UNIVERSITÀ DI PARMA

UNIVERSITÀ DEGLI STUDI DI PARMA

Dottorato di Ricerca in Tecnologie dell'Informazione

XXXVII Ciclo

Control of Impact-aware Collaborative Robots

Coordinatore:

Chiar.mo Prof. Marco Locatelli

Tutor:

Chiar.mo Prof. Corrado Guarino Lo Bianco

Dottorando: *Shabnam Shakourzadeh*

2022/2025

To my husband and family,
who are always by my side
in all circumstances

Abstract

The development of collaborative robots (cobots) capable of operating safely and efficiently alongside humans is a critical area in robotics. This dissertation addresses the challenges of impact-aware control in collaborative robotics, focusing on sensorless methodologies to detect and respond to intentional and unintentional impacts. Three key innovations are presented.

First, a novel Hybrid Super-Twisting Differentiator (HSTD) is introduced, enabling accurate velocity and acceleration estimation from position data without requiring dedicated sensors. This method demonstrates superior noise resilience and minimal phase lag, enhancing real-time performance in dynamic environments.

Second, a new Extended State Acceleration Observer (ESAO) is developed for rapid and reliable collision detection. By extending the robot's state-space dynamics and using the error of acceleration signal, this method achieves faster detection times compared to state-of-the-art algorithms, offering robust noise handling and adaptability.

Third, the application of these methods is demonstrated through a dual-arm robot setup designed for object grasping and manipulation tasks. Challenges such as weak external torque signals and redundant impact detection are addressed by leveraging combined external torque norms and time window frame calibrated thresholds, ensuring reliable trajectory planning and control. Experimental validation confirms the effectiveness of the proposed approach in improving task success rates and operational stability.

This work advances the state of impact-aware cobot control, laying the groundwork for safer and more efficient human-robot collaboration.

Keywords: Collaborative robots, Extended state acceleration observer, Hybrid super-twisting differentiator, Impact-aware control, Sensorless methodologies

Contents

List of Figures

List of Tables

Introduction

The automation of industrial processes has long been a crucial factor in the quest for improving both operational and financial efficiency in factories. By automating tasks, companies can either accelerate production or enhance the precision of processes, which ultimately reduces overall costs. Since the mid-2010s, automation has expanded beyond traditional manufacturing to impact sectors like logistics and parcel delivery [1]. This began with the introduction of automated distribution centers and soon advanced to the use of pick-and-place robots for sorting [2]. A key development in this area has been the emergence of collaborative robots, or cobots. These robots are designed to work alongside human employees in shared workspaces, enhancing efficiency and safety. The research and advancement of such technologies have gained significant momentum, driven by the rapid growth of the online retail sector. As e-commerce continues to rise, the demand for more efficient and automated solutions in logistics has become increasingly critical. Consequently, developing cobots and other automation technologies has turned into a priority to meet the needs of this evolving market.

The control of collaborative robots, particularly in scenarios involving impact, has been the focus of numerous research efforts [3–5]. However, despite these studies, there is still a need to enhance the robots' ability to respond with greater accuracy and speed. Two critical qualities must be taken into account when designing and improving collaborative robots. First, the robot must be capable of detecting both intentional and unintentional impacts with precision. Second, the detection process must occur rapidly and with minimal error to ensure safety and efficiency in collaborative

environments. Achieving these objectives is essential for optimizing the performance of collaborative robots, particularly in industries where human-robot interaction is frequent. Enhanced impact detection and fast response times will not only improve operational efficiency but also ensure the safety of workers sharing the workspace with these machines. As the development of cobots continues to evolve, further research and innovation are needed to address these challenges and bring collaborative robots to their full potential.

Since impacts can occur at any joint of the robot, relying solely on force/torque sensors for detection is not a viable option, as equipping every joint with such sensors would be impractical and costly. Therefore, detecting impacts should be carried out using sensorless methods. These methods utilize the robot's internal data, such as motor currents, velocities, and accelerations, to infer when an impact has occurred. Sensorless detection not only reduces the complexity and expense of the robot's hardware but also allows for more seamless integration into various industrial settings, where flexibility and cost-effectiveness are crucial. Developing robust sensorless techniques is key to ensuring that collaborative robots can quickly and accurately respond to impacts without the need for extensive physical sensors, enhancing both safety and efficiency in human-robot interaction.

In this thesis, three critical phases related to the control of impact-aware collaborative robots have been thoroughly examined. Given the necessity of sensorless impact estimation, it is essential to address the specific requirements for achieving accurate and reliable estimations. As mentioned earlier, since collaborative robots are not equipped with direct velocity or force/torque sensors, it becomes crucial to estimate key parameters such as velocity, acceleration, and external torque through alternative methods. These estimations are vital for enabling the robot to detect and respond to impacts without relying on external sensing hardware.

The process of estimating these parameters involves leveraging the robot's internal data, such as motor currents and joint positions, to compute the necessary values in real time. Achieving accurate sensorless estimation is essential not only for maintaining the robot's functionality in detecting both intentional and unintentional impacts, but also for enhancing the overall safety and efficiency of human-robot collab-

oration. This becomes especially important in dynamic environments where humans and robots work closely together, as the robot must respond quickly and accurately to prevent potential harm. Consequently, the methods used for sensorless impact estimation must be refined to ensure minimal error, fast response times, and adaptability to various operational conditions.

For accurate estimation in impact-aware collaborative robots, it is essential to have reliable information about the system's states. This means that, based on the current states of the cobot, the differentiation of these states, such as velocity and acceleration, must be readily accessible.

Using sensorless estimation methods, impacts should be detected with minimal delay and noise. The estimated signals play a crucial role in designing controllers and planning trajectories for collaborative tasks such as object grasping and manipulation. For pick-up applications with dual collaborative arms, it is essential to develop a robust decision-making framework to ensure accurate phase transitions during the pre-impact, object grasping, and post-impact stages.

Based on previous discussions, the thesis is divided into three main parts:

- **Estimation Method as a Differentiator**: The first part introduces an estimation method based on the hybrid Super-Twisting differentiator. Unlike previous methods, which rely on the system's dynamics, this new method is designed to handle dynamic changes during the impact phase, making it more suitable for our specific application. Traditional methods, such as the Kalman filter—although effective for noisy signals—are not applicable here due to the dynamic shifts caused by impacts and the real-time nature of industrial applications, where time efficiency is critical. The newly designed method is independent of system dynamics, allowing for signal detection during the impact phase, and exhibits less phase shift compared to the standard Super-Twisting methodology. The proposed differentiator methodology is crucial for impact detection in collaborative robots that operate without dedicated velocity or acceleration sensors.

- **Reliable Impact Detection Observer**: The second part focuses on developing

a reliable impact detection observer using a new model-based collision estimation algorithm. This novel algorithm quickly identifies collisions without the need for force/torque sensors. It is easy to tune, produces stronger signals for impact recognition, and reduces the delay compared to state-of-the-art algorithms. The method extends the robot's state-space dynamics and incorporates the Hybrid Super-Twisting differentiator into the algorithm. The differentiator's parameters have been optimized to reduce noise and delay, ensuring faster and more accurate signal detection. The new method features linear gains, finite-time convergence, and stable observer characteristics, which can be easily obtained and analyzed.

- **Application of the Methods**: This section describes the practical implementation of the proposed impact detection method using a dual-arm robot setup for grasping and lifting an object. The trajectory planning and controller application are outlined, with the trajectory divided into pre-impact and post-impact phases based on the introduced impact detection method. The control strategy combines feedback linearization with a reference spreading approach to ensure stable and successful grasping and lifting. Two main challenges are addressed: First, the weak external torque signal from the end effector during collision with the object. This is because the end effector has less movement during grasping compared to other joints of the arm, which complicates accurate impact detection. Second, ensuring that impact is only detected the first time both arms engage with the object, thus avoiding false detections during the grasping process that could lead to incorrect trajectory and control commands. Solutions to these challenges are proposed, providing a reliable impact detection approach that supports precise and effective grasping and lifting tasks.

This structure allows for a comprehensive approach to planning and controlling collaborative robots, ensuring both accuracy and efficiency in real-world applications.

Chapter 1

Velocity Estimation Using Second-order Hybrid Super-twisting

1.1 State of the Art

This section is dedicated to the prerequisites for impact detection methodology which will be explained in the second chapter. Since robots are not equipped with velocity or acceleration sensors, it is essential to utilize a differentiator that is suitable for impact-aware robots.

In automation contexts, it is often necessary to estimate, through a dynamic observer, the states of dynamic systems. A very common estimation problem is represented by signals differentiation: given an input signal, a system is implemented for the computation of its first time derivative. The presence of disturbances on the measured variable complicates such apparently simple task, since differentiators amplify the input noise and introduce delays.

The estimation of the velocity of a dynamic system, starting from a position signal, probably represents the most frequent application of differentiators. Several alternative approaches are proposed in the literature: the brief overview next proposed

lists the most known of them.

A finite difference differentiator, immediately followed by a linear low-pass filter, is probably the most widely adopted solution, because of its simplicity. However, the drawback of such method is given by the introduced phase shift, which negatively impacts on the performances of feedback controllers. Delays can be reduced by enlarging the filter bandwidth, at the cost of an increased output noise, or by choosing adequate filter typologies. Finite difference derivators, despite their inherent problems, return acceptable estimates, so that they are widely adopted by the industry.

Much better outcomes have to be expected by computing the speed through a Kalman filter [6]. It is one of the most known state observers, particularly suited for the estimation of signals in presence of noise. The Kalman filter exploits the covariance of the noise in order to reliably predict the system state [7]. The main limit of such method is given by the fact that states can be correctly identified only if the dynamic model of the system is known with a sufficient accuracy [8]. Consequently, if the model is uncertain or unavailable, the Kalman filter do not represent the most appropriate choice, which is the case in impact/collision situations.

An alternative differentiation technique, which has become very popular in recent years, is based on the so called Super Twisting Differentiator (STD) originally proposed by Levant [9]. Such differentiation method is very robust being based on sliding mode techniques. Unlike the Kalman filter, it does not make use of the system model, therefore it can be considered a general purpose differentiator. Another positive characteristic owned by STDs, which is very important if their output signals are used for feedback control loops, is represented by the limited estimation delay they introduce. The literature proposes many papers on the STDs. Some of them concern the second order version, other extend the discussion to higher order releases [10], [11]. Moreover, in [12], a discontinuous STD has been introduced.

STDs are affected by two main cons. For the best performances, their tuning requires a particular care: a manual tuning imposes an iterative process, so that, more frequently, their parameters are found through optimization procedures. Another problem is represented by the huge influence the acquisition noise has on the estimated signal. The sliding mode nature of the differentiator exacerbates such neg-

ative influence, thus limiting the usability of the STDs in practical applications. It is well known that STDs, at rest conditions, return a particularly noisy output which solicits the feedback controller and, in turn, the overall system [13].
Since in industrial environments disturbances represent the norm, alternative Super Twisting structures have been proposed. The most interesting of them is given by the Hybrid Super Twisting Differentiator (HSTD) [13], [14], [15]. As suggested by its name, it is a hybrid implementation of the STD, whose structure changes depending on the noise on the input signal: it exactly acts as an STD if the noise has a limited amplitude, otherwise it behaves as a second order linear low-pass filter. In case of intermediate situations the HSTD assumes hybrid behaviours. The idea at the basis of the approach is that, under low-noise operating conditions, the tracking performances of the STD must be fully exploited, while in noisy environments delays represent the price to be paid for a cleaner signal.
In the HSTD proposed in [13], the system switches, with continuity, between a pure sliding mode controller and a linear one, as function of the noise content of the measured variable. Depending on the application, this can be a limit. In facts, in many cases the noise content of the acquired variable is marginally influenced by the operating conditions, so that the HSTD would maintain an almost constant structure. Conversely, the effects produced by such noise on the feedback controller change depending on the working conditions, so that a variable structure HSTD would be preferred. This is the case, for example, of the joints of collaborative robots in this work, in which the position measurement is affected by an almost constant noise which causes evident vibrations when the system is at rest, while it has negligible effects during the movement: the strategy proposed in [13] would lead to a constant-structure HSTD while, for the best performances, its structure should change depending on the status of motion.
For the above mentioned reasons, at low speeds the HSTD proposed in this work acts as a linear system, while at high speeds – when phase lags on the velocity estimation can be dangerous and noise effects are smoothed by the system inertia – the sliding mode behavior is adopted. The HSTD performances are further improved by online changing its tuning parameters depending on the motion conditions.

1.2 Methodology Definition

As described in the state of the art, the STD method is a robust nonlinear algorithm used to estimate the derivative of a signal while rejecting noise. It is a second-order sliding mode technique that ensures finite-time convergence and improved accuracy over traditional differentiation methods. STDs are variable-structure observers designed to estimate the state of the following system

$$\dot{x}_1 = x_2, \tag{1.1}$$

$$\dot{x}_2 = u, \tag{1.2}$$

$$y = x_1 + w. \tag{1.3}$$

Clearly, (1.1)–(1.3) represent a chain of 2 integrators whose input is represented by u – the acceleration – and whose output y is given by x_1 – the position – plus an additive noise w.

As stated in the state of the art, standard STDs provide derivative estimations which are characterized by very low phase shifts. Unfortunately, their behavior worsens depending on the amplitude of the noise affecting their input signals. For this reason, the following HSTD was proposed in [13]

$$\dot{\bar{x}}_1 = \bar{x}_2 + \chi_1 |e_{1w}|^{\alpha} \operatorname{sgn}(e_{1w}), \tag{1.4}$$

$$\dot{\bar{x}}_2 = \alpha\chi_2 |e_{1w}|^{2\alpha-1} \operatorname{sgn}(e_{1w}), \tag{1.5}$$

$$e_{1w} = y - \bar{x}_1, \tag{1.6}$$

$$\bar{z} = \bar{x}_2, \tag{1.7}$$

where $\operatorname{sgn}(\cdot)$ is the sign function defined as follows

$$\operatorname{sgn}(x) := \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}.$$

Input y of (1.4)–(1.7) is given by the noisy output of (1.1)–(1.3). A schematic representation of this method is shown in Figure 1.1. The estimator state is given by

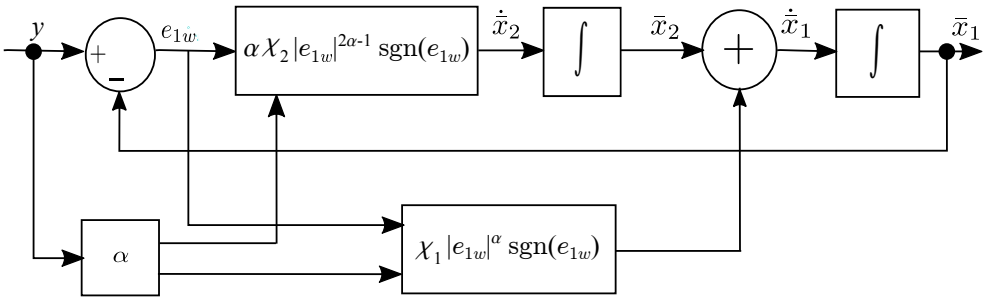


Figure 1.1: Block diagram of STD methodology.

$\bar{\mathbf{x}} := [\bar{x}_1\,\bar{x}_2]^T$, $\bar{z}$ is the estimated velocity, while χ_1 and χ_2 are tuning parameters. The main characteristic owned by the HSTD is that its behavior changes depending on the selection of $\alpha \in [0.5, 1]$. More precisely, when $\alpha = 0.5$ it behaves as a standard STD, while, for $\alpha = 1$, (1.4) and (1.5) can be rewritten as follows

$$\dot{\bar{x}}_1 = \bar{x}_2 + \chi_1(y - \bar{x}_1), \tag{1.8}$$

$$\dot{\bar{x}}_2 = \chi_2(y - \bar{x}_1), \tag{1.9}$$

i.e., the HSTD acts as a linear system. The rationale of the approach is that, in case of a small amplitude input noise, values of α close to 0.5 return good estimates of the speed, while, if w increases, $\alpha = 1$ allows a better noise rejection paid through a higher phase shift. According to this idea, in [13] α was selected on the basis of a noise measure (see [13] for more details). However, depending on the application, such choice could not give the desired results. For example, if the measurement noise associated to y does not depend on the operating condition, α remains constant so that the advantages of the hybrid approach are lost. For this reason, an alternative HSTD is proposed in this thesis.

1.2.1 The Novel HSTD

When motors of the robot's joints are running at sufficiently high speeds, the noise induced by w on the output of the HSTD has normally a minimal effect on the performances of feedback controllers. Conversely, still considering high speeds, phase lags on $\bar{z}$ can consistently worsen the system response. For this reason, during the motion,

α should be chosen close to 0.5. On the contrary, when the speed is small or equal to zero, the noise on $\bar{z}$ causes evident vibrations on the robot's actuator, while phase shifts have a lower impact on the controller behavior: $\alpha = 1$ should be the preferred choice.

Such considerations suggested an alternative implementation of the HSTD, obtained by introducing two major changes to the structure originally proposed in [13].

First of all, by still assuming a core given by (1.4)–(1.7), the selection of α is based on a rough estimation of the system speed obtained through the following equations

$$\dot{z} = \tau\dot{y} - \tau z. \tag{1.10}$$

$$\alpha = \begin{cases} \dfrac{\varepsilon}{\varepsilon + |z|} & \text{if } |z| \leq \varepsilon \\ 0.5 & \text{if } |z| > \varepsilon \end{cases}. \tag{1.11}$$

Secondly, parameters χ_1 and χ_2 are expressed as function of α according to the following expression

$$\chi_i(\alpha) = \underline{\chi}_i + \frac{\overline{\chi}_i - \underline{\chi}_i}{1 - \alpha_{th}}(\alpha - \alpha_{th}), \quad i = 1,2, \tag{1.12}$$

where $\underline{\chi}_i$ and $\overline{\chi}_i$ represent the minimum and the maximum values for $\chi_i(\alpha)$, respectively, while $\alpha_{th} \in [0.5, 1]$ is a proper threshold.

Equation (1.10) is a first order low-pass filter, whose pole is placed in $-\tau$, with $\tau > 0$. Its input, which is indicated by $\dot{y}$, is obtained from y through a finite difference derivative. Output z is a rough estimate of the the system speed, used to select α through (1.11): as desired, at low speeds $\alpha = 1$, while for high speeds $\alpha = 0.5$. In case of intermediate velocities, a proper tuning of ε returns appropriate values for α. Term ε can be seen as the threshold speed over which $\alpha = 0.5$ represents the preferred choice in terms of performances. Parameter τ is chosen so as to minimize the noise on z. Evidently, (1.10) returns an estimate of z which is affected by delays. However, such delays do not represent a problem since (1.10) is only used to detect the motion conditions of the system: the actual speed estimation of the HSTD is provided by (1.7), so that phase lags caused by (1.10) have a negligible impact on the behavior of the overall system [16].

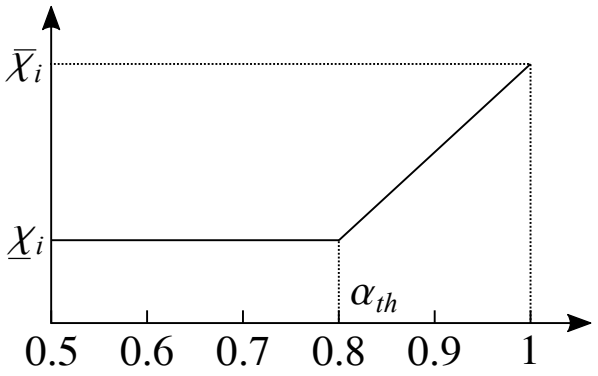


Figure 1.2: Function $\chi_i(\alpha)$.

