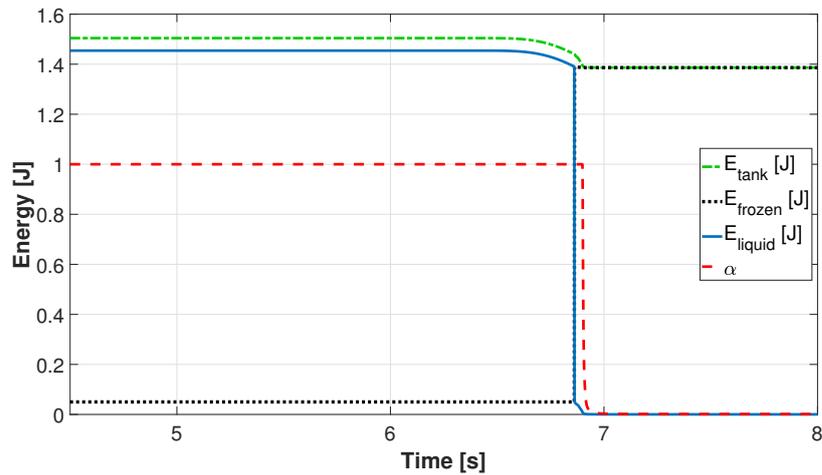


FIGURE 6.11: Behavior of the energy tank safety extension in the event of rotational contact loss (scenario 5). *Energy freezing* enabled, *setpoint returning* disabled.

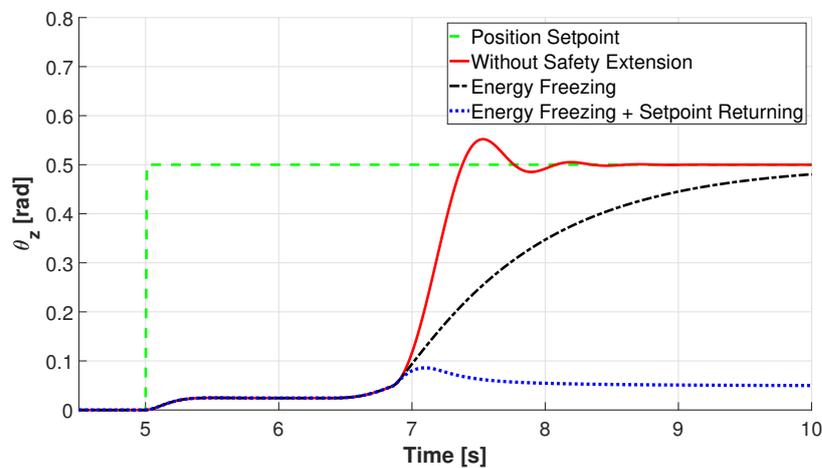
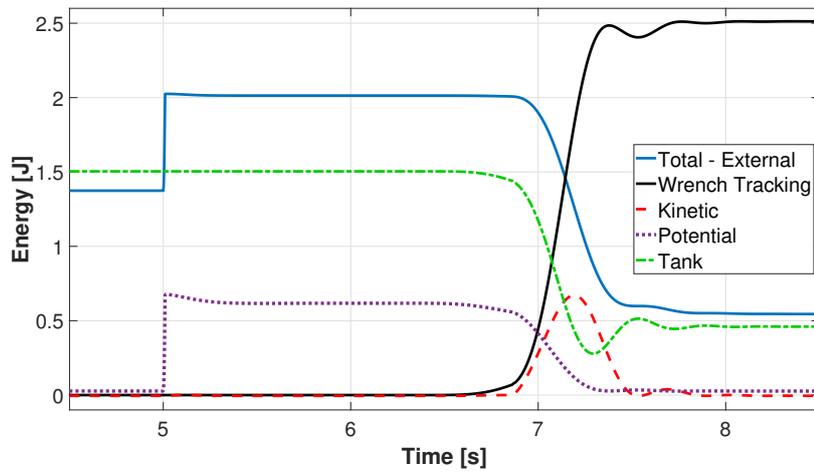


FIGURE 6.12: Comparing the rotational displacement of the UAV end-effector around the z-axis, θ_z , in the event of rotational contact loss. A comparison between three options for the safety extension.



(A) Without safety extension

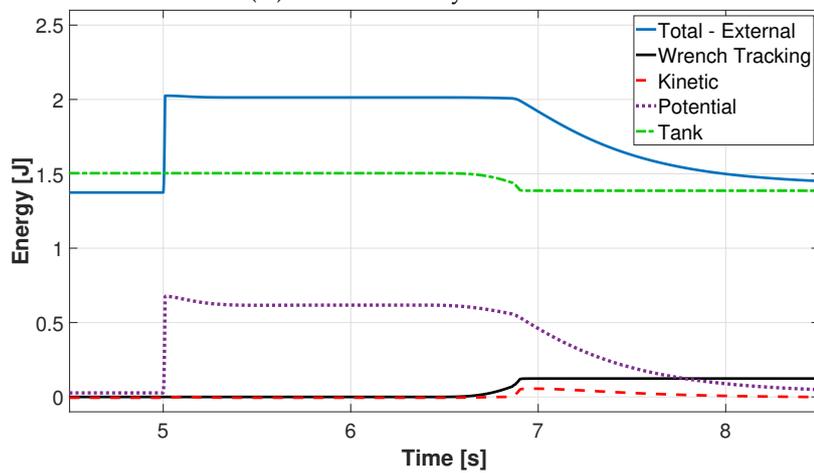
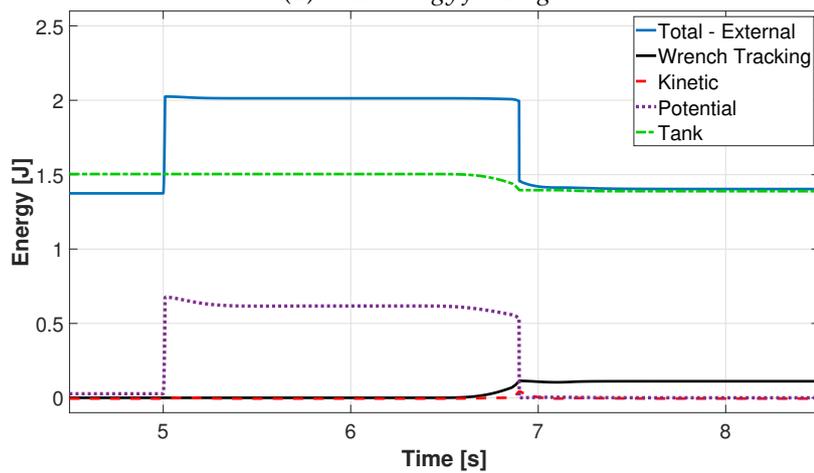
(B) With *energy freezing*(C) With *energy freezing* and *setpoint returning*

FIGURE 6.13: A comparison of the energy levels between three different options for the safety extension, in the event of rotational contact loss (scenario 5).

6.4 Conclusion

The focus of this chapter has been to validate the safety extension designed in chapter 5. This safety extension is supposed to deal with the safety issues that arise when the robot starts regulating the interaction wrench with the environment. It has been noted that, after the event of contact loss, the robot should be prevented from gaining a large amount of kinetic energy, as well as from displacing too much with respect to its intended pose. Next to being effective in this regard, other design objectives include general applicability, and affecting the wrench tracking performance during normal (safe) operation as little as possible. The safety extension has been tested on each of these aspects, using three different simulated scenarios.

