UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Head Office: Università degli Studi di Padova

Department of Industrial Engineering

Ph.D. COURSE IN INDUSTRIAL ENGINEERING
Curriculum MECHANICAL ENGINEERING

Series XXXV

COLLISION MANAGEMENT IN THE EXECUTION OF ROBOTIC TASKS

Coordinator: Prof. Giulio Rosati

Supervisor: Prof. Giulio Rosati
Co-Supervisor: Prof. Alberto Doria

Ph.D. Student: Giulio Cipriani

... you can't connect the dots looking forward; you can only connect them looking backward. So you have to trust that the dots will somehow connect in your future. You have to trust in something: your gut, destiny, life, karma, whatever. This approach has never let me down, and it has made all the difference in my life.

- Steve Jobs

ABSTRACT

The increasing demand in the manufacturing market for highly customizable products requires high agility in the production systems. High agility means that these systems must be flexible enough to adapt quickly to product changes, ensuring, at the same time, high quality of the products.
In response to these new requirements, the actual trends among robotic researchers search for ways to reintroduce human operators' intrinsic flexibility in the production system. In particular, the actual trends regard using artificial intelligence and collaborative robotics. The former can be employed to learn a task directly from the human operator by direct demonstration or imitation. In contrast, the latter reintroduces the human operator in the workspace, making them work side by side with the robots. Both trends can successfully increase the flexibility of the robotic cell, but they come with some downsides. In particular, by using artificial intelligence, the task is no longer hard-coded in the robot program but it results from the predictions of dedicated algorithms. On the other hand, the reintroduction of human operators in the workspace implies that their safety must be ensured. Therefore, the added stochasticity and unpredictability of these systems demand the introduction of an additional safety layer to limit possible risks of collisions between the robot and the environment in which it moves, comprising the human operator eventually.
In this scenario, this work aims at providing possible solutions to collisions. Firstly, a mechanical approach is deployed to study mono-dimensional impacts between the robot and movable objects and between the robot and fixed obstacles. A mathematical model of the impacts is developed, comprising robot compliance, end-effector compliance, and impact characteristics. A mechanical solution, in the form of a non-linear bi-stable mechanism, is proposed to mitigate these collisions, and experimental validation corroborates the validity of the models. Two prototypes are designed for mono-dimensional tasks, and one other prototype is designed for extending these concepts to bi-dimensional impacts. All these prototypes confirm the usefulness of the proposed mechanical system in mitigating collisions. Then, a multi-physics approach is adopted to design a hydraulic system as an alternative to the mechanical one. A comparison between the two solutions is proposed, showing the pros and cons of adopting either one or the other system.
Finally, a collision avoidance algorithm is designed to tackle the problem of avoiding human operators during a collaborative task. The system adopts a state-of-the-art algorithm to calculate the trajectory of

the robot. However, it uses a novel preprocessing phase to eliminate the robot from the images and avoid the estimation of the distances between the human operator and the robot.

Keywords: Industrial robot, flexible systems, smart robotics, collaborative robotics, collision mitigation, collision avoidance

CONTENTS

1 INTRODUCTION

1.1 CHANGES IN ROBOTICS TRENDS

In the last decades, industrial robotics has been facing the challenge of coping with a rapidly changing environment, driven by the demand for small lots of highly customizable products. This new type of demand requires the adoption of highly flexible and agile systems. In other words, there is a need to secure both quick market responses and high-quality products [1].

In the past, automation substituted the highly flexible human assembly system to reduce the cost of labor in Western countries and increase the productivity and competitiveness of the companies. With the increase in product variety demand, different designs have been proposed to increase the flexibility of the robotic system. One example is the so-called Flexible Assembly System (FAS), where the human is substituted with a manipulator and a series of reconfigurable bowl feeders. However, despite the very short cycle time, their cost and set-up time are directly proportional to the complexity of the assembly that can influence the final dimension of the system and the number of feeders. Moreover, changing the product could require a complete transformation of the system. Therefore, Fully Flexible Assembly Systems (F-FAS) [2] have been designed to overcome some limitations.

The overall system consists of a manipulator, a single flexible feeder, and a vision system. This constructive simplicity eliminates the necessity for modifications to the system at product change but reduces the throughput with respect to FAS. As an intermediate solution, Hybrid Flexible Assembly System (H-FAS) [3] reintroduced the bowl feeders for the most complex or numerous parts of the assembly, reducing the cycle times but incrementing the cost and set-up time.

These systems are examples of the effort of the researchers to overcome the new market trends. However, the flexibility is still much lower than the manual assembly system.

Thus, the researchers' interest shifted toward finding ways to reinsert the human operators in the task, trying to instruct the systems based on their expertise, or using them to cover for the most complex phases of the task. Consequently, the trends turned toward intelligent and collaborative robotics [4]. The former can increase flexibility through the ability of the robot to learn directly from the operator [5] or to generalize its expertise to different tasks [6]. Therefore, considering that the cost of a robotic application is related 50/70% to the robot programming and integration [7], the use of intelligent robots

can reduce the time and resources required to implement a new solution.

The second trend sees collaborative robotics as a way to increase production flexibility by reintroducing the human worker in the task [8]. In this way, humans' flexibility and complex reasoning ability are mixed with the precision and repeatability of the robots.

These new trends can be seen as a direct evolution of the F-FAS [2] or the H-FAS [3] systems to increase their flexibility further. For example, an artificial intelligence (AI) algorithm can improve those systems by detecting multiple 3D grasping poses in real-time. Then, it can send them to an autonomous robot that can select and reach the most suitable pose [9]. Without an AI, all those poses must be hard-coded in the robot program slowing down the robot integration and the overall performance.

The increase in flexibility consequent to adopting either of these two systems is undoubted. However, since they can be considered new frontiers in robotics, there is still much work to be done from a regulation point of view [7, 10]. Therefore, current regulations lengthen the time necessary to implement these solutions, making them less attractive than traditional systems and limiting their application. The main issue with these solutions is related to safety reasons. Indeed, the possible inaccuracy of the predictions or the unpredictability of the human movement can put in danger the various components of the robotic system, including the human operator, if present.

In the next paragraph, an overview of the two trends is proposed, and safety considerations are drawn.

1.2 LEARNING ALGORITHMS: LEARNING FROM DEMONSTRATION

The literature about applications of artificial intelligence algorithms in industrial robotics is mainly focused on vision systems [6]. These systems are designed to replace the human operator in tasks of detection, localization, recognition, and scene understanding. For example, they can be used to inspect parts for defects [11], to predict grasping poses of objects disposed on a plane [12, 13], or objects poses even when overlapped [14].

Aside from vision systems, the literature offers interesting applications where authors made use of the so-called learning from demonstration (LfD) approach [5]. Recalling that collaborative robotics is born to take advantage of human dexterity during a task, with LfD, the authors are trying to learn that ability and transmit it to the robots.

Humans use demonstrations to teach other humans.

With the same logic, robotic tasks can be taught by observing a human operator executing the task [15]. The Learning from Demonstration (LfD) approach is fascinating because the teaching phase does

not require specific programming skills from the person that executes it. Indeed, a robot can learn the task through direct demonstration or imitation. The direct approach demonstrates the task using teleoperation or kinesthetic movements. The mapping between demonstration and execution is one-to-one. Hence, transformations are not required. On the other hand, in the imitation framework, the learning system observes the human executing the task. Thus, it does not influence normal behavior.

A basilar approach uses kinesthetic movements to show the system how to grasp an object. In [16], the authors fed a Convolutional Neural Network (CNN) with all the robot and gripper poses and all the data available from the sensors. The resulting trained model generates a heat map over the objects representing how suitable a particular position is for grasping. Consequently, the algorithm can generalize the demonstration over the entire workspace and object orientations.

Furthermore, demonstrations can also be used to learn trajectories. A common approach is to use Dynamic Movement Primitives (DMP) [17] to generalize the trajectories learned from the operator. With only fixed obstacles, one single demonstration is required to obtain a model able to generalize to different operative conditions (i.e., different starting or ending positions) [18].

Another interesting way to demonstrate the task to the robot is through imitation. In [19], the operator executes an assembly task while being recorded by an optical device. This method prevents workers' natural assembly motion from being modified. The only difference from everyday work is the presence of markers on the operator's hand that are necessary to track the hand movements.

Although these examples generalize the operator work, in most cases, they will not improve over it. Therefore, the resulting policies can be sub-optimal. One way to do this is by implementing demonstration in a Reinforcement Learning (RL) framework [20]. This family of learning algorithms learns through a trial and error approach, trying to maximize a reward function. However, considering the high amount of variables to be controlled in a robotic cell (for example, all the robot joints' positions and velocities), the reward function requires a lot of effort from a design point of view to be effective, and the training requires a lot of time to achieve high accuracy. For example, in [21], the authors taught a robot to assemble a circuit breaker without any prior knowledge. The teaching lasted 1380 minutes to achieve 94% of accuracy.

Due to the amount of training necessary and the high number of variables to tweak, the training is often executed in simulators to prevent damage to the system. However, the resulting accuracy is higher than the one shown in the real world [20].

A possible solution to speed up the training and increase the accuracy is to use human demonstrations to show a plausible policy to the robot. This demonstration can be used to generate a reasonable initial policy [22]. Thanks to demonstrations, the RL algorithm can achieve a high success rate in a reasonable time [23]. Moreover, thanks to its exploratory nature, it can improve over the demonstration, increasing its efficiency and coping with unexpected and untrained scenarios.

In conclusion, artificial intelligence algorithms are stochastic, and a minimum variation in the operative conditions or sensor readings can take to a completely different outcome. Therefore, their predictions are not always accurate enough to rely on them entirely, and errors can occur. Consequently, industrial environments require an additional safety system to prevent collisions and costly damages to the robotic system [24].

1.3 COLLABORATIVE ROBOTICS

As mentioned in the first paragraph, human-robot interaction (HRI) can take advantage of both the dexterity and flexibility of human operators and the repeatability, speed, accuracy, and strength of robots [10]. As a result, higher agility is achieved, and complex tasks can be completed without the need to hard-code them in the robot control program.

Even if traditional robots can be employed in a collaborative environment, a new family of robots, called cobots, is born to answer this specific purpose. Cobots are equipped with sensors and procedures to detect, prevent and react to impacts [25, 26], or their frame can be designed for this purpose [27]. In addition, thanks to their ability to work with human operators, they are not restricted inside closed cells. Therefore, they can be mounted over mobile robots and moved around factories with extreme simplicity to, for example, cover for ill workers or assist inexperienced operators [28].

This freedom comes at a cost. The operators' safety must be ensured, and the risk of a specific application must be assessed [29, 30]. The risks can be categorized into three different classes [30]: robot-related hazards, hazards related to the robot system, and application-related hazards. Moreover, due to the unpredictability of the human worker and the general dynamics of a collaborative task, the safety systems must assess all these risks in real-time to be effective.

Consequently, it is not a coincidence that the current research trend in the field is related to the development of systems to be paired with cobots to decrease the risk of their use [31, 32]. In addition, pure collaboration is mainly avoided to reduce that risk intrinsically. Indeed, in general, the Human-Robot Interaction can be divided into five categories [4]: cell with a fenced robot, coexistence, sequential collaboration, cooperation and responsive collaboration. Most works are

related to coexistence and sequential collaboration. The former sees operators and robots execute tasks in a separated workspace, whereas the latter sees them execute tasks at different times. Therefore, there is no direct human-robot interaction.

This limitation is mainly due to the impossibility of the robot to perceive the environment in which it moves. Indeed, the robot should master the three fields of obstacle avoidance, predictive control, and task recognition to be fully autonomous [33]. As an example, the human can interfere in the robot workspace as a result of a trajectory error or of a time sequence error [34]. In the former, the robot should predict the probability of a possible collision. In the latter, it should redefine its task and warn the operators of their poor performance. Knowing or predicting operators' intentions prevents useless movements, increasing productivity [35].

1.4 AIM OF THE WORK

The new industrial robotics trends increase the unpredictability in an industrial setting and, consequently, the number of risky scenarios during the execution of a robotic task.

This research aims to provide solutions to compensate for these new challenges, reducing the possible damages to the robot or the environment in which it moves.

The main part of this thesis adopts a mechanical approach to design systems that ensure a passive and instantaneous response to unpredicted collisions at the end-effector level. In other words, it assumes that the control algorithm can be faulted or that the sensors can be inaccurate enough to cause an imprecise analysis of the possible risks [24]. Moreover, it can also be exploited in a collaborative task. Indeed, in general, aside from the built-in features of cobots, collision avoidance approaches use vision systems [36]. However, they can be faulty due to possible occlusions when multiple persons or objects are present in the scene. In that cases, the human worker can go unseen and be hit by the moving robot.

The final part looks at the problem of preventing collisions between the robot and human operators. Therefore, the solution must detect and react to human movements in real-time, modifying the robot's trajectory when the operator is too close to the robot.

The works presented in this dissertation can answer to a "safety through control" approach to human-robot interaction. Indeed, four types of safety can be identified in collaborative robotics [37]: safety through control, motion planning, prediction, and consideration of psychological factors. Control methods can be divided into two categories: pre and post-collision. Pre-collision methods increase safety by trying to avoid collisions or limiting robot speed and energy in specific risky areas. However, if the system fails and a collision oc-

curs, post-collision methods must detect and minimize the damage to the human and the robot. Therefore, this work will present, firstly, a post-collision mechanical approach to mitigate collisions between the robot's end-effector and the environment and, finally, a pre-collision approach to collision avoidance.

1.5 OVERVIEW OF THE DISSERTATION

Aiming to address the problem of collisions raised in the previous Sections, this dissertation presents solutions to both collision mitigation and collision avoidance problems. In particular, the collision mitigation solutions are novel since they employ a mechanical approach to the problem, whereas typically, the approaches are related to improving the control systems. On the other hand, the collision avoidance solution proposes novel alternatives to the preprocessing phase of state-of-the-art algorithms.

Chapter 2 presents a study of mono-dimensional collisions between the robot and objects that are free to move. The study considered the influence of robot and end-effector compliance during the collision between the end-effector and the environment. The study brought to the design of a novel end-effector that can mitigate the collisions between the robot and objects disposed on a plane, reducing the momentum transferred to them. In particular, the end-effector uses a non-linear bi-stable mechanism to decouple the tool from the robot. The experimental validation of the models by means of a prototype of this end-effector is proposed.

Chapter 3 extends the previous study by considering the impacts of the robot against hard surfaces. Again, the bi-stable mechanism is proposed as a solution. A new prototype has been designed to validate the models experimentally. Finally, the Chapter presents an extension of the bi-stable mechanism to 2D by employing a rotational device that aligns the bi-stable mechanism along the direction of the impacts. The tool has been validated experimentally, demonstrating its functioning.

Chapter 4 addresses the collision mitigation problem with a multi-physics approach. A hydraulic solution has been designed, and an analytical comparison with the bi-stable mechanism is proposed. The two solutions present peculiarities that make them suitable for different situations and must be considered in the choice of the mitigation system.

Chapter 5 shifts the focus to collision avoidance. A novel approach to the problem is presented that employs Radial Basis Functions to remove the robot from depth images of the workspace and generate a repulsive function to divert the robot's trajectory from the possible obstacles.

Finally, in Chapter 6, conclusions and future work are presented.

PUBLISHED WORK *Parts of this dissertation, including figures, have been published or will be published in the future. The published works can be found in [6, 38, 39, 40, 41].*

2

MITIGATE COLLISION BETWEEN ROBOTS AND OBJECTS

This Chapter presents the study of monodimensional impacts between the robot and objects disposed on a plane and free to move. A mathematical model of impacts comprising robot dynamics, end-effector compliance, and impact characteristics is proposed. This study brought to the design of an end-effector able to reduce the momentum transferred to the objects. This end-effector can be helpful, for example, to speed up positioning tasks in teleoperation without increasing the risks of the operation. In addition, the model exposed in this Chapter will be the basis for the one in Chapter 3 where it is integrated to mediate collisions with hard surfaces.

2.1 IMPACT BETWEEN THE ROBOT AND ITS ENVIRONMENT

Recalling the LfD approach exposed in Chapter 1, one possibility to teach a task to a robot is by teleoperating it. Teleoperation sees a human operator moving a robot remotely by means of a dedicated joystick. Its effectiveness relies on the control system that tries to achieve the goals of stability and telepresence [42, 43]. Stability means that the system remains stable independently of an operator and environment behavior. In contrast, telepresence, or transparency, is the ability of the system to render the environment to the operator accurately. The main issues with this approach are related to the delay in the communication channel or the operator's inexperience.
Therefore, in the LfD framework using teleoperation, two scenarios can bring to a collision:

1. The inexperienced operator misreads the situation or gets distracted;

2. The AI prediction is inaccurate.

In both cases, the collision must be detected and mitigated to reduce possible damage to the robot or the environment.
The literature offers different solutions that focus on the improvement of the control system, such as wave variables [44], adaptive [45], and model predictive [46] algorithms. As control algorithms, they modify the system's response depending on the external perturbation. Therefore, they can be considered active responses.
On the other hand, a mechanical approach can be used to develop passive systems able to react almost instantly to an unexpected collision or perturbation. For example, in [47], the authors proposed a simple variable stiffness mechanism whose compliance can reduce

unexpected peak forces during a robotic deburring. In [48], the end-effector is designed so that its internal compliance can passively reduce the cutting force during a teleoperation task.

These examples show how the impact can be mitigated using a proper tool design or acting on the robot's control system. Many different situations require the mitigation and control of impact.

The study exposed in this Chapter is related to impacts between the robot and objects that are free to move on a plane. In this scenario, if the robot moves too fast, a hard impact will lead to a large final velocity of the object that may separate it from the tool and be shot far away. Therefore, managing the impact force is important because it may damage the object, the robot, and, in general, the work cell. Moreover, the robot control system is usually designed to decelerate a robot when a contact is detected, and measuring the impact force is one of the best ways to detect the impact. The duration of the impact is significant as well. If the impact is short in comparison with the typical response time of the robot, the robot has a short time window to decelerate after impact detection. Conversely, the robot has enough time to decelerate and mitigate the impact if the impact is rather long.

2.1.1 *Instantaneous impact*

If the duration of the impact is very short and the impact force is so large that other forces can be neglected, the impact between the tool of the moving robot and the object can be studied according to the classical impact theory [49]. Two limit cases are represented in Figure 1a and 1b.

In the first limit case, the tool, which has a plane face, is rigidly connected to the robot, and the object is a spherical point mass. Since the mass of the robot with the tool is much larger than the mass of the object, its velocity change is negligible. In other words, the velocity of the tool after the impact (V_t) is equal to the velocity before the impact (v_t) and is imposed by the control system of the robot. The final velocity of the object can be calculated using the restitution coefficient (e), which is the ratio of the relative velocity of the two masses after the impact to that before; $e = 1$ means that the contact is elastic (no kinetic energy is lost), and $e = 0$ means that the contact is completely inelastic.

The following equation holds:

$$V_p = V_t + e(v_t - v_p) \tag{1}$$

in which v_p and V_p are the initial and final velocities of the object, respectively. It is worth noting that the mass of the object has no effect on its final velocity. If the tool is moving with velocity $v_t = 1$ ms^{-1}. and the object is initially stationary, the final velocity of the

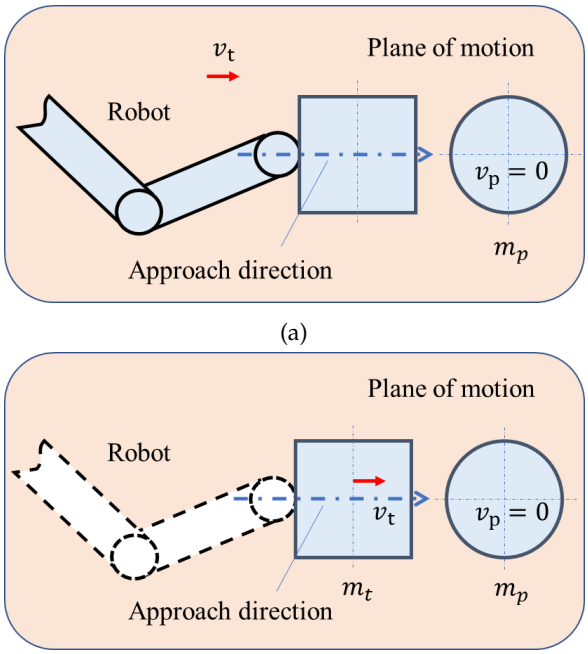


Figure 1: Instantaneous impacts: (**a**) A tool rigidly fixed to the robot; (**b**) a tool decoupled from the robot.

object is $V_p = 2\ ms^{-1}$ with $e = 1$ (pure elastic impact) and $V_p = 1\ ms^{-1}$ with $e = 0$ (completely inelastic impact).

In the second limit case, the tool is completely decoupled from the robot and is represented by a simple point mass m_t with plane faces. The initial velocity v_t of the tool is imposed by the motion of the robot, but it can change after the impact and the tool can move backwards. The conservation of momentum of the mechanical system depicted in Figure 1b leads to the following:

$$m_t v_t + m_p v_p = m_t V_t + m_p V_p \tag{2}$$

The final velocities of the two masses can be found by coupling Equation 2 with Equation 1:

$$V_t = v_t + \frac{m_p}{m_t + m_p}(1 + e)(v_p - v_t) \tag{3}$$

$$V_p = v_p - \frac{m_t}{m_t + m_p}(1 + e)(v_p - v_t) \tag{4}$$

Figure 2 shows the effect of e on the final velocities. A tool with $m_t = 0.125$ kg and $v_t = 1\ ms^{-1}$ and an object with $m_p = 0.25$ kg and $v_p = 0\ ms^{-1}$ are considered. The increase in the restitution coefficient leads to increases in the final velocity of the object and in the velocity variation of the tool.

The effect of tool mass is depicted in Figure 3. A decrease in the mass of the tool causes a decrease in the final velocity of the object for every value of the restitution coefficient.

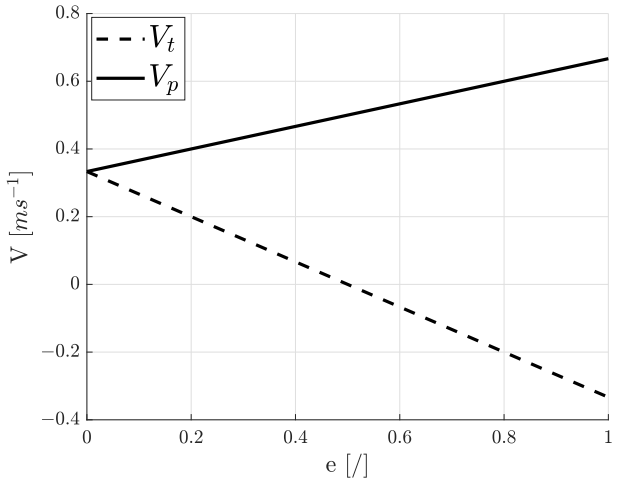


Figure 2: Influence of e on the end velocities of the tool and of the object.

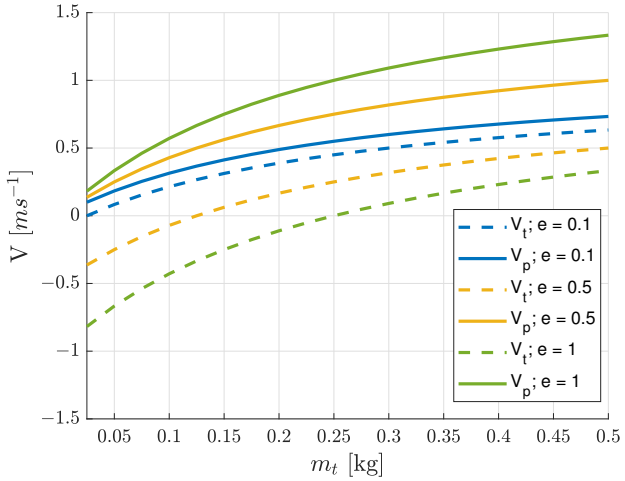


Figure 3: Influence of moving mass on end velocities.

The comparison between the final velocities of the object calculated in the two limit cases shows that the variation in the velocity of the tool leads to a large reduction in the final velocity of the object. In fact, the tool is neither rigidly fixed to a massive and stiff robot nor completely free to move. Nevertheless, this simple analysis shows that the decrease in the restitution coefficient and the mechanical decoupling between the tool and the robot are needed to minimize the object's final velocity.

2.1.2 *Non-instantaneous impact*

The theoretical analysis presented in the previous sub-section in many practical cases gives a good prediction of final velocities, but it does not give information about the contact force and duration, which are very important for robot control.

In recent years, many contact force models have been presented [50]. The impact between the two bodies can be schematized as in Figure 4, in which an elastic element and a damping element in parallel are present.

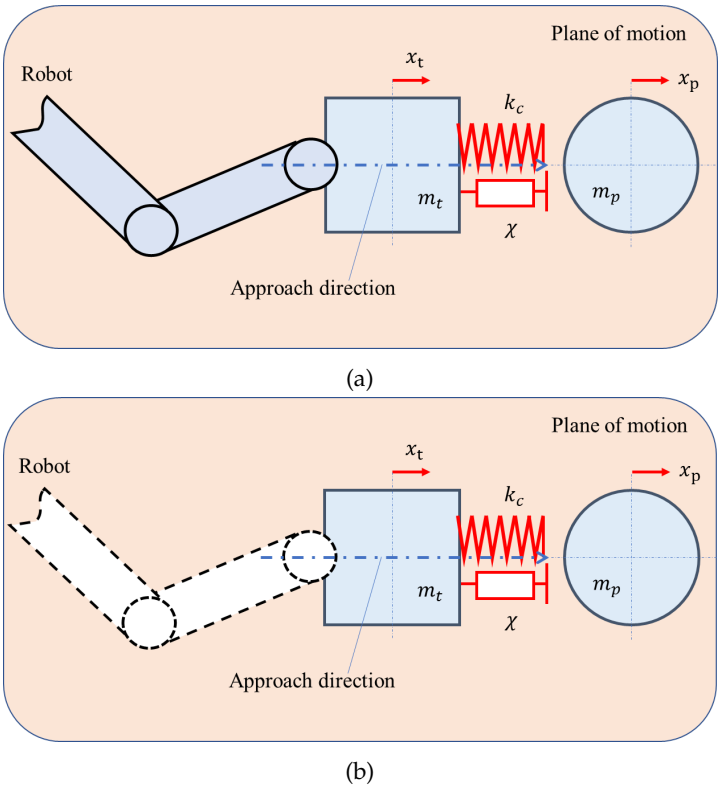


Figure 4: Non-instantaneous impacts: (**a**) A tool rigidly fixed to the robot; (**b**) a tool decoupled from the robot.

The elastic element can be linear or non-linear; the damping element is usually non-linear and the damping force depends both on the deformation velocity $\dot{\delta}$ and on deformation δ. Most of the proposed contact models [50, 51, 52] can be formulated according to this equation:

$$F = k_c \delta^m + \chi_c \delta^n \dot{\delta} \tag{5}$$

in which k_c is stiffness, χ_c is a constant and m and n are two exponents.

In the present case, F is the force that a tool exerts on the object, and $\delta = x_t - x_p$, with $x_t - x_p > 0$ to guarantee the contact.

The motion of the two masses after the impact can be calculated by solving the two differential equations:

$$\begin{aligned} m_t \ddot{x}_t &= -F \\ m_p \ddot{x}_p &= F \end{aligned} \tag{6}$$

If the tool is rigidly connected to the robot, only the second equation has to be solved, because x_t is assigned. Since in general F is non-linear, a numerical integration was carried out by means of Simulink.

It is worth noting that Equation 5 with $m = 1$ and $n = 0$ represents a simple linear spring-damper contact model. The simple linear contact model makes it possible to predict the contact forces and impact duration and leads to a simple relationship between the damping ratio of the relative motion and the coefficient of restitution [49], but it has some drawbacks. The coefficient of restitution does not depend on the relative velocity before the impact (as experimentally assessed [51]). The viscous component of the contact force is discontinuous, since it suddenly increases at the beginning of the contact, is not zero at the end of the contact and is opposed to the separation between the object and the tool. This behavior is not consistent with physical intuition, since at the beginning and at the end of the contact there is no penetration and the force should be zero [51].

A non-linear contact model overcomes most of the above-mentioned limits. In the framework of this research, a contact model with $m = \frac{3}{2}$ and $n = \frac{3}{2}$ was considered. Since the tool and the object can be schematized as a plane surface and a sphere, respectively [53], the exponent of the elastic force comes from the well-known Hertz contact law [51]. The calculation of the energy lost in the hysteresis loop of the damping force shows that exponent n has to be set equal to m [51]. The damping constant χ is related to the contact stiffness and to the restitution coefficient by this equation, as derived in [54]:

$$\chi_c = \frac{3}{2}\frac{(1-e)}{e} k_c \frac{1}{\dot{\delta}_{in}} \tag{7}$$

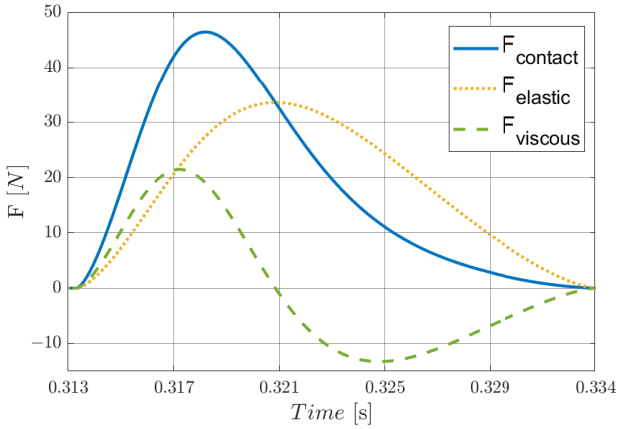
in which $\dot{\delta}_{in}$ is the initial relative velocity. It is worth noting that Equation 7 comes from an improvement of the impact model presented in [51], which extends the validity of the model in the range of small restitution coefficients.

Figure 5 shows the total contact force and its elastic and damping components calculated for the two limit cases of Figure 4. In the simulations, the parameters in Table 1 were adopted.

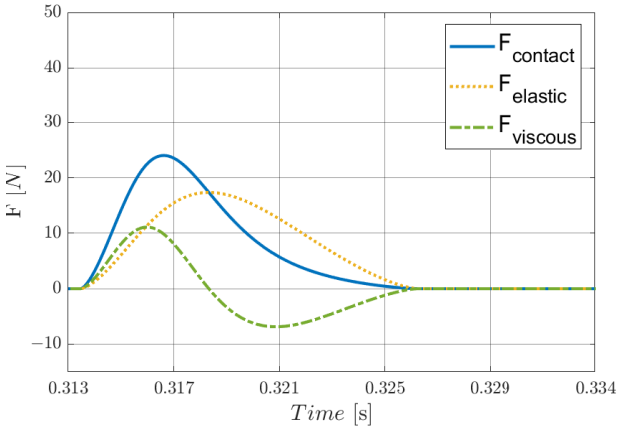
The non-linear model leads to a realistic shape of the damping force (without discontinuities) and of the contact force. If the tool is rigidly connected to the robot, the contact force peak is almost twice that of the peak that occurs when the tool is decoupled. The duration of the contact decreases if the tool is decoupled from the robot.