The experimental evidence shown that better performances can be achieved if the χ_i terms are made function of the state of motion. For this reason, (1.12) modifies the tuning coefficients of the HSTD depending on α. A typical shape of $\chi_i(\alpha)$ is shown in Figure 1.2: $\chi_i(\alpha)$ is constant for a large interval of α, while it linearly changes when α is close to 1 .

1.3 Stability Analyses

This section is devoted to the stability analysis of the proposed second order HSTD based on (1.4)–(1.7) and (1.10)–(1.12). The core of the HSTD is the same already proposed in [13], so that when $\alpha = 1$ it behaves as a linear system, while when $\alpha = 0.5$ it behaves as a pure STD.

Sufficient stability conditions on χ_1 and χ_2 have been given in [17] for $\alpha = 0.5$. Additional conditions can be provided for $\alpha = 1$. Indeed, when $\alpha = 1$, a few manipulations on (1.8) and (1.9) lead to the following transfer function

$$\frac{\bar{x}_1}{y} = \frac{\chi_2 + \chi_1 s}{s^2 + \chi_1 s + \chi_2}, \tag{1.13}$$

so that the stability requires assuming $\chi_1 > 0$ and $\chi_2 > 0$. According to these considerations the stability analysis next proposed will only consider the values of α contained in the open interval $(0.5, 1)$. By bearing in mind (1.11), this implies that

condition $|z| \leq \varepsilon$ applies, so that the derivative of α, obtained from (1.10) and (1.11), can be expressed as follows

$$\dot{\alpha} = \frac{-\varepsilon \dot{z} \operatorname{sgn}(z)}{(\varepsilon + |z|)^2}. \tag{1.14}$$

The HSTD scheme proposed in this thesis differs from the one devised in [13] for the selection method proposed for α. Furthermore, χ_1 and χ_2 change depending the status of motion because of (1.10)–(1.12). Necessarily, an alternative stability proof is required, based on the following hypotheses:

1. w is a bounded-amplitude noise such that $|w| \leq \overline{w}$, where $\overline{w}$ is a known upper bound;
2. τ and ε are constant positive scalars. Additionally, $\varepsilon \in [0.5, 1]$;
3. z, i.e., the speed estimate provided by (1.10), is limited, so that $|z| \leq \bar{z}$, where $\bar{z}$ is a known upper bound;
4. $|u| \leq \overline{u}$, where u is the input of (1.1)–(1.3) and $\overline{u}$ is its known upper bound;
5. $\underline{\chi}_i, \overline{\chi}_i \in \mathbb{R}^+$, so that $\chi_i(\alpha)$, $i = 1, 2$ are always positive;
6. $\alpha \in [0.5, 1]$.

The estimation errors on the first and the second states of the observer are indicated as $e_1 = x_1 - \bar{x}_1$ and $e_2 = x_2 - \bar{x}_2$, respectively. Bearing in mind (1.4) and (1.5), the errors derivatives w.r.t. the time can be written as follows

$$\dot{e}_1 = e_2 - \chi_1 |e_{1w}|^{\alpha} \operatorname{sgn}(e_{1w}) \tag{1.15}$$

$$\dot{e}_2 = u - \alpha \chi_2 |e_{1w}|^{2\alpha - 1} \operatorname{sgn}(e_{1w}) \tag{1.16}$$

A proper Lyapanuv function is required for the stability proof. By assigning $\boldsymbol{\Gamma} := [\Gamma_1 \Gamma_2]^T$, where

$$\Gamma_1 = |e_1|^{\alpha} \operatorname{sgn}(e_1), \tag{1.17}$$

$$\Gamma_2 = e_2, \tag{1.18}$$

the following candidate Lyapunov function can be assumed

$$V = \mathbf{\Gamma}^T \mathbf{P} \mathbf{\Gamma} \,, \tag{1.19}$$

where $\mathbf{P}$ is a positive-definite constant matrix. Clearly, condition $V > 0$ is satisfied $\forall \mathbf{\Gamma} \neq 0$.

The following expressions, which are required for obtaining $\dot{V}$, are obtained by differentiating (1.17) and (1.18)

$$\dot{\Gamma}_1 = |e_1|^{\alpha} \left(\dot{\alpha} \ln(|e_1| + \frac{\dot{e}_1}{e_1} \alpha\right) \operatorname{sgn}(e_1) \,, \tag{1.20}$$

$$\dot{\Gamma}_2 = \dot{e}_2 \,. \tag{1.21}$$

It is worth to mention that $\dot{\Gamma}_1$ is not defined for $e_1 = 0$ because of the discontinuity introduced by the $\operatorname{sgn}(\cdot)$ function. This is not a problem since, as it will soon be shown, condition $e_1 = 0$ is never reached exactly.

As stated in [13], e_{1w} and e_1 are linked together through the following relationship

$$|e_{1w}|^{\beta} = |e_1|^{\beta} \operatorname{sgn}(e_1) + f(e_1, w) \, |w|^{\beta} \,, \tag{1.22}$$

where $f(e_1, w) \in [-2, 2]$ is a bounded function, while $\beta \in \mathbb{R}^+$ is a scalar (for the problem at hand $\beta = \alpha$ or $\beta = 2\alpha - 1$). Bearing in mind (1.14), (1.15), (1.16), and (1.22), equations (1.20) and (1.21) can be rewritten as follow

$$\dot{\Gamma}_1 = \alpha \, |e_1|^{\alpha - 1} \left(f(e_1, z) + \Gamma_2 - \chi_1 \Gamma_1 + \chi_1 f(e_1, w) \, |w|^{\alpha}\right), \tag{1.23}$$

$$\dot{\Gamma}_2 = u - \alpha \, |e_1|^{\alpha - 1} \left(\chi_2 \Gamma_1 - \chi_2 \, |e_1|^{1 - \alpha} f(e_1, w) \, |w|^{2\alpha - 1}\right), \tag{1.24}$$

where $f(e_1, z)$ is defined as follows

$$f(e_1, z) := e_1 \ln(|e_1|) \frac{\tau(z - \dot{y})}{\varepsilon + |z|} \operatorname{sgn}(z) \tag{1.25}$$

A few algebraic manipulations on (1.23) and (1.24) lead to the following representation for $\dot{\mathbf{\Gamma}} := [\Gamma_1 \, \Gamma_2]^T$

$$\dot{\mathbf{\Gamma}} = \Phi(\mathbf{A}\mathbf{\Gamma} + \mathbf{n}) + \mathbf{u}, \tag{1.26}$$

where

$$\mathbf{A} := \begin{pmatrix} -\chi_1 & 1 \\ -\chi_2 & 0 \end{pmatrix} \tag{1.27}$$

$$\mathbf{n} := [n_1\, n_2]^T \tag{1.28}$$

$$n_1 := \chi_1 f(e_1, w)\, |w|^{\alpha} + f(e_1, z) \tag{1.29}$$

$$n_2 := \chi_2\, |e_1|^{1-\alpha} f(e_1, w)\, |w|^{2\alpha-1} \tag{1.30}$$

$$\mathbf{u} := [0\, u]^T \tag{1.31}$$

$$\Phi := \alpha\, |e_1|^{\alpha-1} \tag{1.32}$$

According to assumption 5, $\chi_1 > 0$ and $\chi_2 > 0$ independently from α: the eigenvalues of $\mathbf{A}$ are always positive.

According to [18], if $\mathbf{A}(\alpha)$ is a linear function of $\alpha \in \mathscr{D}$, where $\mathscr{D}$ is a compact and closed interval, and $\mathbf{A}(\alpha)$ is Hurwitz at the extremes of $\mathscr{D}$, then there exists a constant positive-definite matrix $\mathbf{P}$ such that the following equation is satisfied for all $\alpha \in \mathscr{D}$:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q}, \tag{1.33}$$

where $\mathbf{Q}$ is a positive-definite, constant matrix. For the problem at hand, $\mathbf{A}$ is Hurwitz for $\alpha = 0.5$ and for $\alpha = 1$. Furthermore, by virtue of (1.12), it is constant for $\alpha \leq \alpha_{th}$ and it changes linearly for $\alpha \in [\alpha_{th}, 1]$: it would be certainly possible to find a constant $\mathbf{P}$ which satisfies (1.33), $\forall \alpha \in [0.5, 1]$.

By bearing in mind (1.26) and (1.33), the derivative of (1.19) can be written as follows

$$\begin{aligned} \dot{V} &= \dot{\mathbf{\Gamma}}^T\mathbf{P}\mathbf{\Gamma} + \mathbf{\Gamma}^T\mathbf{P}\dot{\mathbf{\Gamma}} \\ &= -\Phi\mathbf{\Gamma}^T\mathbf{Q}\mathbf{\Gamma} + 2\Phi\mathbf{n}^T\mathbf{P}\mathbf{\Gamma} + 2\mathbf{u}^T\mathbf{P}\mathbf{\Gamma}. \end{aligned} \tag{1.34}$$

Many terms in (1.34) are explicitly bounded. For example, the following inequalities apply

$$\underline{\lambda}_Q I_2 \leq \mathbf{Q} \leq \overline{\lambda}_Q I_2, \tag{1.35}$$

$$\underline{\lambda}_P I_2 \leq \mathbf{P} \leq \overline{\lambda}_P I_2, \tag{1.36}$$

where $\underline{\lambda}_Q$ and $\overline{\lambda}_Q$ are the minimum and the maximum positive eigenvalues of $\mathbf{Q}$, while $\underline{\lambda}_P$ and $\overline{\lambda}_P$ are the minimum and the maximum positive eigenvalues of $\mathbf{P}$. Analogously, one of the initial hypotheses (more precisely hypothesis 4) allows one writing

$$\|\mathbf{u}\| \leq \overline{u}. \tag{1.37}$$

An upper bound $\overline{n}$ for $\|\mathbf{n}\|$ can be analogously provided given that the following proposition applies:

Proposition 1 *There always exists a proper bound $\rho \in \mathbb{R}^+$ for e_1 such that if condition $|e_1| < \rho$ is satisfied, the following inequality holds for $\forall \alpha \in [0.5, 1], \forall d \geq 1$*

$$|e_1| \, |\ln(|e_1|)| \leq d \, |e_1|^{(1-\alpha)}. \tag{1.38}$$

Proof – By imposing

$$\nu := |e_1|, \nu \in \mathbb{R}^+,$$

equation (1.38) can be rewritten as follows

$$\nu \, |\ln(\nu)| \leq d \nu^{(1-\alpha)}. \tag{1.39}$$

The proposition is demonstrated if a value of ρ is found such that, $\forall \nu \leq \rho$ and $\forall \alpha \in [0.5, 1]$, (1.39) applies.

Inequality (1.39) can be equivalently posed into the following form

$$\nu^{\alpha} \, |\ln(\nu)| \leq d. \tag{1.40}$$

Figure 1.3 shows some plots of the left-hand side of (1.40) for $\alpha \in [0.5, 1]$. Evidently, ρ can be obtained by numerically solving (1.40) for a given value of d and for $\alpha = 0.5$. In particular, Figure 1.3 shows that, by assuming $d = 1$, (1.40) is satisfied for $\forall \nu \leq 1.763 = \rho$. ■

It should be noted that $\|\mathbf{n}\|$ can be written as follow:

$$\begin{aligned} \|\mathbf{n}\| = \sqrt{(\chi_1 f(e_1, w) \, |w|^{\alpha} + f(e_1, z))^2 + \ldots} \\ (\chi_2 \, |e_1|^{1-\alpha} f(e_1, w) \, |w|^{2\alpha-1})^2. \end{aligned} \tag{1.41}$$

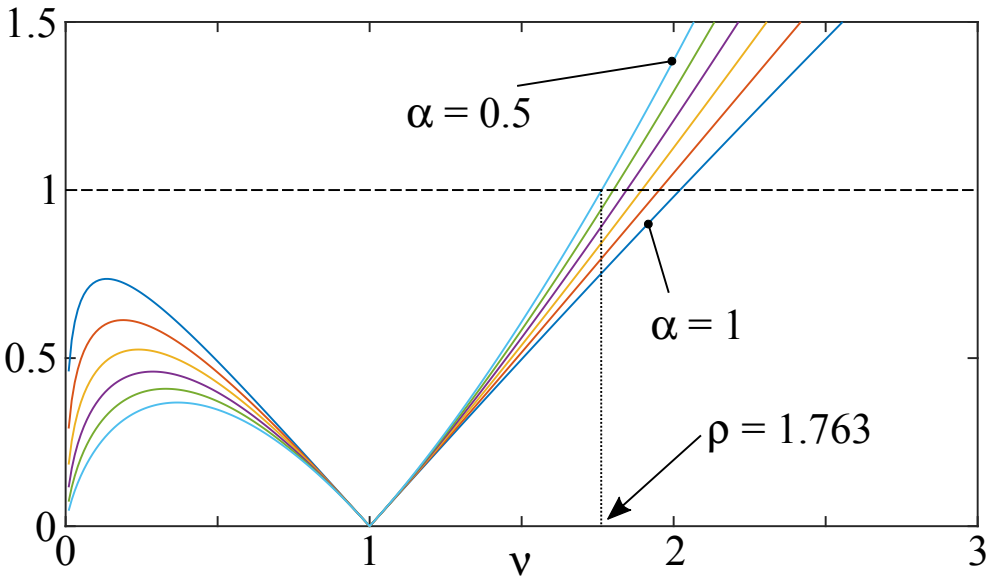


Figure 1.3: Shapes of (1.40) for $\alpha \in [0.5, 1]$.

By virtue of the triangle inequality, the following expression can be written

$$\begin{aligned} \|\mathbf{n}\| \leq \chi_1 |f(e_1,w)| |w|^{\alpha} + |f(e_1,z)| + \\ \chi_2 |e_1|^{1-\alpha} |f(e_1,w)| |w|^{2\alpha-1}. \end{aligned} \tag{1.42}$$

According to the premises many terms in (1.42) admit upper bounds. By defining

$$C_1 := \max\{|f(e_1,w)| |w|^{\alpha}\} \tag{1.43}$$

$$C_2 := \max\left\{\left|\frac{\alpha\tau|z-\dot{y}|}{\varepsilon}\right|\right\} \tag{1.44}$$

$$C_3 := \max\{|f(e_1,w)| |w|^{2\alpha-1}\} \tag{1.45}$$

the following relationship can be obtained from (1.42)

$$\overline{n} \leq \chi_1 C_1 + |e_1| |\ln(|e_1|)| C_2 + \chi_2 |e_1|^{1-\alpha} C_3. \tag{1.46}$$

By further considering (1.38), (1.46) can be finally written as follows

$$\overline{n} \leq \chi_1 C_1 + d |e_1|^{1-\alpha} C_2 + \chi_2 |e_1|^{1-\alpha} C_3. \tag{1.47}$$

Coming back to the original problem, the following inequality can be obtained from (1.34)

$$\dot{V} \leq -\underline{\lambda}_Q \Phi \|\mathbf{\Gamma}\|^2 + 2\Phi \overline{\lambda}_P \|\mathbf{\Gamma}\| \overline{n} + 2\overline{u}\,\overline{\lambda}_P \|\mathbf{\Gamma}\|. \tag{1.48}$$

By virtue of (1.48) and by discarding the obvious solution $\|\mathbf{\Gamma}\| = 0$, condition $\dot{V} < 0$ is achieved if the following condition holds

$$\|\mathbf{\Gamma}\| \geq C_4 \left(\overline{n} + C_5 |e_1|^{1-\alpha} \right), \tag{1.49}$$

where

$$C_4 := 2\frac{\overline{\lambda}_P}{\underline{\lambda}_Q}, \quad C_5 := \frac{\overline{u}}{\alpha}.$$

Equation (1.47) makes it possible to write (1.49) as follows

$$\|\mathbf{\Gamma}\| \geq C_4 \left[C_1 \chi_1 + (dC_2 + C_3 \chi_2 + C_5) |e_1|^{1-\alpha} \right]$$

or, equivalently, as follows

$$\|\mathbf{\Gamma}\| \geq K_1 \chi_1 + (K_2 + K_3 \chi_2) |e_1|^{1-\alpha}, \tag{1.50}$$

where

$$K_1 := C_4 C_1, \quad K_2 := C_4(dC_2 + C_5), \quad K_3 := C_4 C_3.$$

The following inequalities can be directly derived from (1.19)

$$\underline{\lambda}_P \|\mathbf{\Gamma}\|^2 \leq V \leq \overline{\lambda}_P \|\mathbf{\Gamma}\|^2, \tag{1.51}$$

so that one can write

$$\|\mathbf{\Gamma}\| \geq \left(\frac{V}{\overline{\lambda}_P} \right)^{\frac{1}{2}}. \tag{1.52}$$

From (1.17) it immediately descends that $|e_1| = |\Gamma_1|^{\frac{1}{\alpha}}$ and, still bearing in mind (1.51), the following chain of inequalities can be written

$$|e_1|^{1-\alpha} = |\Gamma_1|^{\frac{1-\alpha}{\alpha}} \leq \|\mathbf{\Gamma}\|^{\frac{1-\alpha}{\alpha}} \leq \left(\frac{V}{\underline{\lambda}_P} \right)^{\frac{1-\alpha}{2\alpha}} \tag{1.53}$$

The following inequality is finally achieved by substituting (1.52) and (1.53) into (1.50)

$$\left(\frac{V}{\overline{\lambda}_P}\right)^{\frac{1}{2}} \geq K_1\chi_1 + (K_2 + K_3\chi_2)\left(\frac{V}{\underline{\lambda}_P}\right)^{\frac{1-\alpha}{2\alpha}}. \tag{1.54}$$

The solution of (1.54) provides an estimate of the domain over which condition $\dot{V} \leq 0$ is satisfied. In particular, it can be easily verified that, since $\alpha \in [0.5, 1]$, (1.54) is verified for any value of V larger than a given threshold $\overline{V}$, independently from all the other constant terms. Conversely, (1.54) does not hold for small values of V: the origin is not asymptotically stable, but the system converges to a ball whose radius $\left\|\overline{\mathbf{\Gamma}}\right\|$, according to (1.51), is equal to

$$\left\|\overline{\mathbf{\Gamma}}\right\| = \left(\frac{\overline{V}}{\underline{\lambda}_P}\right)^{\frac{1}{2}}. \tag{1.55}$$

Necessarily, the system stability also requires that the condition posed by Proposition 1 is satisfied. By bearing in mind (1.53), one can assert that

$$\left\|\overline{\mathbf{\Gamma}}\right\|^{\frac{1}{\alpha}} \geq |\Gamma_1|^{\frac{1}{\alpha}} = |e_1|, \tag{1.56}$$

so that, if the following inequality holds

$$\rho \geq \left\|\overline{\mathbf{\Gamma}}\right\|^{\frac{1}{\alpha}}, \tag{1.57}$$

equation (1.38) is satisfied.

Equation (1.54) provides a sufficient condition for the system stability, so that the system could be stable even if such inequality is not satisfied. However, it is important to verify that the feasible set is not totally empty. Indeed, depending on the values assumed by $\overline{\lambda}_P$, $\underline{\lambda}_P$, $\underline{\lambda}_Q$, χ_1, χ_2, K_1, K_2, and K_3, $\overline{V}$ could be so large that (1.57) is never satisfied. By scrutinizing the composition of (1.54), it can be observed that smaller values of $\overline{V}$ are achieved if terms $\overline{\lambda}_P$, χ_1, χ_2, K_1, K_2 and K_3 are sufficiently small, while $\underline{\lambda}_P$ and $\underline{\lambda}_Q$ are sufficiently large. For example, K_2 can be easily reduced by acting on $\overline{u}$ and τ, but for other terms the analysis is more complex. Indeed, $\overline{\lambda}_P$, $\underline{\lambda}_P$, $\underline{\lambda}_Q$, χ_1, and χ_2 are mutually correlated through (1.33): the decrement of one term could cause the increment of another one [16].