In scenario 3, the effectiveness of the safety extension has been evaluated in the context of translational contact loss. It is shown to greatly limit the maximum kinetic energy that the UAV attains, as well as reduce the resulting displacement (in case *setpoint returning* is enabled). Furthermore, the argument has been made that the use of *setpoint returning* would not be necessary if the two virtual spatial springs were not forced to share the same setpoint. In that case, the passive spring setpoint could be placed at (or close to) the point of contact, while using only the variable impedance spring to apply the desired wrench.

In scenario 4, the effect of sensor noise on the safety algorithm has been tested. This was done because it had been identified that a large peak in the velocity estimate could lead to a false detection of contact loss. In such a situation, it would not be desirable for the safety extension to immediately shut off all wrench tracking capabilities as a response. In the simulation, a peak in the velocity estimate has been added, and it is shown how the algorithm does not respond to such a false detection. This is, however, shown to be dependent on how a certain parameter (E_{safe}) has been chosen. In a way, this choice has been shown to lead to a trade-off between safety and performance. Choosing this value higher leads to more displacement after contact loss, while choosing it too low leads to too many responses to false detections.

In scenario 5, the focus was on finding out to what extent the designed safety extension is generally applicable, meaning effective in different types of scenarios and environments. To this end, a different environment has been constructed in the simulation, where the UAV could apply a torque to an object, that resists rotation until a certain interaction torque has been reached (e.g. due to stiction), resulting in rotational contact loss. The exact same version of the safety extension has been used for this scenario as for the previous scenarios, meaning no parameters have been tuned and no information about the environment has been processed. The results have shown that the effectiveness of the safety extension transfers well to a rotational contact loss scenario, despite it being designed in the context of translational contact loss.

In conclusion, it is shown that the designed safety extension can offer an effective solution to contact loss. Furthermore, it proves to be transferable to at least one alternative unsafe scenarios, and shows a certain level of robustness to disturbances in the velocity estimates. Future work lies in testing in many other types of environments, and also while performing more complex tasks. Furthermore, more realistic circumstances could be created, where a limited bandwidth, rotor dynamics, sensor noise, and other disturbances play a role. On top of that, real experiments are needed for further validation, which might expose issues related to such practical conditions.

Chapter 7

Simulations: More Realistic Conditions

Up to now, the validation of the proposed methods for achieving stable and safe interaction have been performed in a rather ideal simulation environment. However, when considering practical implementation, it is important to get an idea of the effectiveness of the designed solution in less ideal circumstances. For example, currently unmodeled behavior such as motor dynamics, aerodynamic effects, delays and other disturbances might significantly affect the actual control wrench that the UAV experiences. If this control wrench differs too much from the assumed control wrench as outputted by the control system, the passivity might be compromised. Regarding the safety extension, conditions such as a limited bandwidth and delays are expected to increase its response time to safety violations, hence reducing its effectiveness.

In order to validate the proposed design more thoroughly, the original plan was to perform several experiments on a real setup that involves the actual hexarotor UAV. However, as mentioned before, unforeseen circumstances have prevented the execution of these experiments, despite having them ready to be performed. Therefore, the choice has instead been made to perform this additional validation in a more realistic simulation environment, that includes a part of the aforementioned practical conditions.

These more realistic simulations have been performed within the *Gazebo* simulation environment (see figure 7.1), using the *RotorS* simulator framework (Furrer et al., 2016). In comparison to the UAV model simulated in chapters 4 and 6, the *RotorS* package includes motor dynamics for each rotor, as well as additional aerodynamic effects, following the model proposed in Martin and Salaun (2010). Furthermore, the ground truth values of the UAV's twist ($\mathbf{T}_B^{I,I}$) and pose (\mathbf{H}_B^I) are retrieved from the simulation at a limited bandwidth: in this case 25 Hz. Note that in case of a real experiment, a state estimator would be used here instead, that fuses IMU and optical tracking data.

The proposed control system, which was designed within a port-based framework, has been translated into discrete-time algorithms implemented in C++. The used control and simulation parameters remain identical to those used in the previous simulations, as summarized in the table in appendix A. However, the contact between the end effector and the wall surface is now simulated by the default gazebo contact model.

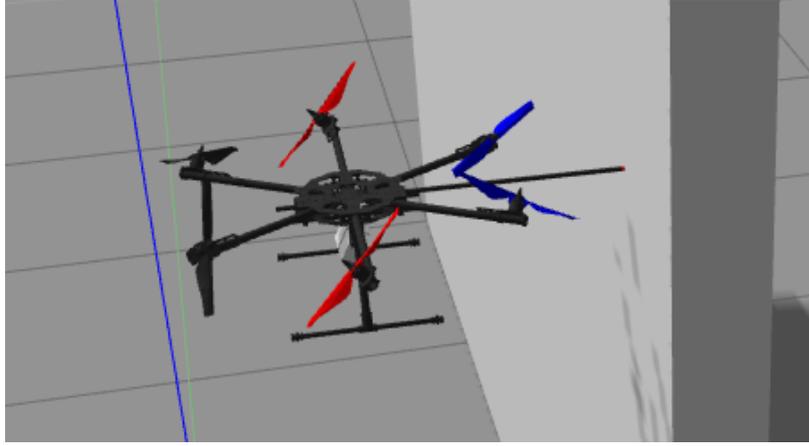


FIGURE 7.1: Screenshot of the *Gazebo* simulation environment, using the *RotorS* UAV simulation framework.

7.1 Validation of the Energy Tank-Based Controller

A correct functioning of the designed interaction controller has been investigated, including its behavior after the tank depletes. To this end, the energy in the energy tank has been initialized at only 0.2 J. The simulation has been executed in a similar fashion to the steps described by *scenario 1* (in §4.1). The illustration of these steps has been repeated in figure 7.2.

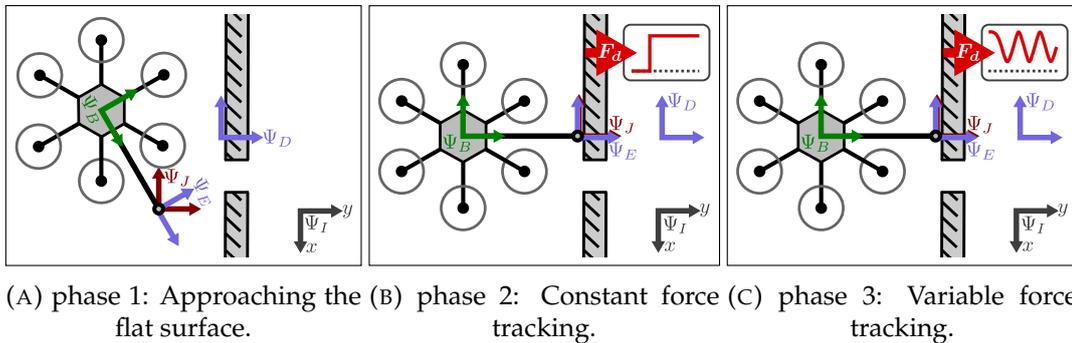


FIGURE 7.2: Description of scenario 1.

7.1.1 Explanation and Discussion of the Results

The resulting behavior of the energy tank, as well as the commanded and observed interaction force, can be found in figure 7.3.