Parameter	Unit	Value
m_t	kg	0.125
m_p	kg	0.25
k_c	$Nm^{-1.5}$	10^5
e	/	0.5
v_t	ms^{-1}	1
v_p	ms^{-1}	0

Table 1: Parameters used to simulate the non-instantaneous impact.



(a)



(b)

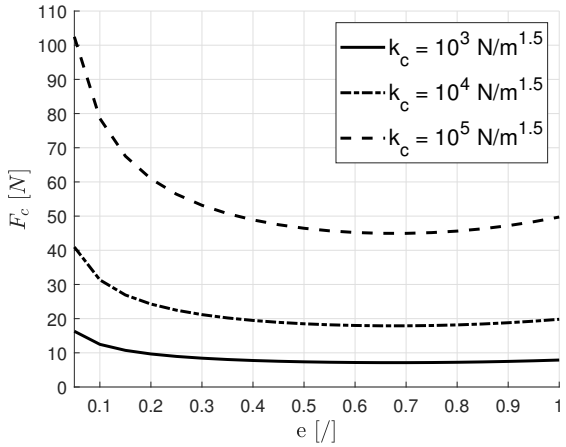
Figure 5: Temporal evolution of contact force for the non-linear model. (**a**) Limit case of Figure 4a; (**b**) limit case of Figure 4b

Referring to Figure 5a, it is worth noting that the damping component of the contact force becomes negative for $t > 0.321$ s. This result does not mean the presence of a force that is opposed to the detachment of the two bodies; in fact, even if in Figure 4 the spring and the damper are depicted as two separate lumped elements, they

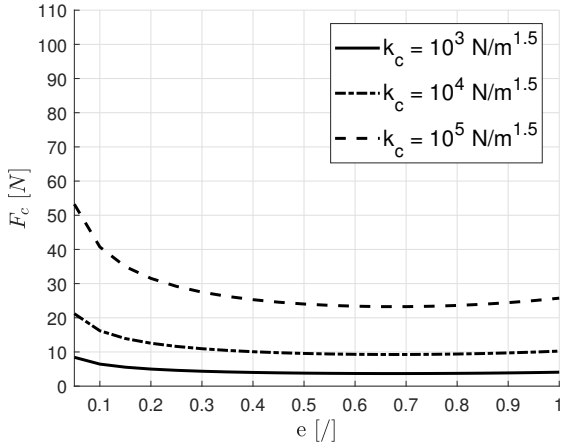
are associated to the deformation of the same body, and a negative damper force only means that damping decreases the contact force with respect to the theoretical value corresponding to a purely elastic contact.

The effects of the contact stiffness k_c and of the restitution coefficient e on the contact force and impact duration were analyzed by carrying out series of numerical simulations with $m_t = 0.125$ kg and $m_p = 0.25$ kg. The initial velocities were set to $v_t = 1\ ms^{-1}$ and $v_p = 0\ ms^{-1}$. The simulations considered both cases, as shown in Figure 4.

Figure 6 shows that, when e is assigned, an increase in contact stiffness k_c leads to an increase in contact force. This behavior is consistent with physical intuition. When k_c increases, the deformation de-



(a)

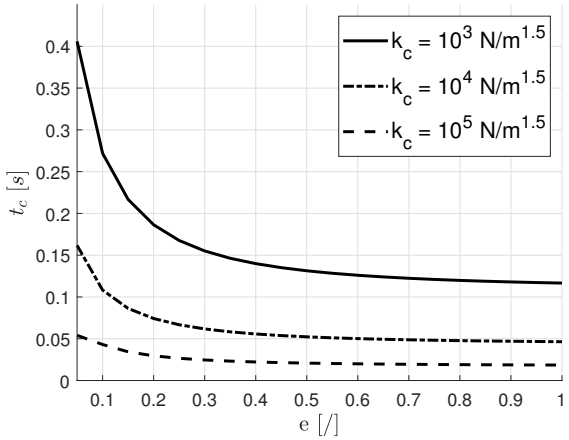


(b)

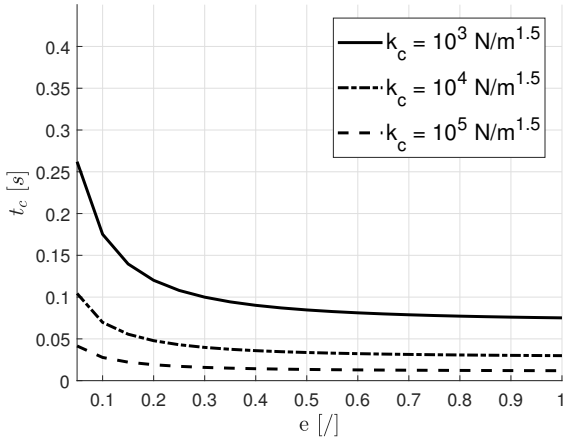
Figure 6: Influence of k_c and e on contact force. (**a**) Limit case of Figure 4a; (**b**) limit case of Figure 4b.

creases, but the elastic energy transferred from the tool to the object has to remain constant, since e is fixed. Thus, the reduction in defor-

mation is compensated by the increase in contact force. If the contact stiffness k_c remains constant, the global contact force increases as the restitution coefficient e decreases, owing to the increasing contribution of the damper force.
Figure 7 shows that an increase in contact stiffness k_c leads to a decrease in the impact duration. This effect takes place because deformation decreases when k_c increases. The increase in the impact duration with the decrease in restitution coefficient e is related to the fact that if e tends to zero (inelastic impact), the two masses move together after the impact.



(a)



(b)

Figure 7: Influence of k_c and e on contact duration. (**a**) Limit case of Figure 4a; (**b**) limit case of Figure 4b.

2.2 EFFECTS OF ROBOT COMPLIANCE

In the analysis of the interaction between the tool of the robot and an object, it is important to consider three essential aspects of robot

dynamics. Firstly, as the mass of colliding objects dramatically influences the momentum transfer (see Figure 3), it is important to quantify the robot mass involved in the collision. Secondly, since industrial robots are characterized by an intrinsic compliance [55], this feature could be exploited to mitigate the impacts. Finally, the combination of joint elasticity and link inertia induces vibrations in the robot that affect the motion of the tool and the impact velocity. Thus, this section aims to develop a one-DOF robot model that is useful to evaluate the contribution of the inertia and elasticity of a robot along the approach direction during the collision.

2.2.1 *Model of robot compliance*

The mathematical model of Figure 8 was developed. The slider moves

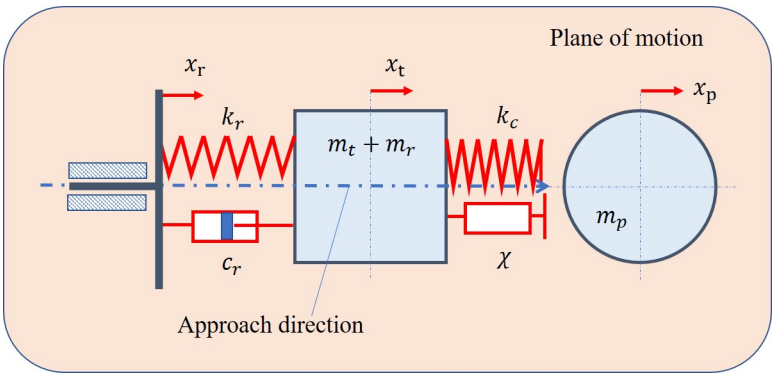


Figure 8: Scheme of the impact between the compliant robot and an object.

along the approach direction with the motion law imposed by the robot control system (coordinate x_r). It is connected to the motors of the robot by means of ideal transmissions without compliances and clearances. The spring with a stiffness of k_r and the damper with a constant of c_r represent the compliance and the damping of the robot in the approach direction. A share (m_r) of the mass of the robot is added to the tool. The friction force between the object and the plane of motion is not taken into account, but it may cause small (and beneficial) reductions in the final velocity of the object.

During the collision, the system is described by the following equations of motion:

$$(m_r + m_t)\ddot{x}_t = c_r(\dot{x}_r - \dot{x}_t) + k_r(x_r - x_t) - k_c\delta^m - \chi\delta^n\dot{\delta} \tag{8}$$

$$m_p\ddot{x}_p = k_c\delta^m + \chi\delta^n\dot{\delta} \tag{9}$$

Equations 8 and 9 hold if $\delta > 0$.

The main issue is the identification of the robot's parameters, which appear in this model (k_r, c_r, m_r). In this research an Omron Adept Viper s650 is considered. The robot is represented in Figure 9.

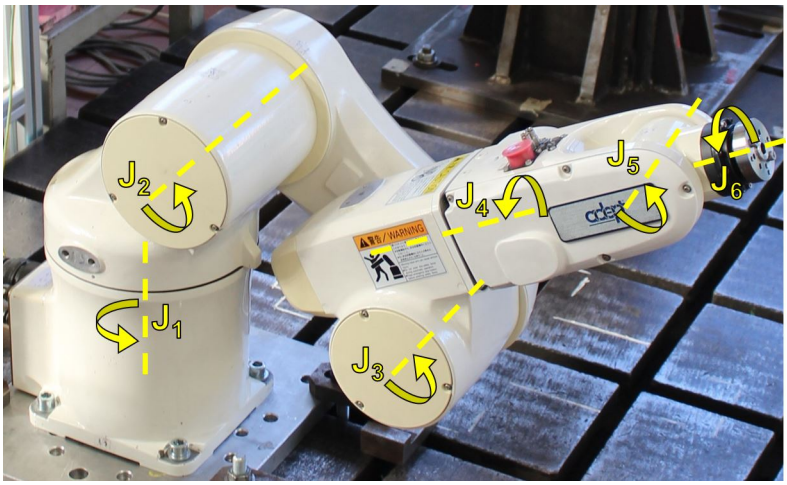


Figure 9: Robot: Adept Viper s650.

It is assumed that robot compliance is due to the compliance of the joints around their rotation axes, while the links and bearings are considered perfectly rigid. With these hypotheses, if the i-th actuator imposes a rotation $q_i(t)$ on the i-th joint, the actual rotation $\vartheta_i(t)$ of the joint does not coincide with $q_i(t)$. In order to consider this behavior, each joint was schematized by means of a spring-damper model, so that the torque produced by the actuator was transmitted to the joint by means of these elements. The motion of a serial six-DOF robot with compliant joints is represented by a system of six second order differential equations:

$$\boldsymbol{M_q}(q)\,\ddot{\vartheta} + \boldsymbol{C}(q,\dot{q}) + \boldsymbol{D_q}(\dot{\vartheta}-\dot{q}) + \boldsymbol{K_q}(\vartheta - q) + \boldsymbol{G}(q) = 0 \qquad (10)$$

where

q is the vector which contains the joint variables imposed by actuators;

ϑ is the vector containing actual joint variables;

$\boldsymbol{M_q}(q)$ is the robot mass matrix in the joint space;

$\boldsymbol{C}(q,\dot{q})$ is the term related to centrifugal and Coriolis forces;

$\boldsymbol{D_q}$ is the robot viscous damping matrix in the joint space;

$\boldsymbol{K_q}$ is the robot stiffness matrix in the joint space; and

$\boldsymbol{G}(q)$ is the term related to gravity forces.

It is possible to neglect centrifugal and Coriolis forces and the effect of gravity when dealing with small oscillations [56]. Therefore, the

equations of motion of the robot in the joint space can be simplified as follows:

$$\boldsymbol{M_q}(q)\ddot{\vartheta} + \boldsymbol{D_q}(\dot{\vartheta} - \dot{q}) + \boldsymbol{K_q}(\vartheta - q) = 0 \tag{11}$$

The mass matrix $\boldsymbol{M_q}(q)$ can be estimated using the data provided by the computer-aided design (CAD) model of the robot, retrieved from the website of the manufacturer. The matrices $\boldsymbol{D_q}$ and $\boldsymbol{K_q}$ are diagonal and independent of the robot configuration $\boldsymbol{q}$. In particular, the terms of these matrices represent the viscous damping coefficient and the stiffness of each joint around its rotation axis, and they can be identified by means of an impulsive modal analysis of the robot [56, 57]. The movement of the tool and its collision with an object in the plane of motion are described in the operational space, so it is necessary to transfer the description of robot dynamics from the joint space to the cartesian space. Thus, the mass, stiffness and viscous damping matrices in the cartesian space ($\boldsymbol{M_X}$, $\boldsymbol{D_X}$, $\boldsymbol{K_X}$) have to be calculated. The relationships which correlate the matrices in the two spaces [58] are the following:

$$\boldsymbol{M_X}(q) = \boldsymbol{J}(q)^{-T}\, \boldsymbol{M_q}(q)\, \boldsymbol{J}(q)^{-1} \tag{12}$$

$$\boldsymbol{D_X}(q) = \boldsymbol{J}(q)^{-T}\, \boldsymbol{D_q}\, \boldsymbol{J}(q)^{-1} \tag{13}$$

$$\boldsymbol{K_X}(q) = \boldsymbol{J}(q)^{-T}\, \boldsymbol{K_q}\, \boldsymbol{J}(q)^{-1} \tag{14}$$

where $\boldsymbol{J}(q)$ represents the robot's Jacobian matrix. It is worth noting that both stiffness and viscous damping matrices ($\boldsymbol{K_X}$, $\boldsymbol{D_X}$) are dependent on the robot configuration $\boldsymbol{q}$. Consequently, it is possible to write the equations of motion in the cartesian space:

$$\boldsymbol{M_X}(X_r)\ddot{X}_t + \boldsymbol{D_X}(X_r)(\dot{X}_t - \dot{X}_r) + \boldsymbol{K_X}(X_r)(X_t - X_r) = 0 \tag{15}$$

In Equation 15, $\boldsymbol{X_r}$ is the vector containing the coordinates of the imposed position of the tool in the cartesian space, whereas $\boldsymbol{X_t}$ is the vector containing the coordinates related to the actual position of the tool. The robot configuration $\boldsymbol{q}$ is a function of the imposed position $\boldsymbol{X_r}$, obtained through the equations of the inverse kinematics of the robot. Thus, the mass, stiffness and damping matrices can be expressed as functions of $\boldsymbol{X_r}$ as in Equation 15.

Equation 15 makes it possible to describe the dynamics of the robot along the imposed tool trajectory. However, it is assumed that the robot control system is able to exactly generate the desired trajectory. In fact, this is not possible in real applications, due to a series of small-entity causes, such as structural imperfections and external disturbances, which introduce small position errors. As already stated,

the aim of this model is to estimate the influence of joint compliances and robot inertia properties during a collision, so the hypothesis of perfect control is reasonable.

In the one-DOF model, only the cartesian coordinate that describes the motion of the tool in the direction of approach (x) is considered. Therefore, the equations of motion of the robot are reduced to a single second-order differential equation:

$$M_{X,11}(x_r)\ddot{x}_t + D_{X,11}(x_r)(\dot{x}_t - \dot{x}_r) + K_{X,11}(x_r)(x_t - x_r) = 0 \quad (16)$$

where x_r is the x-coordinate imposed by the robot control system, x_t is the actual x-coordinate of the tool, and $M_{X,11}$, $D_{X,11}$ and $K_{X,11}$ are the elements (1,1) of mass, viscous damping and stiffness matrices, respectively.

The mass, damping and stiffness terms of this equation physically represent the lumped element components (m_r, c_r, k_r respectively), which are added to the impact model represented in Figure 8. Mass $M_{X,11} = m_r$ simply adds to the tool mass.

The terms in Equation 16 depend on the robot configuration. For this reason, a MATLAB script was developed to calculate the robot configuration from the current position in the cartesian space and to determine the parameters of the one-DOF model. The script follows the steps reported in the flow chart presented in Figure 10.

2.2.2 *Numerical results*

In the simulations, the tool approached an object along a linear trajectory. Moreover, it was assumed that the tool moved between the starting and final point following a trapezoidal velocity profile, and a collision occurred when the robot moved at a constant velocity. The position and velocity are referred to the fixed reference frame of the first joint of the robot (see Figure 9). Figure 11 represents the imposed position (x_r) and the imposed velocity ($v_r = \dot{x}_r$) of the robot flange before the collision. In order to evaluate the behavior of the robot during the interaction with an object, the robot was assumed to be equipped with a control system that was capable of detecting the contact force and ordering the robot to start braking. The force detection threshold was set to 1 N. The reaction to the impact was not instantaneous, but it was assumed that the robot began to perform braking after 16 ms in order to take into account the delay in processing the external input signal [59]. The circular marker (Figure 11) indicates when the impact happens.

The trends of the lumped element parameters of the robot m_r, c_r, k_r as functions of the imposed position x_r are depicted in Figure 12. The circular marker highlights the values of these parameters when the impact occurs. Several scenarios were simulated, with the aim of evaluating the effects of robot properties during the collision. The sim-

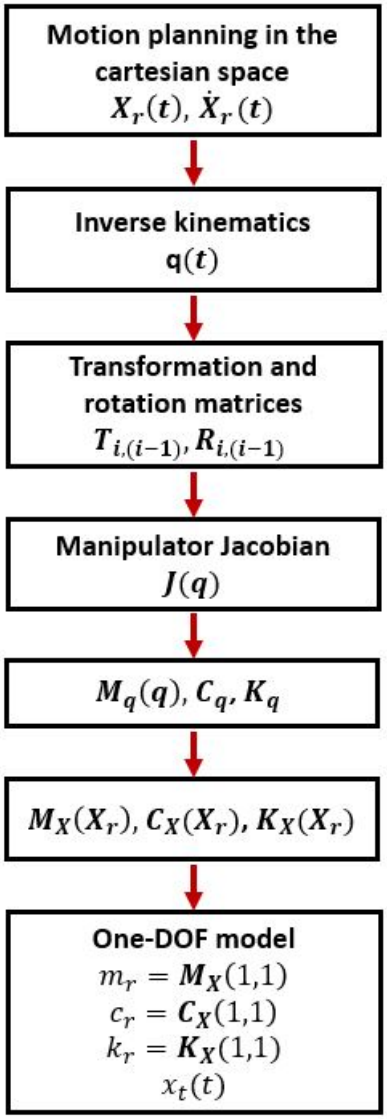


Figure 10: Flow chart of the steps taken to evaluate the terms $M_{X,11}$, $D_{X,11}$ and $K_{X,11}$ along the trajectory of the tool. DOF: degree of freedom.

ulation results depicted in Figure 13a and 13b represent the velocities of the tool and of the object when the robot had rigid and compliant joints, respectively. A perfectly elastic collision was considered ($e = 1$) with $k_c = 10^5$ $Nm^{-1.5}$; the masses of the tool and of the object were $m_t = 0.125$ kg and $m_p = 0.25$ kg, respectively.

In Figure 13a, the object velocity is exactly twice the approach velocity of the robot and the collision is so short that the robot actuators start braking after the impact. In this condition, as the impact occurred in a period of time in which the mass $(m_r + m_t)$ maintained the same velocity, the robot with rigid joints was equivalent to a slider moving at constant velocity. The final velocity of the object confirmed this assumption (see Section 2.1.1).

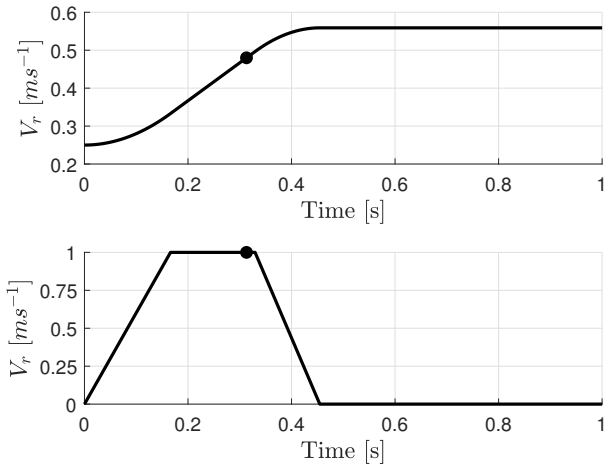


Figure 11: Position and velocity profiles imposed on the tool by the robot control system. The circular dot indicates the impact time.

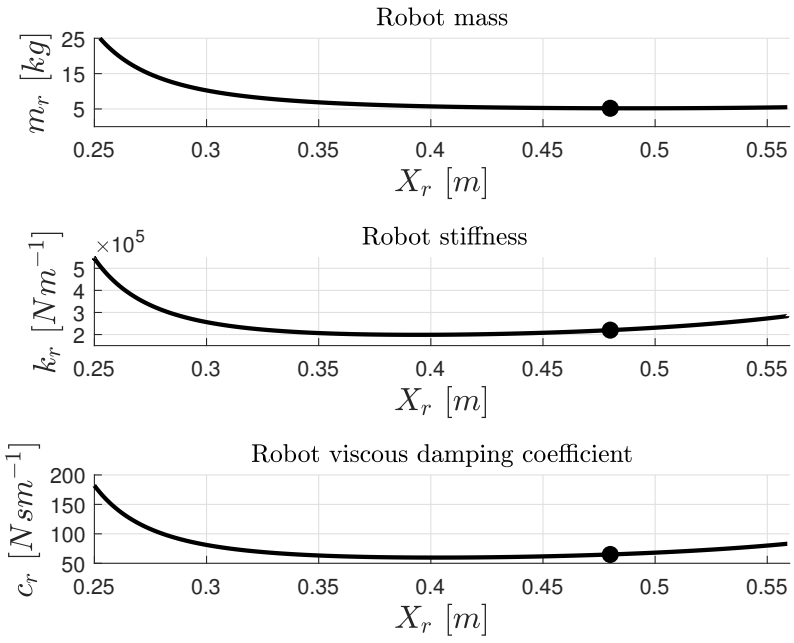
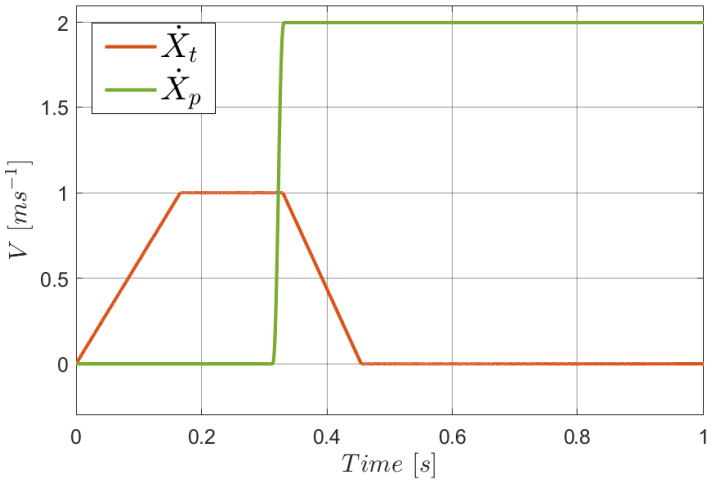
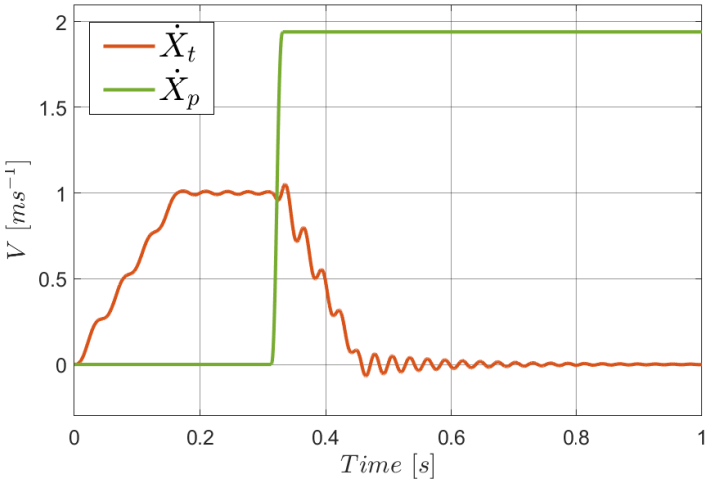


Figure 12: Lumped components m_r, c_r and k_r as a function of the imposed position x_r. The circular dot indicates the impact time.

In the case of compliant joints (Figure 13b), the impact velocity is about 0.98 ms^{-1} due to the vibrations of the tool (the impact takes place when there is a local minimum of robot velocity), and the velocity of the object is about 1.94 ms^{-1}, which is almost twice the approach velocity of the robot. Although the total colliding mass of the robot $(m_r + m_t)$ is not rigidly coupled with the imposed motion, due to the compliances represented by the elements k_r and c_r of the model in Figure 8, the actual behavior is nearly the same as the robot



(a)

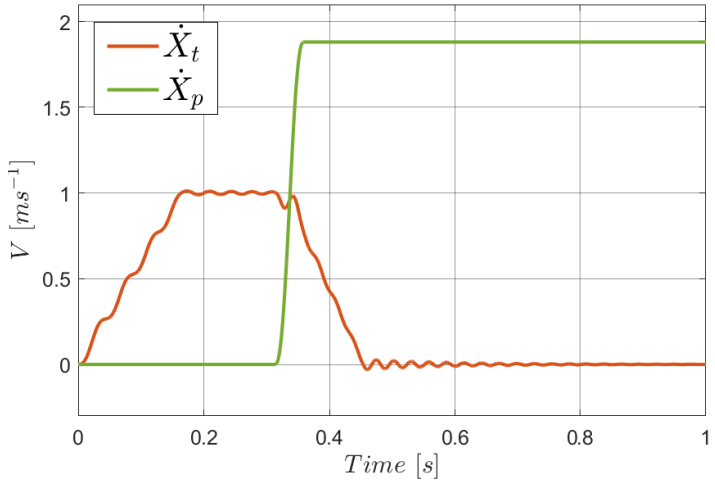


(b)

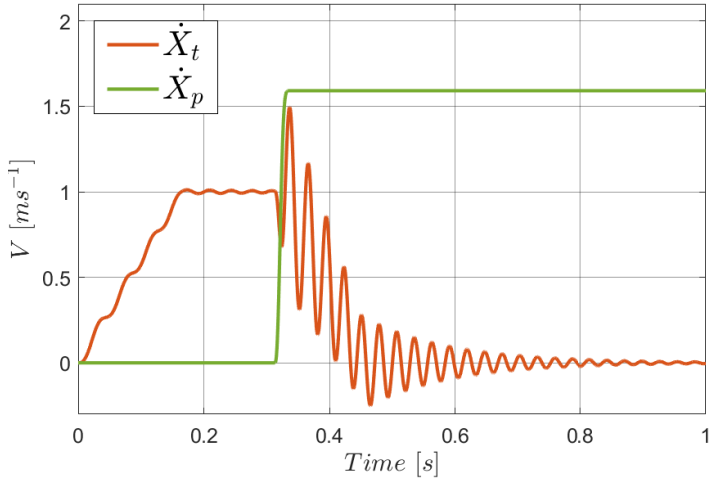
Figure 13: Time history of tool and object velocity. (**a**) Robot with rigid joints; (**b**) robot with compliant joints.

with rigid joints. Two factors explain these results: firstly, the colliding masses are very different, since the term $(m_r + m_t)$ is equal to 5.35 kg just before the impact, and Equation 4 shows that object velocity increases asymptotically towards the limit case, when the tool mass becomes large; secondly, since the contact force is small, joint compliances have a small influence.

In Figure 14a, a heavy object with a mass of 2.5 kg is considered. This mass roughly corresponds to the rated payload of the robot and is about 48% of the term $(m_r + m_t)$, which is equal to 5.35 kg just before the impact. The contact stiffness is set to $k_c = 10^5\ Nm^{-1.5}$. In Figure 14b the impact against a heavy object ($m_p = 2.5$ kg) is still simulated, but the contact stiffness is set to $k_c = 10^6\ Nm^{-1.5}$. The collision is perfectly elastic ($e = 1$) in both cases.



(a)



(b)

Figure 14: Time history of tool and object velocity with robot with compliant joints and: (**a**) $k_c = 10^5 Nm^{-1.5}$; (**b**) $k_c = 10^6 Nm^{-1.5}$.

With $k_c = 10^5 Nm^{-1.5}$, the final velocity of the object is about 1.88 ms^{-1}; this value is very close to that in Figure 13b, despite the difference in the masses of the objects. The impact velocity is about 0.95 ms^{-1}, due to a small deformation of the robot and to the vibration of the tool. Although the momentum transferred to the robot is not negligible, the contact force is not able to significantly deflect the robot arm; therefore, the robot is perceived by the tool as a slider moving with constant velocity.

The effect of the robot compliance is more relevant when the object is heavy and the contact stiffness is large ($k_c = 10^6 Nm^{-1.5}$), which is the case depicted in Figure 14b. These conditions lead to a larger contact force that is able to deflect the joints of the robot. The comparison between Figure 14a and 14b shows that in the second scenario,

the tool is subjected to a significant deceleration during the impact, which means that the robot has been deformed. The deformation implies that a portion of the mechanical energy involved in the collision is stored as elastic energy inside the joints, and this phenomenon generates an increment in the amplitude of vibrations of the tool, whereas the velocity of the object decreases with respect to the previous cases. However, the elastic response is so quick, due to the high stiffness, that the restitution of part of the stored elastic energy occurs during the collision and the object is pushed away.

These results highlight that in most cases, the robot is perceived by the object as a slider moving with an imposed motion. In conclusion, joint compliances cannot significantly reduce the velocity of the object.

2.3 EFFECTS OF TOOL COMPLIANCE

Section 2.2 shows that the inherent elastic and inertial properties of the robot have a small effect on the momentum transfer between the tool and the object. Therefore, in order to compensate for the small compliance of the robot, it is necessary to decouple the tool from the robot. The decoupling system and its effect on the final velocity of the object are discussed in this section.

2.3.1 *Model of tool compliance*

If robot compliance is negligible, the model that describes the collision comprises the rigid robot, the decoupling elements and the tool (see Figure 15).

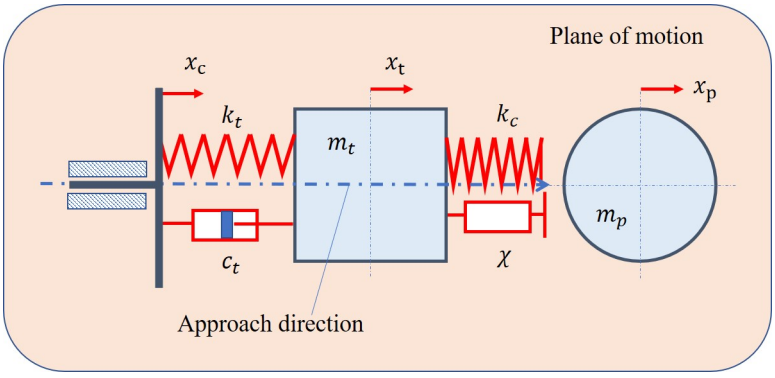


Figure 15: Model of the collision between a tool with decoupling elements and an object.