1.4 Experimental Results and Comparison

The novel HSTD has been tested with the aid of a BrushLess DC motor (BLDC) in joints of a collaborative robot and controlled by a Raspberry Pi 4 board equipped with a real-time Xenomai kernel. A field-oriented controller was implemented for the torque loop, while standard PID controllers were used for the position and the velocity loops. It is important to clarify that while FOC is commonly associated with Permanent Magnet Synchronous Motors (PMSM)—which have a sinusoidal back-EMF—it can also be applied to BLDC motors when precise current regulation is needed. In this case, FOC is employed specifically for the current loop regulation and to ensure a smooth and dynamic torque response, rather than for sinusoidal control of the motor itself. This approach helps mitigate torque ripples and improves overall motor performance.

By adopting FOC for the BLDC motor, the system benefits from better torque linearity and responsiveness, making it suitable for applications requiring high-performance actuation. However, the motor itself retains the operational characteristics of a BLDC motor, rather than being operated as a PMSM. The sampling rate of the torque control loop was posed equal to 10 kHz, while position and velocity loops run at 2 kHz. The position reference signal is obtained through a point-to-point polynomial trajectory. The same control structure and tuning were used for all the tests, so that the experiments described in this section differ each other only for the adopted velocity estimator. As a consequence, no further details on the controller are provided.

The position signal is affected by a noise whose measured maximum amplitude does not depend on the operating conditions and it is equal to $\overline{w} = 0.008$ rad. The velocity loop requires accurate speed estimates obtained from the position signal: noise and delays worsen the performances. As stated in the introduction, the velocity estimates obtained through a Filtered Finite Difference Derivative (FFDD) amplify the noise and generate delayed signals. The use of an STD partially improves the estimate quality by reducing phase lags, but, as early stated, under steady-state conditions its output noise may destabilize the feedback controller.

The adopted FFDD uses a first order filter with a pole equal to 30. The STD

assumes $\chi_1 = 50$ and $\chi_2 = 100$. The HSTD is tuned as follows: $\tau = 30$, $\varepsilon = 0.5$, and $\alpha_{th} = 0.8$. Parameters χ_1 and χ_2 are selected according to (1.12), by imposing $\overline{\chi}_1 = 200$, $\overline{\chi}_2 = 10000$, $\underline{\chi}_1 = 50$, and $\underline{\chi}_2 = 100$. It should be noted that, when $\alpha = 0.5$, the values of χ_1 and χ_2 are the same for both the STD and the HSTD. For the best performance, STDs and HSTDs require high sampling rates (see also [19]) so that they were updated at 10 kHz.

A first comparison between the standard STD, the novel HSTD, and an FFDD can be done by means of Figure 1.4. Such figure shows the reference velocity signal compared with its estimates obtained through the three techniques. Under dynamic conditions $\alpha = 0.5$, so that the STD and the HSTD have equivalent behaviors. As can be seen from the detail in Figure 1.4a, both techniques return estimates affected by small delays. Conversely, the FFDD provides a delayed and noisy output: by lowering its cut-off frequency noise reduces, but delay increases, and vice versa. The FFDD pole that was chosen for the experiment represents a compromise between tracking performances and the robot's actuator noise.

Figure 1.4b shows the behavior of the three strategies under stationary conditions: as desired, the noise on the velocity signal provided by HSTD is almost negligible, while an evident noise affects the STD output. Again, the FFDD returns the worst performances.

Such conclusion is even more evident by observing Figure 1.5, which shows the velocity errors with respect to the reference trajectory. During the transient, similar errors are detected for the STD and the HSTD, while slightly larger values are obtained for the FFDD. Conversely, when the robot's joint is at rest, the average error is close to zero in the three experiments, but its standard deviation shows evident differences: it is equal to $5.24 \cdot 10^{-2}$ rad s^{-1} for the FFDD, to $3.85 \cdot 10^{-2}$ rad s^{-1} for the STD, and to $4.3 \cdot 10^{-3}$ rad s^{-1} for the HSTD. As desired, when the velocity is zero, the HSTD noise is roughly one order of magnitude smaller than the ones achieved with the other methods [16].

The proposed methodology introduces a simple yet effective differentiator that can be used not only to calculate velocity but also to obtain all necessary derivatives in sensorless collaborative robot applications. In the next chapter, a new impact de-

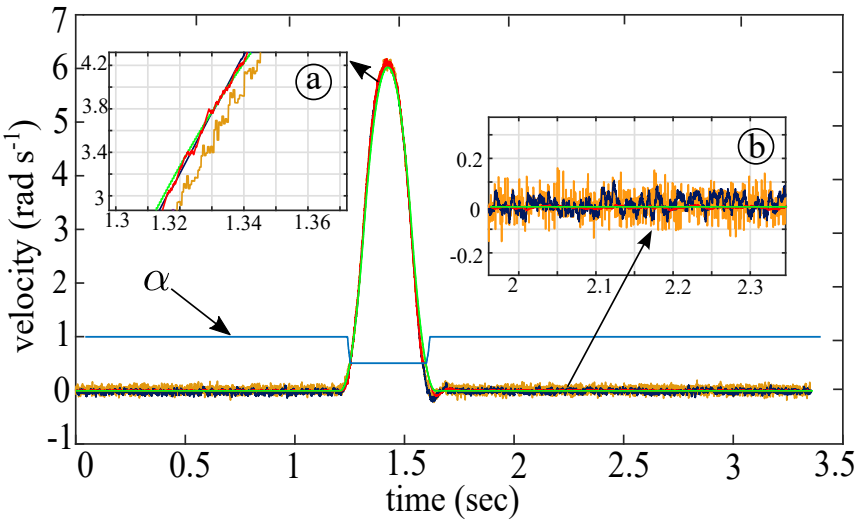


Figure 1.4: Comparison between the velocity reference signal (green line) and the outputs of the STD (blue line), of the novel HSTD (red line), and of the FFDD (orange line). The shape of α is represented through a light blue line.

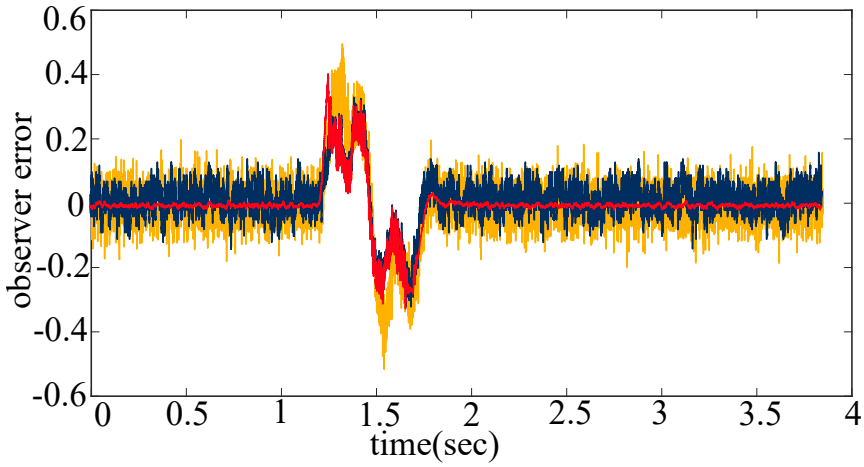


Figure 1.5: Difference between the velocity reference signal and the estimated velocities for the HSTD (red line), the STD (blue line), and the FFDD (orange line).

tection methodology is introduced through the application of the proposed Hybrid Super-Twisting approach.

Chapter 2

Estimation of Impact Using Extended State Acceleration Observer

2.1 State of the Art

Collaborative robots, also known as cobots, have become increasingly common in the industry over the past few decades. They are designed to work alongside human workers, assisting with repetitive or dangerous tasks, and can increase productivity and safety in many different industries [20]. Cobots share their workspace with humans and as a result, it is important to have safe collaboration work. The applications of these robots usually are repetitive works, without adding further value to the product that is handled, such as assembly, packaging, material handling, quality control, and pick and place [21], which require high speed of their movements. Therefore, any collision between humans and this type of robot needs to be considered and detected as fast as possible to avoid any damage and injury to humans.

It is worth pointing out that an impact may occur not only on the end-effector, but also randomly on one of robot's links. As a result, monitoring and detecting them on all joints are essential. Since sensors are quite expensive instruments, most industrial

robots are not equipped with force/torque sensors on all of their joints to perceive external forces. Therefore, the usage of sensorless methods for collision detection is inevitable.

Accordingly, designing a proper estimation for sensorless collision has become a challenging area in industry [22]. The requirements of estimating collision in collaborative robots are: 1) detecting fast with less delay 2) detecting without fault 3) estimating with less noise. The primary consideration is to prioritize the prevention of harm or injury to humans. For the second feature, since there is always uncertainty in the model and parameters, it is important to not send faulty detection to the controller. For the third requirement, since in most industrial places and models there always exists noise and disturbances from the environment or measurement sensors, it is also necessary to have an accurate estimation to not get the fault command.

Lots of works have been dedicated to use sensorless collision estimations. One of the common methods is the so-called Generalized Momentum (GM), which has been introduced in [23]. This method is based on knowledge of the dynamics of the robot and its passivity property. Since the methodology is based on a linear filter, it will introduce phase shift in the estimation of external torques. As a result, it is not the best and accurate method to calculate the external torque in case of noise in the measured signals. In other words, in case of noise, by increasing the cut-off frequency of this filter (gain of the observer) the signal becomes noisier and it will not be accurate. Conversely, in case of low gain the result will be a delayed external torque estimation, which is not proper for collision detection purposes.

Following the GM methodology, many papers have focused on using this method based on the idea of having less mathematical calculation and simple strategy. In [24], the authors propose to use the GM property and design a sliding mode observer for the new model. As well as using a Luenberger Observer (LO) to find stability conditions for the dynamic of estimator. Additionally, in [25] the authors used the idea of GM and newton Euler method to replace the derivative of momentum with a new algorithm, utilizing a friction-based filter to compensate for the effects on gain parameter of GM. The same concept of GM and Sliding mode observer has been used in [26], in which the authors considered linear and non-linear estimation to have

finite time convergence. However, the problem of these methods is chattering and noisy signals which make the estimation less reliable.

Two different algorithms for collision detection have been introduced in [22]. The crucial part of these estimation methods is the knowledge of the dynamics of the system. In addition, all these methods have used conventional estimation techniques which are considered a correction term in the algorithm by considering the residual term, mainly called Luenberger Observer. One of these methods is velocity-based observer [27]. In this method, the estimated torque is a variable filter frequency. However, this algorithm exploits a non-linear filter, which estimates the external torque with delay, and, moreover, the computational time is not negligible. Another algorithm is inverse dynamic identification, which depends on the control method and its parameters, and needs to be tuned accurately. As stated in [28], it depends on the executed trajectory which needs to be smooth.

Recently, other observer methods have been used for calculating the external torque, such as Sliding Mode [26] and Extended State Observer [29, 30]. Even these methodologies leverage the system model. However, since measurement noises are not avoidable in real environment, in these methods it is difficult to reach a good trade off between delay and increasing noises of the observer. Therefore, the accuracy and speed in recognizing properly the collision will not be achieved.

In order to minimize the aforementioned problems that affect the previously mentioned methods, in this paper a new model-based collision estimation algorithm has been introduced. The suggested method is a new algorithm for fast recognition of the collision without any force/torque sensors, which is simple to tune and produces stronger signals for recognition of the impacts with less delay compared to the state-of-the-art algorithms.

The introduced method is based on the idea of extending the state-space of the robot's dynamics by exploiting the Extended State Observer (ESO) [31] structure, and Hybrid Super-twisting application that has been introduced in the previous chapter. The latter has been used to achieve less noise and delay in estimating the new method required signals while the ESO observer has been adopt thanks to its features: it has linear gains, it converges in finite time, and its characteristic parameters,

for having a stable observer, can be obtained and analyzed trivially.

2.1.1 Already Existed Methods

In this section three different collision detection methods will be described. Each of them uses an alternative model-based approach. As a result, firstly, the general dynamical model of the robot will be explained.

Most of the works that has been done to detect collisions without sensors has been focused on improving the delay in detecting collision and finding a trade-off between having an appropriate estimation and less noise in detection.

The Dynamical Model of a Manipulator

A proper and complete model is a crucial aspect of model-based estimations. In order to reconstruct the external torque signals, and identify unexpected collisions, the following model of n Degree-of-Freedom (DoF) robot in joint space has to be considered:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_f = \boldsymbol{\tau}_j + \boldsymbol{\tau}_{ext} \tag{2.1}$$

where $\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}} \in \mathbb{R}^n$ are position, velocity and acceleration of the joints, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n\times n}$ is the positive definite inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \in R^{n\times n}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ are centripetal/Coriolis matrix and gravitational vector respectively. $\boldsymbol{\tau}_j \in \mathbb{R}^n$ and $\boldsymbol{\tau}_{ext} \in \mathbb{R}^n$ are joint actuated torque and external torque respectively. Supposing that the robot moves in free space, the external torque consists solely of collision effects perceived by the joints of the robot. Finally, $\boldsymbol{\tau}_f \in \mathbb{R}^n$ is the torque friction, which has been modelled as follows:

$$\tau_{f_i} = \mu_{v_i}\dot{q}_i + \mu_{c_i}\mathrm{sgn}(\dot{q}_i), \quad i = 1,\ldots,n, \tag{2.2}$$

where μ_{c_i} and μ_{v_i} are the Coulomb and the viscous friction coefficients for the i-th joint, respectively. The friction parameters can be estimated by means particular tests, thus exploiting the least square method. Hence, by fitting data with a polynomial curve, the parameters regarding viscous and coulomb friction can be identified.

Generalized Momentum

In [23, 32, 33] it was suggested to use the Generalized Momentum (GM) of a robot for collision detection purposes. As known, the GM can be expressed as follows

$$h = \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}. \tag{2.3}$$

By virtue of the passivity property of the robot, the following formula can be written:

$$\dot{\mathbf{M}}(\mathbf{q}) = \mathbf{C}^T(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}). \tag{2.4}$$

Let us define the residual of the system as $\mathbf{r} := \mathbf{K}(h - \hat{\mathbf{p}})$, where $\hat{\mathbf{p}}$ is the estimated generalized momentum, $\mathbf{p}$ is the actual one, and $\mathbf{K} := \text{diag}(k_1, k_2, \ldots, k_N)$ is a diagonal, positive definite matrix. By recalling (2.1) and (2.4), and after a few algebraic manipulations, it is possible to calculate the the system residual through the following equation (more explanations can be found in [32]).

$$\mathbf{r} = \mathbf{K}\Big[\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} - \textstyle\int_0^t (\boldsymbol{\tau}_j - \boldsymbol{\tau}_f + \mathbf{C}^T(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) + \mathbf{r})\, ds - \mathbf{p}(0)\Big] \tag{2.5}$$

The Laplace transform of the derivative of (2.5) reveals the following relationship existing between the components of $\mathbf{r}$ and the ones of the actual external torque $\boldsymbol{\tau}_{ext}$

$$r_i = \frac{k_i}{s + k_i}\tau_{ext_i}\,,\; i = \{0, 1, \ldots, N\} \tag{2.6}$$

i.e. r_i is a low pass filtered estimate of the external torque τ_{ext_i} affecting joint i. The low pass filter action minimizes the high-frequency noise which affects the torque estimate, but induces undesired delays between such estimate and the actual external torque.

Gains $k_i\,,\; i = \{0, 1, \ldots, N\}$ need to be chosen sufficiently high in order to guarantee an accurate estimate of $\boldsymbol{\tau}_{ext}$ ($\mathbf{r} \to \boldsymbol{\tau}_{ext}$). However, such high values would amplify the estimation noise. Conversely, low gains cause undesired delays in the collision detection mechanism.

Velocity Extended State Observer

Another method for calculating the collision torque is based on velocity signals. In [31], a Velocity Extended State Observer (VESO) was proposed for the estimation of the external torque. As can be deduced by its name, it exploits each joint velocity signal. This approach is based on an Extended State Observer, which uses the model of the system as well, and, as an extra input of the model, the extended state.

Let us consider the following dynamic system

$$\ddot{\mathbf{y}} = \boldsymbol{f}(t,u,\mathbf{y},\dot{\mathbf{y}}) + \boldsymbol{\delta}(t,\mathbf{y},\dot{\mathbf{y}}) \tag{2.7}$$

where $\boldsymbol{\delta}(t,\mathbf{y},\dot{\mathbf{y}})$ represent an external unknow disturbance, and $\mathbf{y}$ and $\dot{\mathbf{y}}$ are joint position and joint velocity vectors respectively. By defining $\mathbf{x}_1 := \mathbf{y}$, $\mathbf{x}_2 := \dot{\mathbf{y}}$, and $\mathbf{x}_3 := \boldsymbol{\delta}(t,\mathbf{y},\dot{\mathbf{y}})$, (2.7) can be posed in state space form, according to the following equation

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 = \mathbf{x}_3 + \boldsymbol{f}(t,u,\mathbf{y},\dot{\mathbf{y}}) \\ \dot{\mathbf{x}}_3 = \dot{\boldsymbol{\delta}}(t,\mathbf{y},\dot{\mathbf{y}}) \end{cases} . \tag{2.8}$$

Accordingly, the following extended state observer can be proposed for the computation of its state

$$\begin{cases} \dot{\hat{\mathbf{x}}}_1 = \hat{\mathbf{x}}_2 - \boldsymbol{\beta}_1 \mathbf{e}_p \\ \dot{\hat{\mathbf{x}}}_2 = \hat{\mathbf{x}}_3 + \boldsymbol{f}(t,u,\mathbf{y},\dot{\mathbf{y}}) - \boldsymbol{\beta}_2 \mathbf{e}_p \\ \dot{\hat{\mathbf{x}}}_3 = -\boldsymbol{\beta}_3 \mathbf{e}_p \end{cases} \tag{2.9}$$

where $\mathbf{e}_p = \mathbf{y} - \hat{\mathbf{x}}_1$ is the position error and $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2$, and $\boldsymbol{\beta}_3$ are positive definite diagonal matrices, which coefficients are chosen so as to guarantee a proper dynamic answer of the estimator (for more details, see [31]).

As shown in [31], since the observation of the position signal is not strictly required for the the estimation of the impact torque, a reduced order observer, based on the velocity error signal can be adopted. It can be expressed as follows

$$\begin{cases} \dot{\hat{\mathbf{v}}}_1 = \hat{\mathbf{v}}_2 + \boldsymbol{f}(t,u,\mathbf{y},\dot{\mathbf{y}}) - \boldsymbol{\gamma}_1 \mathbf{e}_v \\ \dot{\hat{\mathbf{v}}}_2 = -\boldsymbol{\gamma}_2 \mathbf{e}_v \end{cases} \tag{2.10}$$

where $\mathbf{e}_v = \dot{\mathbf{y}} - \hat{\mathbf{v}}_1$ is the velocity estimation error, while γ_1 and γ_2 are positive definite diagonal matrices, which values need to be tuned to have a proper result.