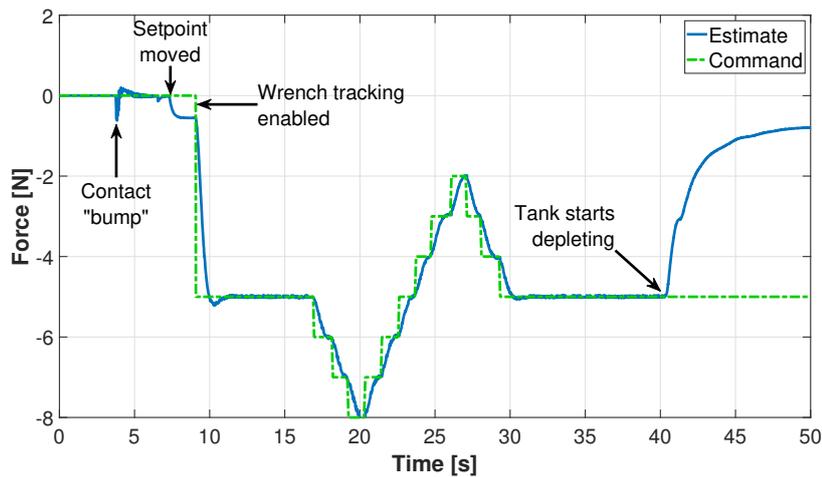
After about 40s, the energy tank starts to reach depletion, causing the valve gain (α) to decline as well. Subsequently, the observed interaction wrench gradually decreases, as should be expected. Also note how the rate at which the tank depletes becomes lower over time. This follows the expectations, because a lower valve gain also leads to a decrease in the wrench tracking power (P_{tr}) drawn from the tank.

One unexpected behavior is the steady decrease of tank energy over time, despite the fact that almost no energy is exchanged with the environment (the wall has almost no compliance). Upon further investigation, the dominant cause for this decline has been found in the twist estimate. To illustrate the problem, a 1D velocity is considered instead of a twist, such that the point can be made more clearly. The

plot in figure 7.4 shows the estimated UAV x -velocity - here the direction normal to the wall - while it applies a constant normal force of 5N and remains in steady state. However, despite the fact that the UAV does not move in this direction, the velocity estimate shows a significant bias. A moving average of this velocity shows that a displacement of more than 1 mm should be expected every second, but clearly the displacement remains zero. Therefore, if the applied tracking force $F_{x,tr} = 5$, this bias would after 20 seconds cause about $0.001 \text{ m/s} \cdot 5 \text{ N} \cdot 20 \text{ s} = 0.1 \text{ J}$ of energy drawn from the tank. This indeed matches the rate of energy loss that is actually observed.

This bias in the twist measurements is already present in the simulated ground truth values given by the *gazebo* environment, and is only present during physical interaction with the wall. It is therefore attributed to some kind of numerical phenomenon in the simulated contact. This problem would not show up in the twist estimate if it were coupled to the pose estimate (assuming no drift in the pose estimate). In this way, such a bias could be mitigated.

After solving this problem, information about the energy flows involving the physical interaction might start to become visible in the energy tank, as was shown to be the case in a previous simulation (§6.1). In practical implementations, this will obviously depend on the quality of the sensor measurements and the subsequent state estimation.



(A) Commanded and estimated interaction force

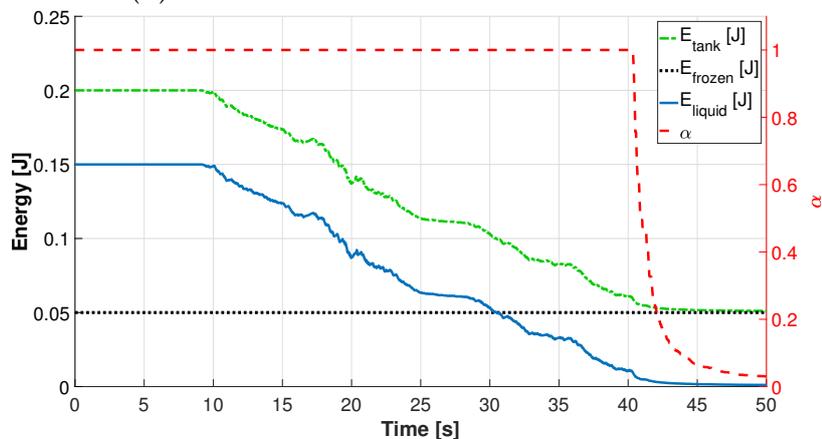
(B) Tank energies + resultant valve gain α

FIGURE 7.3: Behavior of the energy tank and the resulting observed interaction force, before and after the energy tank depletes.

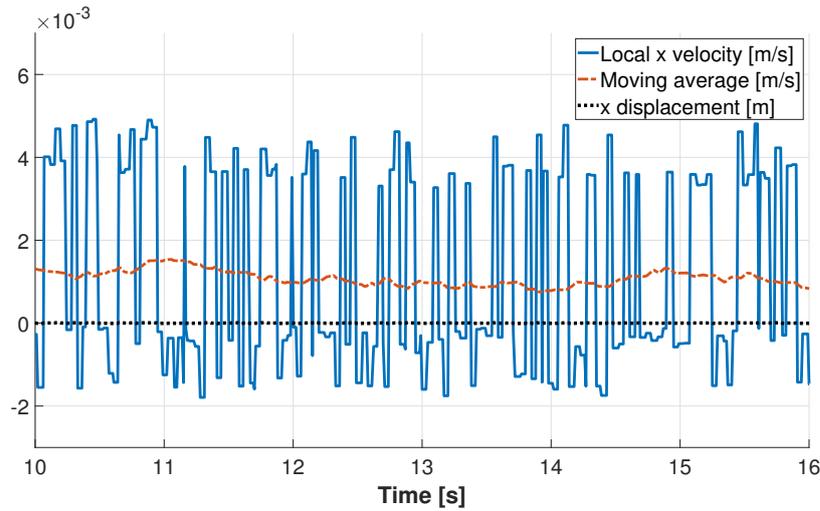
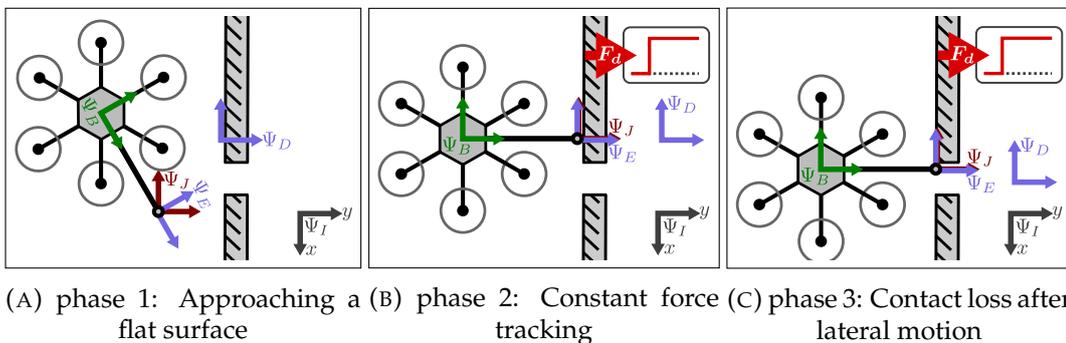


FIGURE 7.4: A significant bias in the local x velocity estimate of the end effector while in steady state, when a 5 N force is applied to the wall surface.

7.2 Validation of the Safety Extension

The performance of the designed safety extension can be expected to decrease in the presence of conditions that increase its response time. In the current simulation environment, such conditions include the limited bandwidth (25 Hz) on which the sensors operate and the delay caused by the additional rotor dynamics.

In the following simulations, it is investigated how the behavior of the UAV in the event of contact loss is affected by moving towards such less ideal conditions. The steps described by *scenario 3* (in §6.1) have been followed, such that the results of both simulation environments can be directly compared. The description of scenario 3 has been shown once more in figure 7.5. It starts with an approach to the surface of the wall, after which the desired frame is placed behind this surface. Then the interaction wrench regulation is enabled, which realizes a commanded normal force of 5 N. Finally, the desired frame is moved laterally, such that a while later the UAV's end effector loses contact with the wall surface. This behavior can be seen in the resulting estimated force plot in figure 7.6.