The corresponding equations of motion during the collision are as follows:

$$m_t \ddot{x}_t = c_t(\dot{x}_c - \dot{x}_t) + k_t(x_c - x_t) - k_c \delta^m - \chi \delta^n \dot{\delta} \tag{17}$$

$$m_p \ddot{x}_p = k_c \delta^m + \chi \delta^n \dot{\delta} \tag{18}$$

The rigid body moving with imposed motion is the robot flange. The parameters k_t and c_t represent the stiffness and damping characteristics of the system that decouple the tool from the robot. The set of the decoupling system and the tool represents the end-effector. It is worth noting that the mass-spring-damper system that represents the end-effector is excited by the imposed motion x_c, which represents the x-coordinate of the robot flange.

If the two equations are summed, the impact force disappears:

$$m_t \ddot{x}_t + m_p \ddot{x}_p = c_t(\dot{x}_c - \dot{x}_t) + k_t(x_c - x_t) \tag{19}$$

If Equation 19 is integrated between the instant (t_1), in which the contact between the tool and the object begins, and the instant (t_2), in which the contact ends, the following result is obtained:

$$m_t V_t + m_p V_p = m_t v_t + m_p v_p + c_t \int_{t_1}^{t_2} (\dot{x}_c - \dot{x}_t) dt + k_t \int_{t_1}^{t_2} (x_c - x_t) dt \tag{20}$$

Equation 20 shows that the linear momentum of the system is no longer conserved owing to the effect of the impulses of the damping and elastic forces. It is worth noting that the elastic and damping forces increase the final momentum as long as the terms in the integrals are positive. The total impulse generated by the decoupling elements is

$$I = c_t \int_{t_1}^{t_2} (\dot{x}_c - \dot{x}_t) dt + k_t \int_{t_1}^{t_2} (x_c - x_t) dt \tag{21}$$

Introducing Equations 1 and 21 into Equation 20 leads to the following equation:

$$V_p = v_p - \frac{m_t}{m_t + m_p}(1 + e)(v_p - v_t) + \frac{I}{m_t + m_p} \tag{22}$$

Equation 22 highlights that the total impulse generated by the decoupling system increases the final velocity of the object.

2.3.2 *Simulated results*

The effects of the stiffness and damping of the decoupling system on impact mechanics were analyzed by carrying out parametric simulations with $m_t = 0.125$ kg, $m_p = 0.25$ kg, $v_t = 1\ ms^{-1}$ and $v_p = 0\ ms^{-1}$. The robot was assumed to start braking after the collision.

The damping coefficient was defined by means of the viscous damping ratio [60]

$$\zeta = \frac{c_t}{2\sqrt{k_t m_t}}. \tag{23}$$

Figure 16 represents the final velocity of the object for different values of contact stiffness k_c, as a function of the stiffness k_t and considering a constant damping ratio, which is equal to the critical value $\zeta = 1$. A perfect elastic impact ($e = 1$) is assumed.

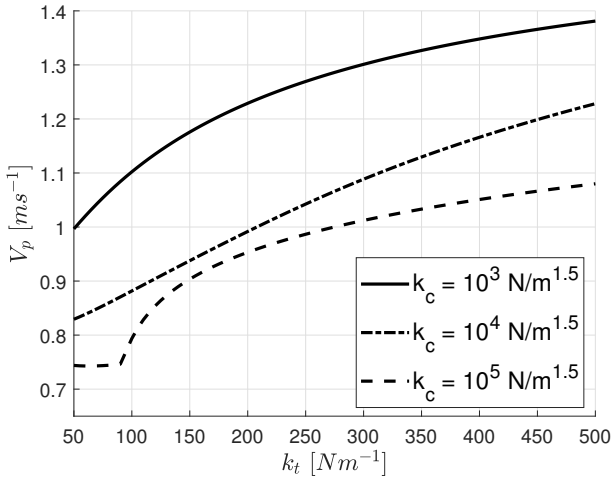


Figure 16: Effects of the stiffness and of the damping of the decoupling system on the final velocity of the object with $e = 1$.

The comparison between Figures 13a and 16, shows that the decoupling system leads to a significant reduction in the velocity of the object after the collision. The increase in stiffness k_t and in damping c_t of the decoupling system increases the final velocity of the object, in agreement with Equations 20 and 22. Moreover, Figure 16 shows that an increase in k_c leads to a decrease in the final velocity of the object. This phenomenon is related to the fact that an increase in k_c produces an increase in the contact force which, in turn, generates a large compression of the decoupling system, which moves backwards. Since $e = 1$, the relative velocity V_{rel} before and after the impact are equal and, when there is a backwards motion of the tool, the absolute final velocity of the object $V_p = V_t + V_{rel}$ is reduced.

When k_t is about 90 Nm^{-1}, the curve corresponding to the largest contact stiffness suddenly changes its slope. After the collision, the elastic energy stored inside the spring of the decoupling system is released and the tool is subjected to a rebound. If the impact duration is short, the restitution of the elastic energy starts very quickly. In this condition, multiple impacts can occur. Therefore, the sudden increment in slope highlights the presence of multiple collisions, which leads to an increment in the final velocity.

Figures 17a and 17b represents the robot, tool and object velocities with $k_t = 150\ Nm^{-1}$, $\zeta = 1$, $e = 1$. In Figure 17a, the contact stiffness is set to $k_c = 10^5\ Nm^{-1.5}$, whereas in Figure 17b, the contact stiffness is set to $k_c = 10^4\ Nm^{-1.5}$. The contact force applied to the object and the force generated by the decoupling system on the tool are depicted in Figures 17c and 17d, respectively.

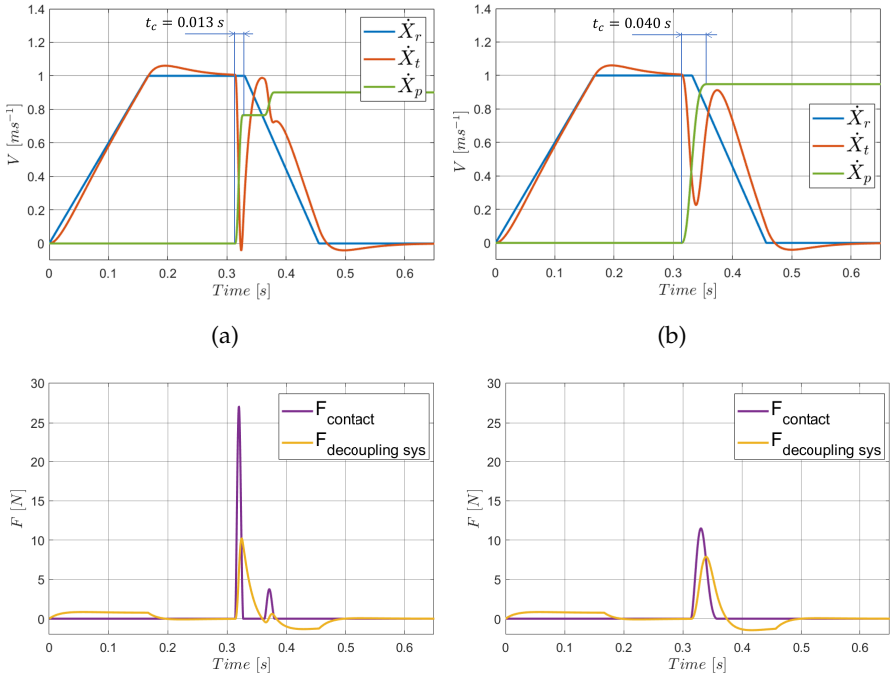


Figure 17: Velocity comparison: (**a**) $k_c = 10^5\ Nm^{-1.5}$; (**b**) $k_c = 10^4\ Nm^{-1.5}$. The contact force and total force generated by the decoupling elements are as follows: (**c**) $k_c = 10^5\ Nm^{-1.5}$; (**d**) $k_c = 10^4\ Nm^{-1.5}$. $e = 1$.

The end-effector is compressed by the inertia force during the acceleration of the robot. Therefore, the velocity of the tool is slightly higher than the robot velocity when the latter moves at a constant velocity due to the restitution of the elastic energy absorbed during the acceleration. The increase in contact stiffness increases the deceleration of the tool. Figure 17c highlights two clear peaks in the contact force profile and confirms the presence of multiple impacts.

When inelastic impacts are considered, the results show that the final velocity of the object decreases (see Figure 2). Furthermore, the decrease in contact stiffness k_c leads to a reduction in the final velocity of the object; this behavior is opposite to that found with $e = 1$. Parametric simulations were carried out to analyze this aspect considering $m_t = 0.125$ kg, $m_p = 0.25$kg, $v_t = 1ms^{-1}$ and $v_p = 0ms^{-1}$. The robot was assumed to start braking after the collision.

Figure 18 represents the final velocity of the object for different values of contact stiffness k_c as a function of the stiffness k_t considering a constant damping ratio equal to the critical value ($\zeta = 1$). A nearly completely inelastic impact ($e = 0.1$) was simulated.

The comparison between Figures 16 and 18 highlights that the reduction in the coefficient of restitution inverts the effect of the contact stiffness. A coefficient of restitution near to zero means that the tool and the object move almost synchronously after the collision, because the final relative velocity is only a small fraction of the initial relative velocity. There are two contributions to the final velocity of the object: the momentum transferred during the collision and the momentum transferred during the transient after the impact, when the robot brakes.

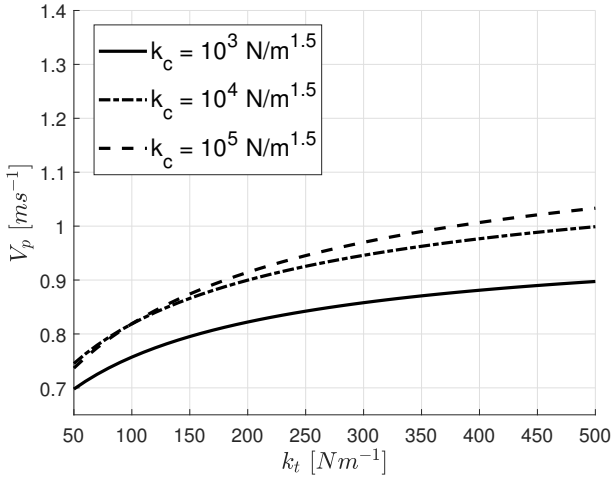


Figure 18: Effects of the impulses of the damping and elastic forces on the final velocity of the object with $e = 0.1$.

Figures 19a and 19b represent the velocities of the robot, tool and object. Figures 19c and 19d show the contact force and the total force generated on the tool by the decoupling elements. All simulations were carried out with $k_t = 150\ \mathrm{Nm}^{-1}$, $\zeta = 1$, $e = 0.1$. The contact stiffness was set to $k_c = 10^5\ \mathrm{Nm}^{-1.5}$ in Figures 19a and 19c and to $k_c = 10^3\ \mathrm{Nm}^{-1.5}$ in Figures 19b and 19d, respectively.

Furthermore, in this case, the increase in contact stiffness increases the deceleration of the tool. When the contact stiffness is low (Figure 19b), even if the deceleration of the tool is reduced, the duration of the total momentum transfer is made longer, and the robot is therefore able to markedly reduce its velocity during the contact between the tool and the object. An inelastic impact makes the dynamics of the object strictly dependent on the robot control system. Thus, the robot control system can be exploited to reduce the final velocity of the object. Figures 19c and 19d show that the total force applied on the tool by the decoupling elements is positive during the contact with

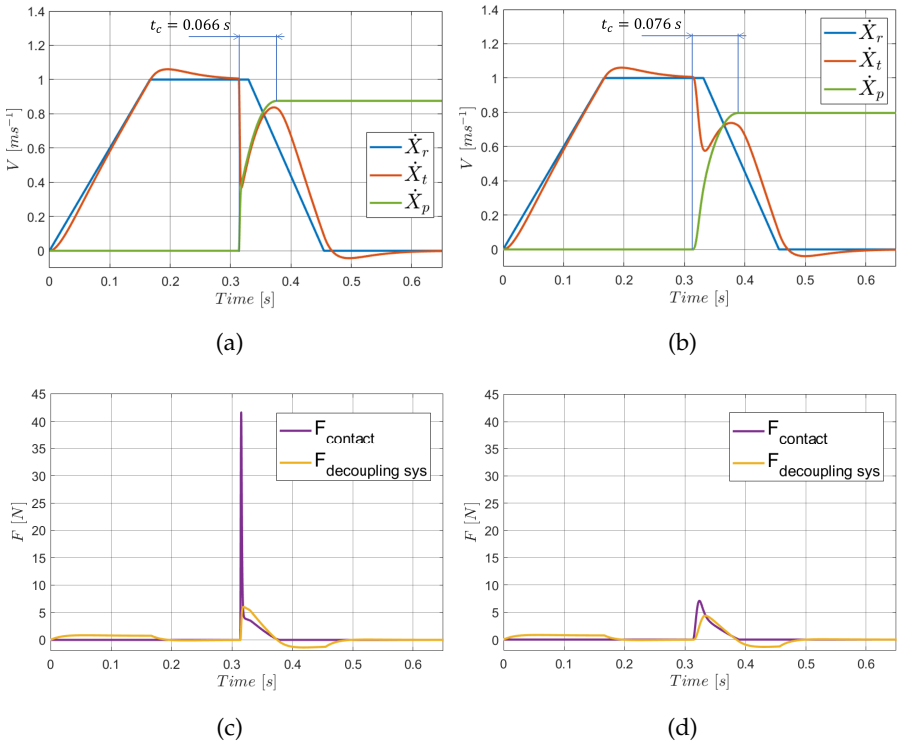


Figure 19: Velocity comparison: (**a**) $k_c = 10^5\ Nm^{-1.5}$; (**b**) $k_c = 10^3\ Nm^{-1.5}$. Contact force and total force generated by the decoupling elements: (**c**) $k_c = 10^5\ Nm^{-1.5}$; (**d**) $k_c = 10^3\ Nm^{-1.5}$. $e = 0.1$.

the object, since the decoupling system is compressed and the velocity of the tool is lower than the robot velocity. Figure 20 compares the total impulse generated by the decoupling system during the contact in the two cases analyzed in Figure 19.

It is worth noting that the larger impulse corresponds to a larger final velocity of the object, in agreement with Equation 22.

Since the impulse generated is equal to the integral of the force applied during the contact time between the tool and the object, the force profile generated by the end-effector during the contact represents a key factor in the optimization of the decoupling system.

2.4 END-EFFECTOR DESIGN

2.4.1 *A bi-stable mechanism to mitigate collisions*

In order to optimize the force profile generated by the decoupling system, non-linear solutions were analyzed to meet the following requirements:

1. The system has to be able to exert a minimum force on the tool to keep it in the desired position during the acceleration of the robot;

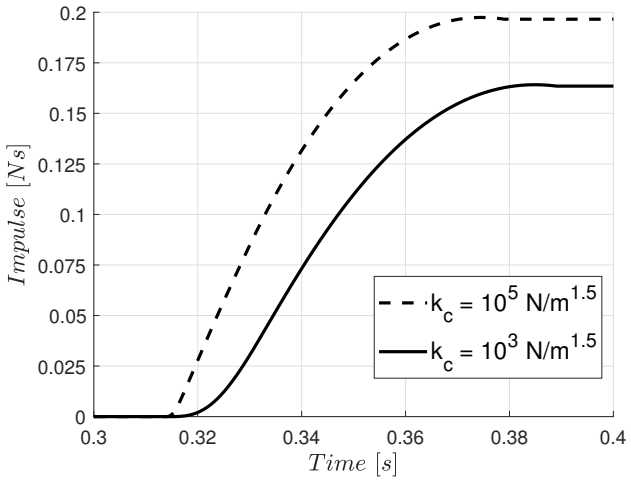


Figure 20: Impulse generated by the decoupling system.

2. The system has to prevent rebounds of the tool in order to avoid multiple collisions and the restitution of the energy absorbed in the impact;

3. The impulse transmitted to the object should be as small as possible, according to Equation 22.

The bi-stable mechanism shown in Figure 21 is considered.

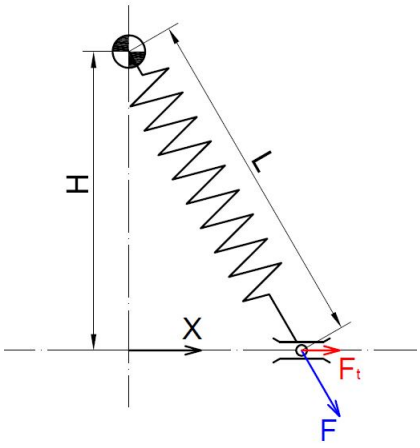


Figure 21: Schematic representation of a bi-stable mechanism with a single spring.

This system is characterized by two stable equilibrium positions and, in general, consists of a spring connected to a sliding element, which is constrained to move along the x-axis. The spring generates an elastic force on the sliding element which has two components: the first is parallel to the x-axis (F_T) and influences the dynamics of the slider; the second is orthogonal to the sliding path and does not

have important effects on the movement of the slider. The component F_t generated by the spring is given by the following equations:

$$F_t = k_b(L_0 - L)\frac{x}{L} \tag{24}$$

$$L = \sqrt{H^2 + x^2} \tag{25}$$

where k_b is the stiffness of the spring, L is its actual length, L_0 is the spring free length, x is the position of the slider relative to the vertical alignment position of the spring and H is the distance between the upper revolute joint of the spring and the sliding direction. In real applications, the slider could be constrained to move between two mechanical constraints that introduce a limit to its movement [61, 62]. The trends of the elastic force F_t as a function of the distance from the vertical alignment position are reported in Figure 22a in the case with $L_0 = 0.07$ m and $k_b = 50$ Nm^{-1}. In the same figure, the effect on the force profiles of two mechanical constraints are schematized through the two orange areas, which identify the off-limit zones. The scheme depicted in Figure 22b shows a physical representation of the mechanical constraints.

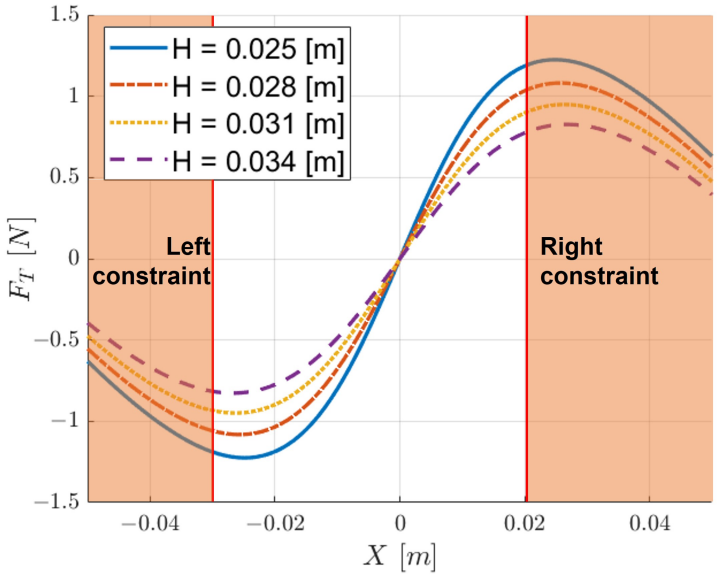
Figure 22a shows for $x > 0$ that the force is positive, whereas for $x < 0$, the force is negative. The limitation of the movement of the slider between the two mechanical constraints leads to a system that is characterized by a specific portion of the force profile. If the tool is leaning against the constraint placed on the right of the $x = 0$ position and an external force pushes it back towards the negative direction of the x-axis, the system reacts with a positive elastic force that tends to counterbalance the external force. However, when the vertical alignment position of the spring is crossed, the tool is pulled back against the left constraint and is kept there by the elastic force. The figure shows that it is possible to adjust the maximum positive and negative values of the force profiles modifying the geometrical parameter H, but also to vary the vertical alignment position of the spring in relation to the mechanical constraints. The maximum positive value of the elastic force should be large enough to keep the tool in position against the right constraint before the impact, in order to avoid oscillations, but, at the same time, it should be as small as possible to minimize the impulse transmitted to the object. The maximum negative value of the elastic force should be large enough to pull the tool back and keep it against the robot flange (left constraint).

This bi-stable mechanism was integrated in the numerical model of the end-effector. The new lumped model can be seen in Figure 23.

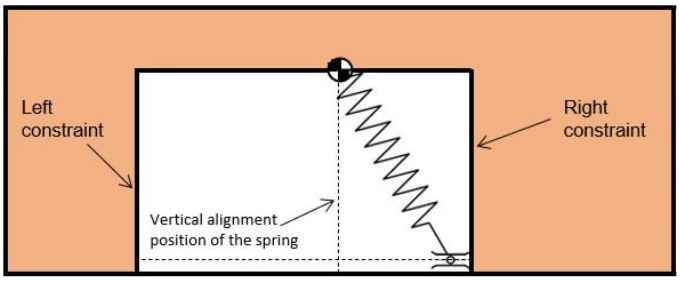
Following the new model, the equation of motion 17 and 18 becomes:

$$m_t\ddot{x}_t = F_b - k_c\delta^m - \chi_c\delta^n\dot{\delta} + F_e \tag{26}$$

$$m_p\ddot{x}_p = k_c\delta^m + \chi_c\delta^n\dot{\delta} \tag{27}$$



(a)



(b)

Figure 22: (**a**) Tangential force profiles for different values of the parameter H; (**b**) physical representation of mechanical constraints.

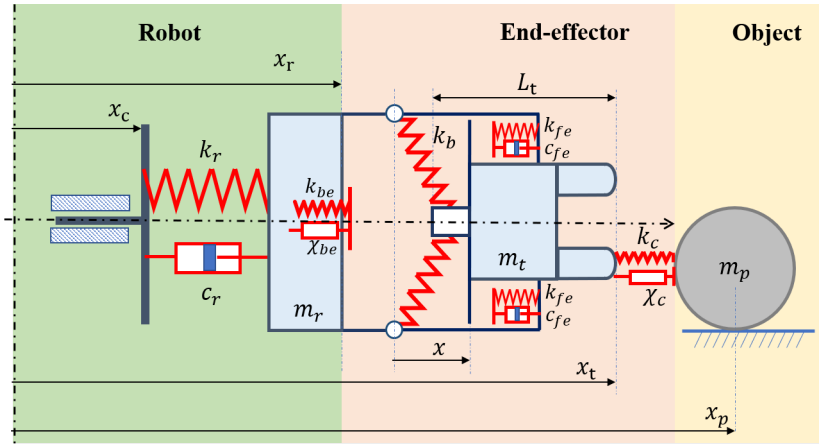


Figure 23: Lumped parameter model of the interaction between the robot and objects free to move.

where

- F_b is the force exerted by the non-linear spring in the direction of the impact. It is equal to the tangential force expressed in equation 24;
- F_e represents the contact force on the tool due to the impact with one of the two end-stops.

If the tool impacts against the front end-stop, F_e follows:

$$F_e = -k_{fe}(x - C_1) - c_{fe}\dot{x} \tag{28}$$

whereas, if it impacts against the back end-stop, F_e follows:

$$F_e = k_{be}(|x| - C_2)^m + \chi_{be}(|x| - C_2)^n|\dot{x}| \tag{29}$$

As can be inferred from the two equations above, the front end-stop follows a linear contact law, whereas the back end-stop follows a non-linear one. This choice was done to speed up the simulation. Indeed, the tool is in contact with the front end-stop before the impact, and it can oscillate due to the robot's movements. This small vibration of the tool against the front end-stop is computationally expensive, and the use of a linear contact law can save up some time. Moreover, due to the magnitude of the vibrations, the two laws return the same result.

Figure 24a shows the effect of the bi-stable mechanism on the velocities of the tool and of the object, and this is compared with the end-effector comprising a linear spring-damper system (Figure 24b). Figure 24c and 24d show the contact force and the total force generated on the tool by the end-effector. The parameters of the simulation are listed in Table 2.

Parameter	Unit	Value
H	m	0.028
e	/	0.1
k_c	$Nm^{-1.5}$	10^3
L_0	m	0.07
k_b	Nm^{-1}	50
k_t	Nm^{-1}	50
ζ	/	1
m_t	kg	0.125
m_p	kg	0.25

Table 2: Parameters used to simulate the bi-stable mechanism.

Figure 24a shows that the tool follows the movement of the robot during the acceleration phase, without oscillating, owing to the positive elastic force applied by the spring on the tool (see Figure 24c). The

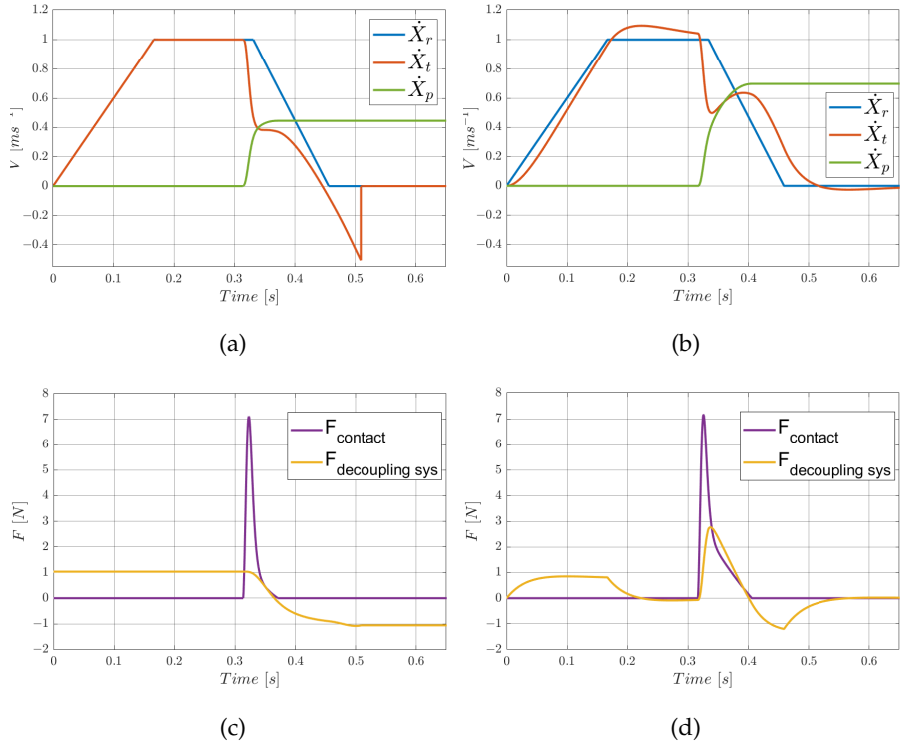


Figure 24: Velocity comparison: (**a**) end-effector with bi-stable mechanism; (**b**) end-effector with a linear spring-damper system. Contact force and total force generated by the end-effector: (**c**) bi-stable mechanism; (**d**) linear spring-damper. ($e = 0.1$).

tool is subjected to a hard deceleration during the collision, since its velocity changes nearly instantaneously from 1.0 ms^{-1} to 0.4 ms^{-1}. When the tool reaches 0.4 ms^{-1}, the effect of the collision is complete, but, when the velocity seems to stabilize, the tool crosses the vertical alignment position of the spring and is pulled back towards the robot flange. The velocity of the tool is subjected to a second rapid change. Then, the tool velocity suddenly adapts to the robot velocity when the tool hits the robot flange. Figure 24c shows that the force applied on the tool by the bi-stable mechanism is lower than the total force generated by the the spring and the damper in the end-effector with the linear behavior (Figure 24d): the force generated in the bi-stable mechanism decreases when the end-effector is compressed. Moreover, the tool is separated from the object after the collision, and this feature leads to the interruption of the momentum transfer. The comparison between the impulses generated by the bi-stable mechanism and by the linear spring-damper system is represented in Figure 25, which clearly shows that the bi-stable mechanism leads to a reduction in the impulse transmitted to the object.

In order to show the effect of the bi-stable mechanism on the velocities of the tool and of the object, Figure 24a is compared with the case in which the tool is rigidly connected to the rigid robot, which is

depicted in Figure 26. This comparison shows that the bi-stable mechanism leads to a 51% reduction in the final velocity of the object.

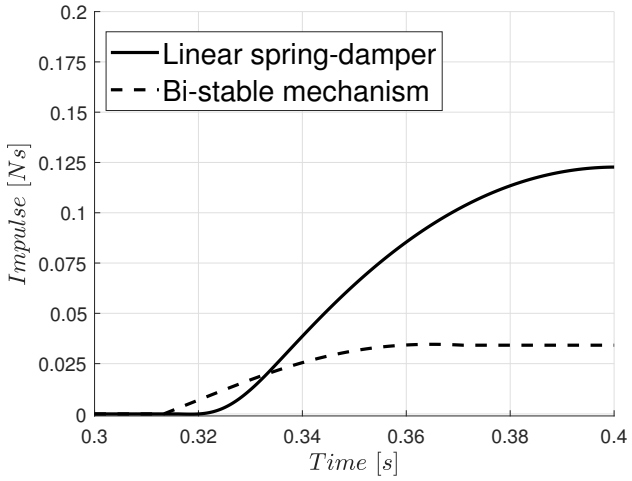


Figure 25: Impulse generated by the decoupling system.

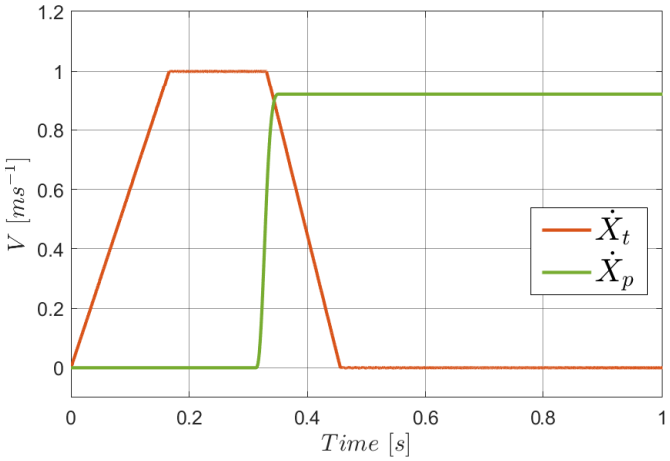


Figure 26: Tool rigidly connected to the rigid robot with $e = 0.1$, $k_c = 10^3$ Nm$^{-1.5}$, $m_t = 0.125$ kg and $m_p = 0.25$ kg.

2.4.2 *End-effector prototyping*

A prototype has been developed to validate the bi-stable mechanism design of the previous section. The task considered to design the end-effector is sorting small components positioned on a plane and free to move. Consequently, the tool design sees the bi-stable mechanism coupled with a paddle. The geometric model can be seen in Figure 27.