Now, if we specialize the observer by considering the model of the robot (2.1) as the required system, it is possible to reformulate the equations stated in (2.10) by applying the following substitution: $\boldsymbol{f}(t,u,\mathbf{y},\dot{\mathbf{y}}) = \mathbf{M}(\mathbf{q})^{-1}\Big[\boldsymbol{\tau}_j - \boldsymbol{\tau}_f - \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q})\Big]$. Finally, the external torque can be simply computed by keeping in mind the following equation: $\boldsymbol{\delta}(t,\mathbf{y},\dot{\mathbf{y}}) = \mathbf{M}^{-1}\boldsymbol{\tau}_{ext}$.

2.2 Methodology Definition

The method proposed in this work for impact detection, use joint acceleration signals, which are estimated by means of the Hybrid Super-twisitng method. However, the proposed one owns an intrinsic filter behavior, which is helpful to limit noise affecting the estimated acceleration signals. In this section, the proposed algorithm for collision detection will be explained. In the first subsection, the method that has been derived for the first time for detecting incidental collision in a fast way and its properties will be described. In the next subsection, the required components for the collision detection algorithm will be explained and its features will be presented.

2.2.1 Extended State Acceleration Observer Algorithm for a Fast Collision Detection

The proposed new torque estimation technique is model-based, which means it needs to know the complete dynamical model of the system as expressed in equation (2.1). Moreover, this new method necessitates the state-space representation of the entire model.

The estimation method called Extended State Observer (ESO) has been selected for our purposes. ESO is suitable for state estimation in non-linear systems. Moreover, ESO observers allow to tune their bandwidth to obtain the desired stability/robustness properties. Furthermore, the implementation of it is quite simple and it is suitable in a real-time setup, since the calculus are not so much computationally complex. Since

the proposed method is based on exploiting the ESO architecture, all the reasoning may start from its system equations (2.8) introduced in 2.1.1.

The purpose of the estimation is to estimate non available auxiliary state of the dynamical system. Solely the state $\mathbf{x}_1 = \mathbf{q}$ is considered already known by means specific sensors. Since $\mathbf{x}_1$ is a determined state, considering it for observation is not essential. Therefore, by not considering $\mathbf{x}_1$, it is possible to reduce the state order of the system, that implies a reduced order for the observer as well. The new states can be named as follows: $\hat{\mathbf{x}}_1 = \dot{\mathbf{q}}$ and $\hat{\mathbf{x}}_2 = \mathbf{M}^{-1}\boldsymbol{\tau}_{ext}$. Accordingly, the new state-space representation of the system can be expressed as follows:

$$\begin{cases} \dot{\hat{\mathbf{x}}}_1 = \hat{\mathbf{x}}_2 + \mathbf{f}(t, u, \mathbf{y}, \dot{\mathbf{y}}) \\ \dot{\hat{\mathbf{x}}}_2 = \dot{\boldsymbol{\delta}}(t, \mathbf{y}, \dot{\mathbf{y}}) \end{cases} \tag{2.11}$$

Finally, the new reduced-ESO observer, associated to the above system, has the same structure (2.10) of the one defined in the previous section.

However, in order to detect a collision more quickly, the signals with fast derivatives need to be considered. It is worth pointing out that acceleration is a signal that will bring to faster detection of any changes in the states of the system. Acceleration signals own fast dynamics and have been considered as an input to the observer in order to estimate promptly and accurately the external torque. More specifically, in this work, since acceleration signals have been used even within estimation errors, them play an important role inside the observer. Furthermore, the derivative of external torques need to be evaluated in order to speed up the impact identification.

Therefore, in according to the aforementioned intuitions, new changes need to be applied in the system states as follows: $\bar{\mathbf{x}}_1 = \ddot{\mathbf{q}}$, $\bar{\mathbf{x}}_2 = \dot{\boldsymbol{\delta}}$, where the final state-space of the required system can be written as second order state.

The following formulation, in order to define the new one, should be brought in mind:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{q})\Big[\boldsymbol{\tau}_j - \boldsymbol{\tau}_f - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q})\Big] + \mathbf{M}^{-1}(\mathbf{q})\boldsymbol{\tau}_{ext} \tag{2.12}$$

By computing the derivative of the above equation, and taking into account the passivity property of robots in (2.4), it is possible to achieve the following formula-

tion:

$$\dddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{q})\Big[\dot{\boldsymbol{\tau}}_j - \dot{\boldsymbol{\tau}}_f - (\mathbf{C}^T(\mathbf{q},\dot{\mathbf{q}}) + 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}))\ddot{\mathbf{q}} - \dot{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \dot{\mathbf{g}}(\mathbf{q})\Big] + \mathbf{M}^{-1}(\mathbf{q})\dot{\boldsymbol{\tau}}_{ext} \tag{2.13}$$

Finally, system states can be written as follows:

$$\begin{cases} \dot{\overline{\mathbf{x}}}_1 = \overline{\mathbf{x}}_2 + \dot{\mathbf{f}}(t,u,\mathbf{y},\dot{\mathbf{y}},\ddot{\mathbf{y}}) \\ \dot{\overline{\mathbf{x}}}_2 = \ddot{\boldsymbol{\delta}}(t,\mathbf{y},\dot{\mathbf{y}},\ddot{\mathbf{y}}) \end{cases} \tag{2.14}$$

where, $\dot{\mathbf{f}}(t,u,\mathbf{y},\dot{\mathbf{y}},\ddot{\mathbf{y}}) = \mathbf{M}^{-1}(\mathbf{q})\Big[\dot{\boldsymbol{\tau}}_j - \dot{\boldsymbol{\tau}}_f - (\mathbf{C}^T(\mathbf{q},\dot{\mathbf{q}}) + 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}))\ddot{\mathbf{q}} - \dot{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \dot{\mathbf{g}}(\mathbf{q})\Big]$. The toughest calculations that regard this statement, first of all the inverse of matrix M, can be computed offline in order to reduce the computational time associated to this method, since it has to be used within a real-time system.

The Extended State Observer can be defined with the acceleration error $\mathbf{e}_a = \ddot{\mathbf{y}} - \overline{\mathbf{z}}_1$ as follows:

$$\begin{cases} \dot{\overline{\mathbf{z}}}_1 = \overline{\mathbf{z}}_2 + \dot{\mathbf{f}} - \mathbf{k}_0\mathbf{e}_a \\ \dot{\overline{\mathbf{z}}}_2 = -\mathbf{k}_1\mathbf{e}_a \end{cases} \tag{2.15}$$

As one can see from this new derived method named Extended State Acceleration Observer (ESAO), it is possible to estimate the quick changes of external torques, i.e., their derivatives. This means, if any incidental collision happens, the ESAO method is able to detect it by exploiting the acceleration changes of each robot joint. The schematic of this method can be viewed in Figure 2.1 and Figure 2.2 to appreciate both the insight of the proposed system, and the usage of it within the control system of a Collaborative Robot [34].

Analysis of ESAO's Characteristic Parameters

In order to analyse the parameters of the defined ESAO, the characteristic polynomial of it must be evaluated. Firstly, the observer dynamics should be determined. The mentioned definition will be made by two observer errors as follows:

$$\begin{cases} \overline{\mathbf{e}}_1 = \overline{\mathbf{z}}_1 - \overline{\mathbf{x}}_1 \\ \overline{\mathbf{e}}_2 = \overline{\mathbf{z}}_2 - \overline{\mathbf{x}}_2 \end{cases} \tag{2.16}$$

Then, by taking the derivative of errors, and considering the equation (2.15), the dynamic of the observer can be characterized as follows:

$$\begin{bmatrix} \dot{\bar{\mathbf{e}}}_1 \\ \dot{\bar{\mathbf{e}}}_2 \end{bmatrix} = \begin{bmatrix} -\mathbf{k}_0 & I \\ -\mathbf{k}_1 & 0 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{e}}_1 \\ \bar{\mathbf{e}}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\ddot{\boldsymbol{\delta}} \end{bmatrix} \tag{2.17}$$

where $\mathbf{k}_0, \mathbf{k}_1 \in \mathbb{R}^{n \times n}$ are diagonal matrices. Therefore, in order to exploit the ESO features, the system must be Hurwitz. Hence, in order to satisfy the Hurwitz criterion, we should have: $|SI - A| < 0$, in which A is equal to:

$$A = \begin{bmatrix} -\mathbf{k}_0 & I \\ -\mathbf{k}_1 & 0 \end{bmatrix} \tag{2.18}$$

By using the above definition, the stability can be achieved solely by imposing $\mathbf{k}_0, \mathbf{k}_1 > 0$. In other words $\mathbf{k}_0$ and $\mathbf{k}_1$ must be positive definite.

In the most industrial robotic applications only encoders are available. As a consequence, just $\mathbf{x}_1 = \mathbf{q}$ can be measured. Therefore, for taking advantages of the proposed method's full features, velocity and acceleration of the system must be known. Furthermore, as shown before, ESAO needs also to know the derivatives of the mechanical active torques. Hence, all the aforementioned derivatives must be computed, and they have been accomplished by using the introduced hybrid-super twisting approach.

2.2.2 ESAO Required Signals

This section explains how the Hybrid Super-Twisting Differentiator (HSTD) is used to estimate key derivative signals, including joint velocities, joint accelerations, and joint torque derivatives. As discussed in the previous section, the Extended State Acceleration Observer (ESAO) methodology requires velocity, acceleration, and current derivatives for torque estimation. To generate these signals without additional sensors, the proposed HSTD is employed.

The hybrid Super-Twisting approach depends on selecting appropriate parameters to achieve optimal results. A trial-and-error method for parameter selection is not

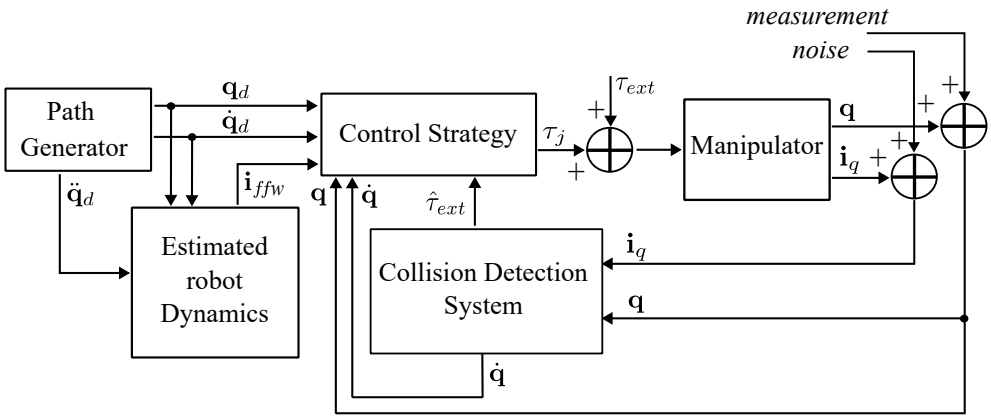


Figure 2.1: Schematic of the usage of collision detection system.

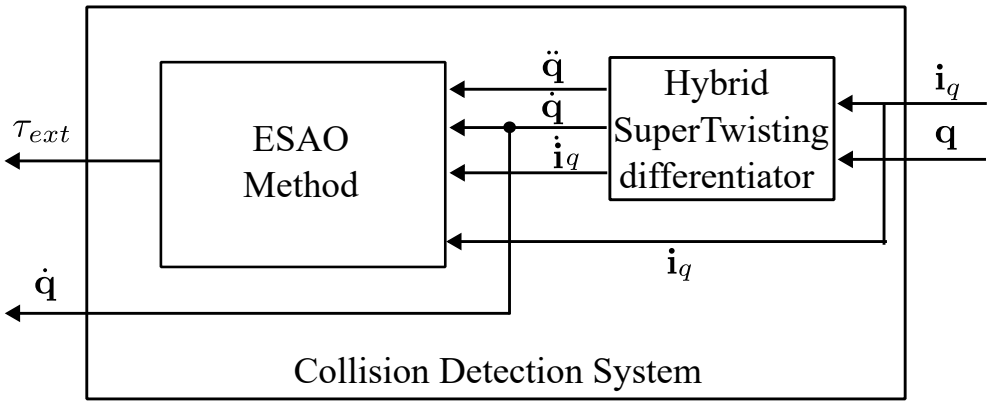


Figure 2.2: Schematic of the Collision detection methodology, Extended State Acceleration Observer (ESAO).

ideal, as finding optimal values that ensure observer stability is challenging, as discussed in the previous chapter. Therefore, a simple optimization method—the Nelder-Mead algorithm (explained in [35])—was applied to determine the best parameters. This approach enhances the accuracy of signal differentiation while minimizing noise and phase lag.

To optimize HSTD parameters, a multi-objective cost function was designed to minimize estimation noise, tracking error, and response delay while ensuring system stability. This function included penalty terms to suppress excessive noise in estimated signals and errors in trajectory tracking. The Nelder-Mead algorithm iteratively adjusted HSTD parameters through closed-loop simulations of the impact detection system, evaluating their influence on the accuracy and robustness of estimated velocity and acceleration signals. To maintain system stability, constraints were imposed based on HSTD stability analysis, ensuring that the optimized parameters met required conditions. The optimization strategy prioritized noise reduction in steady-state conditions while preserving sharp impact detection capabilities, leading to a more reliable and responsive system.

Due to sudden impact forces, signals such as velocity and acceleration change rapidly and sharply. The optimization results indicate that the parameter for the second state of the observer must be at least ten times larger than the first parameter. Additionally, the first parameter should be selected based on stability conditions, as referenced in (1.50), to ensure the observer remains stable. While this thesis does not detail the optimization technique itself, further information can be found in [35].

In the next section, the validation of the introduced collision detection algorithm will be examined, and the ESAO method will be compared to the existing Generalized Momentum (GM) and velocity-based ESO (VESO).

2.3 Simulation Results and Comparison

The proposed collision detection method has been tested in a simulation environment on 2 DoF as a test bench robot which is extracted from the experimental SCARA robot (shoulder and elbow joints). The shoulder and elbow joints are linked to the

Table 2.1: Input signals of the impact torque estimators

	$\mathbf{q}$	$\dot{\mathbf{q}}$	$\boldsymbol{\tau}_j$	$\ddot{\mathbf{q}}$	$\dot{\boldsymbol{\tau}}_j$
GM	✓	✓	✓	-	-
VESO	✓	✓	✓	-	-
ESAO	✓	✓	✓	✓	✓

Table 2.2: Parameters of Hybrid Super-Twisting.

Parameters	χ_1	χ_2
$\dot{\tau}_j$	35	700
$\dot{\mathbf{q}}$	810	85000
$\ddot{\mathbf{q}}$	810	85000
$\dot{\mathbf{i}}$	35	700

respective actuators – two BLDC motors. The motor controllers exploit an inner Field Oriented Control (FOC), which receives current set points by the adopted control strategy. That control strategy, as can be appreciated by inspecting the Figure 2.1, consists of the so-called Feedforward Nonlinear Control (see [36]). The comparative study proposed in the reminder of this section will involve all the detection techniques earlier discussed, i.e., the GM, the VESO and the novel ESAO method.

The input signals required by all the methods are summarized in Table 2.1. Joint variables $\mathbf{q}$ are directly provided by the motor encoders, while, for a fair comparison, the same $\dot{\mathbf{q}}$ were used for all the detection methods. Similarly, the same $\boldsymbol{\tau}_j$, i.e., the joint torque, is shared by all the methods.

The latter is obtained from the motor currents by exploiting the system model, which parameters were preliminarily estimated. The ESAO further requires $\dot{\tau}_j, \dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$, which are obtained by means Hybrid Super-Twisting that has been explained in the previous section. The parameters for Hybrid Super-Twisting have been chosen to have fast estimation, less phase lag, and low noise. In Table 2.2, the parameters for each required signal in Hybrid Super-Twisting have been shown.

The tests followed the procedure next will be described. The manipulator executes a repetitive trajectory in the free space and all the detection methods run in parallel. A random impact is generated and the detection time is measured: lower detection times correspond to better performances.

For the best performances, the GM, the VESO and the ESAO estimators need to be properly tuned. Since the GM, the VESO and the ESAO use only a few tuning parameters, they were selected through a trial-and-error strategy, looking for the same amount of noise within the estimated signals, and the best performances: fast detection of the external torque, good torque estimation, and reduced output noise in case movements in the free space.

Concerning the last aspect, it is known that all the detection methods potentially provide false positives in the absence of impacts. Multiple reasons are the basis of this behavior: model uncertainties, sensors' noises, the use of approximated derivatives, etc... As a consequence, proper thresholds need to be adopted depending on the estimator used. To this purpose, for all estimation methods, a long sequence of movements in the free space was executed and limits were chosen to avoid false positives. Smaller thresholds should be preferred, since, usually, they allow faster reaction times. To compare the effectiveness of each method in the collision detection task, GM, VESO, and ESAO parameters were tuned so as to set the same threshold for all of them. A unique threshold (one for each joint), in addition to the usage of the same input signals, can infer which is the fastest method among them (all bounds can be seen in Table 2.3 and Table 2.4).

For all the impact detection methods, the tuning parameters were chosen through an iterative procedure that aimed at reducing the impact detection times. Since in simulation, the exact time of applied force is detectable, the delay between the real applied force and the estimated external torques will show the performance of each method. The threshold is considered unique for each joint and the same for all the methods in this way since the chosen parameters for all the methods have the same noise in movement free and impact free phase, the comparison will be reliable and fair. The comparison has been done in two elbow and shoulder joints which will be explained in the following.

Table 2.3: Table of external torque thresholds and parameters.

ETE method	Shoulder threshold	Elbow threshold
GM, VESO, ESAO	0.2 N·m	0.3 N·m

Table 2.4: Table of external torque parameters.

ETE method	Shoulder parameters		Elbow parameters	
GM	80	-	80	-
VESO	1450	65000	1450	84000
ESAO	3200	65000	3200	65000

A common characteristic of all the estimation methods (see also [32]) is that collisions that affect link i, also activate the detection system of links $1,2,...,i-1$, while they are not detected for links from $i+1$ to N, where N is the number of links of the manipulator. Consequently, the link involved in the collision can be identified and a proper reaction can be programmed. One of the methods that already has been described in [32] is the energy dissipation control strategy. This method has been used for the test to stop the movement when impact is detected. This method has been utilized based on the slower method to not effect the external torque evaluation. In the energy dissipation method, when the robot's joint experiences an impact, the joint can be stopped by applying torque in the opposite direction of the movement. This is achieved by defining the maximum current required in such a scenario (For more information refer to [32]).

2.3.1 Comparison Results

In this section, the GM, VESO and ESAO will be compared in terms of detection time. Since a specific force can be applied in the simulation, the timing of the collision is known for this test. A force of 3 N·m has been selected for testing, and it is applied to a rigid object at various positions corresponding to each joint. For a fair comparison, all impact detection techniques were executed in parallel. Given that

the manipulator controller was tuned for precise trajectory tracking, unexpected impacts could cause instability in the system. To maintain control of the robotic system while avoiding interference with external torque estimations, the slowest estimator was used as a trigger for the reactive method. This method is activated when the trigger surpasses a sufficiently large threshold. All the triggering thresholds were set larger than the noise bound in the rest phase, the threshold for each joint has been shown in Table 2.3 to allow all torque estimators to overcome the real thresholds.

Shoulder Collisions

The first comparison regards when solely a shoulder collision happened. Both links were moving based on a third-order trajectory planner. This trajectory planner gave set points in position, velocity, and acceleration to each joint. The maximum velocity reached in movements was 4 rad/s. Collision have been tested around the peak of highest velocity movements.