(A) phase 1: Approaching a flat surface (B) phase 2: Constant force tracking (C) phase 3: Contact loss after lateral motion

FIGURE 7.5: Description of scenario 3.

7.2.1 Explanation and Discussion of the results

The resulting response has been shown for two different settings: first with the full safety extension enabled (including *energy freezing* and *setpoint returning*), and then without safety extension. Each plot shows a direct comparison between the current *RotorS* simulation and the previous, more ideal *20-sim* simulation. Their results have been synchronized based on the real moment of contacts loss. Also note that, as shown in figure 7.6, the results only focus on the part before and after contact loss

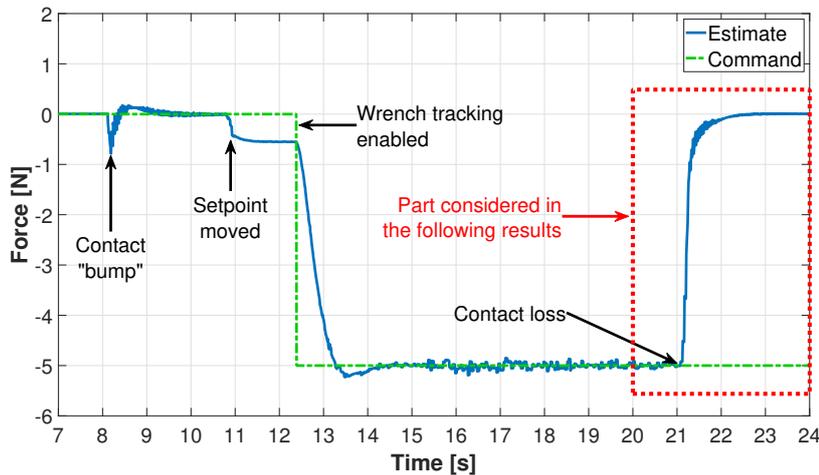


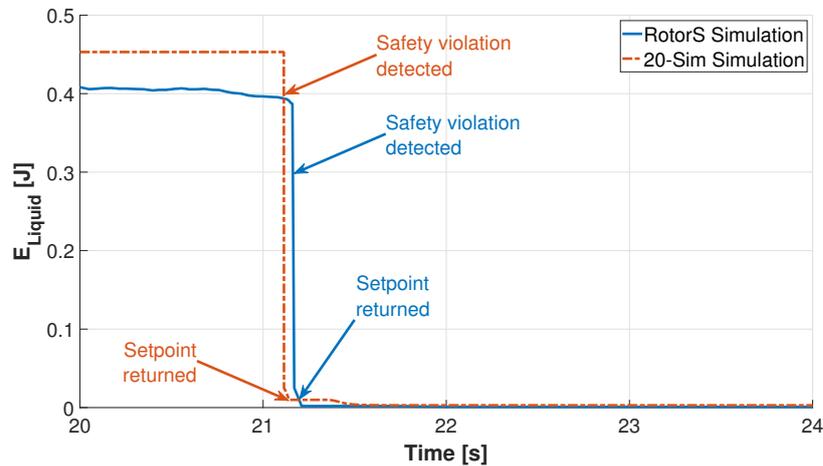
FIGURE 7.6: Commanded and estimated normal force, applied by the wall on the end effector.

Figure 7.7 shows the results in case the full safety extension has been enabled. The resulting liquid energy is shown for both simulations in figure 7.7a, such that the differences in detection and response time become visible. For both, the *RotorS* simulation shows a reaction that is about 0.05 s later. This can be expected, as this is just slightly more than one time-step regarding the sensor bandwidth.

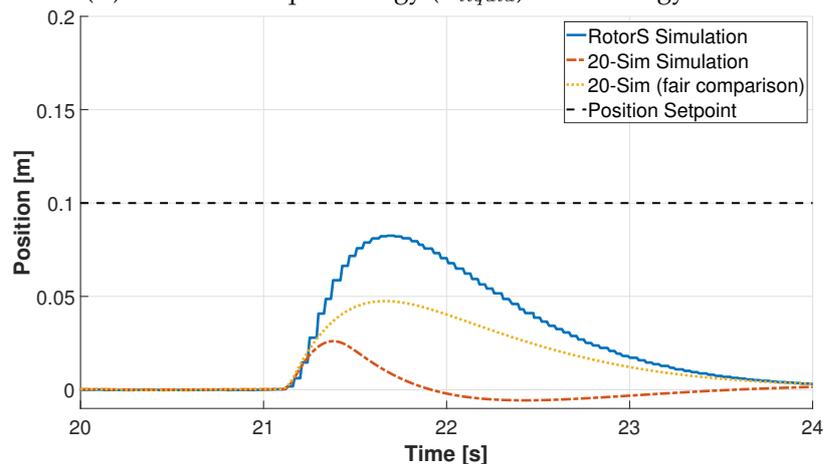
The resulting displacement (figure 7.7b) is clearly higher in case of the *RotorS* simulation. However, after further investigation the safety extension turned out to have an unexpected added enhancement in the *20-sim* simulation. This has to do with the *setpoint returning* algorithm: as soon as $\alpha < 1.0$, the setpoint is placed back to the last safe position. However, this also means that the energy tank shortly stops from being depleted, as the modulated spring force is now applied in the opposite direction. Therefore, the much stiffer modulated spring adds a force component that counteracts the displacement motion. In the *RotorS* simulation, only a fraction of this additional counteracting force is applied, due to the discrete time-steps causing the energy tank to deplete almost instantly before the setpoint is moved back. To make the comparison more "fair", a third simulation result has been added to this plot, showing how the *20-sim* simulation performs without this advantage.

The contact loss behavior in case of no safety extension can be seen in figure 7.8. Moving to the *RotorS* simulation has clearly led to instability when the safety extension is disabled. Strictly speaking, the system should behave passively, and in this case the energy tank should take care of enforcing this, which is why such instability should not be expected. In contrast, the *20-sim* simulations still show a gradual dampening of the oscillations, although this can be attributed to the rather ideal conditions. Apparently, in the more realistic case, certain practical factors prove capable of compromising the passivity after impactful events such as contact loss. Probably, the amount of energy in the tank is not small enough to prevent the control wrench

and associated errors from growing beyond control. For example, it was observed that the initial oscillations eventually led to control forces high enough to cause actuator saturation, which can subsequently lead to a large difference between the assumed and actual control wrench, compromising the passivity. To prevent this, one could enforce a limited control wrench at an earlier stage in the controller, such that this difference do not arise.