Four aspects drove the design of the end-effector:

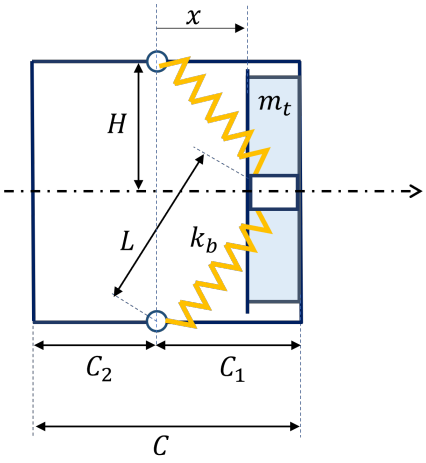


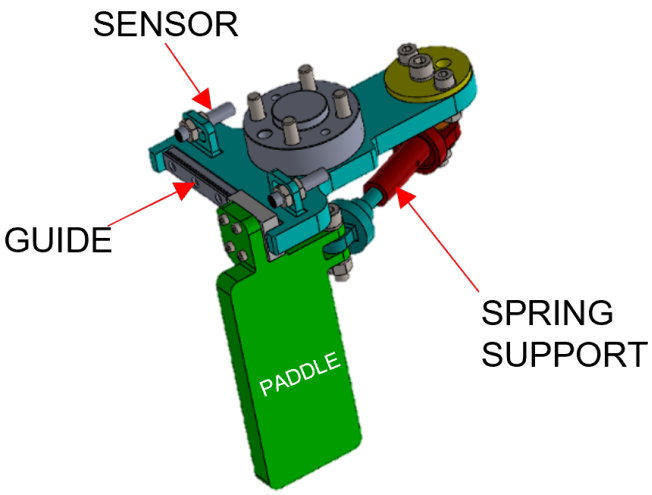
Figure 27: Geometric model of the bi-stable mechanism (vista from above).

1. since the aim is to move small objects, it must be light to reduce the momentum transferred to them;
2. the friction between the various components of the end-effector should be as low as possible;
3. it should be as close as possible to the rotation axis to allow a quick reset to a stable equilibrium position, but it should not interfere with the robot's movements;
4. it is hard to find commercial springs with low stiffness.

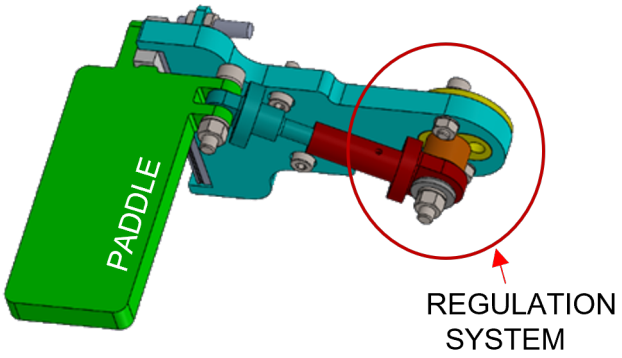
The last aspect infers that the mechanism's length should be enough to house common commercial springs. Therefore, this consideration and the third one led to the tool design shown in Figure 28. The design shows the spring placed horizontally to bring the paddle close to the rotation axis, to allow long lengths of the springs, and to avoid interferences with the joint's rotation. Indeed, if the paddle is placed vertically, a joint rotation can cause the prototype to collide with the link of the robot. In addition, the design exploits a simple way to regulate the height of the bi-stable mechanism and, consequently, the stiffness of the spring. The regulation system is shown in Figure 28b. It comprises an interchangeable 3D printed flange. This flange presents a series of holes that can be used to connect the rod that acts as a hinge for the bi-stable mechanism.

The main focus during the design process was to reduce friction and clearances between the movable parts of the mechanisms to guarantee that the sliding of the paddle was as smooth as possible. In particular, a ball-bearing guide has been used as the sliding element of the prototype. However, they can also be realized in materials with a low coefficient of friction, such as PTFE.

The second aim was to reduce the weight as much as possible. Therefore, the prototype was realized in plastic material (PLA) em-



(a)



(b)

Figure 28: Two views of the CAD model of the proposed prototype.

ploying additive manufacturing. However, this process brought some downsides. Indeed, the small Young's modulus of the PLA can introduce uncontrollable compliance in the system. Consequently, the components must be thick enough to have a reasonable stiffness. Moreover, it is hard to obtain small dimensional and geometric tolerances; generally, the surfaces can be very rough. Since both aspects increase the friction and stick-slip phenomena, the contact PLA-to-PLA was reduced to the minimum and restricted to the hinges of the spring support. The mathematical model must consider this added friction. Equations 17 and 18 can be updated as follows:

$$\begin{cases} m_t \ddot{x}_t = F_b - k_c \delta^m - \chi \delta^n \dot{\delta} + F_e + F_{f_{tool}} \\ m_p \ddot{x}_p = k_c \delta^m + \chi \delta^n \dot{\delta} + F_{f_{floor}} \end{cases} \tag{30}$$

where $F_{f_{tool}}$ is the friction force acting on the sliding component of the paddle, and $F_{f_{floor}}$ is the one on the object due to the plane friction. $F_{f_{floor}}$ is dependent on the object velocity and mass, and on the friction coefficient of the plane on which it moves, whereas $F_{f_{tool}}$ can be defined as follow:

$$F_{f_{tool}} = \mu F_n \tanh\left(\frac{\dot{x}_r - \dot{x}_t}{v_0}\right) \tag{31}$$

From Figure 21 and Equation 24, F_n is defined as:

$$F_n = N k_b \tanh(L_0 - L)\frac{H}{L} \tag{32}$$

where N is the number of springs used.
The final prototype can be seen in Figure 29.

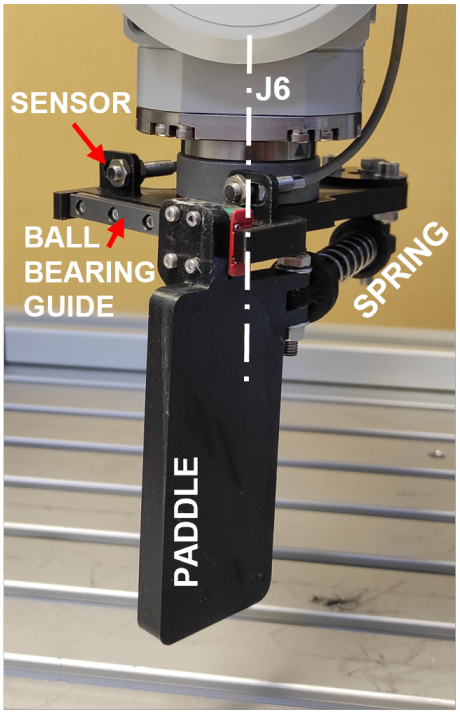


Figure 29: 3D printed prototype

In conclusion, as can be deducted from Figure 29, the impact is detected by means of an inductive sensor positioned in correspondence of the front end-stop. The sensor detects the metallic surface of the cursor of the guide. Therefore, it is on until an impact occurs. After a certain displacement of the paddle, the sensor turns off, detecting a hazardous impact. In the prototype shown in Figure 29 that displacement is equal to 1.5mm from the front end-stop.

2.5 EXPERIMENTAL VALIDATION

Experimental validation of the mathematical model was carried out by means of the prototype exposed in the previous section.

The experimental setup is constituted of a MITSUBISHI RV-4FRL-D robot, a linear guide, and a high frame rate camera, as shown in Figures 30 and 31.

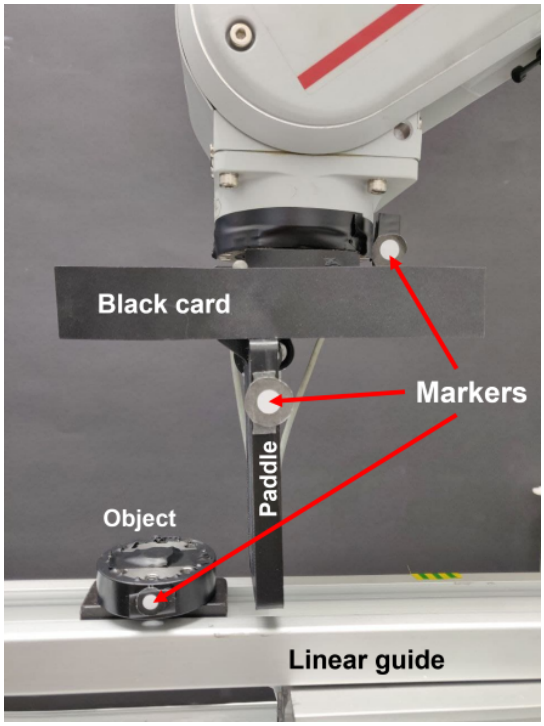


Figure 30: Experimental setup for the validation of the mathematical model

The robot moves linearly along an axis following a trapezoidal velocity profile. It moves from its home position to a specified location after the object. The robot is planned to collide with the object while moving at a constant velocity of $1\ ms^{-1}$. The object is cylindrical to ensure repeatable contact with the paddle. Indeed, a surface-to-surface contact is impossible to carry out experimentally, and there would be no certainty about the actual contact area. Moreover, the object is mounted on the cursor of a linear guide to limit its movement to one direction. This limitation will intrinsically increment the friction on the object's movement, but it will avoid projection errors on the measure of its position.

The tracking is performed utilizing three (3) markers disposed on the three moving components of the system (the robot, the tool, and the object), as shown in Figure 30. The markers are tracked at $300fps$ using the high frame rate camera (TELEDYNE DALSA - Genie Nano GigE M640) shown in Figure 31.

Before the start of the data collection, a calibration process must occur. The calibration estimates the parameters of the camera lens that will be used to remove the effects of lens distortion from an image. It is performed in MATLAB using a standard procedure. The procedure detects a well-defined pattern (i.e., a checkerboard) with known dimensions and estimates the camera parameters. Therefore,

Figure 31: Camera used during the experiment: Genie Nano GigE M640, TELEDYNE DALSA.

a checkerboard is positioned in front of the camera as distant as the markers during the tests. Finally, a series of images are taken with the camera. Figure 32 shows an image used for the calibration.

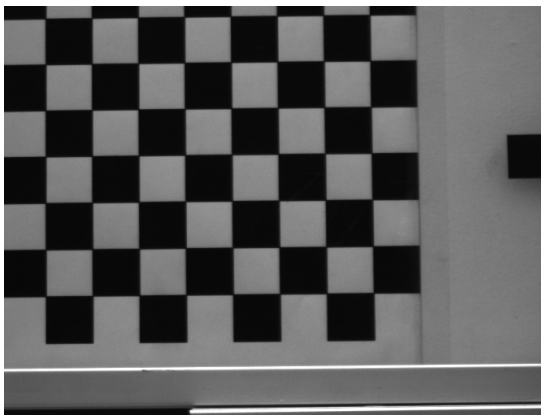


Figure 32: Photo taken by the camera showing the checkerboard necessary for the calibration.

After the calibration procedure, the task starts. The video processing consists of the binarization of every single frame captured by the camera, the detection of the markers, and the determination of their centers. Figure 33 shows the binarization result of a single frame. The threshold was selected by hand since it depended on the amount of light in the scene. Moreover, the frame images were zeroed out above the robot's marker and below the object's marker to delete the robot and the sliding guide.

The markers are identified employing the *regionprops* function in MATLAB. This function detects the white areas in the binarized image and gives the centroid of those areas in the output. The positions of their centers are saved and derived to estimate the velocity of the three components.

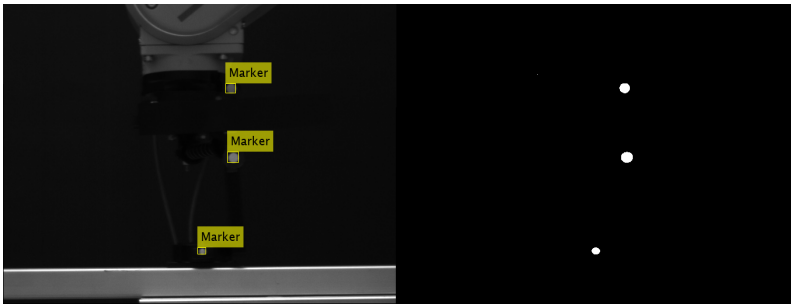


Figure 33: Frame captured from the camera and consequent binarization showing the isolated markers

The tests were carried out using a fixed impact velocity of $1\ ms^{-1}$. The delay in the communication channel, equal to about 60 ms, was measured from the experimental data. The sources of this delay are related to:

1. the scan cycle time of the protocol of the controller connected to the sensors;
2. the time necessary for MATLAB to receive the event from the controller and analyze it;
3. the time necessary for the robot controller to receive the new command from MATLAB.

These three sources have been addressed to reduce the delay as much as possible. In particular, the controller chosen is a BECKHOFF master/slave hardware that, thanks to the EtherCAT protocol, can ensure a scan cycle time of 100 µs. Even if it can go faster, tests have shown that the reception of the signals from MATLAB would be deterministic only above that cycle time. Moreover, the computer used in the tests crashed using faster cycle times. The other two sources have been addressed simplifying and optimizing the code.

Figure 34 shows some of the velocity plots resulting from video processing.

The plots show the velocities of the three markers in 2 of the conditions tested. In particular:

- Figures 34a and 34c are the results with an object of mass $m_{obj} = 136gr$, whereas Figures 34b and 34d use a mass $m_{obj} = 246gr$;
- Figures 34a and 34b are the results with a deceleration of the robot $d_{robot} = 5ms^{-2}$, whereas Figures 34c and 34d use a deceleration $d_{robot} = 9ms^{-2}$.

From the plots, it can be inferred that the deceleration of the robot does not influence the object's final velocity. This fact is consequent to

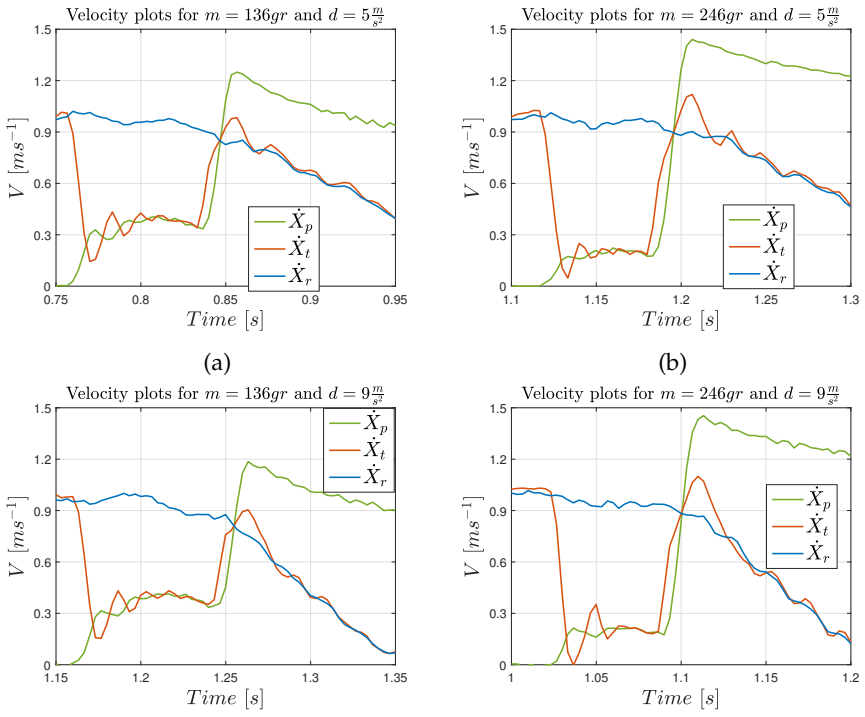


Figure 34: Velocity plot in different conditions: (**a**) $m_p = 136gr$, $d_{robot} = 5ms^{-2}$; (**b**) $m_p = 246gr$, $d_{robot} = 5ms^{-2}$; (**c**) $m_p = 136gr$, $d_{robot} = 9ms^{-2}$; (**d**) $m_p = 246gr$, $d_{robot} = 9ms^{-2}$

the high velocity of the object that detached it from the tool before the deceleration begins. With a lower delay, the deceleration should have a higher effect since it could prevent multiple impacts and reduce the amount of momentum transferred during those impacts.

It is also worth noticing that the impact dynamics last around 0.1 s. Thus, the camera detects around 30 points for every impact. This factor can affect the peak values.

A series of tests has been performed under different properties to perform the validation. In particular, the main tests were carried out using the values shown in Table 3. Each test results from the combination of the possible choices for each variable.

m_{obj}	d_{robot}	H
[kg]	$[ms^{-2}]$	[mm]
0.136	5	45
0.246	7	53
0.336	9	

Table 3: Variable values used in the tests

The tests were carried out using one spring with $k_b = 490 N m^{-1}$. Recalling Figure 21, H is the height of the bi-stable mechanism. Modifying it will change the overall stiffness of the spring.

A qualitative match between the experimental and the analytical results is sought to validate the models. The simulations were carried out using the parameters in Table 4. Some results are shown in Figure 35.

Parameter	Unit	Value
N	/	1
L_0	m	0.062
k_b	$N m^{-1}$	490
C	m	0.047
$C_1 = C_2$	m	0.0235
V_c	$m s^{-1}$	1
a_c	$m s^{-2}$	5
m_t	kg	0.048
k_c	$N m^{-1.5}$	$0.6 \cdot 10^5$
e_c	/	0.8
k_{be}	$N m^{-1.5}$	$1 \cdot 10^6$
e_{be}	/	0.3
k_{fe}	$N m^{-1}$	$1 \cdot 10^6$
c_{fe}	$N s m^{-1}$	235
DT	s	$60 \cdot 10^{-3}$
μ	/	0.23
v_0	$m s^{-1}$	0

Table 4: Values of the parameters used in the simulations for the matching with the real system

Figures 35e and 35f show a test performed using a rigid tool to show the effectiveness in the reduction of the momentum transferred to the objects. These tests have been carried out using the tool shown in Figure 36. Furthermore, since the tool and the robot are rigidly fastened, a single marker attached to the tool is used to track their movements. Therefore, the oscillations visible in the blue line of plot 35e are related to the tool's flexibility consequent to its length. This factor is not considered in the model.

The comparison in Figure 35 shows that the simulated model is able to describe qualitatively and quantitatively the results obtained experimentally. For example, considering the plots in Figures 35a and 35b, the model correctly predicts the number of collisions with the object, the various rebounds of the tool, and almost all the peak val-

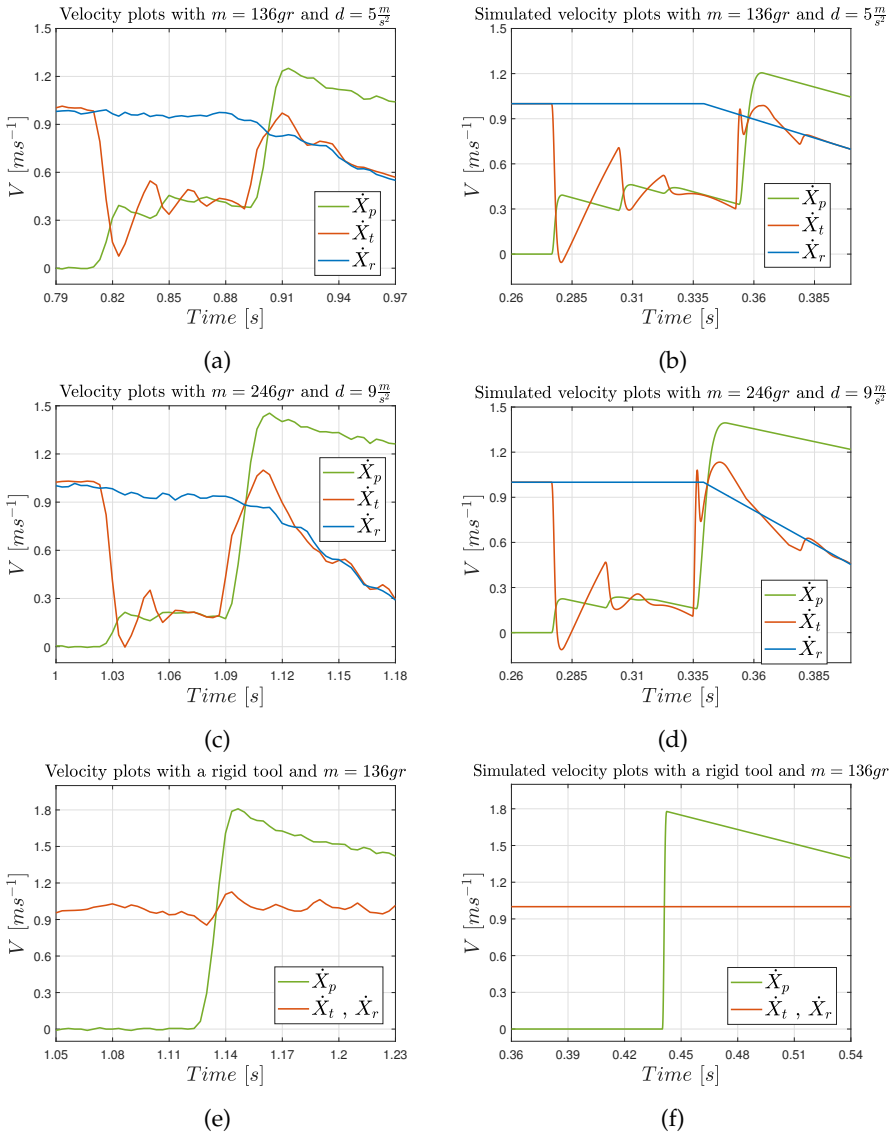
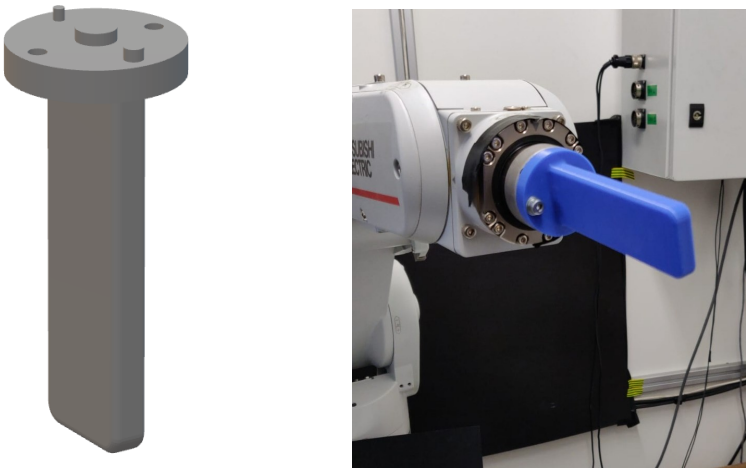


Figure 35: Matching between velocity plots obtained experimentally and in simulation in different conditions: (**a**,**b**) $m_p = 136gr$, $d_{robot} = 5ms^{-2}$; (**c**,**d**) $m_p = 246gr$, $d_{robot} = 9ms^{-2}$; (**e**,**f**) rigid tool with $m_p = 136gr$

ues of the velocities. However, it is worth stating that the peaks of the experimental velocities can be approximated due to the sampling frequency. For example, a statistic analysis for the tests highlighted in Figure 35 returns the following results for the peak velocities of the object after the impact: $1.231 \pm 0.054ms^{-1}$ for $m_p = 0.136kg$, $1.430 \pm 0.040ms^{-1}$ for $m_p = 0.246kg$, and $1.798 \pm 0.037ms^{-1}$ for the rigid tool and $m_p = 0.136gr$.



(a) (b)

Figure 36: Rigid tool realized to evaluate the effectiveness in the reduction of the momentum: (**a**) CAD model; (**b**) 3D printed model mounted on the robot.

2.6 CONCLUSIONS

This section presented a study of 1D impacts that resulted in the design of a novel end-effector that could reduce the momentum transmitted to the objects with which the robot impacts. The fundamental analysis carried out with the classical impact theory highlights that reductions in the mass of the tool and in the restitution coefficient result in significant decreases in the final velocity of the object. A more detailed model of contact mechanics, taking into account contact stiffness and damping, confirms the previous results. Moreover, the model highlights that a decrease in contact stiffness leads to an increase in contact duration, which can be exploited by the robot control system, which can start braking during the contact with a reduction in the momentum transferred to the object. Reductions in the restitution coefficient and in contact stiffness can be achieved by means of the proper selection of the material of the surface of the tool that impacts the object.
The reduction in the mass of the tool is a more complex problem. Numerical simulations have shown that a relevant share of the mass of the robot moves together with the tool in the approach direction and that the intrinsic compliance of robot joints has a small effect on impact dynamics. Therefore, a proper elastic system that decouples the tool from the robot is needed to minimize the impacting mass. The stiffness of the elastic system has to be sufficient to keep the tool in the desired position when the robot accelerates towards the object. Rebounds of the tool, which give back energy to the object and cause multiple collisions, have to be avoided. These requirements are met by the proposed bi-stable mechanism, which can reduce the final ve-

locity of the object by about 50% with respect to the case of a rigid connection between the robot and the tool. The calculated results refer to a tool moving at a velocity of $1\ ms^{-1}$ and impacting a stationary object. The extension of the results to different velocities is complex due to the non-linearities of the system. Some simulations showed that a decrease in the impact velocity may prevent the bi-stable mechanism from reaching the second stable configuration. Conversely, a small increase in the impact velocity does not significantly change the behavior of the system. Therefore, the bi-stable mechanism has to be tuned to the impact velocity.
The results obtained were used to design a prototype of the end-effector with a bi-stable mechanism. The prototype was realized in additive manufacturing to be lightweight. The prototype was experimentally tested to validate the effectiveness of the system regarding the reduction of the momentum transmitted to the object. The model is shown to correctly predict the behavior of the bi-stable mechanism and of the object both qualitatively and quantitatively.
In conclusion, the sorting task studied in this section is very specific. In the next section, an extension of the study is proposed to address the case of collisions with hard surfaces. Those cases are more common and allow the creation of a module to be interposed between the robot flange and the tool to mitigate the collisions in a more general case.

3

MITIGATE COLLISIONS BETWEEN ROBOTS AND OBSTACLES

This Chapter presents an implementation of the study carried out in Chapter 2. Previously, collisions with small objects that are free to move were considered. This Chapter extends the study to consider collisions with fixed obstacles or hard surfaces. This case is more general than the previous one and more severe regarding forces at play and consequent damage to the system.

3.1 HARD IMPACTS WITH FIXED OBSTACLES

In robotics, the risk of collisions with hard surfaces is an important issue when the robot does not carry out repetitive tasks and moves in unknown environments [63]. Methods and algorithms have been developed to create, for example, collision-free paths also in cluttered environments [64] or to execute tasks requiring a controlled contact force [65]. However, independently from the considered task, if those collisions can happen, the velocity of the robot must be limited, and the efficiency of the system will be penalized. Furthermore, as shown in the previous Chapter, the dead time necessary to detect and react to collisions worsens the scenario because no action is taken during this time.

In addition, in the case of teleoperation tasks, hard collisions can generate instability in the control system that can have disruptive consequences (both for the robot and the system on which it acts) [66]. The study of stability is well documented in the literature [67], and various algorithms have been developed to guarantee optimal behavior and reduce the risk of instabilities [42]. These solutions generally work at the control level exploiting a mix of impedance and admittance systems [43].

From a mechanical point of view, many researchers have developed technologies for the mitigation of the effect of hard collisions, developing compliant coverings of the links, back-drivable transmissions, elastic transmissions that are able to decouple joint actuator inertia from link inertia [68], and end- effector air-bags [69].

Mechanisms to mitigate collisions at the end-effector level can be found on the market in the form of compensation units that are placed between the end effector and the robot flange [70]. However, they compensate for imprecise tolerances or small misalignments in the position of the objects to be picked up. Therefore, their movement is limited.

Other examples regard the field of soft robotics [71]. Indeed, the products manipulated in a flexible work cell can present different dimensions or features, and deploying tools with rigid joints and links can lead to a collision. Unless very carefully controlled, these collisions can damage the object or push it out of the desired path. In response, soft grippers present an added compliance given by a particular design or by employing elastic materials that deform under external perturbations. Moreover, the mechanism's stiffness can be manipulated accordingly to the object enabling the same tool to manage very different objects [72].

The study presented in this Chapter integrates the one in Chapter 2. Therefore, a mechanical approach is employed to study hard impacts at the end-effector level. The previous study shows that the impact can be mitigated by exploiting a non-linear decoupling system to delay the instant the robot starts to push the impacted obstacle directly. Consequently, the decoupled movement of the tool masks the dead-time in collision detection and reaction. These findings are implemented in this Chapter considering impacts with fixed obstacles. As with moving objects, hard contact with surfaces will be softened, and the robot will have the time to stop or, at least, slow down and reduce the effects of impacts. Consequently, this solution eliminates extreme limitations of the velocity of the robot, and the overall performance can be increased.

It is worth stating that impacting hard surfaces can be sought for some specific applications. For example, in [73], the stability of an aerial teleoperated robot is studied during controlled contact with a surface. The task requires pushing buttons or keeping the UAV in contact with the surface. Another example is approaching a surface to execute robotic deburring [47].

The introduction of passive compliance in a robotic system has to cope with two opposite requirements: on the one hand, compliance has to mitigate the consequences of unexpected collisions between the end-effector and an obstacle; on the other hand, the robot has to be stiff enough to follow a specific trajectory or perform the task. Therefore, the end-effector has to be characterized by a variable stiffness able to exert the necessary force on the environment before yielding. Consequently, the specifics of the end-effector must respond to these requirements. For example, a linear spring embedded between the robot flange and the end-effector represents a passive compliant mechanical system that increases compliance and reduces the reflected inertia at the interaction between the end-effector of the robot and the environment. However, the compliance and the equilibrium position cannot be adjusted when linear springs are selected. On the other hand, using non-linear springs allows control of both the compliance and the equilibrium position [74, 75].

The bi-stable mechanism, proposed in the previous Chapter, represents a passive adjustable compliant system that acts as a non-linear spring and introduces the following advantages over the linear spring:

- the force applied to the obstacle during the interaction gradually decreases when the compression of the non-linear spring increases. This property prevents overloading the robot and the obstacle when the pre-load is exceeded;
- the second position of stability of the bi-stable mechanism allows detaching the end-effector from the obstacle, interrupting the interaction. This property is critical when the interaction has to be interrupted as soon as possible to preserve the integrity of the obstacle or the end-effector.

This Chapter presents a validation of the new model using the bi-stable prototype used in Chapter 2. Nevertheless, a new prototype has been designed and validated to extend the generality of the bi-stable mechanism. The new model exploits the bi-stable mechanism inside a module to be interposed between the robot flange and the tool. Finally, an extension of the bi-stable mechanism in 2D is proposed and validated experimentally. This solution couples the bi-stable mechanism with a rotational device to align it along the direction of the impact.

The main limitation of the proposed end-effectors is that they are not able to protect the robot from possible collisions of the links with obstacles in the workspace.