Initially, as considered the collision happened to the shoulder at 7.008 seconds which has been considered as the point of collision.

Figure 2.3 illustrates the behavior of each collision detection signal in the event of an incidental impact on the shoulder link. Notably, ESAO detected the unexpected impact earlier than all other algorithms, identifying the collision at 7.0132 seconds. This means it detected the impact within just 5.2 ms after the collision occurred. GM and VESO also successfully detected the impact, with GM responding after 7.0148 ms and VESO after 7.0182 ms.

In order to have reliable results, the test has been done multiple times. In this way, the average timing delay in each method can be observed and compared together. The result can be found in the Table 2.5.

From the aforementioned table it can be deduced that the ESAO averagely works like the GM method. However, in general, ESAO detected collisions with less delay concerning both the GM and VESO methods. VESO ranked as third in detecting impact.

Finally, Figure 2.4 presents the external torque of the elbow during a shoulder collision. As expected from the definition of external torques (2.3) , and since the

Table 2.5: Collision detection timings

ETE method	Mean collision detection timings in Shoulder (ms)	Mean collision detection timings in Elbow (ms)
GM	6.9	8.1
VESO	10.2	11.9
ESAO	5.2	6.4

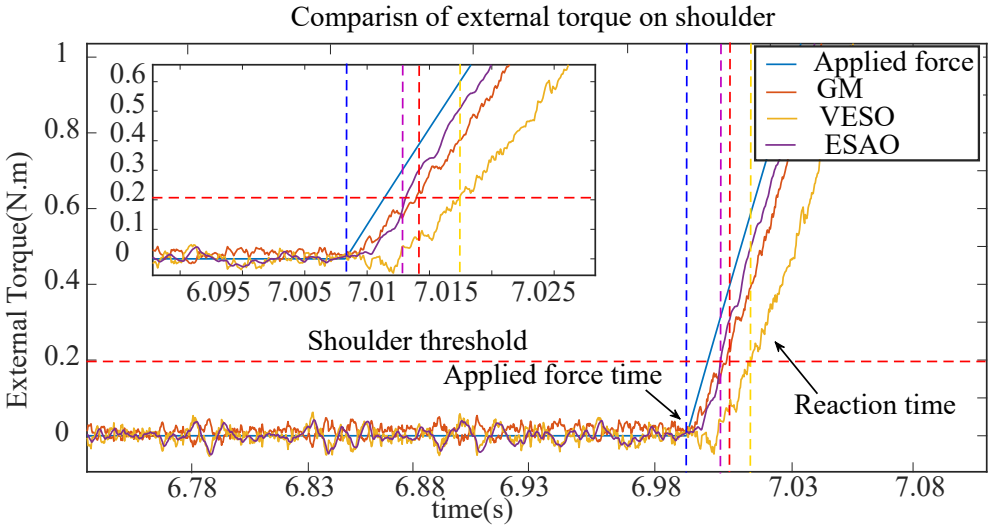


Figure 2.3: Comparison of external torque in impact on shoulder

shoulder joint precedes the elbow joint in the kinematic chain, none of the methods detected a collision at the elbow.

Elbow Collisions

The second scenario involves a comparison where only in elbow collision occurred. This movement consisted of 1.5 rad elbow joint angle displacement. The maximum velocity reached in shorter movements was 4 rad/s. The shoulder movements introduced in the previous subsection were synchronized with the elbow movements

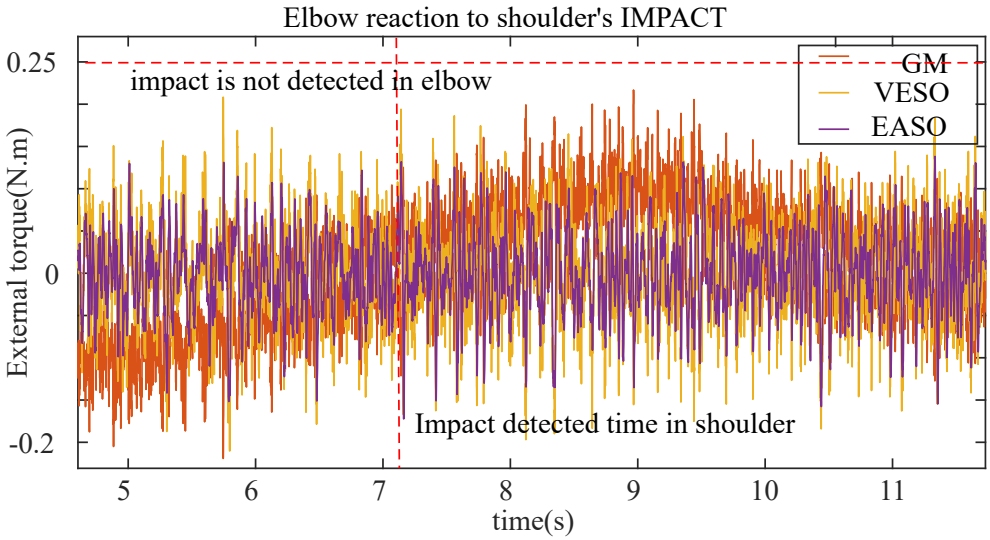


Figure 2.4: Elbow reaction towards shoulder's impact

described here.

The exact collision time has been recorded in the simulation environment as 7.0055 seconds when an elbow collision happened.

By inspecting the Figure 2.5 it is possible to compare the timing of collision identification of the different external torque estimation methods. ESAO detected the collision at 7.0119 seconds with a 6.4-ms delay from the real-time of collision. Other methods, including GM and VESO, detected the collision respectively with 8.3 and 11.6 ms delay. Further collision detection timings on average are available in Table 2.5.

What is shown in Figure 2.5 is a confirmation of the result shown in Table 2.5. Indeed, the latter infers with certainty that ESAO always detected collisions first. Even though the GM method is closest to the ESAO method in detection, still ESAO with average 1.7ms improvements shows a better response. Moreover, the reaction from the shoulder towards collision in the elbow has been demonstrated in Figure 2.6. The time of impact detection varies from the one in the elbow since first it was detected in the elbow, then the shoulder shows its effects. As can be appreciated from the previous figure, and as expected before, even in the shoulder a collision can be identified.

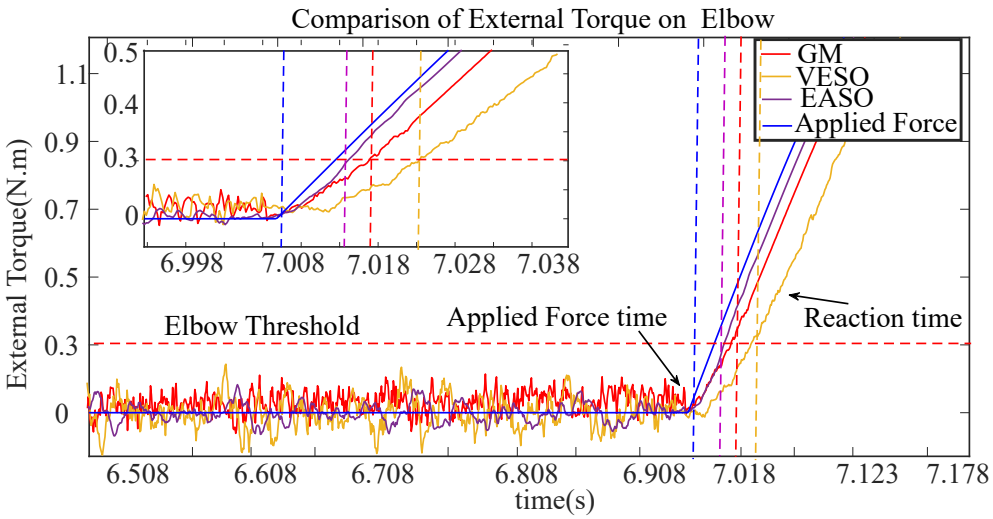


Figure 2.5: Comparison of external torque in collision on elbow

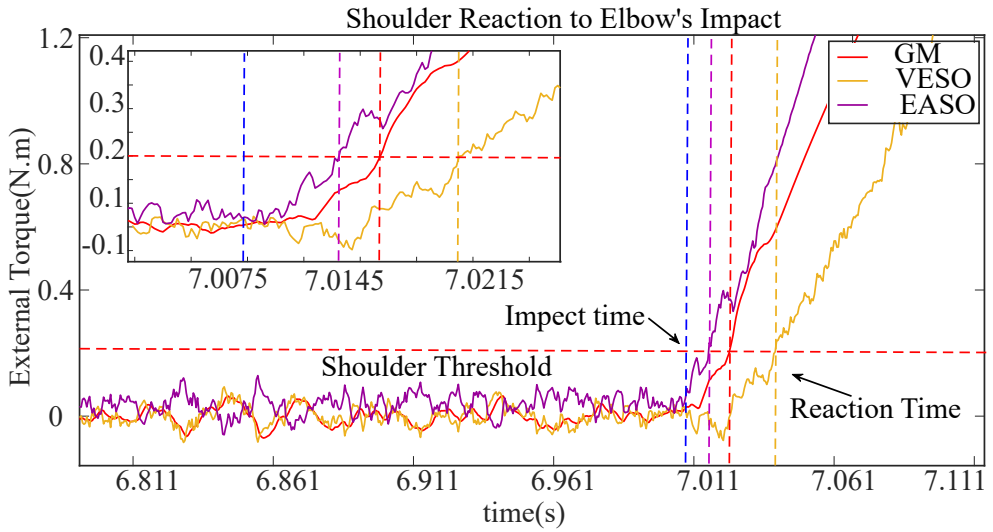


Figure 2.6: Reaction of shoulder in elbow's impact

In conclusion, even considering a collision on the elbow's link, ESAO was the fastest impact detection method, which detected it in almost 5.2 ms, while others, like GM and VESO, identified the same collision with a delay of 1.7 ms and 5.1ms on average compared to ESAO results.

In the next chapter, the application of impact detection methodology has been explained, and a new and simple method has been introduced to be able to use this method in grasping and picking up the object. This approach enhances the system's ability to detect impact and plan the next movement phase, ensuring more reliable and precise manipulation during these operations.

Chapter 3

Control of Impact-aware Robot for Grasping and Picking up Objects

3.1 State of the Art

One of the most important applications of impact detection methodologies in collaborative robots, particularly dual-arm robots, is in object grasping and picking tasks. Collaborative robots, such as dual-arm robots, have been used in industry for decades. The primary challenge in this application is ensuring that collisions are accurately detected without generating false commands. Additionally, it is crucial to distinguish between the first detected impact and subsequent ones. In a typical grasping task, one arm may initially detect contact with the object, and only after both arms have detected the impact can the grasp be completed. Any impacts detected after the grasp is fully established should not be factored into the control planning for the collaborative robots (cobots). While impacts may still be observed in external torque estimations after the grasp is completed, only the first detection should influence control decisions. Proper control planning and path-following are key aspects covered in this section.

Various control approaches have been developed to manage such impacts, contributing significantly to the improvement of reliability and precision in collaborative robot (cobot) operations. However, certain aspects of the existing literature still require further development. One key limitation is that many of the approaches assume the robot's velocity is zero during the moment of grasping. While this assumption simplifies control design, it is unrealistic for fast, dynamic grasping applications. In real-world scenarios, robots need to grasp objects while in motion, and relying on the zero-velocity assumption can lead to instability and unreliable performance. As a result, these methodologies often fail to ensure stable and reliable operation in more dynamic environments. Addressing this limitation is crucial for advancing the performance of cobots in practical, high-speed tasks.

Yang et al. [37] present a new framework to improve the robustness of feedback controllers in legged robots during impact events, where rapid and significant changes in velocity occur. The method involves projecting control objectives onto an "impact-invariant subspace," which remains unaffected by uncertainties related to the impact. This allows the robot to retain control over key dimensions while minimizing the adverse effects of timing discrepancies and unpredictable impacts.

Bombile and Billard [38] propose a unified motion generation algorithm for a bimanual robotic system, enabling two seven-degree-of-freedom robotic arms to swiftly grab and toss objects in motion. The system stabilizes the robots' grip on objects by controlling contact forces before and after impact. By using a sequence of time-invariant dynamical systems (DSs) and a state-dependent modulation function, continuous control of the reach, grab, and toss motion is achieved. the proposed MDS-based algorithm in enabling dynamic grabbing and tossing, resulting in shorter and more energy-efficient pick-and-place tasks.

In [39], the tracking control of fully actuated Lagrangian systems with frictionless unilateral constraints, focusing on both free-motion and constraint-motion phases has been addressed. A switching controller is designed to ensure approximate tracking, with particular attention to transitions between motion phases, such as impacts and detachment.

One of the recent interests of impact-aware control approaches is reference spread-

ing [40]. The primary goal of this method is to extend the velocity trajectories before and after impact for a specified duration around the moment of impact, and to transition to the post-impact state once the impact is detected. This approach helps prevent velocity error spikes caused by discrepancies between the planned and actual impact times [40].

Beumer in [41] addresses the challenge of impact-aware manipulation in robots, where unexpected impacts or timing discrepancies can cause systems to enter unplanned modes. The study explores extending reference spreading to task-based quadratic programming (QP) control, a common approach for complex robots like humanoids and dual-arm manipulators. By defining ante- and post-impact reference trajectories compatible with impact dynamics, the proposed method extends reference trajectories beyond expected impact times to handle perturbations. A procedure is introduced for generating these extended reference trajectories for specific QP tasks in a dual-arm box-lifting application, with a QP control framework based on reference spreading to track these trajectories during perturbations.

The reference spreading method is based on defining pre- and post-impact trajectories. These trajectories should be defined by the user and must account for the velocity jump at the time of impact. The accuracy of the velocity definition in the post-impact phase is crucial for effective planning, as it helps avoid significant errors in the controller's performance. The velocity after impact should be defined using an impact map, which can be determined through testing or based on the robot's kinematics.

In this work, reference spreading in grasping object has been applied to address the previously discussed challenges. However, impact detection approaches in grasping applications present certain problems, making it difficult to detect impacts accurately or leading to the generation of unwanted impact signals. Two new approaches have been introduced to address these issues. These approaches enhance the control strategy and enable accurate trajectory planning, focusing on two key phases: pre-impact and post-impact.

When both arms grasp the same object, their dynamics become inherently coupled, introducing interdependencies in force distribution, movement coordination,

and impact response. To maintain stability and accuracy in the grasping and manipulation process, the control strategy must ensure synchronization between the two arms.

Since any force applied by one arm directly affects the dynamics of the other, a compliant control strategy is essential to regulate interaction forces, preventing excessive stress on the object and ensuring smooth motion. In this work, adding the estimated external torque has the same effect as applying an additional force to the arms, allowing for a more adaptive and stable grip.

From a control perspective, a decentralized control approach enables each arm to operate based on its own force and motion feedback, reducing complexity. However, robust communication is required to prevent inconsistencies between the arms. Alternatively, the Extended State Acceleration Observer (ESAO) proposed in this work can be adapted to monitor interaction forces, ensuring that impacts are detected as a combined effect rather than in isolation. This reduces false positives and facilitates smoother transitions between grasping and manipulation phases.

Regarding trajectory planning, the transition from free movement to grasping must account for velocity synchronization to avoid disturbances upon contact. The Reference Spreading Approach (RSA) applied in this work ensures smooth velocity adaptation during impact, but incorporating an additional synchronization mechanism could further minimize discrepancies between the two arms, enhancing precision in dual-arm manipulation.

3.2 Simulation Analysis of Impact-aware Robots in Picking up Application

Impact-aware robots, such as dual arms, have been used for collaborative applications such as picking up objects. In this application, defining a proper plan is a crucial first step. Planning a pre- and post-impact in trajectory defining and also in controlling strategy depends on the impact, which should be detected as fast as possible and be a reliable signal. A reliable signal of external torque means that the first impact in grasping an object should be considered as the point of changing the trajectory and

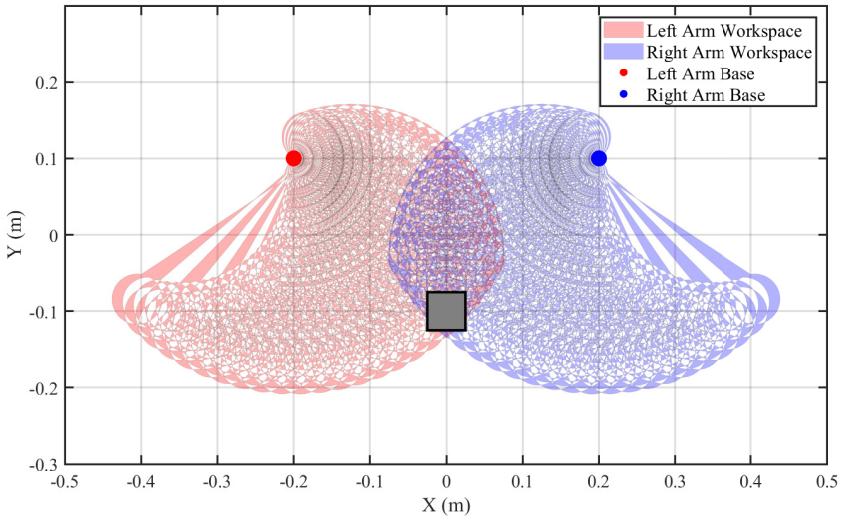


Figure 3.1: Workspace of dual arm

the plan for waiting for the second arm to grasp the object.

To evaluate the application of impact detection introduced in the previous chapter on a dual-arm setup, a simulation analysis was conducted using a dual conventional 3-DOF robot. The tested simulation is visible in Figure 3.2. Multiple tests were performed to assess external torque signals. The workspace of the dual-arm setup, shown in Figure 3.1, was used to define the trajectory, ensuring that the trajectory remains within the workspace.

However, by evaluating the external signal on this application, the following problems have been detected;

- Weak external torque signal at the end effector
- Redundant impact detection has been detected during grasping.

The weak external torque signal observed at the end effector during an object's grasping motion arises primarily due to the low velocity of the end effector. This reduction in velocity is inherent to the grasping process and is a necessary aspect

of achieving a secure hold on the object. Consequently, the external torque signal is close to the predefined thresholds established for non-impact phases, making it challenging to detect impacts using the end-effector signal alone.

During the grasping and lifting phases, the end effector experiences minimal movement, further limiting the strength and reliability of external torque or collision signals. Since all the methods of external torques depend on the dynamic movement of the robot, this problem is undeniable in all methods. This lack of significant signal change highlights a fundamental limitation of relying solely on the end-effector signal for impact detection. Such weak signals can be unreliable, leading to a higher likelihood of missed impact events.

To address this issue, developing and implementing a more robust impact detection method capable of overcoming the challenges posed by low-velocity conditions is crucial.

As from simulation results in fig. 3.4a, one can see the external torque signal from the end-effector, is not a reliable choice for impact detection because it remains too close to the threshold ($0.2N.m$), making it less distinguishable and more prone to noise or uncertainty in identifying an impact. In contrast, the signals from the shoulder and elbow joints exhibit significantly higher amplitudes, demonstrating stronger and more pronounced responses to the impact. These stronger signals indicate greater movement and energy transfer at these joints, making them more robust and reliable for detecting the impact with a clear margin above the threshold. This highlights the importance of selecting signals with higher amplitudes for accurate and consistent impact detection.