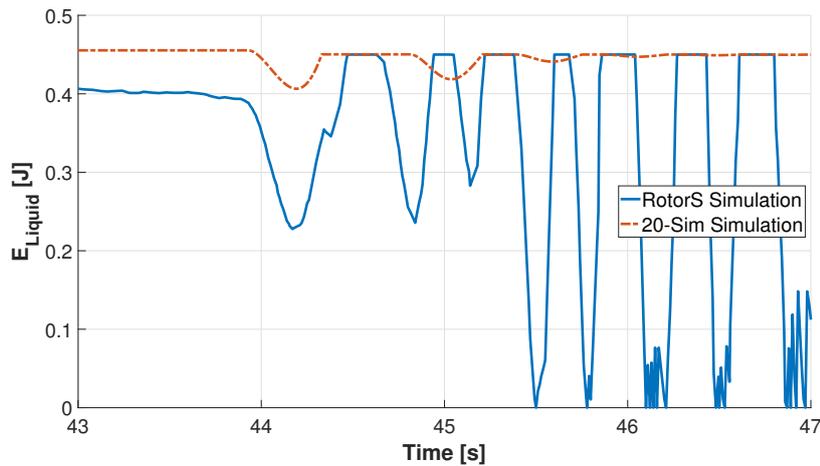
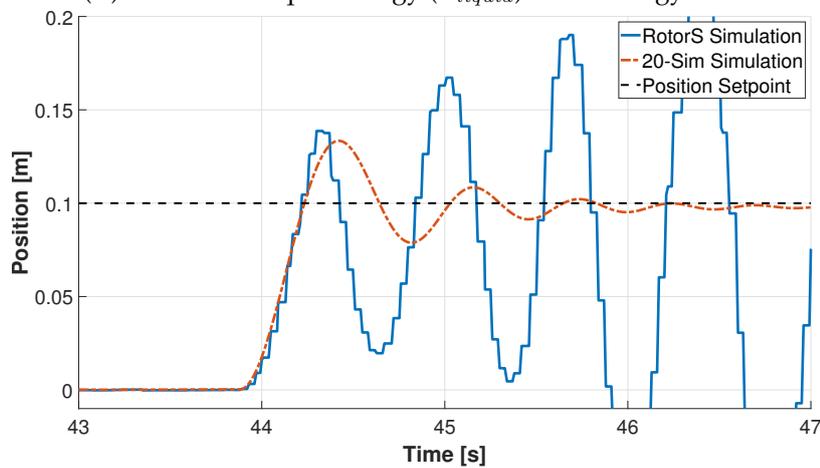


(A) Amount of liquid energy (E_{liquid}) in the energy tank.

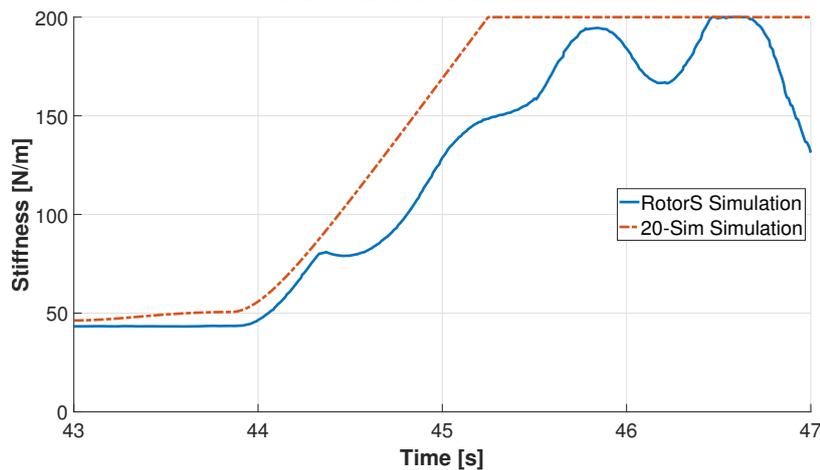


(B) The displacement of the UAV end-effector in the direction normal to the surface of the wall.

FIGURE 7.7: Behavior in the event of contact loss (scenario 3), when the safety extension is enabled (*energy freezing + setpoint returning* enabled): A comparison between the more realistic simulation results ("*RotorS*") and the more ideal simulation ("*20-Sim*").

(A) Amount of liquid energy (E_{liquid}) in the energy tank.

(B) The displacement of the UAV end-effector in the direction normal to the surface of the wall.



(C) The stiffness of the modulated spring in the direction corresponding to the applied normal force.

FIGURE 7.8: Behavior in the event of contact loss (scenario 3), without safety extension: A comparison between the more realistic simulation results ("RotorS") and the more ideal simulation ("20-Sim").

7.3 Limitations of the Safety Extension due to Sensor Noise

In this section, it is examined how noise in the twist estimate ($T_B^{B,I}$) affects the safety extension. In a practical implementation, this noise could cause false detections, and subsequently a false response (i.e. shutting down of the wrench controller when there is no real risk). Therefore, the goal is to find under what noise conditions the safety algorithm would still function properly.

This has been done by adding Gaussian noise to each element of the twist estimate, which runs at 100 Hz. The noise amplitude is defined by a certain standard deviation (SD). Subsequently, it has been identified that only two parameters of the safety extension affect the probability of a false response: the *Melting Rate* and the difference between E_{safe} and E_{window} .

The effect of these parameters on the amount of noise required to cause a false response, while applying a constant normal force of 5 N, has been shown in figure 7.9. Increasing the applied force would decrease the allowed noise level proportionally.

An example of the process of acquiring these results can be found in figure 7.10. Here one can observe the effect of such noise in the twist estimate on the energy freezing and melting behavior, which ultimately determines whether or not the safety extension will respond.

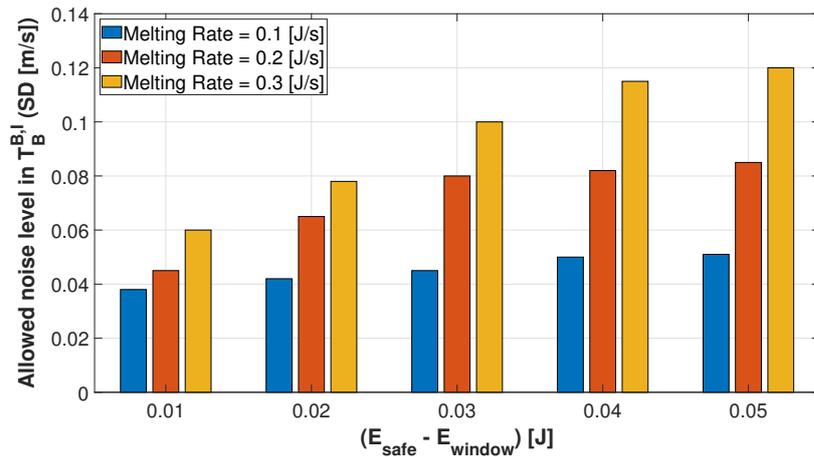
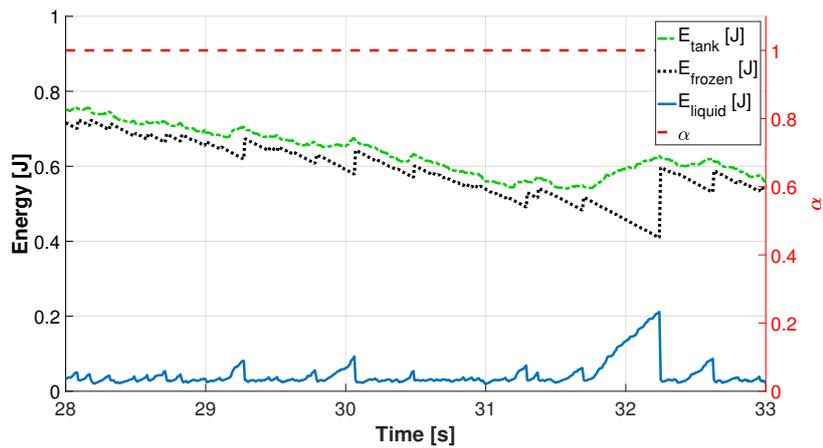
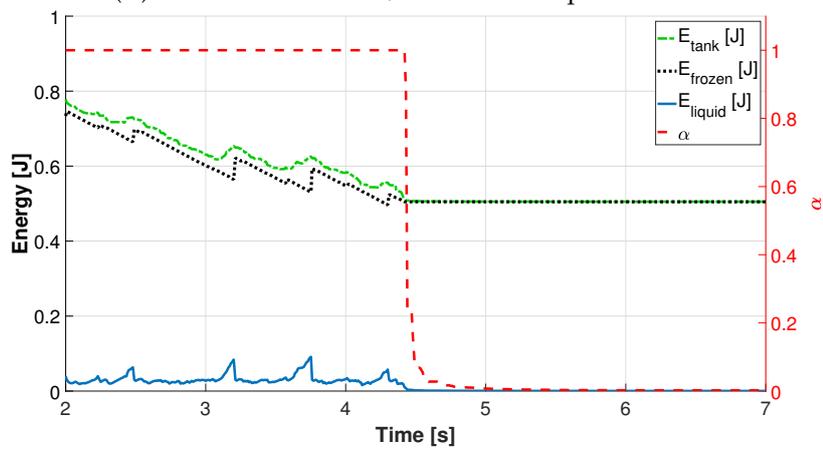