3.1.1 *Mathematical model*

The mathematical model proposed in Chapter 2 considered impacts between the robot and objects disposed on a plane and free to move. The model is now extended to estimate the effects of accidental collisions with hard surfaces on the end-effector and the robot. The mathematical model simulates the behavior of the robotic system in the approach direction. The complete model, represented in Figure 37, consists of three sub-models: the robot, the end-effector, and the obstacle. The analysis assumes a one-dimensional model of the collision. Hence, only planar movements of the robot are considered. In these hypotheses, the multi-degree of freedom (DOF) model of the robot is reduced to a one-dimensional model, represented by a base excited mass-spring-damper system. The resulting model considers the inertia and compliance of the robot along the approach direction.
As stated in Chapter 2, the compliance of the robot is engaged when the robot impacts heavy objects. Since hard surfaces, such as walls, can be seen as an object with infinite weight, the current model considers the compliance of the robot.

The mechanical properties of the robot were identified by means of impulsive modal analysis [56, 76], then, the measured properties were reduced to the trajectory of the robot flange in the Cartesian space [38].

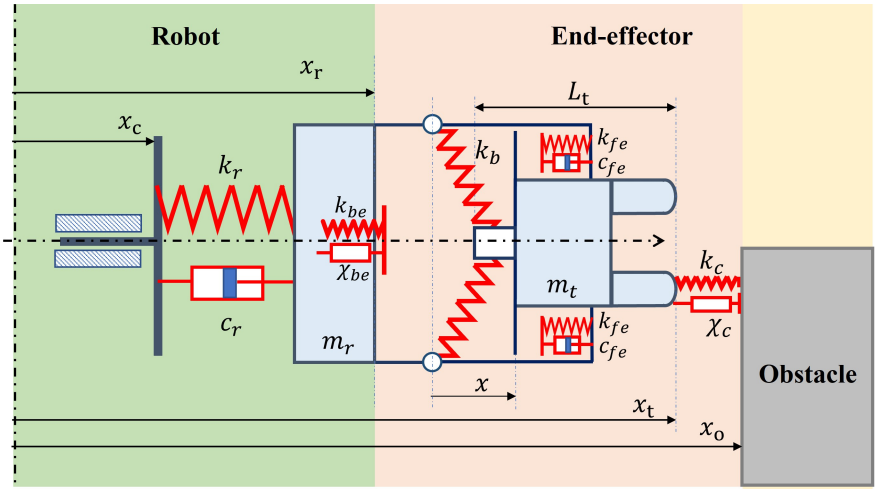


Figure 37: Lumped parameter model of the interaction between the robot, the end-effector and the obstacle

The lumped parameters m_r, k_r and c_r represent the reduced mass, stiffness and damping of the robot, respectively. Coordinate x_c represents the motion law of the robot flange imposed by the control system, i.e., the motion of the robot flange which is generated by the robot with ideal motors, without compliances and clearances. The coordinate x_r represents the actual motion of the robot flange, accounting for the inertia and the compliance of the robot. The end-effector, schematized through the mass m_t, is connected to the robot flange by means of a bi-stable mechanism, as proposed in Chapter 2.

Bi-stable mechanisms are widely used in micromechanics, space and energy harvesting applications for their unique properties [77], [78], [79]. Bi-stable mechanisms are mechanical systems equipped with flexible elements (springs, membranes or foils) which have two stable equilibrium positions separated by an un-stable equilibrium position. When the mechanism in a stable equilibrium position is stimulated by an external force, first the elastic elements deflect and store energy, then, after the un-stable equilibrium position is reached, the stored energy is released and the system is attracted towards the second stable equilibrium position.

The bi-stable mechanism considered in this analysis is represented in Figure 38. It is a chevron-type mechanism [77] and is equipped with two simple linear springs connected to a sliding element, which is constrained to move along the x-axis.

The component F_b of the force generated by the springs of the bi-stable mechanism in the sliding direction is calculated using Equation 24.

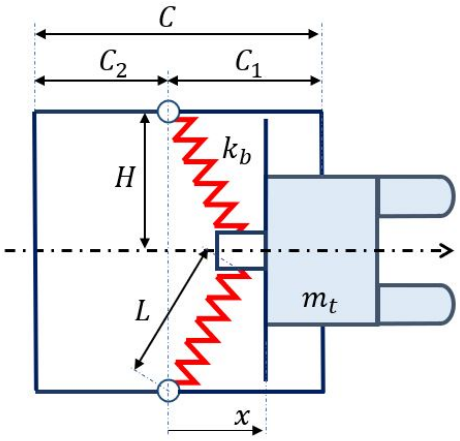


Figure 38: Geometric model of the bi-stable mechanism. C is the total stroke. In the first part (C_1) of the total stroke the bi-stable mechanism generates a positive force, in the second part (C_2) the bi-stable mechanism generates a negative force

In the configuration of Figure 37 and 38 the spring forces perpendicular to sliding direction cancels out and no net force is transmitted to the robot in this direction.

The geometric parameters assumed for the bi-stable mechanism are listed in Table 5.

Parameter	Unit	Value
N	/	2
k_b	Nm^{-1}	200
H	m	0.03
C	m	0.03
C_1	m	0.015
C_2	m	0.015
L_0	m	0.05

Table 5: Reference values of the parameters of the bi-stable mechanism (Figure 38)

The impact between the mass m_t of the end-effector and the obstacle is schematized by means of an elastic element and a damping element in parallel, see Figure 37. The contact force depends on the deformation $\delta = (x_t - x_o)$ and the deformation velocity $\dot{\delta} = \dot{x}_t$, where x_t is the coordinate of the end-effector and x_o is the position of the obstacle in the cartesian space.

The contact force is formulated according to Equation 5. The equations of motion of robot and end-effector during the collision are:

$$m_r\ddot{x}_r = c_r(\dot{x}_c - \dot{x}_r) + k_r(x_c - x_r) - F_b \tag{33}$$

$$m_t\ddot{x}_t = F_b - k_c\delta^m - \chi_c\delta^n\dot{\delta} \tag{34}$$

They are non-linear owing to the formulations of the contact force and of mechanism force. Equation 33 shows that the force exerted on the robot is F_b , which can be calculated from equation 34 that represents the dynamic equilibrium of the end-effector.

The movement of the slider of the end-effector, inside the bi-stable mechanism, is restricted by the robot flange on one side and by a mechanical constraint on the other side. In order to simulate the contact between the slider and these end-stops, each constraint is modelled by means of an elastic element in parallel with a damping element. The end-stop on the robot flange uses the same formulation of Equation 5. It activates when $|x| > C_2$ and physically represents a layer of elastomeric material with equivalent stiffness k_{be} and coefficient of restitution e_{be}. The other end-stop is modelled using a simple linear spring-damper system. This approach reduces the computational effort of the simulation. This end-stop consists in a layer of elastomeric material, with an equivalent contact stiffness k_{fe} and damping coefficient c_{fe} (Figure 37). It activates when $x > C_1$.

When an end-stop is impacted the equations of motion become:

$$m_r\ddot{x}_r = c_r(\dot{x}_c - \dot{x}_r) + k_r(x_c - x_r) - F_b - F_e \tag{35}$$

$$m_t\ddot{x}_t = F_b - k_c\delta^m - \chi_c\delta^n\dot{\delta} + F_e \tag{36}$$

where F_e represents the contact force due to the impact with an end-stop. F_e is given by:

$$F_e = k_{be}(|x| - C_2)^m + \chi_{be}(|x| - C_2)^n|\dot{x}| \tag{37}$$

when the end-effector impacts with the back end-stop. F_e is given by:

$$F_e = -k_{fe}(x - C_1) - c_{fe}\,\dot{x} \tag{38}$$

when the end-effector impacts with the forward end-stop.

3.1.2 *Numerical Simulations*

The mathematical model of Figure 37 was implemented in MATLAB and Simulink, which is suited to carry out non-linear simulations. Simulations are aimed to verify that the bi-stable mechanism is able to mitigate the contact force and the force on the robot when an accidental collision occurs. To verify the effectiveness of the bi-stable mechanism, two cases are considered:

1. the end-effector is rigidly fastened on the robot flange. In this case the bi-stable mechanism is not adopted and the mass m_t of the end-effector is added to the reduced mass m_r of the robot;

2. the bi-stable mechanism is mounted between the end-effector and the robot flange.

The robot considered in this analysis is the Omron Adept Viper s650. Simulations assume a generic operation of the robot, in which the end-effector is initially stationary and moves towards a target position along a linear trajectory. A trapezoidal velocity profile with acceleration a_c and deceleration d_c is assumed. An accidental collision occurs between the end-effector and an obstacle, when the robot is moving at a constant velocity V_c.

It is assumed that the robot stops when the impact with the obstacle is detected. The impact sensor of the end-effector equipped with the bi-stable mechanism is an inductive sensor. This sensor switches on when the end-effector moves of 5 mm from the forward end-stop. When the end effector is rigidly fastened to the robot, the collision is detected by means of a force sensor embedded in the end-effector, which switches on when the force overcomes 25 N.

When the interaction between the end-effector and the obstacle is analyzed, three events take place: the collision, in which the contact begins; the detection of the collision, when the impact detector is switched on; the reaction, when the robot brakes. It is assumed that the reaction does not coincide with the detection, since the time needed by the robot control system to process the external input from the sensor is considered. External inputs are updated at every *scan cycle*, which is a specific feature of the robot. In the worst case, when the input signal switches on after being requested by the control system, it is necessary to wait a full scan cycle to detect it. Hence, the scan cycle represents the maximum time needed to process the external signal. The difference between the detection and the reaction is defined as *dead time DT*. In this analysis the dead time is assumed coincident with the scan cycle of the robot. The difference between the instant in which the collision begins and the instant in which the robot brakes is defined as *reaction time* t_r.

The values of the parameters adopted in the simulations are listed in Table 6.

The configuration $\mathbf{q}$ of the robot in the joint space just before the collision and the related reduced mass m_r, stiffness k_r and damping c_r are reported in Table 7.

Figure 39a represents the velocity profiles $\dot{x}_c$, $\dot{x}_r$ and $\dot{x}_t$ when the end-effector is rigidly fastened on the robot flange. Figure 39b represents the corresponding contact force generated in the collision between the end-effector and the obstacle. For the sake of simplicity, the contact force applied on the end-effector is represented in Figure 39b

Parameter	Unit	Value
a_c	ms^{-2}	1
d_c	ms^{-2}	2
m_t	kg	0.75
k_c	$Nm^{-1.5}$	10^8
e_c	/	0.65
k_{be}	$Nm^{-1.5}$	10^6
e_{be}	/	0.6
k_{fe}	Nm^{-1}	10^6
c_{fe}	Nsm^{-1}	278
DT	s	$16 \cdot 10^{-3}$

Table 6: Reference values of the parameters assumed in the simulations

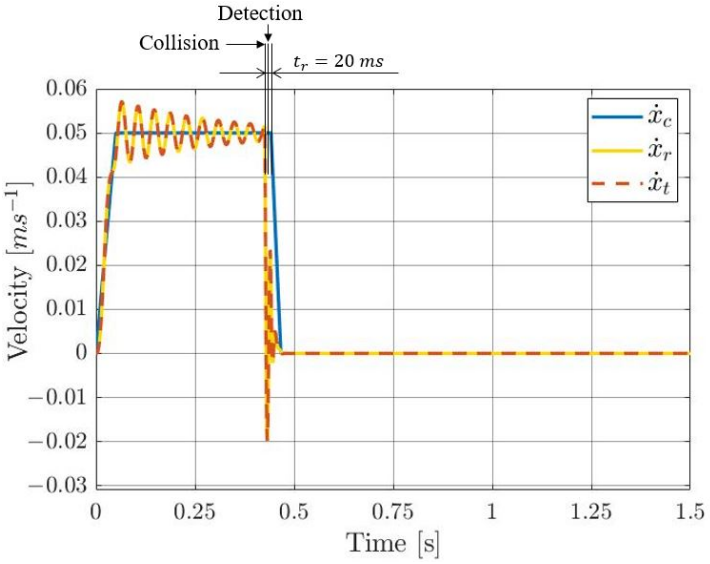
Parameter	Unit	Value
$\mathbf{q}$	°	[0; -87; -105; 0; -78; 0]
m_r	kg	8.4
k_r	Nm^{-1}	$2.3 \cdot 10^5$
c_r	Nsm^{-1}	71

Table 7: Values of the parameters correlated to the robot configuration $\mathbf{q}$ in the joint space just before the collision

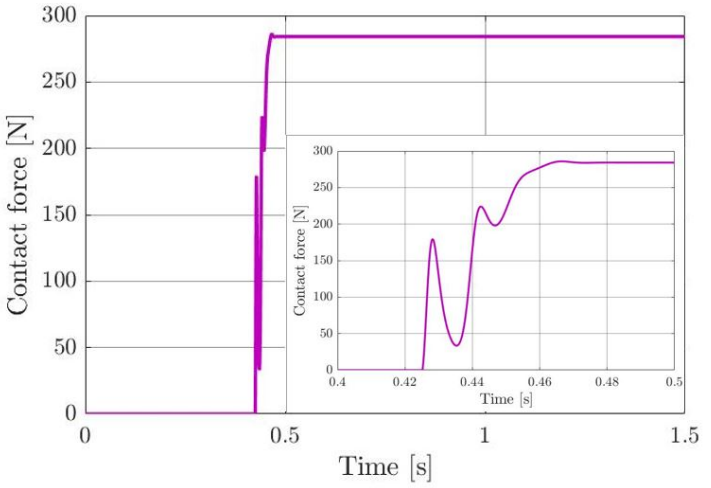
as a positive force, although it is negative in the mathematical description. It is worth noticing that, in this case, the contact force coincides with the force which directly excites the robot. Robot flange velocity is $V_c = 0.05\ ms^{-1}$.

The velocity of the robot mass is characterized by a damped oscillation when the robot reaches V_c, since the spring and the damper, which represent the mechanical properties of the robot, are compressed during the acceleration phase. The collision between the end-effector and the obstacle causes the sudden deceleration of the robot mass. However, since the reaction time of the robot is not fast enough, the motors of the robot keep running. The collision, detection and reaction of the robot are highlighted in Figure 39a. The detection occurs in 4 ms, whereas the reaction time t_r is 20 ms. The contact force profile in Figure 39b reveals that the robot deforms, as the contact force increases far beyond the force threshold of 25 N.

Figure 40a represents the velocity profiles $\dot{x}_c$, $\dot{x}_r$ and $\dot{x}_t$ when the bistable mechanism is adopted. Figure 40b represents the contact force generated in the collision between the end-effector and the obstacle, and the force applied on the robot. For the sake of simplicity, the



(a)

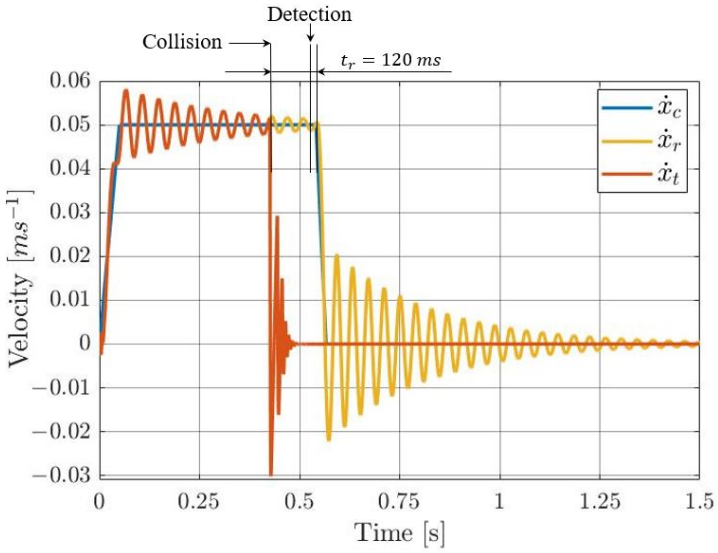


(b)

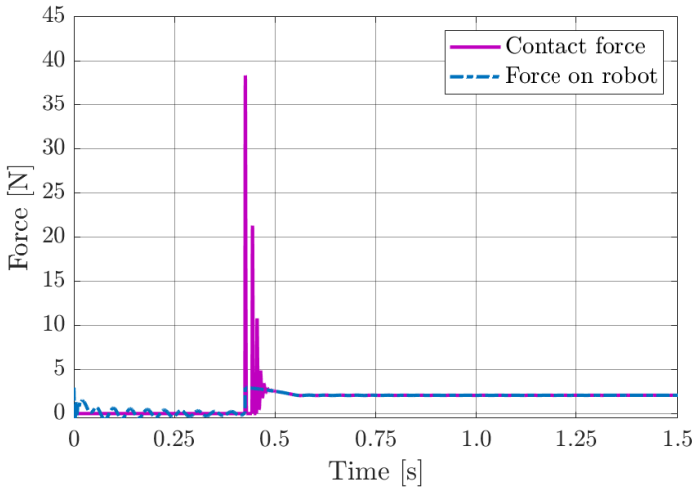
Figure 39: End-effector rigidly fastened on the robot flange: (a) velocity profiles; (b) contact force profile. Parameters of the simulation are listed in Table 6

contact force and the force on the robot are represented in Figure 40b as positive forces, although they are negative in the mathematical description. It is worth noticing that in this case the contact force does not coincide with the force which directly excites the robot. The parameters of the simulation are listed in Table 5 and 6. Robot flange velocity is $V_c = 0.05\ ms^{-1}$.

Figure 40a highlights that the force generated by the bi-stable mechanism is enough to counterbalance the force of inertia acting on the end-effector and, before the collision, the end-effector moves with the robot. The collision causes a sudden deceleration of the end-effector, while the robot mass is moving forward smoothly. It is worth notic-



(a)



(b)

Figure 40: End-effector with bi-stable mechanism: (a) velocity profiles; (b) profiles of the contact force and the force on the robot. Parameters of the simulation are listed in Table 5 and 6

ing that the reaction time t_r of the robot with the bi-stable mechanism ($t_r = 120$ ms) is longer than the one of the robot with rigidly fastened end-effector ($t_r = 20$ ms). Actually, the end-effector takes more time to move of 5 mm from the forward end-stop than the contact force to rise up to 25 N. The time needed to detect the collision is an important parameter that has to be taken into account in the design of such mechanisms. The increase in the reaction time requires larger decoupling distance, hence a longer stroke of the mechanism. In Figure 40a the bi-stable mechanism allows to decouple the movement of the robot from the movement of the end-effector, because the robot mass does not change its velocity during collision, but only when

the impact is detected and the robot brakes. After braking the robot oscillates because it is not in direct contact with the obstacle.

Figure 40b highlights that the end-effector bounces many times on the obstacle. The robot equipped with the bi-stable mechanism reduces the contact force by about 85%.

The decoupling between the end-effector and the robot absorbs the collision, preventing that the large inertia and stiffness of the robot are involved in the collision.

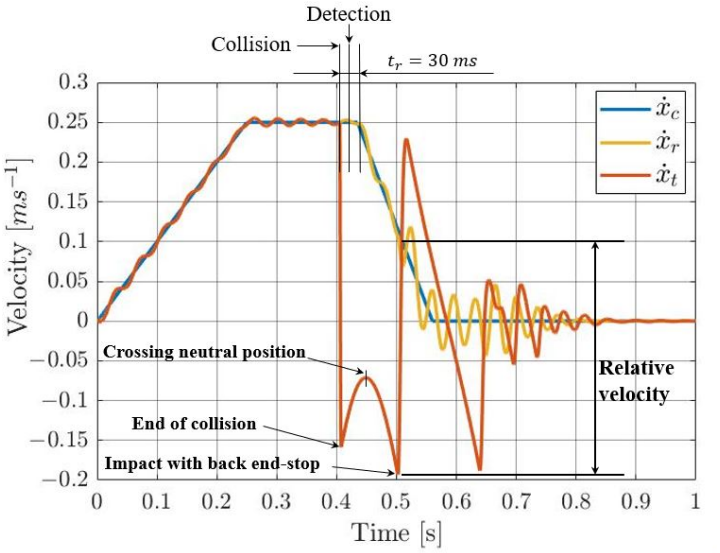
In the case analysed in Figure 40 the end-effector does not cross the neutral position of the spring in the bi-stable mechanism, since after the collision end-effector velocity with respect to the robot ($\dot{x}_t$-$\dot{x}_r$) does not show a negative mean value corresponding to the attraction towards the second equilibrium position. Therefore, the bi-stable mechanism behaves as a spring with variable stiffness. In this condition the maximum force applied to the robot coincides with the maximum force generated by the springs of the bi-stable mechanism, since there is no impact with the back end-stop.

However, if the end-effector crosses the neutral position of the system ($x = 0$), the end-effector moves towards the robot flange and a collision occurs when the end-effector reaches the end-stop. This behavior is highlighted in Figure 41a, which represents the velocity profiles $\dot{x}_c$, $\dot{x}_r$ and $\dot{x}_t$, assuming a higher velocity of the robot ($V_c = 0.25\ ms^{-1}$) with respect to the one of Figure 40. Figure 41b represents the corresponding contact force and total force on the robot.

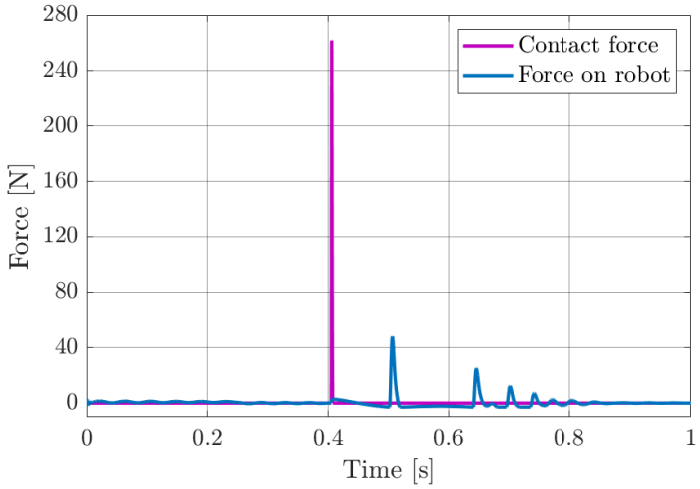
Figure 41a shows that after the collision the velocity of the end-effector becomes negative, then it increases as $F_b > 0$ in the first portion of the stroke. When the neutral position of the spring ($x = 0$) is crossed the velocity of the end-effector decreases, until the back end-stop is reached.

Actually, after the collision, the negative mean value of end effector velocity with respect to the robot corresponds to the attraction towards the second equilibrium position.

Figure 41b shows that the contact force has a single peak, as the end-effector directly moves towards the robot flange after the collision with the obstacle. The change in the sign of the robot force highlights the passage through the neutral position of the bi-stable mechanism. The force on the robot is characterized by several peaks, due to multiple bouncing of the end-effector on the back end-stop. However, the force applied on the robot is much smaller than the contact force. In this case the reaction time t_r is shorter than in the previous case, as the higher velocity V_c and contact force reduce the time needed by the end-effector to move away from the forward end-stop. It is worth noticing that the mechanism allows to decouple the end-effector from the robot only if the stopping distance of the robot is shorter than the total stroke of the system. If this condition is not fulfilled, a second collision occurs and this time the inertia and the stiffness of the robot



(a)



(b)

Figure 41: End-effector with bi-stable mechanism: (a) velocity profiles; (b) profiles of the contact force and the force on the robot. Parameters of the simulation are listed in Table 5 and 6

are involved, as the end-effector is laying directly against the robot flange.

3.2 PARAMETRIC ANALYSIS OF THE MECHANISM

In the mitigation of an accidental collision, the main goals are:

- minimization of the contact force, to prevent damages of the end-effector;
- minimization of the force applied to the robot, as the robot is very expensive to repair or replace.

This section analyses the effects of different parameters on the contact force and the force applied on the robot.

The parameters of the simulations listed in Tables 5 and 6 refer to end-effector properties corresponding to general purpose grippers and to impact with hard obstacles, which can be found in the industrial environment.

3.2.1 *Effect of parameters on contact force*

In the collision of two masses, the contact force mainly depends on the momentum of the masses, just before the collision, and on the contact stiffness k_c [49]. The decrease in the mass of the end-effector allows the robot to move at higher velocity, without increasing the contact force due to accidental collisions.

Figure 42 shows the maximum value of the contact force as a function of the mass m_t and the velocity of the robot V_c.

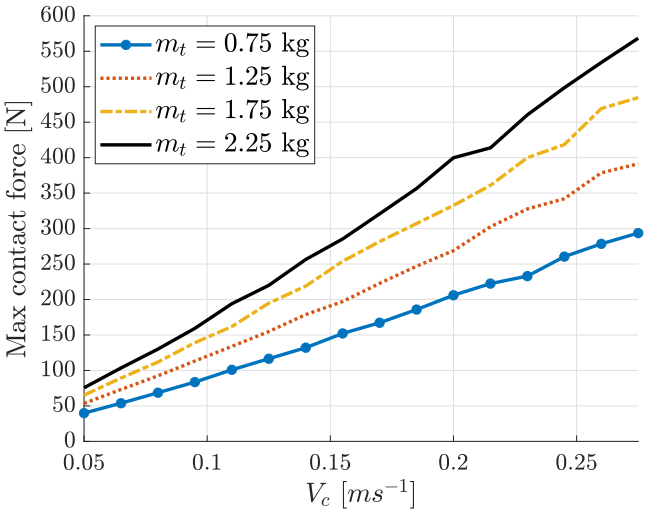


Figure 42: Maximum value of contact force as a function of robot velocity V_c and end-effector mass m_t

The contact force increases almost linearly as a function of the velocity of the robot, considering a certain mass m_t.

The reduction in the contact stiffness decreases the maximum value of the contact force, allowing to increase the velocity of the robot. Figure 43 represents the maximum value of the contact force as a function of velocity V_c and contact stiffness k_c.

The increase in k_b leads to a slightly increase in the peak of the contact force. Figure 44 represents the peak value of the contact force as a function of the velocity of the robot V_c and the stiffness k_b of the bi-stable mechanism.

The dead time DT is related to the effect of the robot control system on the interaction between the robot, the end-effector and the obstacle.

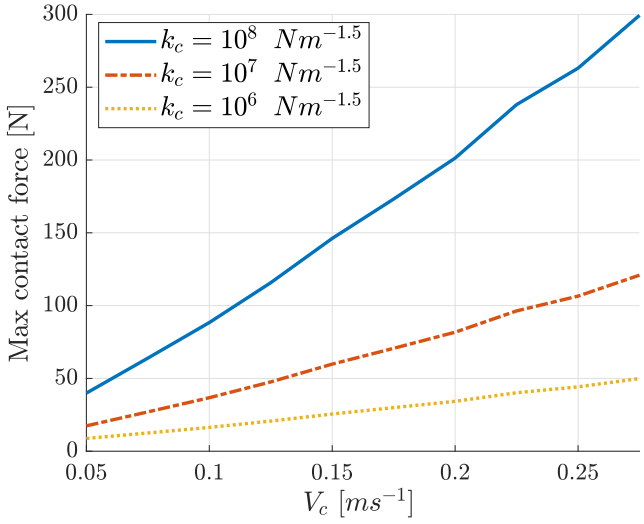


Figure 43: Maximum value of contact force as a function of robot velocity V_c and contact stiffness k_c

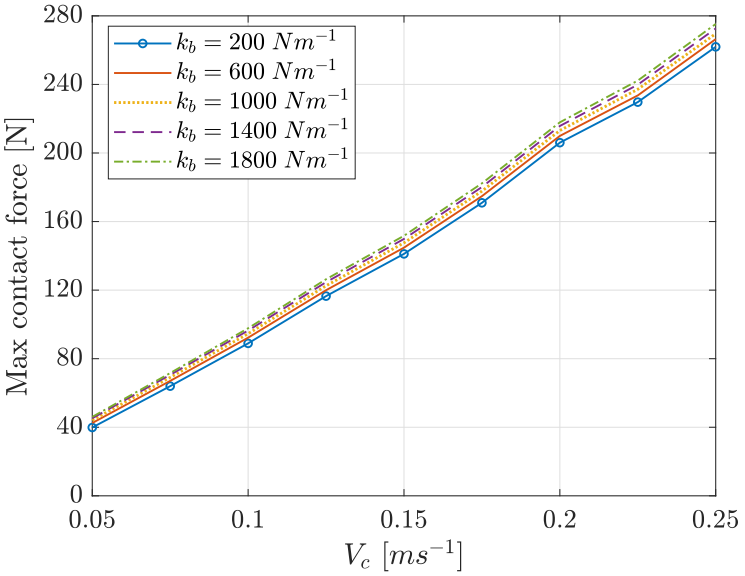


Figure 44: Maximum value of contact force as a function of robot velocity V_c and stiffness k_b of the bi-stable mechanism

Figure 45 represents the maximum value of the contact force as a function of the velocity of the robot V_c and the dead time DT.

The maximum value of the contact force is independent from the time delay *DT*. It is worth noticing that this result holds true when the total stroke of the mechanism C is larger than the stopping distance. On the contrary, a collision between the robot flange and the obstacle occurs. Actually, the stroke of the bi-stable mechanism is just enough to decouple the end-effector from the robot for $V_c = 0.25 \ ms^{-1}$ and $DT = 0.05$ s. In these conditions, the robot flange reaches the obsta-

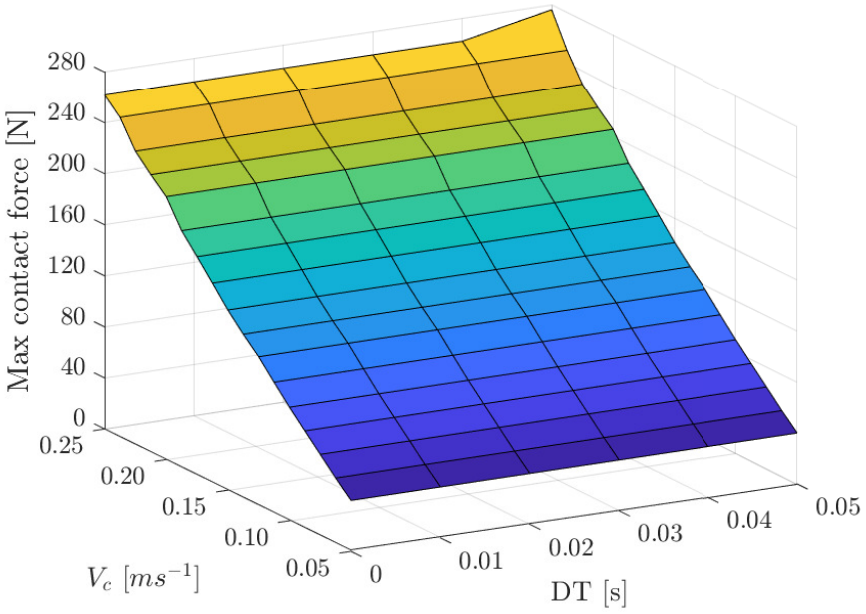


Figure 45: Maximum value of contact force as a function of robot velocity V_c and dead time DT

cle, but a small collision occurs, as the robot has already reduced its velocity. This small collision is highlighted by the peak in the upper right corner in Figure 45.