Table 3.1 presents the average of multiple the simulation results comparing velocity, required thresholds, and the success rate of impact detection, providing a clear analysis of the current method's limitations and the necessity for improvement. For better analysis, objects with different weights were used, and the threshold was set to the same value, 0.2 N.m, to estimate the detection rate under both conditions. These data underscore the need for alternative approaches that can function effectively even when external torque signals are weak or ambiguous. The results indicate an average success rate of 40% to 60% for impact detection based solely on the end-effector's

Table 3.1: Table of comparison results in a weak end effector signal.

Objects	Weight(kg)	V_{ee}(m/s)	Threshold	Rate of impact detection %
Light object	0.2	2	0.2 N.m	40
Heavy object	2	3	0.2 N.m	60

external torque signal, which is a very low rate of success.

The issue of redundant impact detection arises when the two arms involved in grasping are not well-synchronized ($\Delta(t)_i$ time interval between two arms impact detection), or they are in grasping and moving in the post-impact phase. In such cases, the first arm detects an impact during the initial collision, but a short time later, when the second arm collides with the object, the first arm registers a second impact. This lack of coordination between the arms leads to repeated detections of what is essentially the same event.

Additionally, during the grasping and lifting process, both arms remain in constant contact with the object. As a result, external torque variations occurring during this time may generate impulses that exceed the threshold limit, causing them to be misclassified as collisions. These misclassifications further contribute to redundant impact detections, reducing the precision of the overall detection process in planning the trajectory and controller commands.

To better understand and address this problem, Table 3.2 presents data on the number of redundant impact detections and the time intervals between arms. For comparison, two different lengths of objects with the same weight of 0.2 kg were considered. The average impact detection signals show that although small time differences in grasping between both arms were acknowledged, in both cases, three redundant impacts were detected. This analysis highlights the need for improved detection algorithms that can distinguish between genuine collisions and torque-induced impulses. By refining the detection system, it would be possible to minimize false positives and ensure that impact detection remains both accurate and reliable.

Table 3.2: Table of comparison results in redundant detection.

Object length(mm)	$\Delta(t)_i$ impact detection(ms)	Number of impact detected
50	4	3
60	3	3

3.2.1 Proposed Methodology

As mentioned, to detect the impact on the end-effector, relying solely on the external torque as a signal is insufficient, as the lack of significant movement at the moment of impact results in a weak torque signal. Additionally, the estimated collision torque is not a reliable indicator in noisy environments, because the torque threshold is often close to the noise amplitude. This makes it difficult to differentiate between actual impacts and noise, leading to potential inaccuracies in detecting the impact. Therefore, alternative or supplementary methods must be employed to accurately detect the impact in such scenarios.

As discussed in the previous chapter, the collision on the i^{th} link will be reflected in the external torque readings of the preceding links. Since joints closer to the base, such as the shoulder (L_1) and elbow (L_2), experience greater movement during impact, they generate stronger external torque signals compared to the end-effector. As a result, the external torque signals from these joints are more reliable indicators of impact events.

To improve detection, the combined effect of the external torques from all three joints—shoulder, elbow, and end-effector—can be considered. By calculating the norm of the external torques, $\|r\| = \sqrt{r_1^2 + r_2^2 + r_3^2}$, where r_1, r_2, and r_3 represent the torque at the shoulder, elbow, and end-effector, respectively, the overall impact can be assessed. Setting an appropriate threshold for this norm allows for more accurate detection of impact across all joints, ensuring that even subtle impacts at the end-effector are captured while minimizing the influence of noise. This method offers a more robust solution for detecting impacts during manipulation tasks, especially in cases where individual joint signals may be weak or unreliable.

Simulation results show the percentage of the possibility of recognizing the col-

lision is increased with the proposed method, the evaluated signal can be seen in fig. 3.4b.

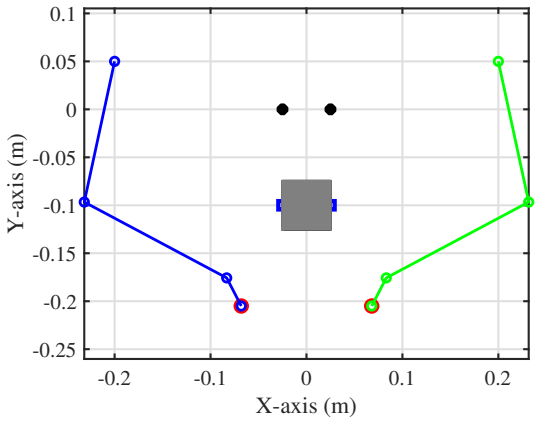
As mentioned the second problem is redundant impact detection. It is essential to make sure the impact is the first impact that happened in the grasping, so in this way, we can avoid any false commands of the controller and change of the trajectory definition. This means while both arms are in grasp or only one has been detected for the first time and the second arm is reaching the object by some small delays and grasping happens in both arms the impact will not be considered for the next time later.

If, within a pre-defined time window from the estimated impact time (Δt_{bound}), the external torque measured is higher than the threshold and, as previously explained, the estimated external torque at the actual moment of impact, then the impact is classified as the first impact for grasp detection. Relying only on threshold in noisy and uncertain environment of the tests cannot be a proper option for detecting impact.

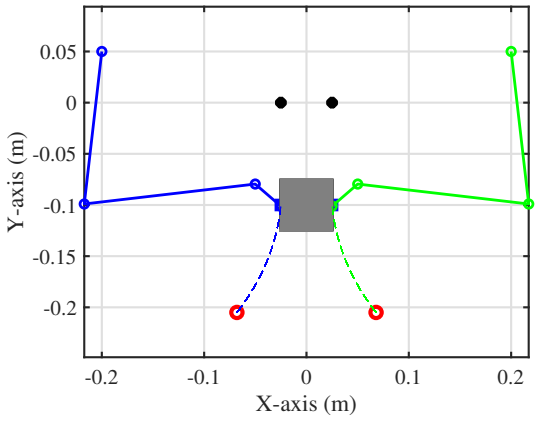
The reasoning behind this approach is that the threshold ensures that no significant external torque occurred in the immediate past. This means that the sudden spike in the torque signal is the first detected impact, not the result of any earlier disturbances or minor contacts. Thus, the torque signal's impulse is treated as the initial and valid impact event.

However, a crucial aspect of this detection process lies in selecting an appropriate time boundary (Δt_{bound}). The choice of this boundary is critical because it defines the time window within which the system expects to observe the first impact. If the boundary is too small, the system might miss the actual impact event, leading to unreliable detection. Conversely, if the boundary is too large, the system may mistakenly treat prior noise or minor disturbances as valid impacts, potentially causing misalignment in the manipulation task.

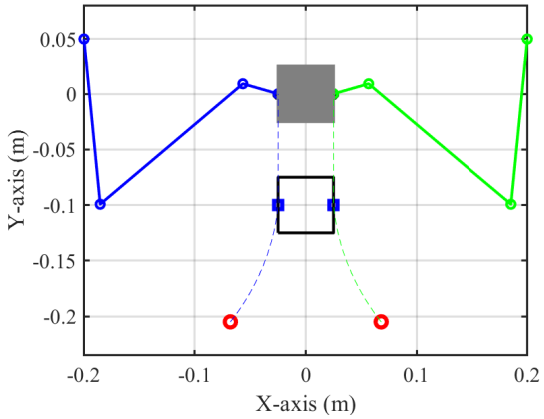
Choosing the time window (Δt_{bound}), should consider the dynamics of the robot and the specific task requirements. For instance, in high-speed tasks or when manipulating lightweight objects, the time window must be short enough to capture rapid changes in the torque signal but long enough to filter out minor oscillations or sensor noise. In contrast, for slower tasks or when dealing with heavier objects, a longer



(a)



(b)



(c)

Figure 3.2: Dual arm in picking up a box in simulation: (a) Initial position, (b) Grasping position, (c) Picking up position

window may be more appropriate to account for the more gradual buildup of torque. The equation below defines the relationship between the torque signal, the threshold, and the time window (Δt_{bound}), providing a formal description of this grasp detection strategy.

$$\|\tau_{i,est}(t-\Delta t_{bound})\| > b_{th} \tag{3.1}$$

In which i is the number of the robot joints, $i \in 1,2,3$.

The simulation results show that the proposed methodology increases the chance of recognizing the first impact correctly in the grasping point to 90%.

In Figure 3.2, steps of pre-, post-, and in-impact have been shown. The position starts from the initial point which shown with red circle and it continues to the edge of the box in the second figure and finally in the third figure picking up object has been completed. The dashed lines show the trajectory that robot follows and it is inside of the workspace as shown in Figure 3.1.

Furthermore, the angles regarding each joints in pre-impact and after impact shown in Figure 3.3. The box size in this simulation is considered to be $5cm$ with a weight of $0.15kgr$ and the grasping completed in 1sec.

The evaluated external torque in detecting impact has been shown in fig. 3.4b. With the introduced methods $\Delta t_{bound} = 4ms$ and threshold has been considered $0.5N.m$.

In fig. 3.4b, the first spike in the signal, which exceeds the threshold, is the optimal choice for impact detection. This is primarily because it is the most significant and prominent peak in terms of amplitude, reflecting the highest energy or power of the external torque. Choosing this spike ensures reliable detection of the impact event as it represents the most pronounced response of the system, minimizing the risk of false positives and accurately identifying the moment of impact. The subsequent spikes do not possess the same features as the first one, as they do not reflect the same energy or power within the introduced time frame. The evaluated external torque correctly detected the impact in both arms without any redundant impacts, increasing the likelihood of correct detection to 90%.

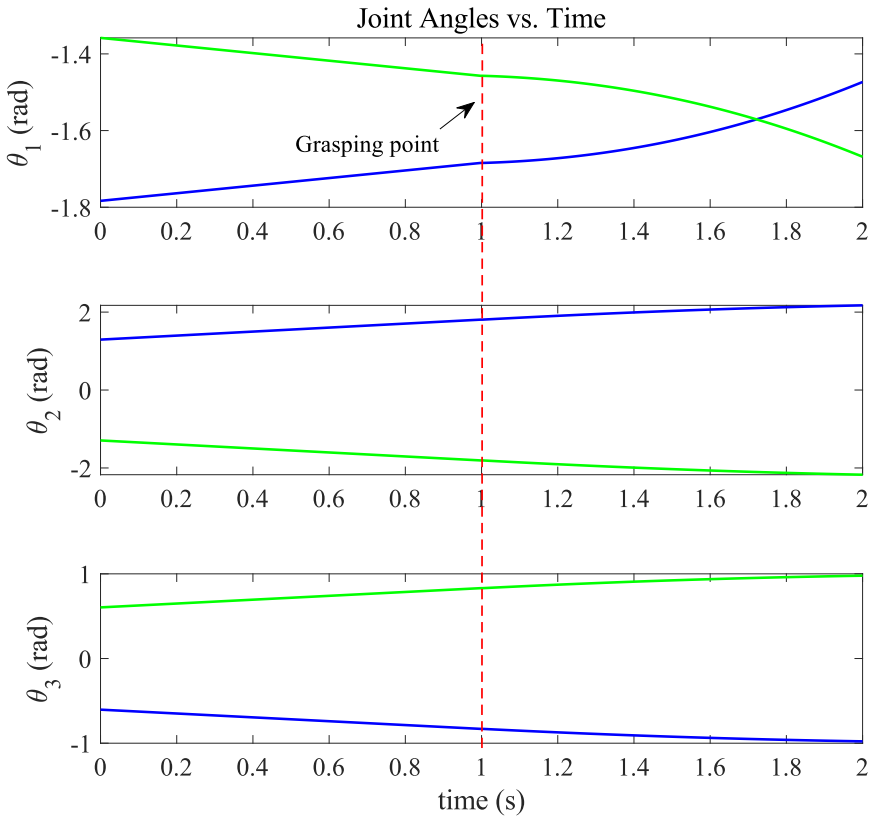
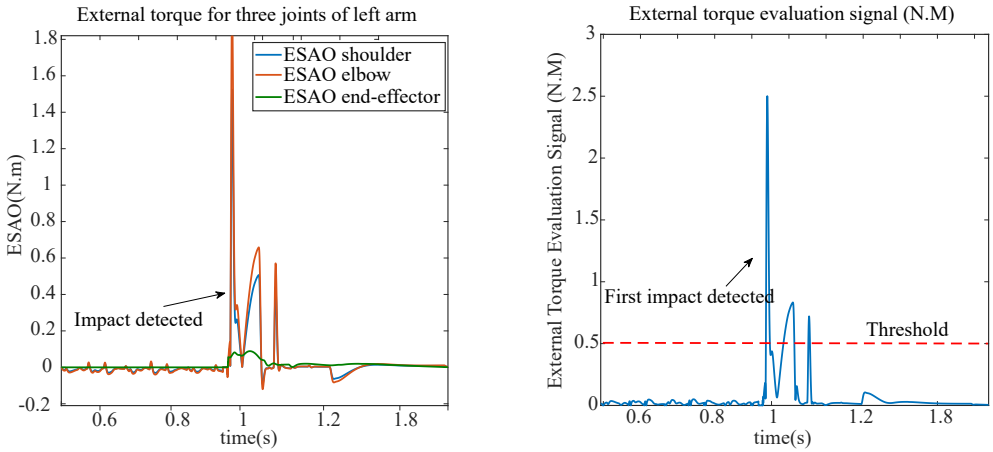


Figure 3.3: Angles of the joints for both arms during the procedure of picking up an object with dual arm collaborative robot in simulation; Right arm highlighted in green and left arm highlighted in blue.



(a) (b)

Figure 3.4: (a) External torque signals for three joints of the left arm, (b) Evaluated signal based on proposed methodology in simulation.

3.3 Experimental Results

In this section, experimental tests of the proposed algorithm are presented. The planning strategy was evaluated on a planar dual-arm setup in the robotics laboratory of Eindhoven University of Technology. The main purpose of the test was to grasp the object, toss it, and then re-grasp it. However, due to mechanical issues with the 3D-printed setup and limitations of the electrical boards on each joint, the test was restricted to grasping and picking up the object.

The dual-arm system consists of two 3-DoF planar robots, each powered by BLDC motors as joint actuators. Each BLDC motor is controlled by an inner Field-Oriented Control (FOC) loop. The only available sensors are current sensors and encoders. The system includes two planar arms with lead screws and a back drivability mechanism. For the test, the object has been chosen as a light square box with lenght 6cm with weight of 0.15 kg. Due to the unique mechanism of the setup, transformations from joint space to linear and then actuator space and vice versa are required. The explanations regarding the transformation are available in Appendix A.

Trajectory Definition of the Test

The pre- and post-impact trajectories were defined based on the position of the square object, which was placed in the center of the workspace between the two robot arms.

Also, regardless of the initial position of the arms, all trajectories of the tests began from the same starting point. This consistent initial position was set to ensure uniformity across all tests.

The trajectory in the workspace is defined based on the initial position of the arm and the object's position as the final target for the pre-impact phase. The initial position of the end effector is determined by zeroing the joints. Using the positions of the joints and kinematic equations, the end effector's position is accurately determined. To calculate the end of the trajectory, the object's center of mass and the displacement of the object's edge define the reaching position of the end effector. A more detailed description is provided in Figure 3.5.

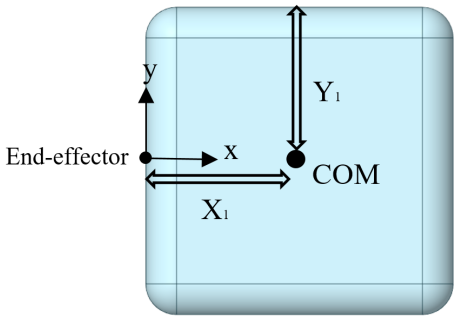


Figure 3.5: Position of end effector

The post-impact trajectory is planned to move the object from its initial pre-located position to a vertical orientation. In post-impact, only the vertical position will change and the horizontal position will remain the same as the pickup object. In this experiment, the object's position in grasping time remains fixed and does not change during the process. The resulting trajectory in the cartesian for the test is illustrated in Figure 3.6.

Through inverse kinematic and inverse transformation as described in Appendix.A,

the required joint positions are calculated.

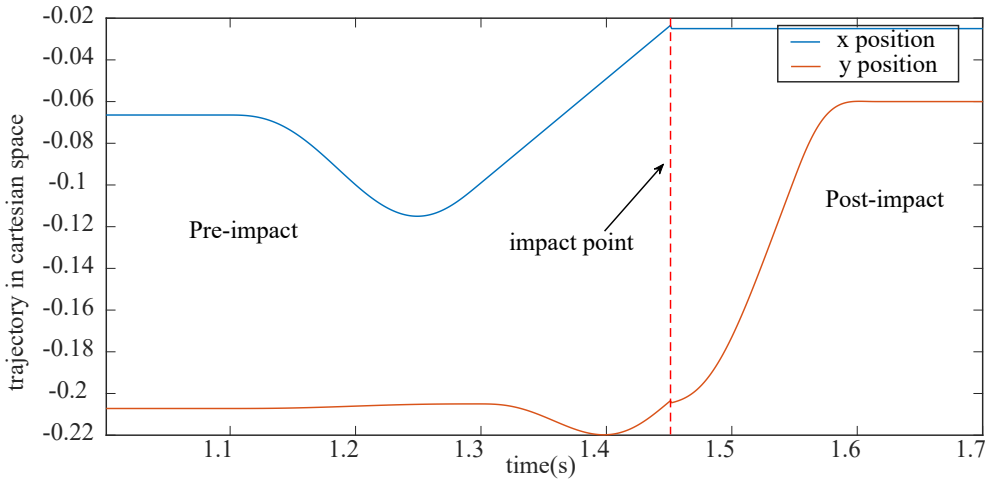


Figure 3.6: Trajectory in cartesian space in two phases with two velocity.

Impact Estimation of the Test

To define the impact detection parameters for the specified methodology, 20 sample tests were conducted. The threshold and time interval parameters, as given in equations (3.1), were determined based on the average external torque observed across both collision-free and collision movements. In this test, as can be seen from fig. 3.8a, end effector has a small amount of external torque compared to other three joints and according to the introduced methodology for avoiding the problem of weak external torque and also redundant impact the proposed method has been used as can be seen from fig. 3.8b. The selected parameters are $b_{th} = 1N.m$ with time interval $\Delta t_{bound} = 5ms$.

The choice of time interval is a critical factor, selected to ensure that the ESAO can detect impacts as quickly as possible. This minimizes the chance of misinterpreting slower, non-impact-related torque spikes that may occur over a longer time interval as actual collisions. Observed test data showed that non-impact torque spikes during complete grasp phase tend to have slower rise times and occur over longer

Table 3.3: Table of required parameters in evaluating impact detection.

Parameters	b_{th}	Δt_{bound}
Values	1N.m	5ms

Table 3.4: Table of ESAO parameters in DUAL ARM setup.

Parameters	Shoulder parameters		Elbow parameters		end-effector parameters	
ESAO	1250	56000	1200	50000	1050	25000

intervals. The required parameters for ESAO are listed in Table 3.3 and Table 3.4.

Control definition of the Test

The controller for the dual-arm setup utilizes an inner Field-Oriented Control (FOC) loop combined with feedback linearization. Since the BLDC motors at the robot's joints accept torque as a control signal, this approach enables the precise application of the required torque and current to the robot's actuators.

$$\boldsymbol{\tau} = \mathbf{M}(q)(\ddot{\mathbf{q}}_{des} + K_p(\mathbf{q} - \mathbf{q}_{des}) + K_d(\dot{\mathbf{q}} - \dot{\mathbf{q}}_{des})) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{3.2}$$

In which K_p and K_d represent the proportional and derivative gains for position and velocity control, respectively. These controller parameters are selected through a trial-and-error approach to achieve optimal tracking performance and system stability. For the dual-arm system, the chosen values are $K_P = 300$ and $K_d = 45$. The same parameters are applied to both arms because they share identical mechanical properties and kinematics, ensuring consistency in their control responses.