FIGURE 7.9: Amount of noise in the estimate of $T_B^{B,I}$ allowed before causing a false response by the safety algorithm, when applying a 5 N normal force (tested for at least 30 seconds). The twist ($T_B^{B,I}$) is estimated at 100 Hz.



(A) Noise SD = 0.065 m/s: No false response occurs



(B) Noise SD = 0.075 m/s: False response occurs

FIGURE 7.10: Example of how noise in $T_B^{B,I}$ may lead to a false response, depending on the noise level. Melting Rate = 0.2 J/s, ($E_{safe} - E_{window}$) = 0.02 J

7.4 Conclusion

In this chapter, the effectiveness of the designed solutions has been demonstrated under more practical circumstances. A discrete-time implementation of the energy-tank based wrench/impedance controller with safety extension has been constructed in C++, that is ready for use on the real hexarotor UAV. Due to aforementioned circumstances, however, experimental validation was no longer possible. As a replacement, this implementation has been tested within a more realistic simulation environment with respect to chapters 4 and 6. It incorporates several added effects, including additional dynamics for the thrust generation by the rotors, as well as a limited sensor update rate.

Under these new conditions, the general behavior of the implemented control system and safety extension has not changed. However, it has been noted how imperfections, in this case a bias in the twist estimate, could lead to a relatively fast depletion of the energy tank. In an experimental setup, it would therefore be worthwhile to closely study how such practical issues might affect the power flow into or out of the tank. The more that such problems can be prevented, the more purely the energy tank behavior is coupled to actual physical interactions.

Furthermore, without safety extension, the high impact of a contact loss event shows to be capable of destabilizing the system, now that the conditions have become less ideal. Therefore, the need for dealing with unsafe behavior becomes even more clear. The safety extension is demonstrated to still be effective in handling the safety issues following from regulation of the interaction wrench. However, its performance in this sense has decreased somewhat, which can be expected when introducing effects in the simulation that affect response time. It also became clear how the safety extension could be improved in terms of preventing unintended displacement, simply by making it respond more strongly. Optimizing this behavior, however, is a point of future work.

Finally, the effect of noise in the twist estimate on the proper functioning of the safety extension has been examined thoroughly, for different values of two relevant design parameters. This gives an idea of how these parameters might be chosen depending on the noise levels in a practical implementation.

In conclusion, the discrete-time implementation of the designed solution shows to remain effective for a less ideal simulation environment. However, experiments with the actual hexarotor UAV can be expected to introduce many more effects that this simulation still does not account for.

Chapter 8

Conclusion

8.1 Conclusions

The goal of this thesis has been to move towards stable and safe physical interaction of a fully actuated aerial robot with an unmodeled environment. Accompanying this goal, the focus throughout this thesis has been on remaining completely within the energy-based paradigm, in the process of designing the solution. This has led to the following formulation of the research questions:

1. How can stable physical interaction with an unmodeled environment be achieved for a fully actuated aerial robot?
2. How can the safety issues be dealt with that arise when regulating the interaction wrench with the environment, without compromising wrench tracking performance during normal operation?
3. How can the complete design be realized as much as possible within the energy-based paradigm?

The proposed method is an energy-based approach to the design of an interaction controller for a fully actuated aerial robot that yields closed loop passivity. Furthermore, it includes integrated algorithms that tackle the safety issues that emerge when the interaction wrench with the environment is being regulated. The complete design is summarized by the overview shown in figure 8.1.

The first part of the solution is the design of a passivity-based interaction controller, which enables stable physical contact with any conceivable passive environment. The design is based on a combination of impedance control and wrench regulation, where the augmentation of an energy tank is used in order to arrive at a passive closed loop system. The wrench regulation has been implemented by modulating the stiffness matrix of a 6-DOF spatial spring, placed parallel to the passive spatial spring used by the impedance controller. Furthermore, a port-based implementation has been proposed for the energy tank and its connection to the rest of the control system. As a result, the complete control system could be fitted into a port-based framework, in which all power flows are explicitly modeled and each subsystem has a physical interpretation.

Subsequently, simulations have been performed in order to validate this energy tank-based wrench/impedance controller. This was done by demonstrating the passivity of the closed loop system, as well as showing a correct functioning of the augmented energy tank near depletion.

The second part of the solution is the introduction of a novel concept called the *safety extension*. It expands the behavior of the energy tank to deal with the safety issues that arise when the interaction wrench with the environment is being regulated. The energy-based approach taken for its design has led to a more generally

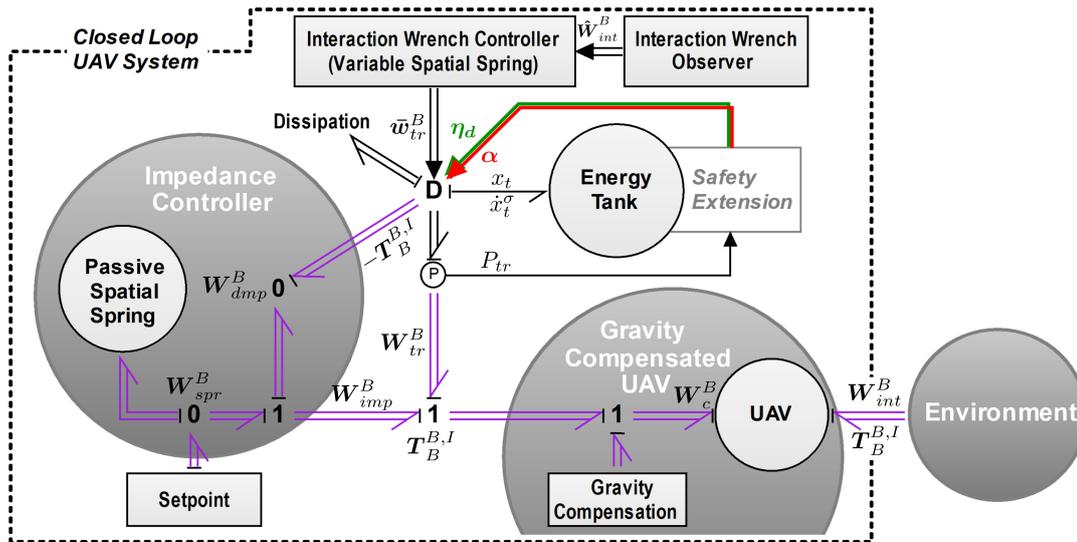


FIGURE 8.1: Overview of the proposed control system, represented using the bond graph notation (see §2.2)

applicable solution compared to a previous approach from literature. In addition to that, it offers more simplicity, as well as effectiveness in uncertain environments, although it lost the capability of enforcing explicit limitations on displacements.