3.2.2 *Effect of parameters on the force applied on the robot*

The end-effector moves towards the robot flange up to reach the end-stop when it crosses the neutral position of the bi-stable mechanism ($x = 0$). The force on the robot caused by the collision between the end-effector and the end-stop depends not only on the mechanical properties of the end-stop and on the momentum of the end-effector before the collision, but also on the dynamics of the robot after the collision. Figure 46 represents the maximum force on the robot as a function of the velocity V_c and the mass m_t. Figure 46 allows to define three main scenarios:

1. the end-effector does not impact with the back end-stop;
2. the end-effector impacts with the back end-stop after the robot stop;
3. the end-effector impacts with the back end-stop before the robot stop.

The three scenarios highlighted in Figure 46 refers to the end-effector with $m_t = 0.75$ kg.

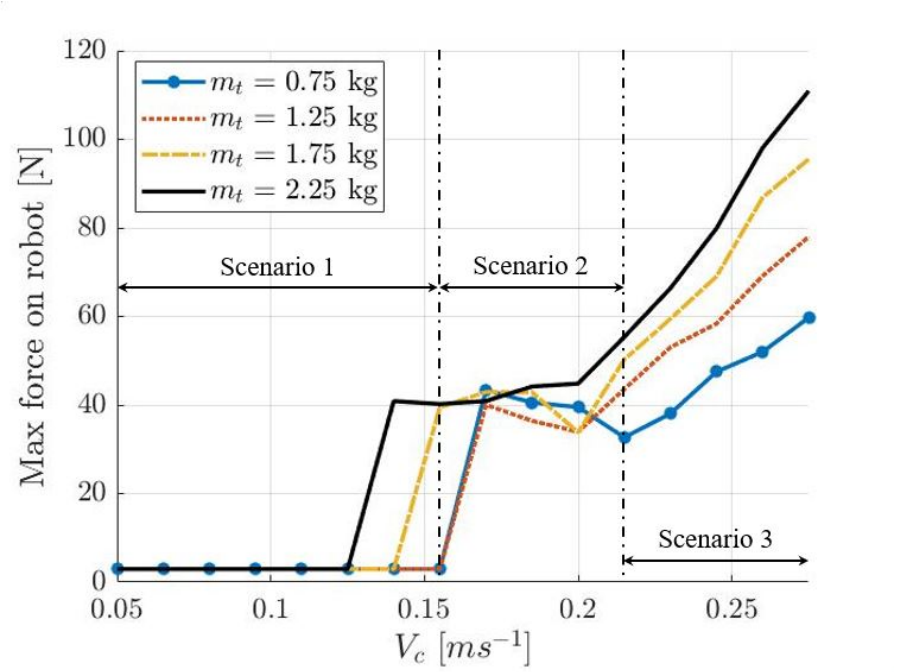


Figure 46: Maximum value of the force on robot as a function of robot velocity V_c and end-effector mass m_t

The first scenario, which is depicted in Figure 40, is characterized by the smallest value of the force on the robot, since the contact force is not enough to move the end-effector beyond the neutral position of the springs.

In the second scenario the increase in the contact force is enough to move the end-effector beyond the neutral position of the springs. Hence, the force on the robot significantly increases, due to the collision of the end-effector with the back end-stop. It is worth noticing that the second scenario represents a transition phase, in which the non-linear interaction between the robot, the end-effector and the obstacle is also affected by the inertia force, due to the braking of the robot. Figure 47 represents the velocity profiles $\dot{x}_c$, $\dot{x}_r$ and $\dot{x}_t$, assuming a velocity of the robot $V_c = 0.20\ ms^{-1}$, which refers to scenario 2.

Figure 47 highlights that the collision between the end-effector and the back end-stop occurs when the robot is stationary. The relative velocity between the robot and the end-effector is smaller in Figure 47 than in Figure 41, causing a smaller force on the robot. It is worth noticing that in Figure 41 the end-effector moves towards the robot flange only when the robot is braking, whereas in Figure 47 a portion of the movement of the end-effector occurs when the robot is stationary. This behavior affects the maximum negative velocity reached by the end-effector before impacting with the back end-stop, hence the relative velocity of impact. The direct comparison between the

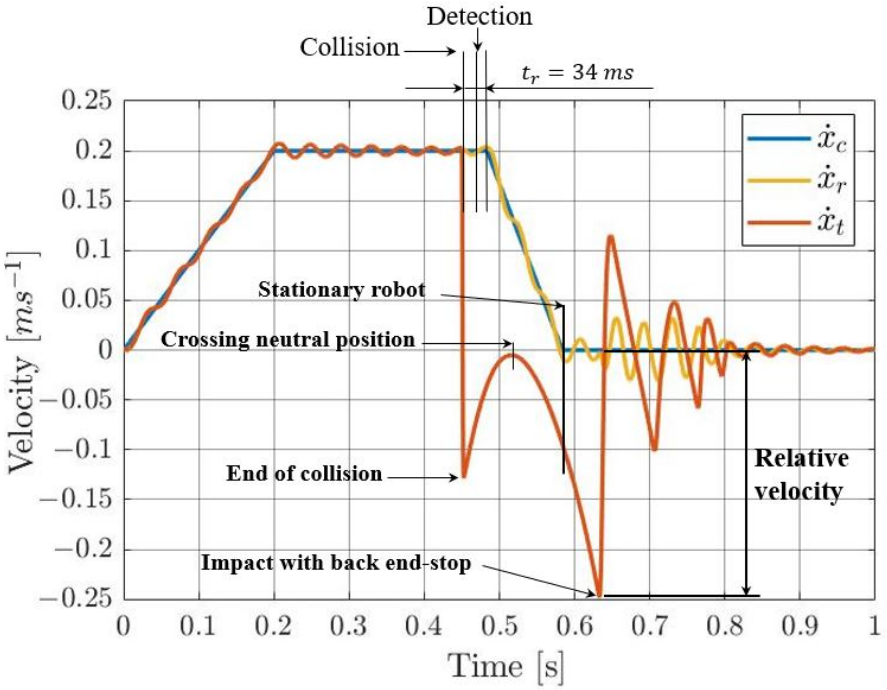


Figure 47: End-effector with bi-stable mechanism: velocity profiles in scenario 2 ($V_c = 0.20\ ms^{-1}$, $m_t = 0.75$ kg). Parameters used in the simulation are listed in Table 5 and 6

velocity of the end-effector in the scenarios of Figures 41 and 47 is represented in Figure 48.

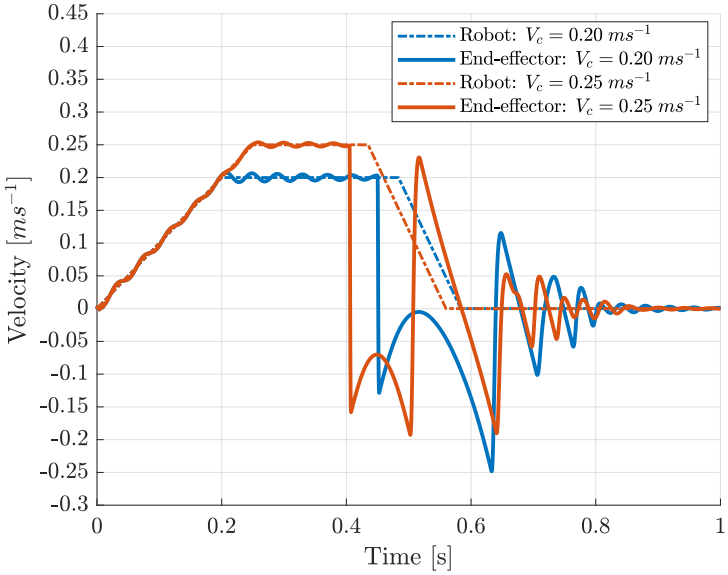


Figure 48: Comparison between the velocity of the end-effector for $V_c = 0.20\ ms^{-1}$ and $V_c = 0.25\ ms^{-1}$ (Scenarios 2 and 3, respectively)

In the third scenario, which is reproduced in Figure 41, the force on the robot increases with the velocity of the robot and the mass of the end-effector. The impact velocity increases for higher velocity of the robot, as the robot takes more time to decelerate.

The force on the robot as a function of the velocity of the robot V_c and the dead time DT is represented in Figure 49. Figure 49 corroborates the existence of three working modes of the end effector. The bi-stable behavior does not occur when the velocity V_c and dead time are small. The end-effector collides with the back end-stop for higher velocities and dead times.

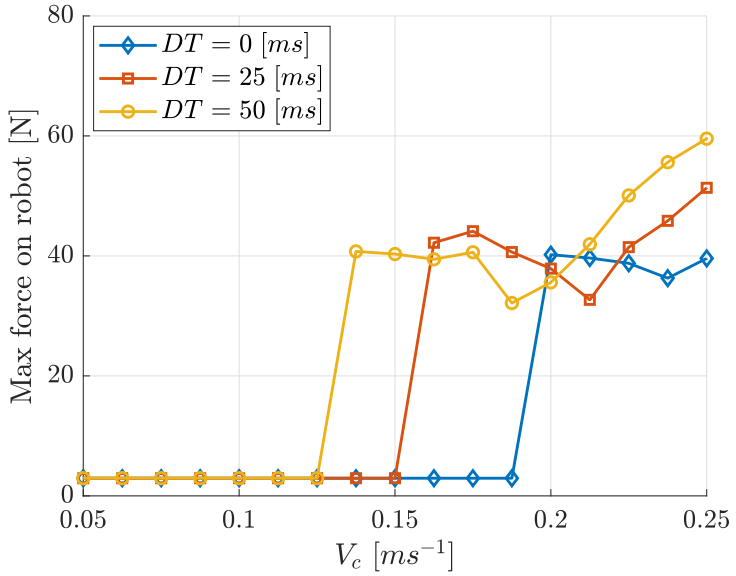


Figure 49: Maximum value of the force on robot as a function of robot velocity V_c and dead time DT, $m_t = 0.75$ kg

In conclusion, numerical simulations showed the presence of three working modes of the end-effector with bi-stable mechanism. With low impact speed and short dead time the bi-stable mechanism does not cross the neutral position and a very large reduction in the force applied on the robot is achieved. Then, there is a range of speeds and dead times in which the end-effector crosses the neutral position and impacts the back end-stop, but the impact takes place when the robot is stationary, and a moderate increase in the force on the robot takes place. For large impact velocities the end-effector crosses the neutral position and impacts the back end-stop when the robot is moving. The force on the robot is larger than in the previous cases and depends on the mass of the end-effector. Nevertheless, even in this working mode the bi-stable mechanism strongly reduces the forces with respect to the values that are found with rigid end-effector.

The mathematical model described in the previous Section is validated by means of a new prototype designed to be adaptable to any linear 1D tool. The prototype is presented in the next Section. Initially, it was decided to perform a pilot validation by means of the same prototype used for the model of impacts with small objects. This decision was done out of two advantages:

- show the validity of the prototype in the mitigation of both type of impacts;
- take advantage of the design of the tool to validate the model's estimation of forces on the tool.

The second point is possible thanks to the reduced velocities of the robot used in the tests. Indeed, the bi-stable mechanism cannot ensure safety at any velocity of the robot. The maximum guaranteed velocity reachable by the robot depends on the efficiency of the control system (i.e., the dead-time) and the length of the tool. Therefore, keeping the total size of the tool reasonable restrained the maximum velocity reachable. Thanks to the lower velocities, an accelerometer could be attached to the paddle to estimate the forces during the impact.

Before showing the validation results, it is worth remembering that the surfaces of the moving components can be very rough due to the use of additive manufacturing in their realization. Therefore, the large clearances and the high roughness of the surfaces increase the friction and stick-slip phenomena. Therefore, the mathematical model described by Equation 36 can be updated as follows:

$$m_r \ddot{x}_r = c_r(\dot{x}_c - \dot{x}_r) + k_r(x_c - x_r) - F_b - F_e - F_f \tag{39}$$

$$m_t \ddot{x}_t = F_b - k_c \delta^m - \chi_c \delta^n \dot{\delta} + F_e + F_f \tag{40}$$

where F_f is the friction force acting along the sliding axis of the paddle and it is defined as follows:

$$F_f = \mu F_n \tanh\left(\frac{\dot{x}_r - \dot{x}_t}{v_0}\right) \tag{41}$$

where μ is an equivalent friction coefficient, v_0 is a constant and F_n is the force component generated by the bi-stable mechanism perpendicular to the sliding direction.

$$F_n = N k_b (L_0 - L)\frac{H}{L} \tag{42}$$

The end-effector is equipped with an inductive sensor, which detects the detachment of the paddle from the forward end-stop due to the collision with the obstacle. Such sensor provides a digital signal once the paddle has slid 1.5 mm from the forward end-stop.

A series of experimental tests were performed in order to validate the mathematical model. The robot used in the tests is a Mitsubishi RV-4FRL-D. As stated in Section 2.2.1, its parameters are estimated using the CAD model of the robot, and an impulsive modal analysis of the robot [56, 80]. The robot moves towards an obstacle, following a linear trajectory with a trapezoidal velocity profile. In order to minimize the risk of robot damage during the tests, a "soft" obstacle made of a polystyrene block was used. This aspect will not affect the generality of the validation since it affects only the value of stiffness and damping of the contact (k_c and χ_c).

The validation is performed tracking two markers attached on the impact actors: the robot, and the paddle. A high frame rate camera is used to track such markers at ~300 fps, during the interaction between the end-effector and the obstacle. Figure 50 represents a snapshot of the experimental setup, where the black cards are used to eliminate reflections and improve the performance of the tracking system. A calibration process was performed by means of the same

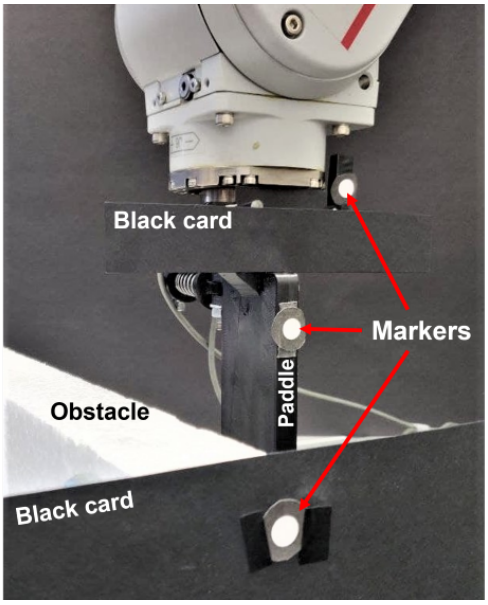
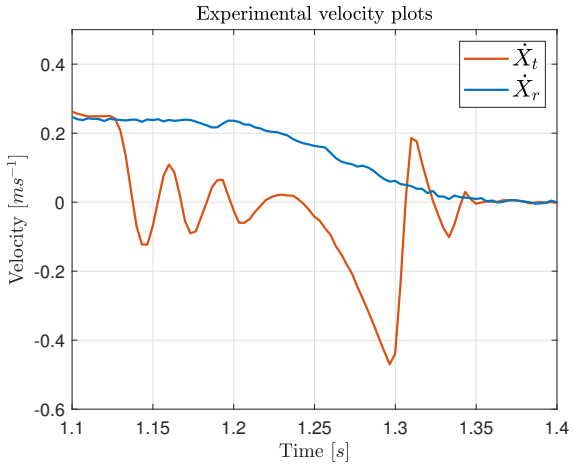


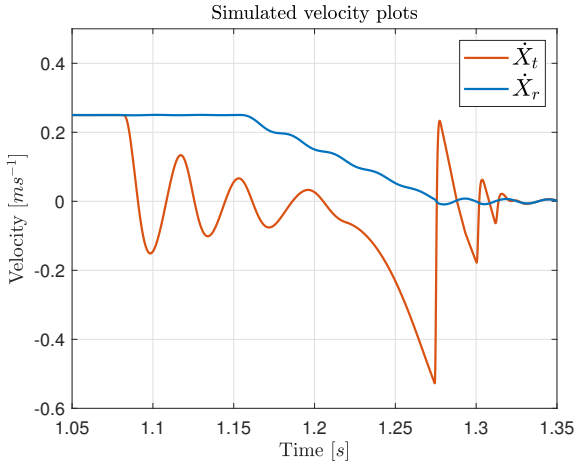
Figure 50: Position of the three markers: on the obstacle, on the paddle and attached to the robot.

standard procedure described in Section 2.5.
The experimental result of a typical tracking and the corresponding simulation can be seen in Figures 51a and 51b. The values of parameters used in the simulation are listed in Table 8. It is worth noticing that the parameters are different from the ones in Table 6 since the prototype was built using the available components and manufacturing technology, and the obstacle is made of polystyrene, which reduces the contact stiffness to $k_c = 0.6 \cdot 10^5$ $Nm^{-1.5}$, in order to avoid possible damages of the robot and the end-effector.

The experimental velocities show the typical features of scenario 3



(a)



(b)

Figure 51: Results of tests with the prototype: (a) experimental velocities; (b) simulated velocities. Parameters of the simulation are listed in Table 8

since the impact with the back end-stop occurs when the robot is still moving. A good agreement between experimental and numerical results is found.

Besides observing the velocities, the aim of this pilot test was the validation of the contact force. The contact force was validated through the measurement of the acceleration of the end-effector. First, the relation between an impulsive force applied on the tip of the paddle and the acceleration of the paddle was determined. Then, the acceleration during the interaction between the end-effector and the obstacle was measured, and the contact force was estimated. These tests were performed by attaching an accelerometer (PCB 352C23) to the upper part of the blade and exerting an impulsive force on the tip of the paddle by means of an instrumented hammer for modal analysis (PCB086C03), as represented in Figure 52.

Parameter	Unit	Value
N	/	1
L_0	m	0.062
k_b	Nm^{-1}	490
H	m	0.053
C	m	0.047
$C_1 = C_2$	m	0.0235
V_c	ms^{-1}	0.25
a_c	ms^{-2}	1
d_c	ms^{-2}	2
m_t	kg	0.048
k_c	$Nm^{-1.5}$	$0.6{\cdot}10^5$
e_c	/	0.8
k_{be}	$Nm^{-1.5}$	$2.25{\cdot}10^6$
e_{be}	/	0.3
k_{fe}	Nm^{-1}	$2.25{\cdot}10^6$
c_{fe}	Nsm^{-1}	235
DT	s	$70 \cdot 10^{-3}$
μ	/	0.12
v_{end}	ms^{-1}	0

Table 8: Values of the parameters used in the simulations of the real system

The tests showed that the peak force, applied by the hammer, and the peak acceleration, measured by means of the accelerometer, are proportional through the mean coefficient of 0.564 $Ng^{-1} \pm 0.19$ (95%).

Figure 53a represents the acceleration profile measured with the accelerometer during the interaction between the end-effector and the obstacle. In contrast, Figure 53b represents the acceleration profile of the end-effector calculated by means of the numerical model.

Figure 53 highlights that the numerical results are in good agreement with experimental measurements. The experimental plot is cut to observe the acceleration after the impact occurs. In the experimental acceleration profiles, the first negative peak, about -6.9 g at 0 s, is due to the collision between the end-effector and the obstacle, whereas the first positive peak, about 65.5 g at 0.23 s, is due to the collision between the end-effector and the back end-stop. The comparison with the numerical simulation highlights that the two collisions are well-described. However, the experimental tests of Figure 53a show a second negative peak (about -14.5 g, just after the collision with the back end-stop), which is not represented in the simula-

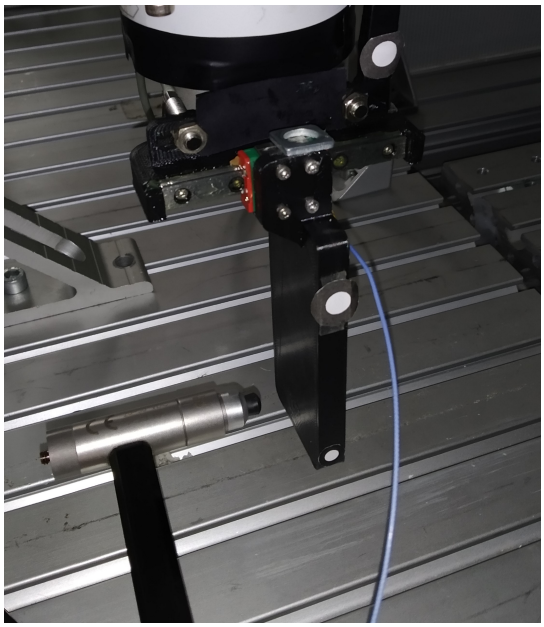


Figure 52: Accelerometer and instrumented hammer

tion of Figure 53b. The reason for that negative peak is related to the force's application point. The general behavior caught by the model considers the impact occurring along the sliding axis of the tool. In the prototype, the impact force is not applied along the sliding axis of the paddle. Therefore, the torque generated during the interaction can generate an oscillation of the paddle when the force applied to the paddle is large. Moreover, since the paddle is realized in additive manufacturing, its vibratory response depend on the internal structure. The numerical model does not include these aspects.

Considering the mean proportionality factor between the force applied on the paddle and the measured acceleration, the experimental acceleration profile allows to estimate the peak in the contact force as follows:

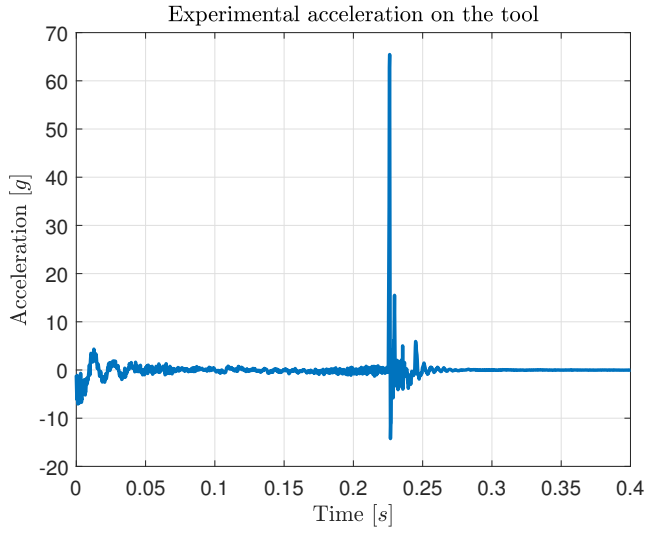
$$max\ contact\ force = 0.564 \mathrm{N g^{-1}} \cdot 6.9\ \mathrm{g} = 3.9\ \mathrm{N} \tag{43}$$

It is worth noticing that the max contact force is much lower than the one found in Figure 41, because the impacting mass of the end-effector and the contact stiffness are much smaller (0.048kg compared to 0.75kg and $0.6 \cdot 10^5\ \mathrm{N m^{-1.5}}$ compared to $10^8\ \mathrm{N m^{-1.5}}$).

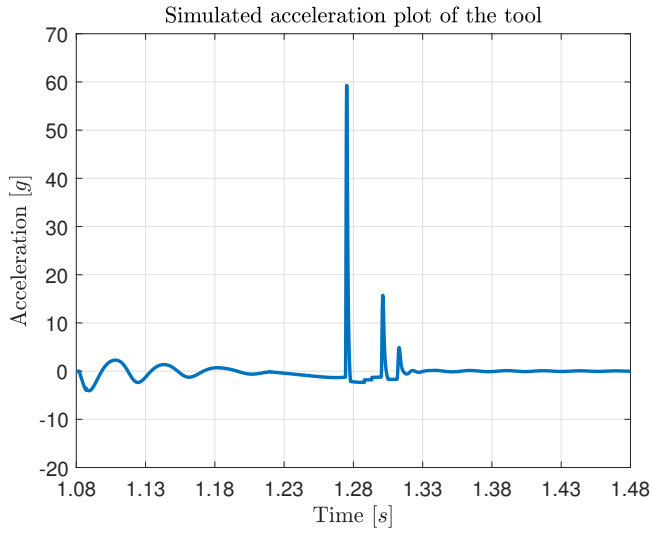
Figure 54 represents the absolute value of the contact force during the interaction between the end-effector and the obstacle, calculated by means of the numerical model.

Figure 54 shows that the numerical simulation is in good agreement with the experimentally estimated peak of the contact force.

In conclusion, as can be inferred from Figures 51, 53 and 54 the model can predict the behavior of the end-effector with a good approximation, both qualitatively and quantitatively. In particular, a dif-



(a)



(b)

Figure 53: Acceleration profiles: (a) from experimental measurement; (b) from the numerical model.

ference of about 5% can be observed in the maximum negative velocity of the end-effector which is influenced by the real friction of the ball bearing guide.

These results are promising and confirm the effectiveness of the end-effector in mitigating the impact of the robot with obstacles.

3.4 GENERALIZATION OF THE DESIGN TO 1D TASKS AND VALIDATION

The validation performed in the previous Section had the focus on the estimation of the contact force. Indeed, the paddle allows the easy

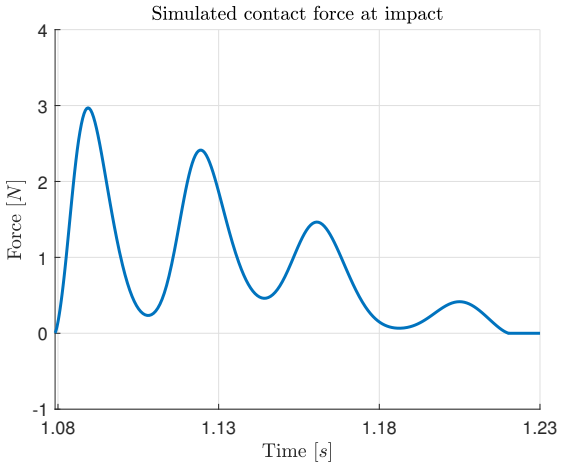


Figure 54: Contact force profile calculated by means the numerical model. The obstacle is made of soft polystirene.

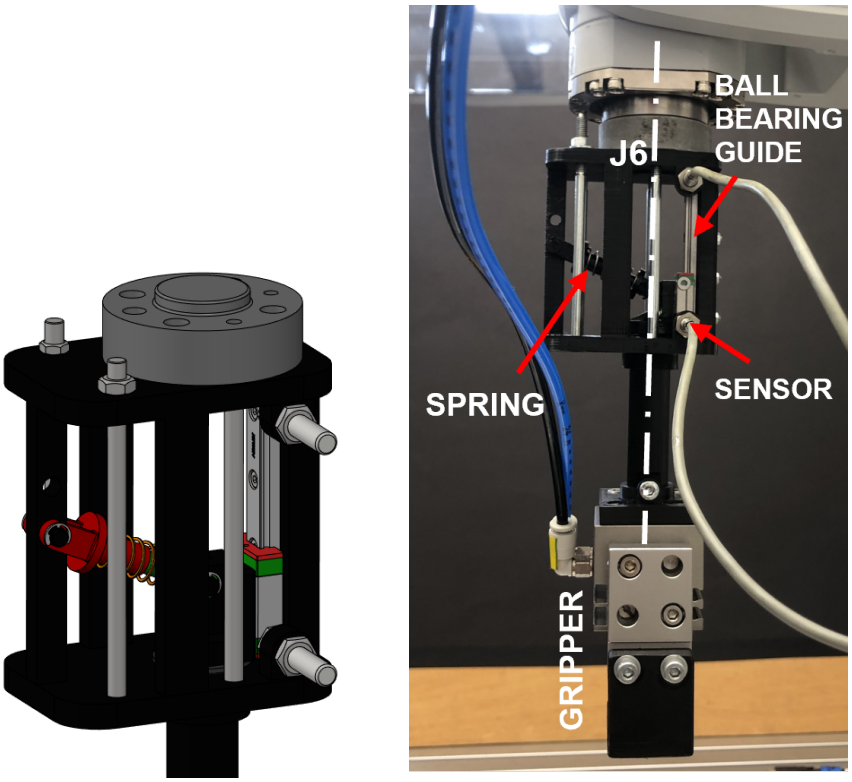
attachment of an accelerometer to its frame. In this Section, a second validation is performed. A new prototype has been realized in the form of a module to be interposed between the robot flange and any linear 1D tool, such as a paddle or a gripper. This design led to a more general system that can be applied to a various scenarios. In this work, a gripper is attached to the tool. The test recalls an erroneous pick-and-place task where the height of the pick/place location is wrongly measured. Another possible example is the execution of the pick-and-place task in teleoperation, where the operator misreads the distance from the surface and hits it.

The new prototype is shown in Figure 55. The movable hinge of the bistable mechanism is connected rigidly to a beam long enough to ensure the complete backward movement of the slider. At the end of the beam, a changeable flange connects the module to the tool.

The system's response can be modified using the regulation system showed in Figure 56. Depending on the desired preload, the upper hinge of the bi-stable mechanism can be moved in one of the holes. More holes can be prepared to increase the system's adaptability to different scenarios.

Figure 56 shows the main difference in the design of this prototype with respect to the one described in Chapter 2. Following Figure 57, the regulation modifies the ratio between the two half strokes, C_1 and C_2. In the previous prototype, the regulation is on the H parameter. This difference is partly due to an optimization of the design to reduce the total size of the system. In addition, this decision enables the system to take advantage of the weight of the attached tool. In particular, two scenarios can take place with respect to the case with $C_1 = C_2$:

1. if $C_1 > C_2$, the initial resistance force offered by the bi-stable mechanism is increased, whereas the force exerted on the back



(a) (b)

Figure 55: Prototype developed for the experimental validation of the models: (a) CAD model; (b) real model.

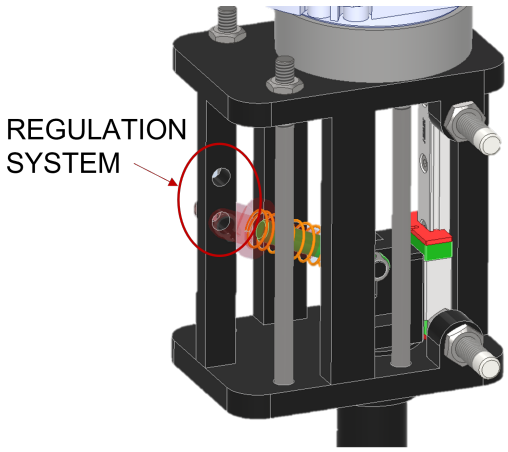


Figure 56: Detail of the regulation system of the bi-stable mechanism in the Z-axis prototype.

end-stop is reduced. Consequently, if the weight of the tool is higher than the force of the spring on the back end-stop, the weight can be used to restore the initial position of the tool once the external perturbation is removed;

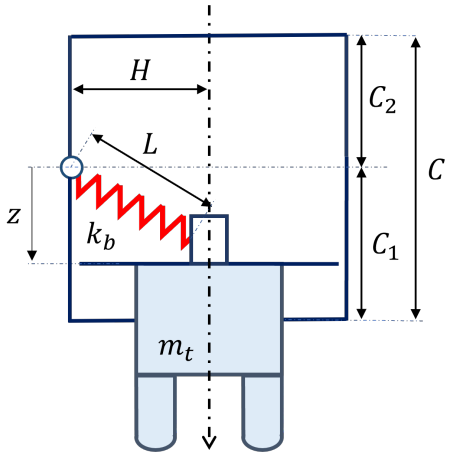


Figure 57: Geometric model of the bi-stable mechanism in the Z-axis prototype. C is the total stroke. The ratio between C_1 and C_2 is modified based on the desired behavior.