However, to reduce the velocity feedback controller's response, K_d is reduced to $K_d = 20$, when an impact is detected in one arm, while waiting for impact detection and grasp completion in the other arm. In a time of grasping, velocity jumps due to impact, and the error of the velocity increases, which makes the controller unstable; therefore, by decreasing the velocity feedback controller, we are able to decrease the error. After the grasp is completed, K_d is reset to its pre-impact value.

Furthermore, during the grasping phase, the reference spreading methodology, as described in [40], was employed to ensure a smooth trajectory reference. This

method extends the trajectory at the time of detecting an impact in one arm. Trajectories should be carefully defined and must account for the velocity jump at the moment of impact. The accuracy of the velocity definition in the post-impact phase is crucial for effective planning, as it helps to avoid significant errors in the controller's performance. The velocity after impact was defined using an impact map, which was determined through testing.

This approach helps maintain uniform behavior across both arms during operation, facilitating better coordination and stability in tasks requiring synchronized movement. Additionally, it helps prevent instability in the controller's response.

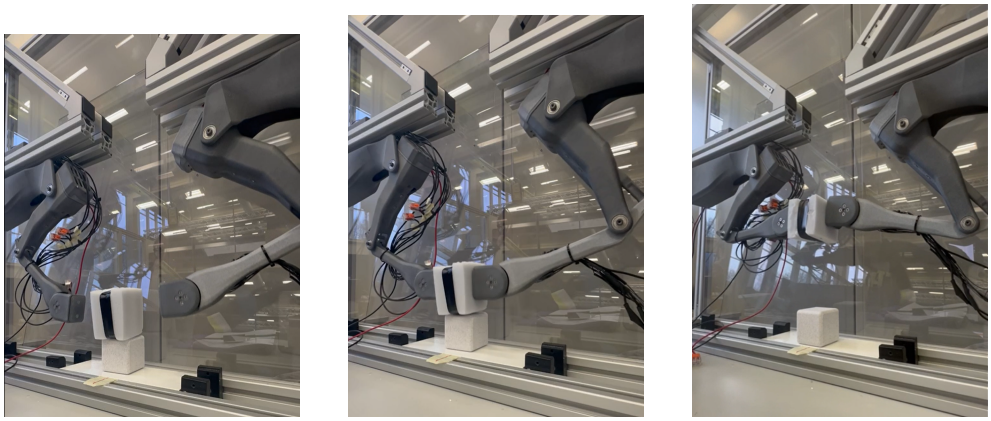
3.3.1 Test Explanation and Result Analysis

Multiple tests were conducted to evaluate the success of grasping the object. To ensure uniformity across all tests, the starting position was kept consistent. This was achieved by initializing the robot arm to move to a predefined starting position before beginning the trajectory.

The test starts with following the defined trajectory as described in Cartesian space. The given trajectory for each joint is followed to reach the desired position in grasping a predefined object. The object is a square 6cm cube with a weight of 150gr, to have a symmetric behavior in both arms.

When both arms detect the impact for the first time the algorithm stores it to avoid any false impact detection for the future estimated external torque signal during the pick-up path. The planning path shows a new pre-defined velocity which has been chosen from 20 tests on the real setup, in this way velocity error in the feedback controller will not spike in the impact point. This helps the controller to not become unstable and keep a stable grasp. When this phase is completed, it shows that both arms are in a complete grasp. The next phase, post-impact will be continued to pick up the object in a vertical position.

A successful test of the methodology has been shown in Figure 3.7. This figure is three screenshots of different phases, which include pre-impact (a) in following the path to reach the object. The second photo (b) shows the grasping phase and the third one (c) shows the picking-up phase.



(a) (b) (c)

Figure 3.7: Overview of picking up object with dual arms collaborative robot: (a) Pre-impact, (b) Grasping, (c) Picking-up.

The results of the tests for both arms are presented in Figure 3.9. Since both arms are symmetrical in terms of trajectory, mechanics, control, and kinematics, the outcomes for both arms are expected to be symmetric; however, due to some uncertainties in the test, there are still differences in following the reference angle and also in a position of each arm which resulted in a time difference of detecting impact.

In fig. 3.8a, three estimated external torques related to three joints and their norm in fig. 3.8b have been shown. As discussed previously, the external torque at the end-effector is a weak signal, making it difficult to distinguish between impact and non-impact points. However, the detected impact is stronger at the other two joints. Therefore, by using the proposed method the norm of external torques will be evaluated instead. This allows for reliable detection of the impact across various movements and grasping trajectories. The evaluated signal in fig. 3.8b highlights the first impact, as well as the threshold used for detection.

The results show that after the object is grasped, the external torque still displays some spikes. However, these do not share the same characteristics as the first impact during the grasping process. The first impact occurs when the end-effectors initially collide with the object, causing the external torque to register a fast, sharp impulse.

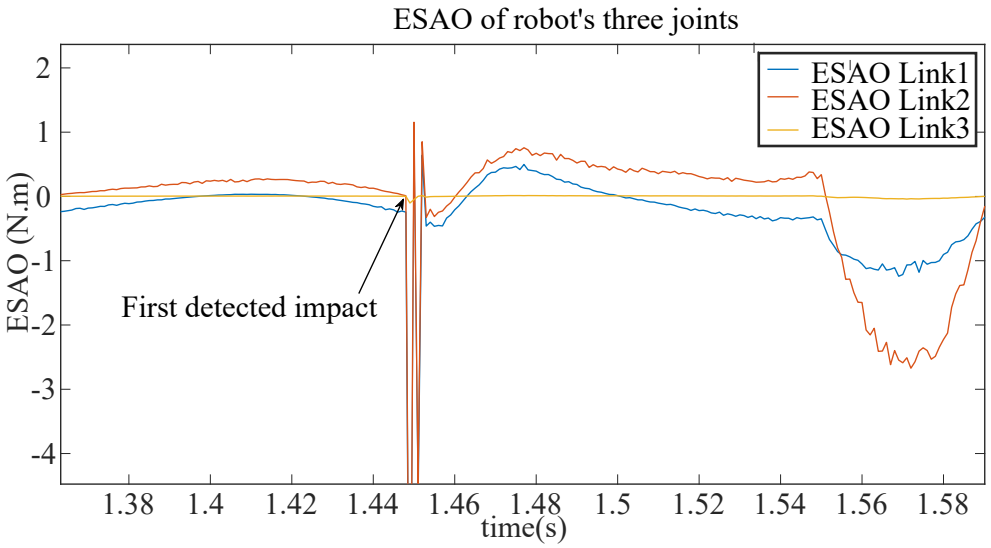
Once the grasp is complete and the system enters the post-impact phase, subsequent impacts produce smaller, slower signals, making distinguishing the initial impact easier and avoiding false detection. The interval timing ($\Delta t_{bound} = 5ms$) used in the detection process is crucial, as it helps differentiate the fast-changing signal of the first impact from other detectable signals during the grasping phase.

The position in Figure 3.9 shows the trajectory following both arms. In joint1 and joint2, 0.05rad difference between both arms can be seen, which means the right arm has reached the grasping point with a delay compared to the left arm. The time of detecting impact has been measured with 2ms difference. However, the average error in both arms is almost the same as specified.

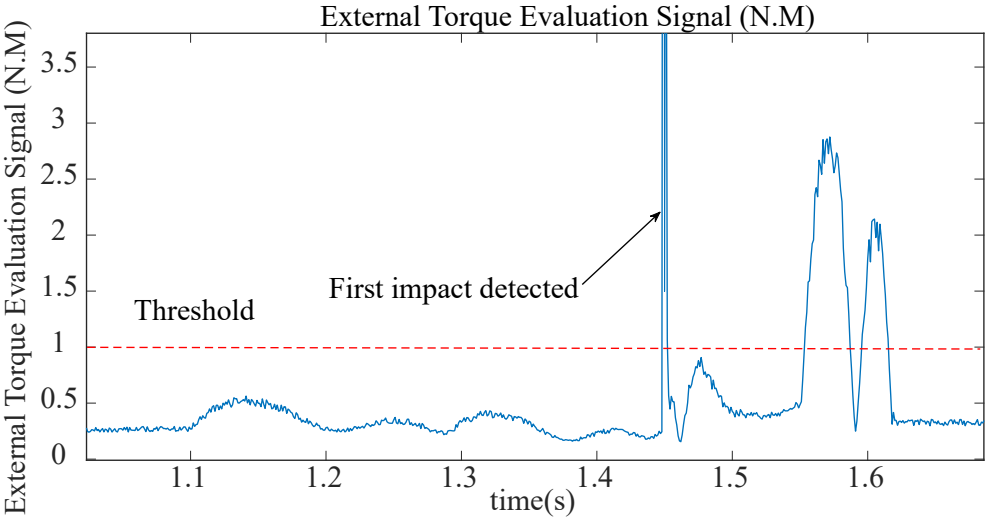
The controller shows a strong performance in tracking the trajectory during both the pre- and post-impact phases. The maximum average error in both phases occurs at the end effector, measuring 0.0052 for both arms. However, the maximum error, 0.0355, is observed in the post-impact phase after the impact is detected at the end-effector. This increase is due to the controller's difficulty in quickly generating the necessary torque to follow the trajectory immediately after the impact, leading to a higher error compared to the pre-impact phase. Despite this, the controller responds well and effectively follows the required trajectory and task.

As seen in the trajectory tracking results, the pre-impact joint space trajectory successfully guides the arm to reach the object, with grasping completed at 1.45 seconds. At this point, the transition from pre-impact to post-impact occurs. The control error at this transition point does not show a significant deviation between the two phases, indicating a successful control performance.

Based on 20 tests conducted for this experiment, the success rate of grasping was 80%, demonstrating a reliable grasping approach. The unsuccessful cases observed were due to a lack of synchronization between the two arms, which is an area that will require further improvement in future work.



(a)



(b)

Figure 3.8: (a) ESAO external torques, (b) Evaluated signal based on proposed method.

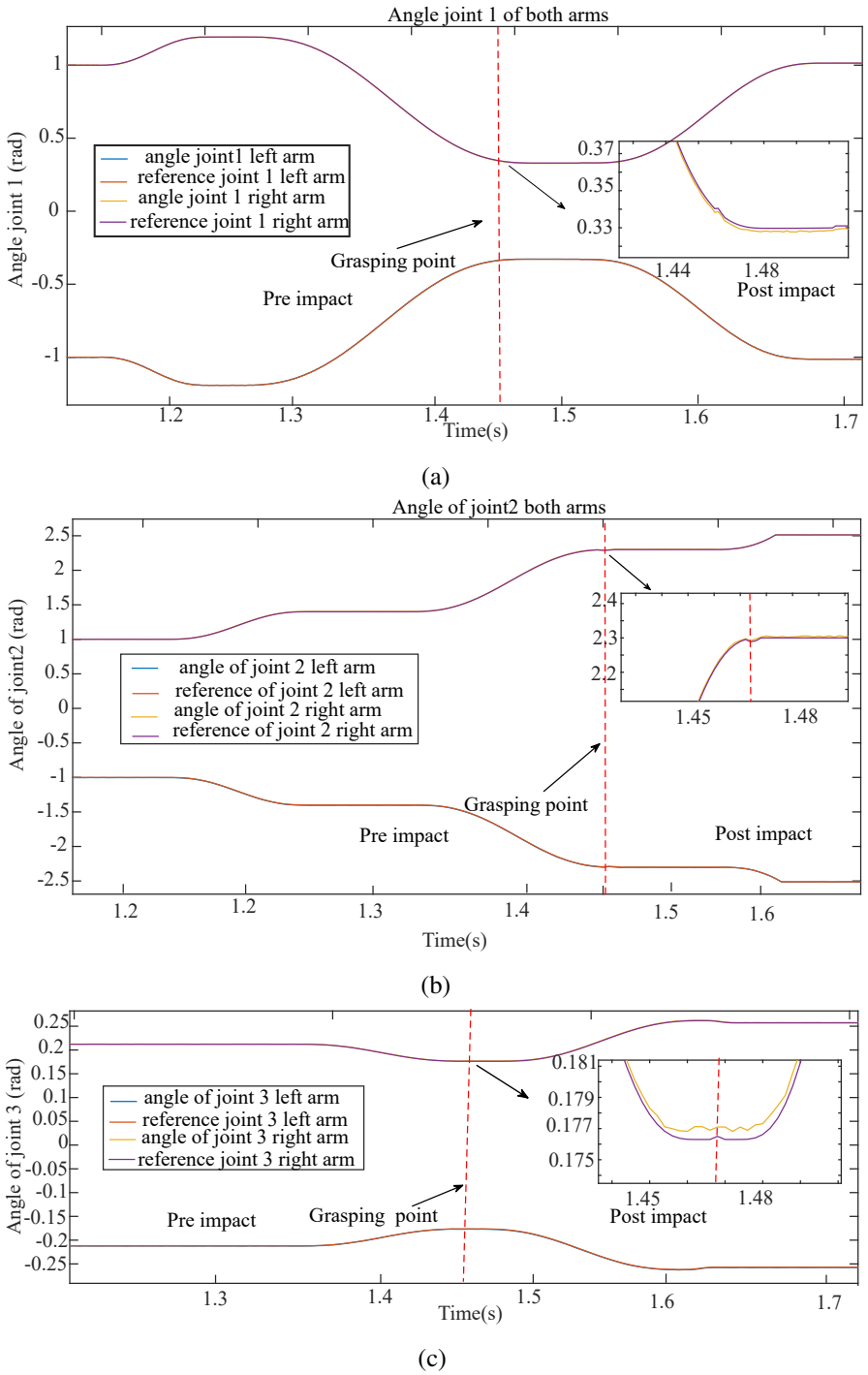


Figure 3.9: (a) Angle of shoulder joint, (b) Angle of elbow joint, (c) Angle of end-effector joint.

Conclusion and Future work

This thesis investigates methodologies for enhancing the control and functionality of impact-aware collaborative robots (cobots). By addressing sensor limitations, proposing advanced estimation techniques, and implementing robust control strategies, the work advances the field of collaborative robotics, particularly in dynamic environments requiring precision and safety.

The primary contributions of this thesis include:

- Hybrid Super-Twisting Differentiator (HSTD): A novel method for estimating velocities and accelerations with reduced noise and phase lag, critical for real-time impact detection and control in sensorless robotic systems. Experimental validations confirm its superiority over traditional techniques like the standard Super-Twisting Differentiator and Filtered Finite Difference Derivatives.

- Extended State Acceleration Observer (ESAO): A fast and robust impact detection method was introduced, offering faster identification compared to existing methods in the literature. This method is based on the Extended State Acceleration Observer (ESAO), which converges in finite time. ESAO extends the system's state to estimate external torques without requiring additional computations, enabling real-time responses. Additionally, the high gains in the ESO observer ensure robust performance against measurement noise and model uncertainties. The method was validated on a two-joint planar robot in simulation, showing that ESAO is, on average, the fastest among the most promising methods in the literature. It is capable of detecting a collision approximately

1.7 ms in advance and reliably identifying impacts with reduced delay times in unexpected collisions compared to other methods.

- Dual-Arm Collaborative Robotic Application: The practical implementation of the proposed methodologies in a dual-arm setup showcased their efficacy in grasp-and-pick tasks. Challenges, such as weak end-effector signals and redundant impact detections, were addressed through innovative solutions like combining joint torque norms and refining detection thresholds.

Overall, the proposed methods significantly improve the precision and reliability of impact detection and response, enabling cobots to operate effectively in environments with human-robot interaction.

Future Work

While this research addresses critical aspects of impact-aware robotics, several approaches for further investigation remain:

- Enhanced Impact Detection Algorithms: Incorporate machine learning models trained on diverse impact scenarios to further improve detection accuracy and robustness, particularly in noisy industrial environments. Investigate hybrid detection strategies combining sensorless methods with low-cost external sensors for enhanced reliability.

- Real-Time Optimization: Develop adaptive controllers that dynamically adjust parameters such as feedback gains based on real-time impact characteristics, further improving system stability during high-speed operations.

- Multi-Robot Coordination: Extend the impact detection and control methods to multi-robot setups, focusing on synchronized actions in shared workspaces to enhance efficiency and safety.

- Broader Applications: Test the methodologies in complex, unstructured environments, such as healthcare or disaster response, where cobots must handle

unpredictable scenarios. Expand the applications to include tasks such as collaborative assembly, welding, and large-scale logistics.

- Hardware and Computational Advancements: Investigate the integration of lightweight, cost-effective sensing technologies that complement the proposed sensorless methods. Optimize computational efficiency to ensure real-time performance on embedded robotic systems.

By addressing these challenges, future work can further refine the capabilities of collaborative robots, solidifying their role in modern industry and beyond. The methodologies introduced in this thesis lay a solid foundation for continued advancements in the field of impact-aware robotics.

Appendix A

Equations for Dual-Arm Robot

A.1 Model of Planar Dual Arm Setup

The setup of the planar dual-arm mechanism at the Eindhoven university of Technology is shown in Figure A.1. As illustrated, the angles of both Link 1 and Link 2 depend on the motion of the leadscrews, while Link 3, representing the end effector, is directly connected to the motor shaft. The linear movement of the leadscrews is directly proportional to the motor rotation, which is determined by the screw pitch [42].

The coordination of each arm's state within the setup, as shown in Figure A.1, can be fully described using Denavit-Hartenberg (DH) parameters, represented as $q_{DH} = \theta = [\theta_1, \theta_2, \theta_3]^T$, measured in radians.