The effectiveness and versatility of the resulting safety extension have been evaluated in simulation. There, it is demonstrated that this solution can indeed effectively deal with contact loss, as it greatly reduces the kinetic energy that the UAV attains, as well as the resulting displacement. Furthermore, it proves to transfer well to at least one different type of unsafe scenario, and shows a certain level of robustness against disturbances in the twist estimate.

In order to validate the design in a more practical context, a discrete-time implementation of the designed solution has been constructed in C++ and tested in a more realistic simulation environment. The behavior remained as expected, although the performance had slightly decreased. Experiments with the actual hexarotor UAV, however, are expected to introduce additional practical conditions that probably further affect performance.

Lastly, one could recognize how the energy-based approach, used during the entire design process, has led to significant advantages over the traditional signal-based approach. Firstly, the port-based framework that was used, gave explicit insight into the power flows between the different interconnected subsystems, as well as a physical interpretation of these subsystems themselves. This especially simplified the passivity analysis, and made clear under what conditions passivity would be preserved. Secondly, fitting the energy tank augmentation in the port-based framework has naturally led to a high level of modularity, compared to previous signal-based implementations that show a significant dependence on the specific choice of wrench controller and observer. Thirdly, for the design of the safety extension, the energy-based generalization of an existing geometry-based safety algorithm has shown to be a promising method for arriving at a more widely applicable solution that can also be implemented elegantly.

8.2 Limitations and Recommendations for Future Work

Additional validation

In order to further validate the effectiveness of the designed controller and safety extension, they must be subjected to less ideal circumstances, with realistic amounts of sensor noise, delay, external (aerodynamic) disturbances and modeling errors. Such phenomena were not sufficiently covered by the additional more realistic simulations performed in chapter 7.

Obviously, an experimental validation would be the best way to prove the practical applicability of the system. It must be noted that such a validation experiment was actually ready to be performed on the real hexarotor UAV. However, this experiment was forced to be cancelled, due to a malfunctioning of the UAV, followed by the recent measures regarding the coronavirus outbreak.

Also, the safety extension needs additional evaluation in terms of its general applicability. As discussed below, there are scenarios conceivable where potentially unsafe events could go undetected. A wider variety of such scenarios could be conceived and constructed in simulation, with a focus on finding the limitations of the safety algorithm.

Problematic scenarios for the safety extension are conceivable

Although the safety extension has been shown to be applicable to different types of unforeseen scenarios, there are also situations conceivable where it would not respond adequately. Problems may arise when the robot already has a significant initial kinetic energy before it starts regulating the interaction wrench. An example of such a scenario is when the UAV interacts with a flat wall surface, while moving at a significant velocity in the lateral direction. Then the UAV reaches the end of the wall surface, after which the wrench tracking controller continues to apply a force that has the direction and magnitude of a centripetal force. In that case, the velocity in the lateral direction would be converted into an equal velocity in the direction normal to the wall surface, without drawing any energy from the energy tank. Therefore, the safety extension would fail to prevent this unintended motion in the normal direction.

Although it is questionable whether there exists a task in which such a scenario can arise, it is still important to realize that there are such conditions under which the safety extension will not respond adequately.

One improvement that could at least be made regarding this issue, is to limit the amount of kinetic energy that the UAV is allowed to have in the first place. In the current situation, a limit on the kinetic energy is not strictly enforced yet, as no restrictions have been defined on the potential energy in the virtual spring of the impedance controller. If the initial setpoint were placed very far from the UAV, there would be a large amount of potential energy in the passive virtual spring, that would eventually be converted into kinetic energy. This in contrast to the wrench tracking controller, of which the total energy that it can inject is limited by the amount of energy in the energy tank.

A potential solution has therefore been found in connecting the action of setpoint moving to an energy tank as well. Note that the exact same solution has been suggested for the passivity problem described below. Therefore a further explanation on this solution can be found there as well.

Moving the setpoint still violates passivity

It has been noted that, in order to maintain a passive system, the pose setpoint should be stationary. If this is not the case, additional energy could be injected by the action of moving the setpoint. Specifically, this would be the case when the motion of the setpoint causes the virtual spring in the impedance controller to extend.

An effective way to both maintaining passivity in this sense and enforcing the kinetic energy limitation mentioned above, could be to couple the action of moving the setpoint to an energy tank as well. This means that the energy that is injected into the system due to the moving setpoint, should be drawn from an energy tank. An empty tank would then need to disable the setpoint from being moved in a direction that would inject energy into the system.

Furthermore, one could either use the same energy tank, or introduce a new energy tank for the action of setpoint moving. An advantage of defining a second energy tank might be that it keeps the setpoint moving process more independent from the process of interaction wrench regulation. In case of introducing this second energy tank, it would probably make sense to connect the energy regenerated by the virtual damper to this 'setpoint moving energy tank', instead of its current connection to the 'wrench regulation energy tank'. The energy within the wrench regulation tank would then solely depend on its initialization and a controlled supply of additional energy by either the user or a higher level controller.

The idea of connecting the action of setpoint moving to an energy tank, and the various choices that can be made here, would therefore be an interesting topic for future research.

The impedance controller spring does not obey the safety extension

Another potential issue that has been noted, is that placing the setpoint too far beyond the contact point gives safety problems. This is in the first place because it causes a larger portion of the interaction wrench to be applied by the passive virtual spring of the impedance controller (W_{spr}), compared to the modulated spring of the wrench tracking controller (W_{tr}). After contact loss, all of the power injected by the passive virtual spring remains unchecked by the safety algorithm. Now if this power is relatively small compared to the power injected by the wrench tracking controller (P_{tr}), then it will not lead to safety issues, but this purely depends on the scale of the extension and the desired wrench.

Placing the setpoint very close to the contact point, on the other hand, is not an ideal solution. This is because it would amplify the effect of noise in the pose estimate on the applied tracking wrench.

A more promising solution is to further separate the concerns of the passive and modulated virtual springs, by allowing their respective setpoints to be defined independently. In an interaction scenario, the passive virtual spring setpoint could be placed near the contact point, while placing the setpoint of the modulated virtual spring way beyond. As an added benefit, this would also eliminate the need for the *setpoint returning* algorithm introduced in chapter 5, reducing complexity and removing a potential point of failure.

It is therefore suggested to investigate whether this idea is indeed a better alternative, and the implications that the presence of two independent setpoints has on the interface with either the user or a higher level controller.

Appendix A

Table of Simulation Parameters

Control parameters		UAV and Simulation parameters	
Passive spring orient. stiff. (\mathbf{K}_o)	$\mathcal{J}^{-1} \text{diag}(\frac{4}{5}, \frac{4}{5}, \frac{1}{2})$	UAV mass (m)	1.84 kg
Passive spring trans. stiff. (\mathbf{K}_t)	$\frac{1}{m} \text{diag}(10, 10, 10)$	UAV moment of inertia (\mathcal{J})	$\text{diag}(0.05, 0.05, 0.094) \text{ kg} \cdot \text{m}^2$
Linear damper, orient. ($\mathbf{K}_{d,o}$)	$\mathcal{J}^{-1} \text{diag}(\frac{1}{4}, \frac{1}{4}, \frac{1}{3})$	g	9.81 m/s^2
Linear damper, trans. ($\mathbf{K}_{d,t}$)	$\frac{1}{m} \text{diag}(10, 10, 10)$	Wall stiffness (normal) [N/m]	Scen. 1: 500 Scen. 2: 200 Scen. 3-5: 2260
Wrench observer gains (\mathbf{K}_{obs})	$5.0 \cdot \mathbf{I}_6$	Wall viscous friction (normal)	123 N / (m/s)
Setpoint distance beyond surface (Δx)	Scen. 1,2: 0.3 m Scen. 3-5: 0.1 m	Wall viscous friction (lateral)	0.1 N / (m/s)
Stiff. modulation P-gain (\mathbf{K}_p)	$\frac{1}{\Delta x} 2.5 \cdot \mathbf{I}_6$	End effector pos w.r.t. CoM (ξ_E^B)	[0.7, 0, 0] m
Tank init. energy	$E_t(0) = 0.2 \text{ J}$	UAV rotor tilt	$\alpha^* = 47^\circ$
Tank max. energy	$E_t^+ = 0.5 \text{ J}$	Rotor distance to center	$L = 0.34 \text{ m}$
Tank min. energy	$E_t^- = 0.05 \text{ J}$	Drag-to-thrust ratio	$\gamma = 0.0134$
Tank depletion smoothing width	$E_{window} = 0.01 \text{ J}$	Maximum rotor thrust	15 N
Amount of energy deemed "safe"	$E_{safe} = 0.05 \text{ J}$ (default value)		
Safety violation power threshold	$P_{unsafe} = 0.5 \text{ J/s}$		
Melting rate	0.1 J/s		