2. if $C_1 < C_2$, the system is more sensitive to external perturbations since the force of the spring against the front end-stop is reduced. In addition, the switching position is closer to the tool.

The second scenario can be helpful when the robot has to manipulate fragile components. In contrast, the first one can be exploited for tasks requiring the application of a force (for example, pressing of button [73]).

Changing the H parameter increases/decreases the force equally on the two end-stops. In the previous prototype, this is a necessary feature to optimize the design and enable its functioning against impacts on both the faces of the paddle (with the opportune rotation of the tool). Figure 58 shows the effect of modifying C_1 on the force expressed by the spring against the front end-stop. In particular, the force on the front end-stop decreases with the distance of the hinge from it (C_1).

Using the bi-stable mechanism along the Z-axis requires a slight modification of the motion laws. The weight of the module and tool are now acting along the direction of the impact. Therefore, they have to be considered, and the motion laws become:

$$m_r\ddot{z}_r = c_r(\dot{z}_c - \dot{z}_r) + k_r(z_c - z_r) - F_b - F_e - F_f - F_w \quad (44)$$

$$m_t\ddot{z}_t = F_b - k_c\delta^m - \chi_c\delta^n\dot{\delta} + F_e + F_f + F_w \quad (45)$$

where F_w is the total weight of tool and module. It is worth noticing that this factor is constant throughout the impact. In the considered case, it works as an added resistance to the upward translation of the tool.

In the design proposed in Figure 55b, the total mass attached to the

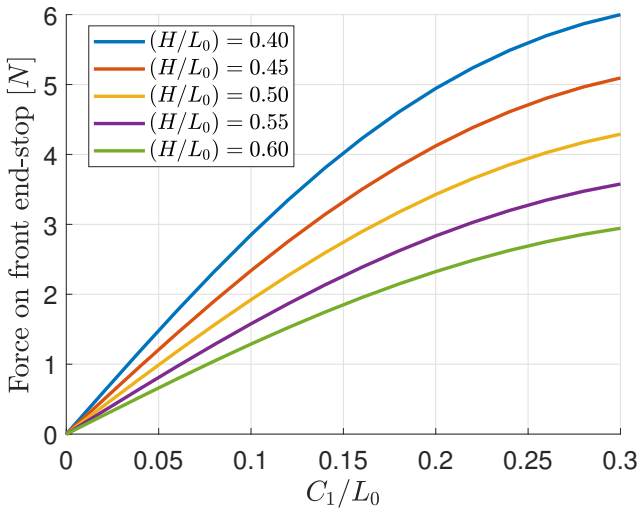


Figure 58: Force expressed by the spring (F_b) on the front end-stop as a function of the rations (C_1/L_0) and (H/L_0), with the parameters of Table 5

robot is equal to about 230gr. Thus, the total weight of the compliant system is about 2.26N. The pneumatic tubes and sensor's cable contribution is considered negligible.

Experimental tests were carried out with the aim of validating the velocities estimated by the model. First, the robot (Mitsubishi RV-4FRL-D) approaches the fixed obstacle linearly using a trapezoidal velocity profile. Then, it hits the surface at a constant velocity. Finally, the sensor detects the impact after a displacement of the tool of about 0.5 mm.

The object is realized in polystyrene to prevent damage during the validation due to a malfunctioning of the prototype. The characteristics of the impacted object do not interfere with the validation. They affect only the value of k_c and χ_c. Figure 59 shows the experimental setup.

The validation is performed by tracking two markers attached to the two actors involved in the task: the robot and the tool (i.e., the gripper). The markers are highlighted in Figure 60.

The tests were carried out by combining the parameters in Table 9.

v_0	C_1
$[ms^{-1}]$	mm
0.150	15.25
0.240	25.25

Table 9: Variable values used in the tests.

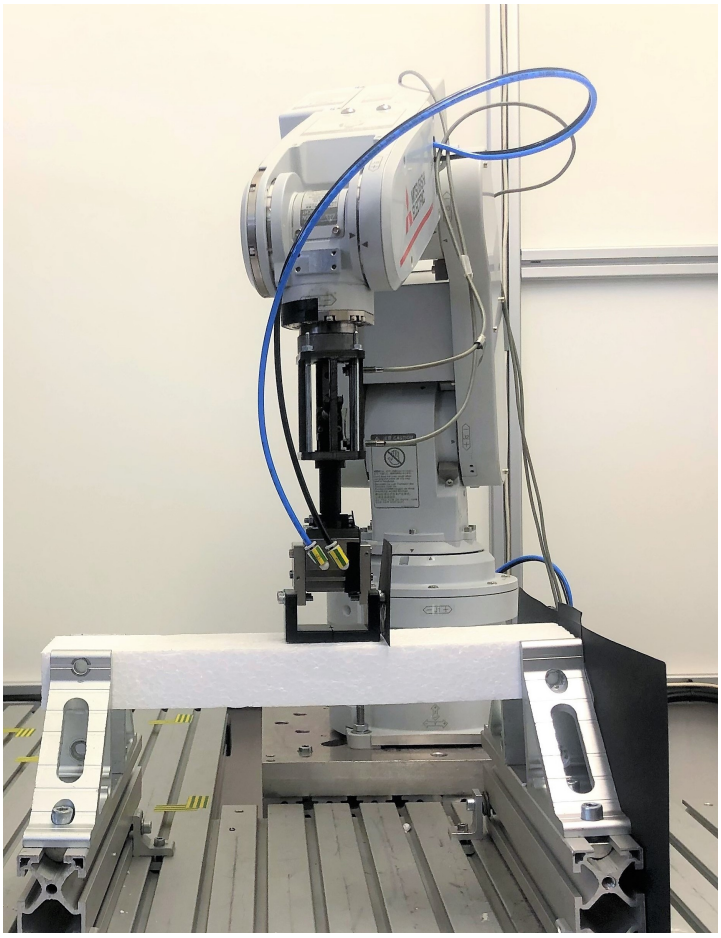


Figure 59: Experimental setup for the validation of the Z-axis prototype.

Two different springs were employed thanks to the reduced velocity of the robot and the increased forces. In the previous case of moving small objects, a stiffer spring would have prevented the bi-stable mechanism from switching, limiting the functionality of the mechanism.

k_b	L_0
$[Nm^{-1}]$	$[m]$
446	34.1
477	50.0

Table 10: Characteristics of the two springs used for the validation.

Figure 61 shows the experimental results and the corresponding simulated matching in some conditions. The tests are identified by the values $[N_i, C_{1,i}, v_{0,i}]$, representing the spring, the frontal half stroke of the mechanism and the impact velocity of the robot, respectively. i represents the row of the parameter in the corresponding table (see

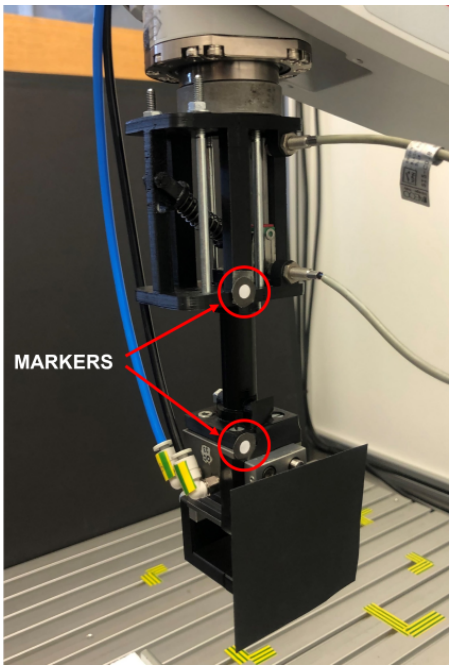


Figure 60: Detail of the markers on the robot and the tool tracked during the experimental validation.

Tables 9 and 10). From Figure 61, it can be inferred that the bi-stable mechanism switches only in the third comparison. In the other two, the spring is too weak to overcome the weight of the tool and push it against the robot flange. Consequently, the tool hops against the impacted surface in the first two plots until the robot stops.

The simulations were carried out with the parameters listed in Table 11. Therefore, the three simulations in Figure 61 simulate the same system in three different conditions. This aspect prevents a perfect matching between tests and simulations. Indeed, a perfect matching would have required modifying the parameters to match every situation better. For example, in Figure 61e, the dead-time appears to be lower than in the other tests. The corresponding simulation in Figure 61f sees the tool impacting with the robot at a higher velocity. Moreover, the second spring is longer than the first one. Consequently, it is more compressed during the impact and can generate more friction on its support system.

Even considering these considerations strictly related to the experimental setup, Figure 61 shows good consistency between the experimental results and the simulated ones. Therefore, the model is able to qualitatively predict the behavior of the tool, as for the case of impacts with small objects. In addition, it is worth remembering from Section 2.5 that the experimental velocity peaks values depend on the tracking frequency.

Concluding, experiments corroborate the validity of the model and the effectiveness of the proposed solution for mitigating unexpected

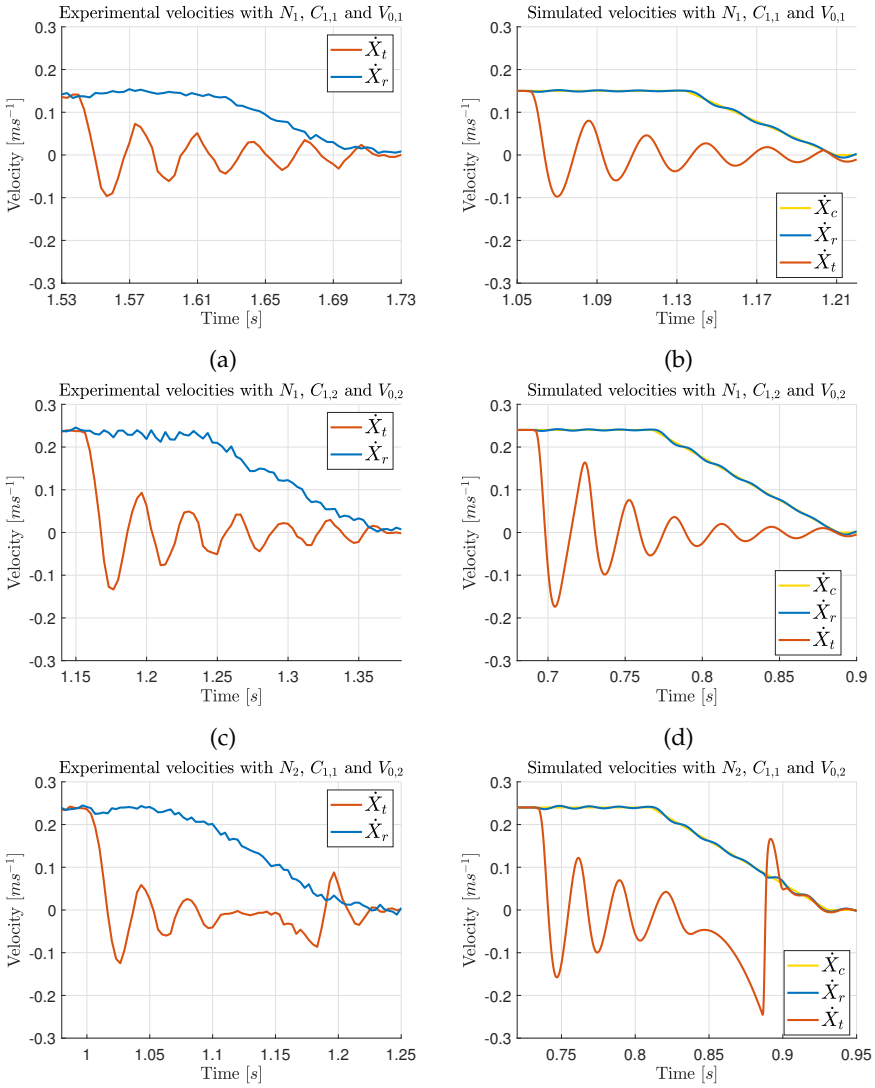


Figure 61: Comparison between experimental and simulated velocities for the Z-prototype.

collisions with hard surfaces. Finally, in the next Section, a prototype is proposed to extend the bi-stable mechanism to 2D.

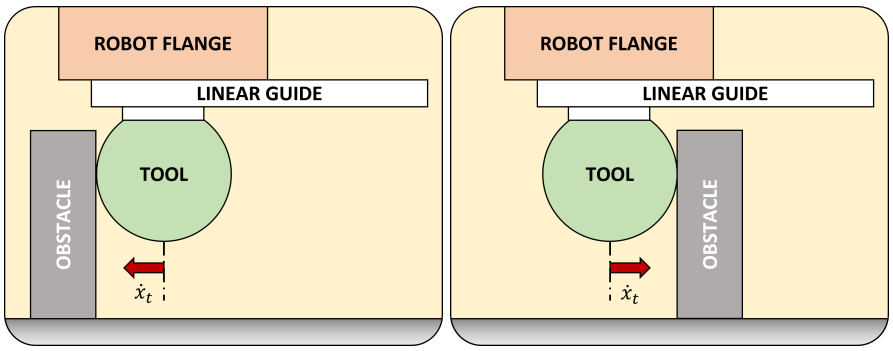
3.5 EXPLOITATION OF THE BI-STABLE MECHANISM IN 2D

This Section presents the design and experimental validation of a module to extend the use of the 1D bi-stable mechanism to two dimensions. The design assumes planar movements of the robot. Therefore, the prototype should be able to mitigate collisions along the XY plane. The proposed solution is constituted of a rotational module to be interposed between the robot flange and the tool able to align the

Parameter	Unit	Value
H	m	0.035
a_c	ms^{-2}	1
d_c	ms^{-2}	2
m_t	kg	0.230
k_c	$Nm^{-1.5}$	$3{\cdot}10^5$
e_c	/	0.85
k_{be}	$Nm^{-1.5}$	$2.25{\cdot}10^6$
e_{be}	/	0.3
k_{fe}	Nm^{-1}	$2.25{\cdot}10^6$
e_{fe}	/	0.4
DT	s	$70 \cdot 10^{-3}$
μ	/	0.15
v_{end}	ms^{-1}	0

Table 11: Values of the parameters used in the simulations of the real system

bi-stable mechanism along the direction of the impact. A system of this kind is useful if we consider the cases in Figure 62.



(a) (b)

Figure 62: Possible case scenarios for 1D impact with bi-stable mechanism: (a) correct impact direction; (b) incorrect impact direction.

The 1D impact can occur in two directions. However, the bi-stable mechanism can mitigate only one. In particular, in Figure 62a, the bi-stable mechanism can mitigate the collision with the obstacle. In contrast, in Figure 62b, the bi-stable mechanism is at the end of the possible stroke. As a result, the tool appears rigidly fastened to the robot, and it cannot mitigate the collision. Attaching the tool to the center of the linear guide can prevent this scenario, but it also increases the size of the mechanism to allow the same response. Moreover, the non-linear spring has to prefer one of the two directions to guarantee an initial resistance along that direction. Otherwise, for

symmetry reasons, it must be set perpendicular to the direction of the impact resulting in no initial force along that direction. The design of the prototype of Chapter 2 reflects these considerations.
A system able to orient the bi-stable mechanism can guarantee the optimal functioning of the tool. Moreover, it can make it functions in 2D.

Another example for which a 2D mitigation system can be useful is the task of bin picking. In this task, the robot has to approach and pick objects overlapped and chaotically arranged in a bin. The will to automate this task is related to the necessity to increase the flexibility of the existing production systems. In particular, it will help to reduce space, avoiding the need for parts feeders [81]. However, the automation is complex due to the infinite scenarios that can occur. Moreover, the robot's motion must be planned carefully to prevent impacts with one of the bin's four borders or any other object while reaching its goal.

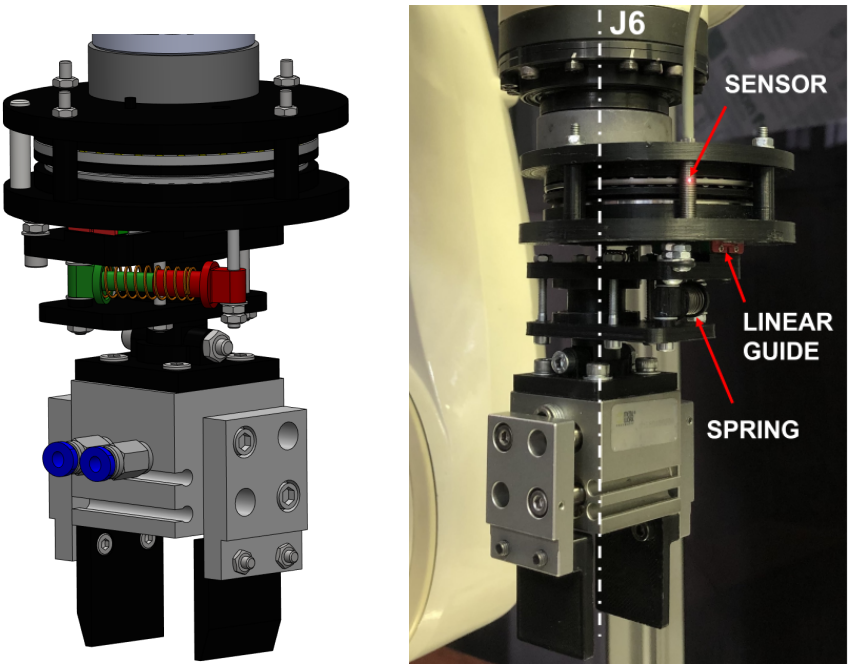
On the other hand, some scenarios require the search for a collision. For example, the robot can face the necessity to move the objects to put them in a graspable position [82] or to free up the targeted object (when different components are overlapped) [83, 84].

A compliant end-effector, such as the one proposed in the previous Section, can easily cover both tasks. In particular, the bi-stable mechanism described above can:

- offer an initial resistance to the tool movement that opportunely tuned can move or push small objects;
- mitigate an unexpected collision with objects (as shown in Chapter 2) and hard surfaces (as shown in Chapter 3.

However, in the current state, these advantages can be exploited for mono-dimensional impacts only, whereas the cases described above are at least bi-dimensional. Hence, the research exposed in this Section extends the bi-stable solution to 2D.
Different designs have been evaluated. The first and more straightforward possible approach follows the idea exposed in [27]. In this paper, the authors designed a compliant frame for a robot by stacking multiple 1D systems together. However, doing the same with the bi-stable mechanism would end up with a bulky system. The solution adopted and developed combines the bi-stable mechanism with a rotational system. This system is interposed between the robot flange and the bi-stable mechanism. During an impact, it orients the bi-stable mechanism along the direction of the impact. The designed system is shown in Figure 63. The module is designed to align the tool along the axis of the last joint of the robot when there are no external perturbations.

The internal design is shown in Figure 64. The system is constituted of three main parts that can freely move one with respect to the other. The part highlighted in blue is attached to the robot flange and fixed.



(a) (b)

Figure 63: Prototype developed for 2D mitigation: (a) CAD model; (b) real model.

The red part is attached to the tool and can rotate thanks to ball bearings. Two ball bearings are employed to keep the beam parallel to the axis of the robot's final joint. The remaining central part houses the bi-stable mechanism and contains a ball bearing and an axial roller bearing to allow the rotation around the blue part.

The two rotating parts are kept in position during free motion by two springs, as shown in Figure 65.

The two springs are also helpful in restoring the initial position of the parts after a weak impact. Indeed, a weak impact does not cause the bi-stable mechanism to switch. If the system switches, an opportune motion or a manual restore has to be performed.

During an impact, the system rotates thanks to a momentum created around its center. The momentum is created by offsetting the tool with respect to the cursor of the ball bearing guide to which it is attached. Figure 66 shows the displacements (identifiable by the letters a and b) between the axis of the tool and the cursor.

Decoupling the beam attached to the tool from the other part of the mechanism is necessary to avoid undesirable movements of the tool that can damage manipulated objects. For example, during an impact, the tool can remain in contact with the cause of the perturbation while the rest of the system rotates.

Experimental tests have been carried out to test the functioning of the prototype. Figure 67 shows the experimental setup for these tests.

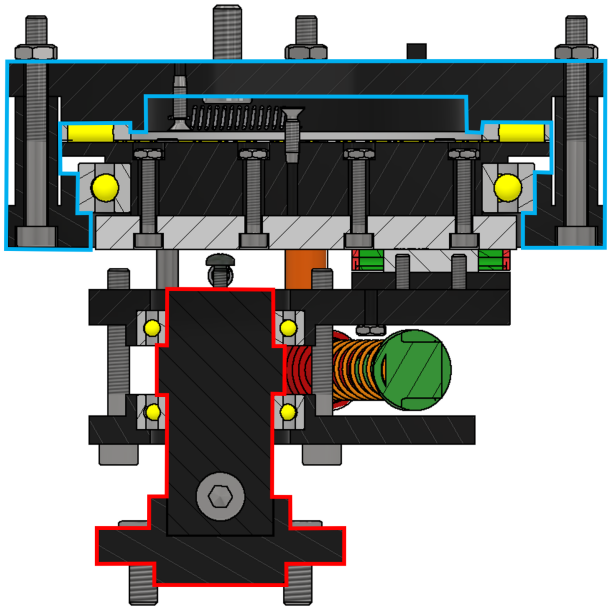
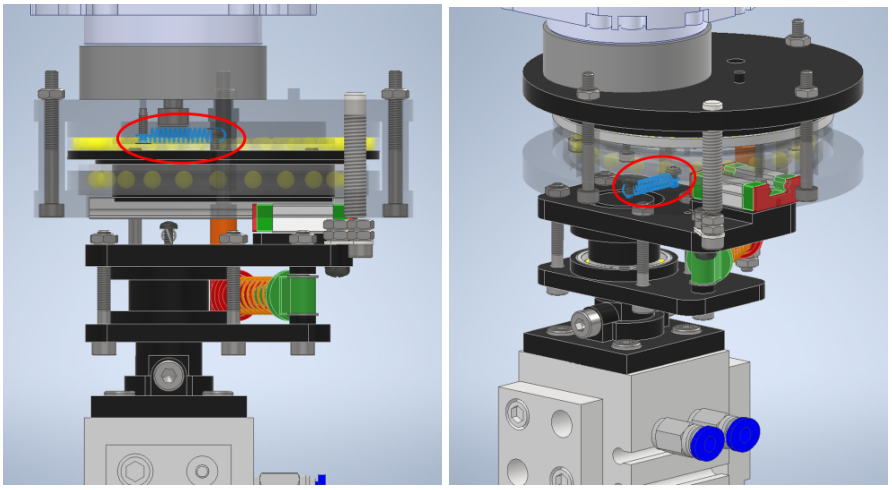


Figure 64: Lateral section of the XY-prototype evidencing the three different parts.



(a) (b)

Figure 65: Detail of springs used to retain the position of the parts during free motion: (a) upper spring; (b) lower spring.

The mitigation system is mounted on an Adept Viper s650. The robot hits a polystyrene beam at a constant speed of 100 ms^{-1}. A marker is attached to the moving part of the mitigation system and is tracked at $\sim$ 300 fps by a high frame-rate camera. In Figure 68, frames captured during the worst-case scenario are shown. In this scenario, the rotational system must rotate 180° to align the bi-stable mechanism to the direction of the impact.

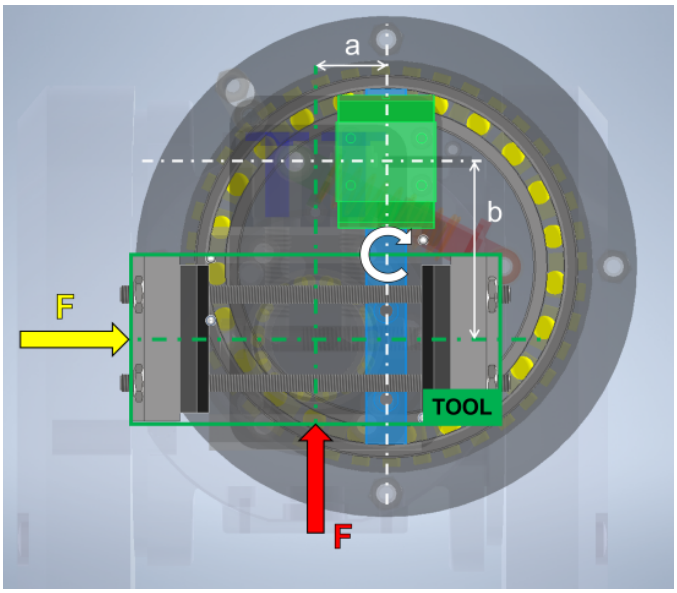


Figure 66: Detail of the displacements between the tool and the cursor to generate the momentum to rotate the mechanism

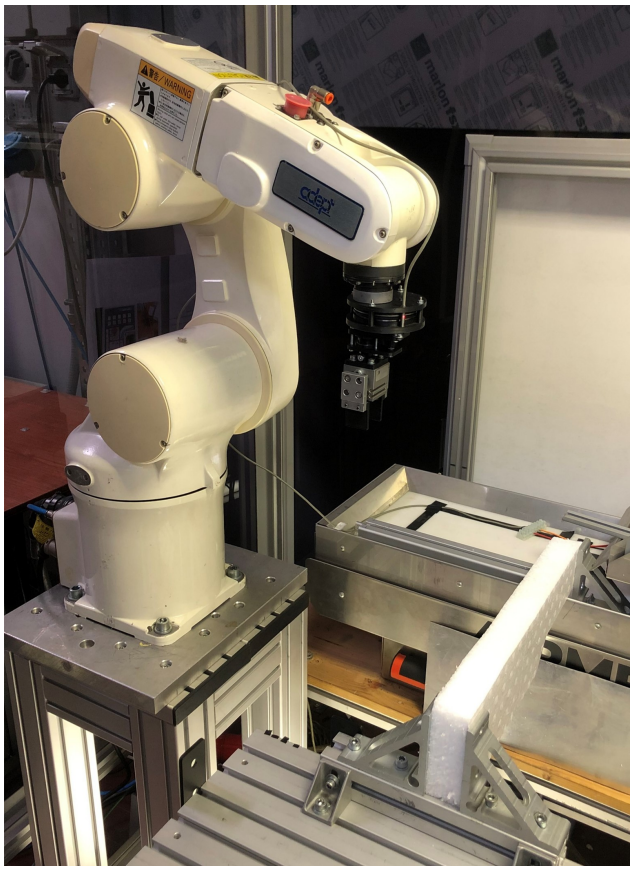
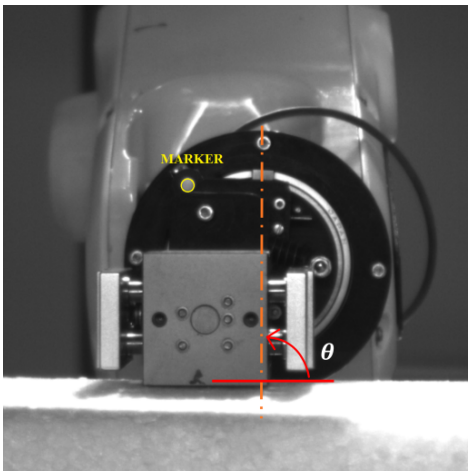
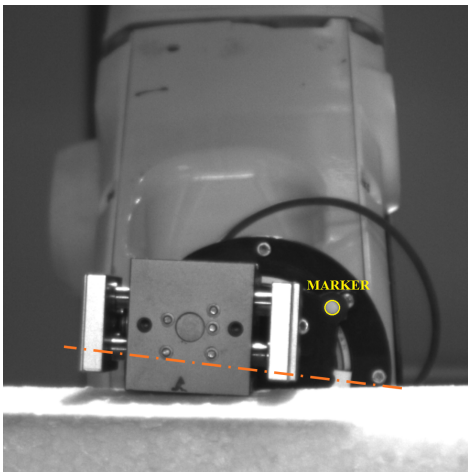


Figure 67: Experimental setup for XY-prototype validation.

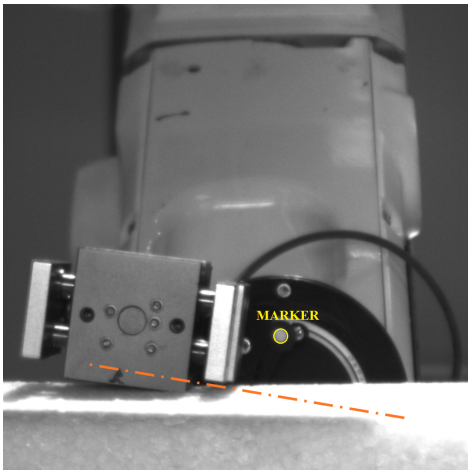
The sequence of frames shows that the system successfully delays the collision with a hard surface. However, it is worth noticing that, in



(a)



(b)

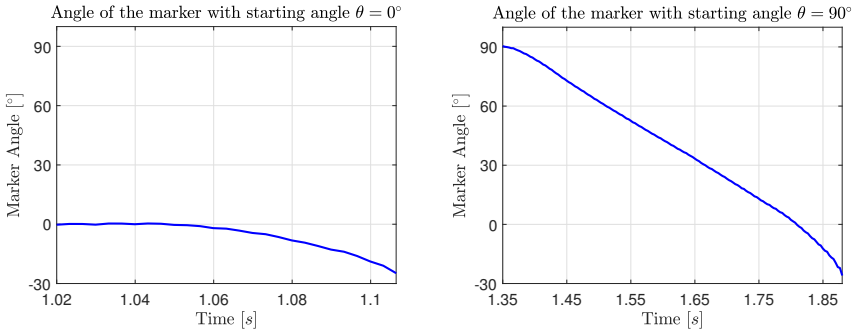


(c)

Figure 68: Succession of frames in the worst-case scenario (rotation of 180° to align bi-stable mechanism along the direction of the impact): (a) instant of the impact; (b) instant in which the bi-stable mechanism starts to push; (c) bi-stable mechanism fully extended.

the worst-case scenario, the bi-stable mechanism worked only from Figure 68b to frame 68c. Therefore, the rotation alone is sufficient to avoid damage. However, if the control system is inefficient, the bi-stable mechanism allows for an additional rotation. For example, if Figure 68c is considered, the switching of the bi-stable mechanism grants the robot more space to brake.