The practical implementation of this setup, however, lacks sensors to directly measure all the required angles. Instead, the rotations of the leadscrews in links 1 and 2 are monitored. To accurately represent the state of each arm, a new set of generalized coordinates is introduced. These coordinates are defined as $q_G = [x_1, x_2, \theta_3]$, where x_1 and x_2 denote the displacement of the leadscrews, and θ_3 represents the angle of link 3, which can be obtained from the encoder of motor three. This formulation allows for the conversion of movements from the encoder to leadscrew displacements:

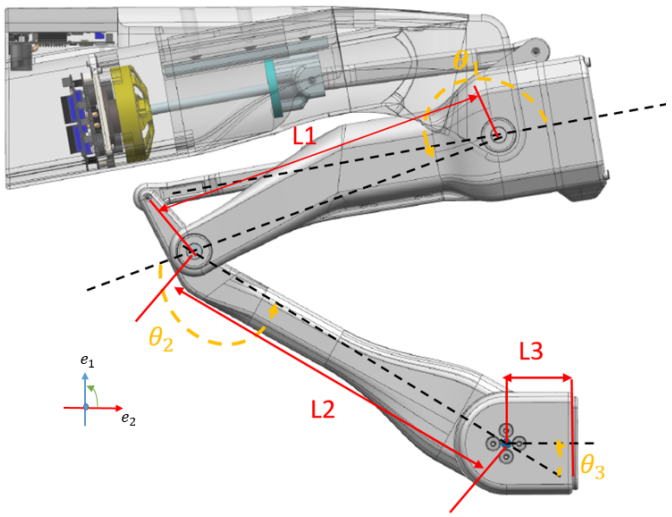


Figure A.1: Visual representation of coordinates which are needed for dynamic model and also have been used in Denavit-Hartenberg

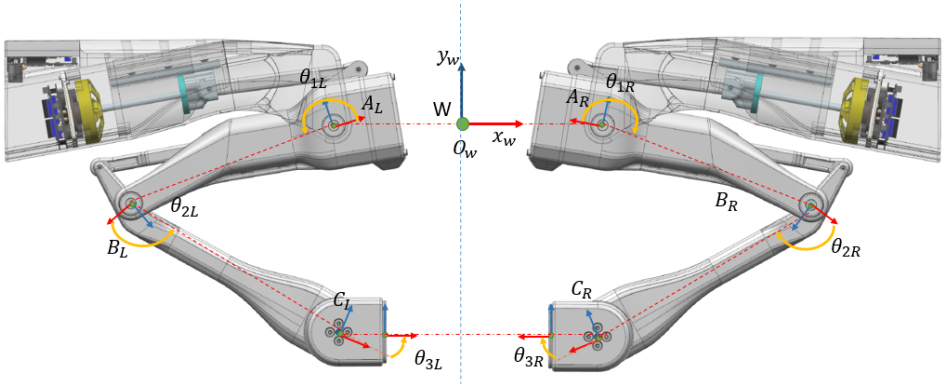


Figure A.2: World W and local frames defining the robotic setup

$$\textbf{linear displacement} = x = \textbf{screw pitch} \times \textbf{encoder angle}, \tag{A.1}$$

The screw pitch required for this project can be obtained from the manufacturer's

datasheet of the leadscrew. In this instance, the manufacturer specifies a screw pitch of 16.25mm. Additionally, to determine the angles of link 1 and link 2 in the robot model, the following geometric calculations based on the Figure A.3 need to be performed:

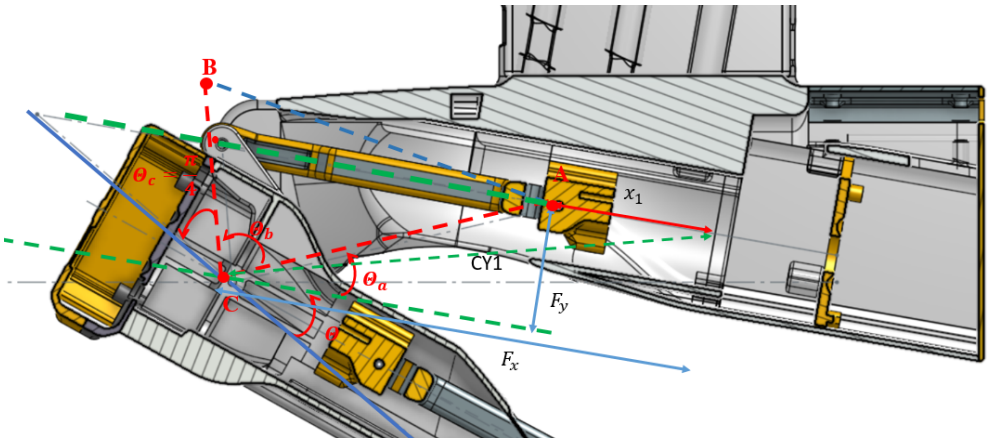


Figure A.3: Transformation from q_G to q_{DH}

$$\theta = \pi - (\theta_a + \theta_b + \theta_c) \tag{A.2}$$

, In which

$$\theta_c = \frac{\pi}{4} \tag{A.3}$$

$$\theta_a = atan\left(\frac{F_y}{F_x - x}\right) \tag{A.4}$$

$$\theta_b = acos\left(\frac{|BC|^2 + |AC|^2 - |AB|^2}{2\,|BC|\,|AC|}\right) \tag{A.5}$$

The parameters of the designed planner arm, have been described in the Table A.1.

The distance $|AB|$, is equal to the length of the spanner, distance $|BC|$ is the distance between the connection point of the spanner on link1 and its hinge point. The variables F_x and F_y are the horizontal and vertical distance from the actuator face to the center of the hinge-point. As indicated in Table A.1, the parameters of AB, BC,

Table A.1: Required parameters for the conversion

Parameter	Description	Value (mm)
AB	Spanner length	104
AC	Spanner to hinge	Variable
BC	Lever length	42
F_x	x act to hinge	152.1
F_y	y act to hinge	40
x	Displacement of leadscrew	Variable
Screw pitch	-	16.25

F_y, and F_x are constants that have been previously computed from the design model. The parameters of AC and x are variables dependent on the actuator or leadscrew movement, since

$$|AC| = \sqrt{\left((F_x - x)^2 + F_y^2\right)}, \tag{A.6}$$

Thus, by employing the aforementioned methodology, we can ultimately convert the computed encoder angle into linear displacement and subsequently into q_{DH}. These q_{DH} parameters will be utilized in computing inertia, Coriolis and centrifugal, gravitational, friction torque, and external torque matrices. The DH angle θ_1 that has been shown in Figure A.1, will be $2\pi - \theta$ that has been calculated previously. The complete model of the planar dual arm is outlined as follows:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) + \tau_f - \tau_{ext} = \tau_j \tag{A.7}$$

In which $M(\theta) \in R^{3\times3}$, $C(\theta, \dot{\theta}) \in R^{3\times3}$, $G(\theta) \in R^{3\times1}$, $\tau_f \in R^{3\times1}$ and $\tau_{ext} \in R^{3\times1}$.

A.2 Inverse Kinematic

To be able to grasp an object the joint positions need to be calculated based on the end effector position in Cartesian space. For this purpose calculating the inverse kine-

matic is inevitable. Inverse kinematics will be calculated based on the schematic that has been provided in Figure A.3 and also by considering the law of cosine.

$$q_1 = 2\pi - (\beta + \gamma + 0.5\pi) \tag{A.8}$$

$$\beta = cos^{-1}(\frac{L_1^2 - L_2^2 + r^2}{2L_1 r}) \tag{A.9}$$

$$r = \sqrt{x^2 + y^2} \tag{A.10}$$

in which x and y are:

$$x = dx - (leg)x - L_3 cos(\theta) \tag{A.11}$$

$$y = y + sin(\theta)L_3 \tag{A.12}$$

in which the variable leg is equal to -1 for the left arm and 1 for the right arm and dx is the distance from the world frame to the first joint of the robot's arm, where movement begins.

$$\alpha = cos^{-1}\left(\frac{L_1^2 + L_2^2 - r^2}{2L_1 L_2}\right) \tag{A.13}$$

$$q_2 = \pi - \alpha \tag{A.14}$$

$$q_1 = -\beta - tan^{-1}(\frac{x}{y}) - 0.5\pi \tag{A.15}$$

$$q_3 = \theta_{ee} - q_1 - q_2 \tag{A.16}$$

In which θ_{ee} is the orientation of the end effector.

A.3 Inverse Transformation

To transfer back the angles of the joints to linear movement and then to motor space in two shoulder and elbow joints, the following need to be considered:

$$\theta_{1ax} = \pi - (-\theta_1 - \frac{\pi}{4}) \tag{A.17}$$

$$x_{ax} = sin(\theta_{1ax})\,|BC| \tag{A.18}$$

$$Y_{ax} = cos(\theta_{1ax})\,|BC| \tag{A.19}$$

$$x_1 = F_x - \sqrt{|AB|^2 - (x_{ax} - Fy)^2} - Y_{ax} \tag{A.20}$$

$$\theta_{2ax} = \theta_2 - |BC| + sin^{-1}\left(\frac{CY_2/L_1}{\pi}\right) \tag{A.21}$$

$$x_{2ax} = sin(\theta_{2ax})\,|BC_2| \tag{A.22}$$

$$Y_{2ax} = cos(\theta_{2ax})\,|BC_2| \tag{A.23}$$

$$x_2 = \sqrt{L_1^2 - CY_2^2} - \sqrt{Fx^2 - (CY_2 - x_{2ax})^2} - Y_{2ax} \tag{A.24}$$

In which $x_1$andx_2 are linear movements of the shoulder and elbow respectively. In these equations the definition of and CY_2 and CY_1 can be seen as has been shown in Figure A.3. In order to make the linear movement to motor rotation angle the followings should be consider:

$$x_{L1} = \frac{x_1 - x_{1,0}}{P} \tag{A.25}$$

$$x_{L2} = \frac{x_2 - x_{2,0}}{P} \tag{A.26}$$

In which $x_{1,0}, x_{2.0}$ are offset of the encoder in the initial position.

Bibliography

[1] Alessandro Farinelli, Elena Zanotto, Enrico Pagello, et al. Advanced approaches for multi-robot coordination in logistic scenarios. *Robotics and Autonomous Systems*, 90:34–44, 2017.

[2] Tomás Lozano-Pérez, Joseph L. Jones, Emmanuel Mazer, and Patrick A. O'Donnell. Task-level planning of pick-and-place robot motions. *Computer*, 22(3):21–29, 1989.

[3] Danica Kragic, Joakim Gustafson, Hakan Karaoguz, Patric Jensfelt, and Robert Krug. Interactive, collaborative robots: Challenges and opportunities. In *IJCAI*, pages 18–25, 2018.

[4] Ayesha Hameed, Andrzej Ordys, Jakub Możaryn, and Anna Sibilska-Mroziewicz. Control system design and methods for collaborative robots. *Applied Sciences*, 13(1):675, 2023.

[5] Fahad Sherwani, Muhammad Mujtaba Asad, and Babul Salam Kader K Ibrahim. Collaborative robots and industrial revolution 4.0 (ir 4.0). In *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pages 1–5. IEEE, 2020.

[6] Rudolph Emil Kalman. A New Approach to Linear Filtering and Prediction Problems. *Trans. of the ASME-Journal of Basic Eng.*, 82(Series D):35–45, 1960.

[7] Lingyi Zhang, David Sidoti, Adam Bienkowski, Krishna R. Pattipati, Yaakov Bar-Shalom, and David L. Kleinman. On the Identification of Noise Covariances and Adaptive Kalman Filtering: A New Look at a 50 Year-Old Problem. *IEEE Access*, 8:59362–59388, 2020. doi:10.1109/ACCESS.2020.2982407.

[8] Yalcin Bulut and Oguz Bayat. Kalman filtering with model uncertainties. *Topics in Modal Analysis I, Volume 5*, pages 447–455, 2012.

[9] Arie Levant. Sliding order and sliding accuracy in sliding mode control. *Int. J. of Control*, 58(6):1247–1263, 1993.

[10] K. Kumari, A. Chalanga, and B. Bandyopadhyay. Implementation of Super-Twisting Control on Order Perturbed Integrator System using Higher Order Sliding Mode Observer. *IFAC-PapersOnLine*, 49(18):873–878, 2016. 10th IFAC Symp. on Nonlinear Control Syst., NOLCOS 2016. doi:10.1016/j.ifacol.2016.10.276.

[11] Asif Chalanga, Shyam Kamal, Leonid M. Fridman, Bijnan Bandyopadhyay, and Jaime A. Moreno. Implementation of Super-Twisting control: Super-Twisting and higher order sliding-mode observer-based approaches. *IEEE Trans. on Ind. Electr.*, 63(6):3677–3685, Feb. 2016. doi:10.1109/TIE.2016.2523913.

[12] N Orani, Alessandro Pisano, and Elio Usai. On a new sliding-mode differentiation scheme. In *2006 IEEE Int. Conf. on Ind. Tech.*, pages 2652–2657, 2006.

[13] M. Ghanes, J. P. Barbot, L. Fridman, and A. Levant. A novel differentiator: A compromise between super twisting and linear algorithms. In *2017 IEEE 56th Annu. Conf. Dec. Contr. (CDC)*, pages 5415–5419, 2017. doi:10.1109/CDC.2017.8264460.

[14] Malek Ghanes, Jean-Pierre Barbot, Leonid Fridman, Arie Levant, and Robert Boisliveau. A new varying-gain-exponent-based differentiator/observer: An efficient balance between linear and sliding-mode algorithms. *IEEE Trans. on*

Autom. Control, 65(12):5407–5414, Feb. 2020. doi:10.1109/TAC.2020.2973609.

[15] Wen-Bin Lin and Huann-Keng Chiang. Super-twisting algorithm second-order sliding mode control for a synchronous reluctance motor speed drive. *Mathematical Problems in Engineering*, 2013, 2013.

[16] Shabnam Shakourzadeh, Giammarco Tonti, and Corrado Guarino Lo Bianco. An alternative hybrid super-twisting differentiator for a motor control system. In *2023 9th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 2002–2007, 2023. © 2023 IEEE. Reprinted, with permission, from Shakourzadeh, Shabnam and Tonti, Giammarco and Lo Bianco, Corrado Guarino, "An Alternative Hybrid Super-Twisting Differentiator for a Motor Control System," 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). doi:10.1109/CoDIT58514.2023.10284113.

[17] Jaime A Moreno and Marisol Osorio. Strict Lyapunov functions for the super-twisting algorithm. *IEEE Trans. on Autom. Control*, 57(4):1035–1040, 2012.

[18] Robert N Shorten and Kumpati S Narendra. Necessary and sufficient conditions for the existence of a common quadratic Lyapunov function for a finite number of stable second order linear time-invariant systems. *Int. J. of Adapt. Control and Signal Proc.*, 16(10):709–728, 2002.

[19] Vinay Chawda, Ozkan Celik, and Marcia K. O'Malley. Application of Levant's differentiator for velocity estimation and increased Z-width in haptic interfaces. In *2011 IEEE World Haptics Conf.*, pages 403–408, June 2011. doi:10.1109/WHC.2011.5945520.

[20] Danica Kragic, Joakim Gustafson, Hakan Karaoguz, Patric Jensfelt, and Robert Krug. Interactive, Collaborative Robots: Challenges and Opportunities. In *IJCAI*, pages 18–25, 2018.

[21] Samir Vojić. Applications of collaborative industrial robots. *Machines. Technologies. Materials.*, 14(3):96–99, 2020.

[22] Sami Haddadin, Alessandro De Luca, and Alin Albu-Schäffer. Robot Collisions: A Survey on Detection, Isolation, and Identification. *IEEE Transactions on Robotics*, 33(6):1292–1312, 2017. doi:10.1109/TRO.2017.2723903.

[23] Alessandro De Luca and Raffaella Mattone. Sensorless robot collision detection and hybrid force/motion control. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, pages 999–1004. IEEE, 2005.

[24] Suneel Kumar Kommuri, Seungyong Han, and Sangmoon Lee. External Torque Estimation Using Higher Order Sliding-Mode Observer for Robot Manipulators. *IEEE/ASME Transactions on Mechatronics*, 27(1):513–523, 2021.

[25] Jianhua Wu, Zixuan Liao, Yong Han, and Zhenhua Xiong. An Integral Design for High Performance Sensor-less Collision Detection of Serial Robots. In *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1714–1721, 2022. doi:10.1109/AIM52237.2022.9863373.

[26] Gianluca Garofalo, Nico Mansfeld, Julius Jankowski, and Christian Ott. Sliding Mode Momentum Observers for Estimation of External Torques and Joint Acceleration. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6117–6123, 2019. doi:10.1109/ICRA.2019.8793529.

[27] Sami Haddadin. *Towards safe robots: approaching Asimov's 1st law*, volume 90. Springer, 2013.

[28] Sami Haddadin, Alin Albu-Schaffer, Alessandro De Luca, and Gerd Hirzinger. Collision detection and reaction: A contribution to safe physical human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3356–3363. IEEE, 2008.

[29] Haoyu Wang, Zhiqiang Zuo, Yijing Wang, Hongjiu Yang, and Shaoping Chang. Composite nonlinear extended state observer and its application to unmanned ground vehicles. *Control Engineering Practice*, 109:104731, 2021.

[30] Yunfei Dong, Tianyu Ren, Dan Wu, and Ken Chen. Compliance control for robot manipulation in contact with a varied environment based on a new joint torque controller. *Journal of Intelligent & Robotic Systems*, 99(1):79–90, 2020.

[31] Tianyu Ren, Yunfei Dong, Dan Wu, and Ken Chen. Collision detection and identification for robot manipulators based on extended state observer. *Control Engineering Practice*, 79:144–153, 2018.

[32] Alessandro De Luca, Alin Albu-Schaffer, Sami Haddadin, and Gerd Hirzinger. Collision Detection and Safe Reaction with the DLR-III Lightweight Manipulator Arm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1623–1630, 2006. doi:10.1109/IROS.2006.282053.

[33] A. De Luca and R. Mattone. Actuator failure detection and isolation using generalized momenta. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 1, pages 634–639 vol.1, 2003. doi:10.1109/ROBOT.2003.1241665.

[34] Giammarco Tonti, Shabnam Shakourzadeh, and Corrado Guarino Lo Bianco. A fast collision detection system based on an extended state observer. In *2024 20th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, pages 1–8, 2024. © 2024 IEEE. Reprinted, with permission, from Tonti, Giammarco and Shakourzadeh, Shabnam and Lo Bianco, Corrado Guarino, "A fast Collision Detection System based on an Extended State Observer," 2024 20th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA). doi:10.1109/MESA61532.2024.10704895.

[35] Giammarco Tonti, Shabnam Shakourzadeh, and Corrado Guarino Lo Bianco. Optimal tuning of high-order super-twisting differentiators. In *2023 IEEE*

19th International Conference on Automation Science and Engineering (CASE), pages 1–6. IEEE, 2023.

[36] J. J. Craig. *Introduction to robotics mechanics and control*. Pearson Prentice Hall, Upper Saddle River, New Jersey, third edition, 2005.

[37] William Yang and Michael Posa. Impact invariant control with applications to bipedal locomotion. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5151–5158. IEEE, 2021.

[38] Michael Bombile and Aude Billard. Dual-arm control for coordinated fast grabbing and tossing of an object: Proposing a new approach. *IEEE Robotics & Automation Magazine*, 29(3):127–138, 2022.

[39] Irinel Constantin Morărescu and Bernard Brogliato. Trajectory tracking control of multiconstraint complementarity lagrangian systems. *IEEE Transactions on Automatic Control*, 55(6):1300–1313, 2010.

[40] Mark Rijnen, Alessandro Saccon, and Henk Nijmeijer. Reference spreading: Tracking performance for impact trajectories of a 1dof setup. *IEEE Transactions on Control Systems Technology*, 28(3):1124–1131, 2020. doi:10.1109/TCST.2019.2898953.

[41] Casper Beumer. Impact aware robot manipulation via task-based reference spreading. *Dept. of Mechanical Engineering. Report locator DC*, 2019.

[42] L.A. Veens. *Impact-aware grabbing control for a planar dual arm robot setup*. PhD thesis, Master's thesis, Department of Mechanical Engineering, Eindhoven University ..., 2022.

Acknowledgements

First and foremost, I would like to express my gratitude to my supervisor, Prof. Corrado Guarino Lo Bianco, for his guidance, insightful advice, and support throughout the course of my PhD. His expertise and encouragement have been fundamental to my academic development and the successful completion of this work.

I am also sincerely thankful to Dr. Alessandro Saccon for his trust and support during my visiting research at Eindhoven University of Technology. I am deeply appreciative of the opportunity to conduct experiments in his laboratory and to work with the Dual Arm setup, which played a crucial role in my research. His generosity in providing both the resources and a collaborative environment made my experience there enriching and productive.

Additionally, I would like to extend my thanks to my colleague Giammarco Tonti for his invaluable help and advice throughout my studies. His contributions and help have been a constant source of motivation and improvement in my work. I am also grateful to Marina Rainary for the joy and positivity she brought into my PhD journey, brightening even the most challenging moments.

Lastly, and most importantly, I would like to express my deepest gratitude to my husband, Saeid. His unwavering emotional and mental support has been a cornerstone of my journey, helping me navigate the ups and downs of this challenging path. I am incredibly fortunate to have had him by my side every step of the way.