TABLE A.1: Table of the simulation parameters (scenario 1 to 5)

Bibliography

- Breedveld, P.
2008. Modeling and simulation of dynamic systems using bond graphs. In *Control Systems, Robotics and Automation*. Encyclopedia of Life support systems (EOLSS).
- Dietrich, A., X. Wu, K. Bussmann, C. Ott, A. Albu-Schäffer, and S. Stramigioli
2017. Passive hierarchical impedance control via energy tanks. *IEEE Robotics and Automation Letters*, 2(2):522–529.
- Duindam, V. and S. Stramigioli
2004. Port-based asymptotic curve tracking for mechanical systems. *European Journal of Control*, 10(5):411 – 420.
- Fasse, E. D. and J. F. Broenink
1997. A spatial impedance controller for robotic manipulation. *IEEE Transactions on Robotics and Automation*, 13(4):546–556.
- Ferraguti, F., N. Preda, A. Manurung, M. Bonfè, O. Lambercy, R. Gassert, R. Muradore, P. Fiorini, and C. Secchi
2015. An energy tank-based interactive control architecture for autonomous and teleoperated robotic surgery. *IEEE Transactions on Robotics*, 31(5):1073–1088.
- Ferraguti, F., C. Secchi, and C. Fantuzzi
2013. A tank-based approach to impedance control with variable stiffness. In *2013 IEEE International Conference on Robotics and Automation*, Pp. 4948–4953.
- Folkertsma, G. A. and S. Stramigioli
2017. Energy in robotics. *Foundations and Trends® in Robotics*, 6(3):140–210.
- Furrer, F., M. Burri, M. Achtelik, and R. Siegwart
2016. *RotorS—A Modular Gazebo MAV Simulator Framework*, Pp. 595–625. Cham: Springer International Publishing.
- Gawthrop, P.
1991. Bond graphs: A representation for mechatronic systems. *Mechatronics*, 1(2):127 – 156.
- Groothuis, S. S., G. A. Folkertsma, and S. Stramigioli
2018. A general approach to achieving stability and safe behavior in distributed robotic architectures. *Frontiers in Robotics and AI*, 5:108.
- Hogan, N.
1985. Impedance Control: An Approach to Manipulation. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):1–24.
- Jiang, G. and R. Voyles
2014. A nonparallel hexrotor uav with faster response to disturbances for precision position keeping. In *2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)*, Pp. 1–5.

- Kamel, M., S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. Taylor, R. Siegwart, and I. Gilitschenski
2018. The voliro omniorientational hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle. *IEEE Robotics & Automation Magazine*, 25(4):34–44.
- Martin, P. and E. Salaun
2010. The true role of accelerometer feedback in quadrotor control. In *2010 IEEE International Conference on Robotics and Automation*, Pp. 1623 – 1629.
- Newman, J. A., N. Shewchenko, and E. Welbourne
2000. A proposed new biomechanical head injury assessment function - the maximum power index. *Stapp car crash journal*, 44:215–47.
- Nguyen, H. and D. Lee
2013. Hybrid force/motion control and internal dynamics of quadrotors for tool operation. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pp. 3458–3464.
- Park, S., J. Lee, J. Ahn, M. Kim, J. Her, G. Yang, and D. Lee
2018. Odar: Aerial manipulation platform enabling omnidirectional wrench generation. *IEEE/ASME Transactions on Mechatronics*, 23(4):1907–1918.
- Paynter, H. M.
1960. *Analysis and Design of Engineering Systems: Class Notes for M.I.T. Course 2,751*. Cambridge, MA: M.I.T. Press.
- Pounds, P. E. I. and A. M. Dollar
2011. Uav rotorcraft in compliant contact: Stability analysis and simulation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pp. 2660–2667.
- Raiola, G., C. A. Cardenas, T. S. Tadele, T. de Vries, and S. Stramigioli
2018. Development of a safety- and energy-aware impedance controller for collaborative robots. *IEEE Robotics and Automation Letters*, 3(2):1237–1244.
- Rashad, R., J. B. C. Engelen, and S. Stramigioli
2019. Energy tank-based wrench/impedance control of a fully-actuated hexarotor: A geometric port-hamiltonian approach. In *2019 International Conference on Robotics and Automation (ICRA)*, Pp. 6418–6424.
- Ruggiero, F., V. Lippiello, and A. Ollero
2018. Aerial manipulation: A literature review. *IEEE Robotics and Automation Letters*, 3(3):1957–1964.
- Ryll, M., D. Bicego, and A. Franchi
2016. Modeling and control of fast-hex: A fully-actuated by synchronized-tilting hexarotor. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Pp. 1689–1694.
- Ryll, M., H. H. Bühlhoff, and P. R. Giordano
2012. Modeling and control of a quadrotor uav with tilting propellers. In *2012 IEEE International Conference on Robotics and Automation*, Pp. 4606–4613.

- Ryll, M., G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, D. Bicego, and A. Franchi
2019. 6d interaction control with aerial robots: The flying end-effector paradigm. *The International Journal of Robotics Research*, 38(9):1045–1062.
- Ryll, M., G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, and A. Franchi
2017. 6d physical interaction with a fully actuated aerial robot. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Pp. 5190–5195.
- Schindlbeck, C. and S. Haddadin
2015. Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, Pp. 440–447.
- Shahriari, E., L. Johansmeier, and S. Haddadin
2018. Valve-based virtual energy tanks: A framework to simultaneously passify controls and embed control objectives. In *2018 Annual American Control Conference (ACC)*, Pp. 3634–3641.
- Shahriari, E., A. Kramberger, A. Gams, A. Ude, and S. Haddadin
2017. Adapting to contacts: Energy tanks and task energy for passivity-based dynamic movement primitives. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, Pp. 136–142.
- Stramigioli, S.
2015. Energy-aware robotics. In *Mathematical Control Theory I*, M. K. Camlibel, A. A. Julius, R. Pasumathy, and J. M. Scherpen, eds., Pp. 37–50, Cham. Springer International Publishing.
- Stramigioli, S. and H. Bruyninckx
2001. Geometry and screw theory for robotics. In *2001 International Conference on Robotics and Automation (ICRA)*.
- Stramigioli, S. and V. Duindam
2001. Variable spatial springs for robot control applications. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, volume 4, Pp. 1906–1911 vol.4.
- Tadele, T. S., T. J. A. de Vries, and S. Stramigioli
2014. Combining energy and power based safety metrics in controller design for domestic robots. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Pp. 1209–1214.