The tests were carried out by modifying the approaching angle θ, shown in Figure 68a. θ is the angle between the impacted surface and the linear guide's axis. The system is useful when $\theta \geqslant 0^\circ$. If $\theta < 0^\circ$, the bi-stable mechanism is immediately engaged in the impact, without the need for rotations. If $\theta \geqslant 0^\circ$, the system must rotate to free the bi-stable mechanism. Figures 69a and 69b show the variation of the angle of the mechanism in the two extreme cases that requires a rotation: $\theta = 0^\circ$ and $\theta = 90^\circ$. $\theta = 90^\circ$ is the worst-case scenario depicted in Figure 68.



(a) (b)

Figure 69: Rotation angle of the mechanism during the collision with starting angle (a) $\theta = 0^\circ$; (b) $\theta = 90^\circ$

The plots are cut in the instant in which the bi-stable mechanism starts to move. It is interesting to observe that the final angle is comparable. Therefore, the system shows repeatability in the results, with a mean angle equal to $-24.73 \pm 0.64^\circ$. The final angle corresponds to the instant in which the force of the external perturbation, exerted along the guide's axis, surpasses the force opposed by the bi-stable mechanism.

In conclusion, the 2D mitigation prototype is able to exploit the rotational system to correctly align the bi-stable mechanism along a direction in which it can work. However, it is essential noticing that due to the force necessary to reorient the system, the proposed solution is suitable only for impacts with hard surfaces. Nevertheless, if the robot is required to sort objects arranged on a plane, its final joint can be oriented to optimize the use of the bi-stable system. Consequently, limiting the 2D system to impacts with hard surfaces is acceptable. Indeed, it can function when unexpected impacts occur in other directions. Moreover, its rotation can be used to overcome inaccurate measures of the orientation of the objects to be moved by

rotating the tool accordingly. Finally, it can be inferred that the prototype can be extended to three dimensions by superimposing the Z prototype developed in the previous Section.

3.6 CONCLUSIONS

This Chapter proposed an extension of the mathematical model of Chapter 2. The new model considers impacts with hard surfaces. These impacts are more severe than those with small objects since higher forces are at play.

Experimental tests carried out on a prototype impacting a soft obstacle corroborated the validity of numerical results. It is worth noticing that the novel solution has a positive effect on the operating velocity of the robot since high velocities, that can damage the robot with a rigid end-effector in case of collision, can be tolerated if the solution with the bi-stable mechanism is adopted. The designed bi-stable mechanism has some significant advantages with respect to other collision-mitigation technologies. In particular, the proposed solution can be introduced in existing robots with only minor modifications in the control system. Moreover, building the new solution is relatively simple and does not require expensive components and sensors. In addition, a new prototype is designed in the form of a module to be interposed between the tool and the robot flange. Any tool (e.g., a gripper for assembly operations, a paddle) can be attached to it to take advantage of the mitigation of collisions provided by the bi-stable mechanism. The generalization of the solution to multiple tasks increases the flexibility of the bi-stable mechanism. However, according to the envisaged application, the end effector design can be optimized according to the stiffness of the potential obstacles and the operational velocities. Indeed, the maximum impact velocity depends on these factors as well as on the efficiency of the control system. The latter affects the dead-time between the impact and the reaction. Therefore, depending on these parameters, the module requires modifications to its geometric parameters to ensure a better response.

Finally, an extension to 2D is proposed. The bi-stable mechanism is coupled with a rotational device that reorients it along the direction of the impact. The 2D prototype has been tested experimentally, validating its functioning. In its actual design, the prototype is suitable to mitigate impacts with hard surfaces only due to the forces necessary to reorient the tool. However, it can also be helpful against impacts with small objects when the object's orientation is inaccurate.

In conclusion, the next Chapter presents a different approach to mitigating impacts. A multi-physics approach is employed to evaluate the effectiveness of the bi-stable mechanism with respect to other solutions and get more insights on the impacts in general.

4

MULTI-PHYSICS APPROACH TO COLLISION MITIGATION

This study presents an alternative solution to the bi-stable mechanism proposed in the previous Chapters. Using a multi-physics approach to collision mitigation, this study presents a hydraulic solution to mitigate collisions between the robot and either a moving or a fixed object. A comparison with the bi-stable mechanism is proposed to look at the pros and cons of the two solutions.

4.1 HYDRAULIC COMPLIANT SYSTEM

In Chapters 2 and 3, collision mitigation is achieved by employing a bi-stable mechanism that decouples the tool from the robot. The mechanism exploits a non-linear spring to achieve the opposite goals of offering an initial resistance to the impact to perform the task and mitigating forces when a certain threshold is achieved.
The study in this Chapter develops an alternative solution to the mitigation to evaluate the bi-stable mechanism's effectiveness or have insights on how to improve it. The novel solution adopts a multi-physics approach in the form of a hydraulic system. Again, initially, impacts with moving objects are considered. Then, the model is improved considering hard surfaces.
The hydraulic system is installed between the robot flange and the end-effector. It consists of a main hydraulic cylinder connected to an external auxiliary cylinder by means of an orifice. The auxiliary cylinder is divided into two chambers by a floating piston. The first chamber is filled with the liquid that has moved from the main to the auxiliary cylinder. The second chamber is filled with gas pressurized by a pneumatic line. Figure 70 schematizes the hydraulic system.
The functioning recalls the bi-stable mechanism. During the collision, the end-effector moves backward since the dynamics of the end-effector and the robot are decoupled. After the collision, the liquid is forced to flow from the main cylinder to the auxiliary cylinder through the orifice. The pressure drop through the orifice and the increase in pressure due to gas compression gradually reduce the relative velocity between the end-effector and the robot, avoiding hard impacts between the two. Indeed, the movement of the end-effector is restricted by the robot flange on one side and by a mechanical end-stop on the other side.

The compliant system is equipped with an impact detector based on a magnetic sensor, typically used in cylinders to detect the move-

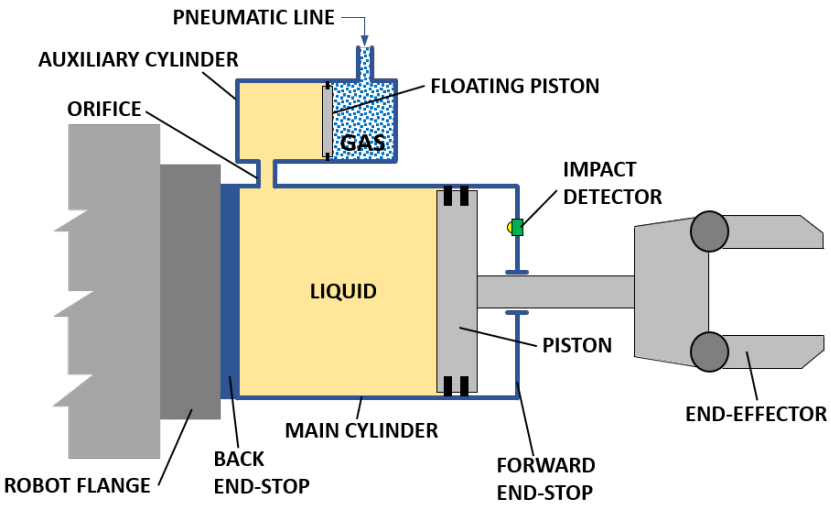


Figure 70: Scheme of the end-effector with the hydraulic compliant system.

ment of the piston. This sensor switches on when the piston moves away by 1.5 mm from the forward end-stop. Therefore, the robot can detect the collision and react, for instance, by stopping its movement. It is worth remembering from the previous Chapters that the robot is characterized by a delay in the reaction to the collision. This time, called dead-time, is due to the time needed for detecting and processing of the signal from the impact detector. The dead time is a feature of the considered robot and signal processing system. As for the bistable mechanism, the dead time is constituted of the following time intervals:

1. the time needed for the system to move enough to activate the signal;
2. the time for the controller to detect the sensor activation and notify the control system;
3. the time for the control system to analyze the sensor signal and deploy a reaction to the impact;
4. the time for the robot to receive the new command and execute it.

The hydraulic system is designed to adapt its compliance to different scenarios. By adjusting the pressure of the gas, the robot can perform specific tasks, such as assembling or plugging, for which a completely compliant system would not be suitable. In addition, the gas in the auxiliary cylinder is pressurized to pre-load the end-effector to prenvent movements due to robot accelerations and restore the initial position of the end-effector after the collision.

The presented design can be extended to a compliant system acting in multiple directions. Relying on [27], in which the mathematical

model of a N-DOF compliant structure is presented, a set of hydraulic systems arranged in series and connected by pre-loaded hinges is able to absorb unexpected collisions from multiple directions.

4.2 IMPACTS WITH SMALL OBJECTS

As for the bi-stable mechanism, the hydraulic solution is first evaluated to mitigate collisions with small objects free to move on a plane. The mathematical model analyses the collision between a small object (hundreds of grams) and the end-effector with the hydraulic compliant system. The simulations aim to highlight the effects of this decoupling system in terms of mitigation of collisions on the robot, the end-effector and the object. The analysis considers only a one-dimensional model of the collision, since it is assumed that the end-effector approaches the object along a linear trajectory. The links of the robot are assumed rigid, and the joints are assumed without compliance and clearances, since the stiffness of links and joints are much larger than the stiffness of the decoupling system [56, 57]. Hence, the robot is equivalent to a rigid plate, moving along the direction of approach with a trapezoidal velocity profile. The end-effector and the object are modelled as lumped masses. The hydraulic decoupling system is schematized using a non-linear lumped parameter model. Three phenomena are considered in the lumped model of the hydraulic system [85]:

- The pressure drop through the orifice, having a diameter much smaller than the diameter of the main cylinder, which is given by:

$$\Delta p = \frac{1}{2} c_0 \rho_L \frac{A_p^2}{A_o^2} (\dot{x}_r - \dot{x}_t)^2 \tag{46}$$

  where c_0 is a coefficient, ρ_L is the liquid density, A_p and A_o are the areas of the main cylinder and the orifice respectively, $\dot{x}_r$ and $\dot{x}_t$ are the velocities of the robot and the tool respectively.

- An adiabatic variation in gas pressure in the auxiliary cylinder is assumed since there is no time to heat exchange due to the impulsivity of the collision. Hence, gas pressure p_1 is given by the following equation:

$$p_1(t) = \frac{p_{10} \cdot V_{g,0}^k}{V_g(t)^k} \tag{47}$$

  where p_{10} and $V_{g,0}$ are the initial pressure and volume of the gas; k is the adiabatic constant of the gas (in this case air with $k = 1.4$); V_g is the effective volume of the gas chamber.

- The friction force between the pistons and the cylinders, which includes both a dry friction term (coefficient d) and a viscous term (coefficient b) [85]:

$$F_f = d \cdot \mathrm{sgn}(\dot{x}_r - \dot{x}_t) + b \cdot (\dot{x}_r - \dot{x}_t) \tag{48}$$

Figure 71 shows the lumped parameter model. The variables x_r, x_t and x_p are the coordinates of the robot flange, the end-effector and the object, respectively.

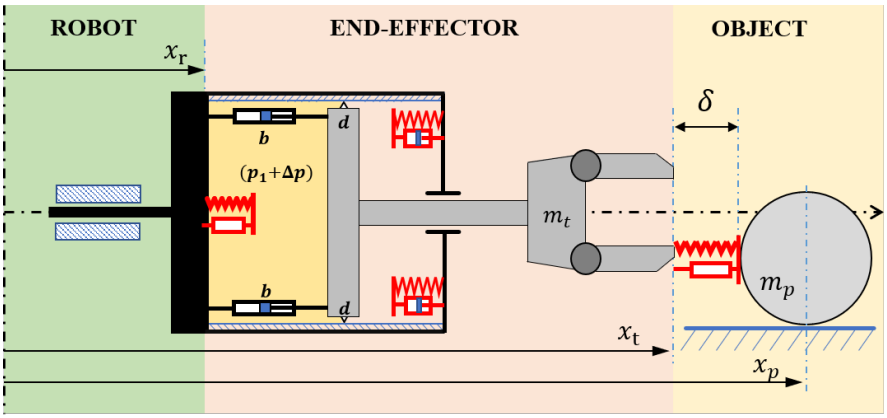


Figure 71: Lumped parameter model of the system for impacts with small objects

The equations of motions of end-effector and object during the collision are:

$$m_t \ddot{x}_t = (p_1 + \Delta p) \cdot A_p - F_c + F_f \tag{49}$$

$$m_p \ddot{x}_p = F_c \tag{50}$$

in which m_t and m_p are the mass of the end-effector and the object, respectively; F_c is the contact force between the end-effector and the object, which is expressed using a non-linear contact model as presented in [38, 50]:

$$F_c = k_c \delta^{\frac{3}{2}} + \chi_c \delta^{\frac{3}{2}} \dot{\delta} \tag{51}$$

where k_c and χ_c are constants.

The force on the robot due to the collision is calculated as follows:

$$F_r = -(p_1 + \Delta p) \cdot A_p - F_f + F_{f,e} + F_{b,e} \tag{52}$$

in which $F_{f,e}$ and $F_{b,e}$ are the contact force due to the impact between the piston of the end-effector and the forward and backward end-stops, respectively. These forces are expressed using the same non-linear formulation presented in Chapter 2.

The non-linear analytical model was implemented in Simulink to carry out the simulations presented in the following Section.

4.2.1 *Numerical results and comparison with bi-stable mechanism*

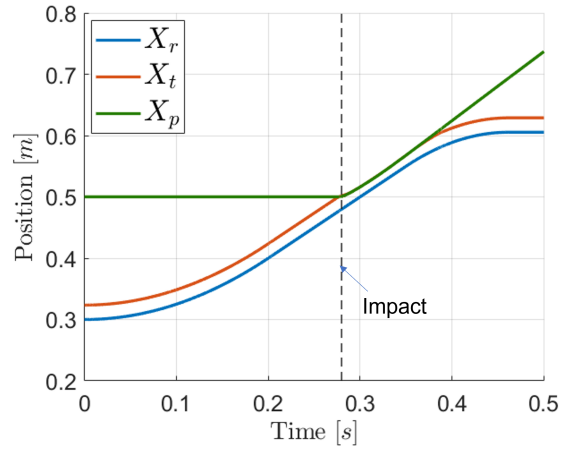
The results of the numerical simulations of the end-effector with the hydraulic system are presented and compared with the ones obtained with the bistable mechanism. It is assumed that the robot approaches the object with a trapezoidal velocity profile and starts to decelerate after a certain dead time (DT) from the collision detection. The final velocity of the robot is set to 0 ms^{-1}. The simulations were carried out using the values in Table 12. The parameters L, R_p and R_o represent the stroke of the piston, the radius of the main cylinder and the radius of the orifice, respectively. $\dot{x}_{r_{max}}$ is the robot velocity before the impact with the objects.

Parameter	*Unit*	*Value*
L	m	0.023
m_t	kg	0.048
R_p	m	0.020
m_p	kg	0.246
R_o	m	0.008
p_{10}	Pa	1000
c_o	/	3
$V_{g,0}$	m^3	$7.63 \cdot 10^{-5}$
b	kgs^{-1}	0.5
ρ_L	kgm^{-3}	900
d	N	0
$\dot{x}_{r_{max}}$	ms^{-1}	1
k_c	$Nm^{-1.5}$	$0.6 \cdot 10^6$
χ_c	$Nsm^{1/2}$	$1.35 \cdot 10^6$
k_{be}	$Nm^{-1.5}$	$1 \cdot 10^6$
e_{be}	/	0.3
k_{fe}	Nm^{-1}	$1 \cdot 10^6$
c_{fe}	Nsm^{-1}	235
DT	s	$70 \cdot 10^{-3}$

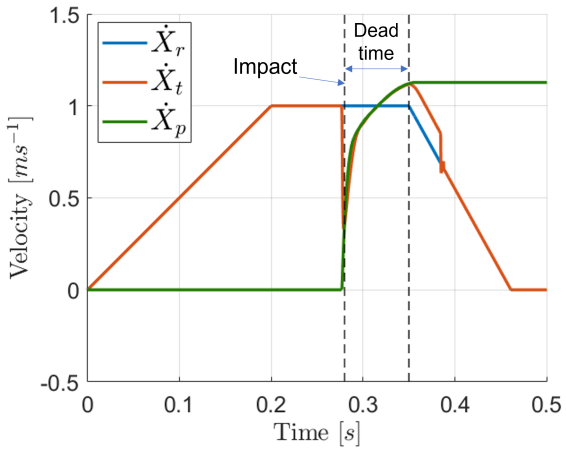
Table 12: Values of the parameters of the model used in simulations of impacts with small objects.

Figure 72 shows the calculated positions and velocities. The vertical line represents the instant at which the impact occurs. After that instant, a dead time DT = 70 ms is assumed.

At the impact, the end-effector is slowed down. Then, after the initial transient, the end-effector and the object move together. Therefore,

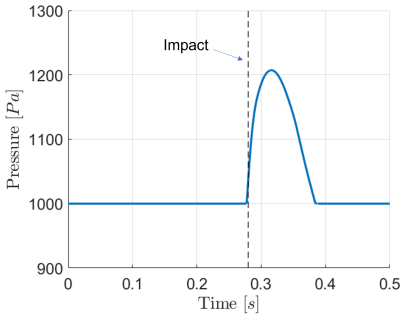


(a)

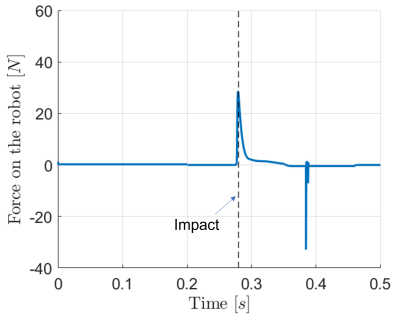


(b)

Figure 72: Simulation results for the hydraulic system with DT = 70 ms: (a) positions and (b) velocities for robot ($X_r, \dot{X}_r$), end-effector ($X_t, \dot{X}_t$) and object ($X_p, \dot{X}_p$)



(a)



(b)

Figure 73: Simulation results for the hydraulic system with DT = 70 ms: (a) pressure inside the gas chamber and (b) force on the robot F_r

Parameter	Unit	Value
N	/	1
L_0	m	0.062
k_b	Nm^{-1}	490
C	m	0.047
$C_1 = C_2$	m	0.0235
m_t	kg	0.048
μ	/	0.23
v_{end}	ms^{-1}	0

Table 13: Values of the parameters used in Chapter 13 to validate the bi-stable mechanism impacting against small objects.

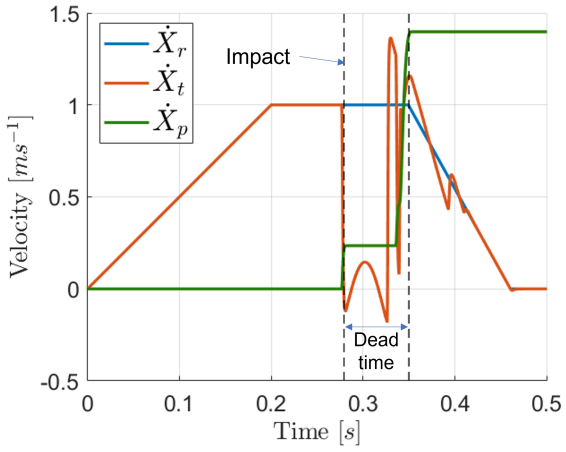
the object is exerting a force on the end-effector that keeps active the flow between the chambers. The end-effector continues moving backward until the pressure in the gas chamber stops the flow. In Figure 73a the pressure inside the gas chamber is depicted. It can be observed that the pressure has a maximum in correspondence to the instant in which the robot starts decelerating. The velocity of the end-effector is larger than the velocity of the robot during braking, hence gas expands.

To analyze the performance of the system, Figures 74a and 74b show the position and velocity plots of the bi-stable mechanism proposed in Chapter 2. The parameters used for the simulations are listed in Table 13.

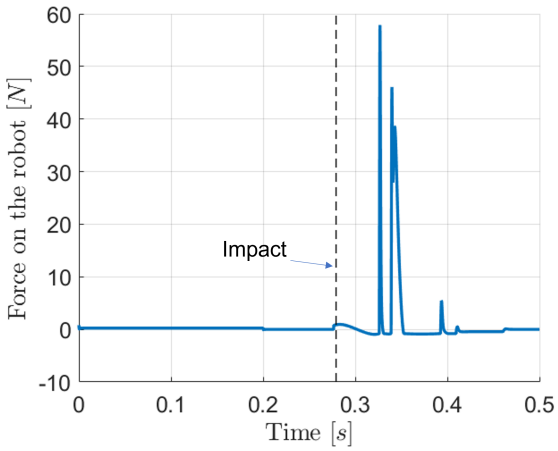
Both systems have a non-linear behavior. However, since the bistable mechanism exploits a mechanical phenomenon, its dynamics are more jerky. Consequently, the forces exerted on the robot include large impulses (see Fig. 74b). Moreover, the final velocity of the object is lower with the hydraulic system than the one reached with the bistable mechanism.

This result is true when there is a certain dead-time between the impact and the reaction. Figures 74 and 75 show that the object velocity with the bistable mechanism depends on the number of collisions with the end-effector. If the control system is improved, consequently reducing the dead time, the bistable mechanism is more efficient regarding to momentum reduction. This change is due to the fact that the subsequent impacts, if they occur, happen when the robot has already decelerated. However, it is still the worst system when forces on robot are considered, as can be inferred from Figure 76.

The forces with the hydraulic system are lower than the ones with the bistable mechanism, as can be inferred from Figure 76. However, it is worth noting that the force peaks occur on different instants. For the hydraulic system, there are two main peaks. The first corre-



(a)

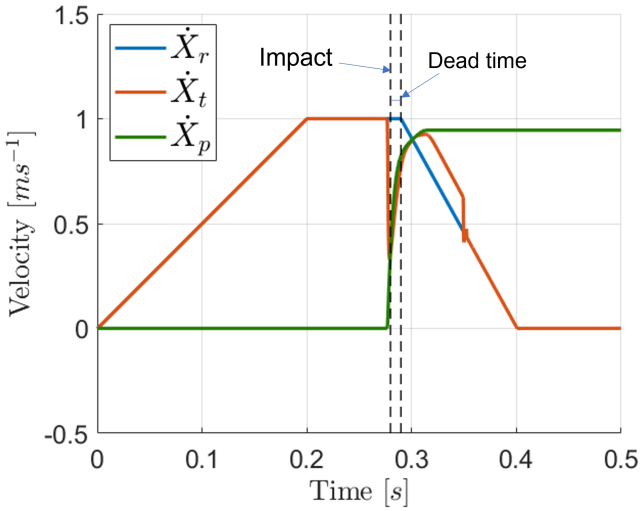


(b)

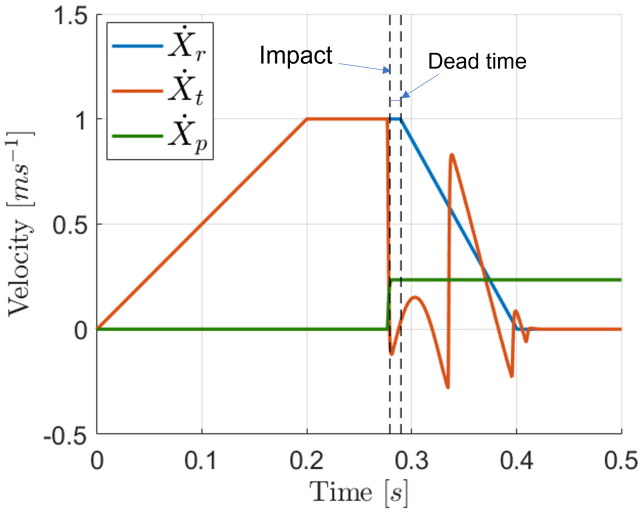
Figure 74: Simulation results for bi-stable mechanism with same general parameters as the hydraulic one, such as approaching velocities, end-stops and objects characteristics. (a) Velocities of the three components of the systems; (b) force on the robot F_r

sponds to the impact with the object, whereas the second (having opposite sign) corresponds to the impact of the end-effector with the front end-stop. Since the end-effector is pushed back to the starting position when the gas pressure is released. Conversely, the bistable mechanism shows three peaks. The first corresponds to the impact with the object and its the minor of the three. The second and the third correspond to impacts with the back end-stop. They are caused by the presence of the spring that pushes the end-effector against the back end-stop.

In summary, with the hydraulic solution, the maximum force is reached when the end-effector impacts the object, whereas with the bistable mechanism, it is reached when the end-effector impacts the back end-stop. Therefore, with low dead time, the bistable mechanism can be



(a)

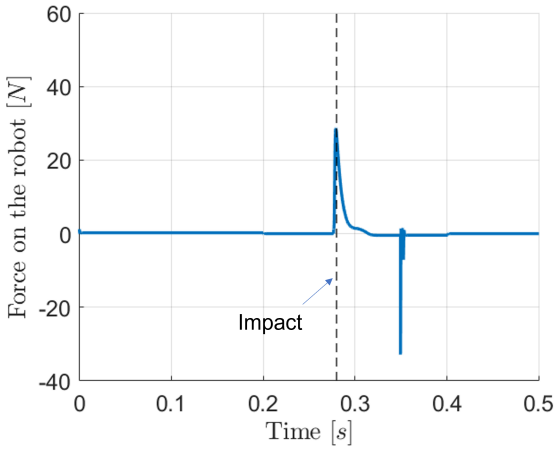


(b)

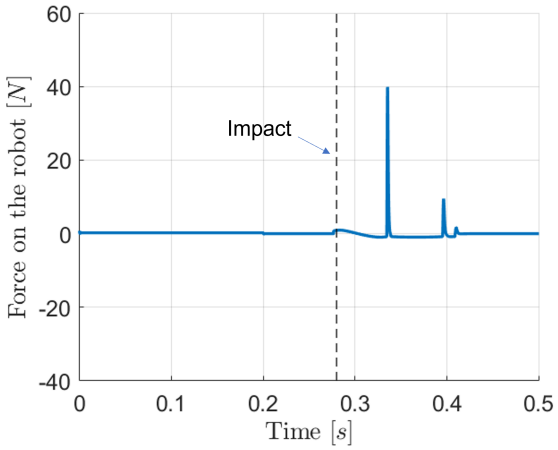
Figure 75: Comparison between velocities with DT $= 10$ ms for the (a) hydraulic system and (b) bistable mechanism

preferred because it transfers less momentum to the object in the first collision.

In conclusion, this study shows that the hydraulic solution is a valid alternative to the bi-stable mechanism in mitigating collisions. Furthermore, the comparison shows how the two different behaviors are reflected in the interaction with the object. The main differences are related to the instant in which the force is transmitted to the robot and the fact that, after the impact, the tool is detached from the object when using the bi-stable mechanism. Therefore, the preferable system depends on the task and the control system's efficiency.



(a)



(b)

Figure 76: Comparison between forces exerted on the robot with DT = 10 ms for the (a) hydraulic system and (b) bi-stable mechanism

4.3 IMPACTS WITH HARD SURFACES

The previous Section presents the work related to impacts with moving objects. As stated in Chapter 2, the collisions with small moving objects are not dangerous for the robot, to the point that its compliance's influence on impacts is negligible. Indeed, a small object is unable to exert a force sufficient to deform the robot or modify its trajectory. However, it is shown how the robot's dynamic is engaged during impacts with hard surfaces due to the increased forces.

In this section, impacts with hard surfaces are considered and the model is modified accordingly. The new lumped parameter model of the system is shown in Figure 77. The variables x_r, x_t, x_o are the coordinates of the robot flange, the end-effector, and the obstacle, respectively. x_c is the desired position of the robot as imposed by the control system, whereas x_r is the actual position of the robot. x_r is

different from x_c since it considers the inertia and compliance of the robot.

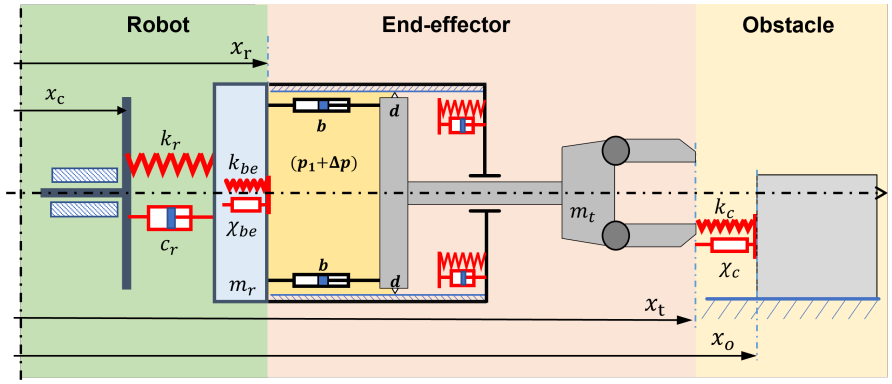


Figure 77: Lumped parameter model of the system for impacts with hard surfaces

The equations of motion presented in Equations 49 and 50 are modified accordingly to the new scenario:

$$m_r\ddot{x}_r = c_r(\dot{x}_c - \dot{x}_r) + k_r(\dot{x}_c - \dot{x}_r) - F_{hyd} - F_f - F_e \tag{53}$$

$$m_t\ddot{x}_t = F_{hyd} + F_f + F_e - F_c \tag{54}$$

where F_e is the force exerted by the end-stop (either the front or the back one depending on the position of the tool), F_f is the friction force described by Equation 48, F_c is the contact force defined with the same formulation as 51, and F_{hyd} is the hydraulic force following from Equations 46,47 and 49:

$$F_{hyd} = (p_1 + \Delta p)A_p \tag{55}$$

$$= \frac{p_{1,0}V_{g,0}^k}{V_g(t)^k}A_p + \frac{1}{2}c_0\rho_L\frac{A_p^3}{A_o^2}(\dot{x}_r - \dot{x}_t)^2 \tag{56}$$

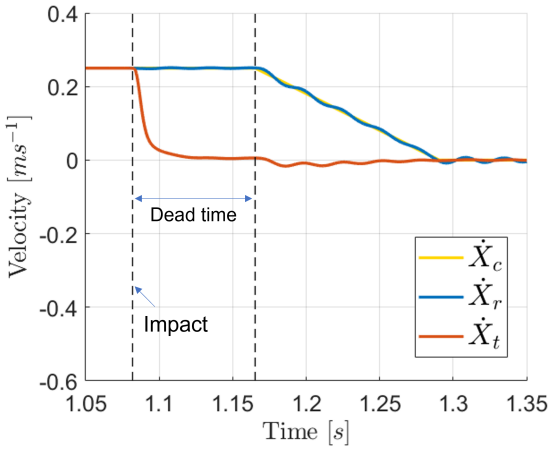
The analytical model is simulated using Simulink. The results and a comparison with the bi-stable mechanism is proposed in the next Section.

4.3.1 *Numerical results and comparison with the bi-stable mechanism*

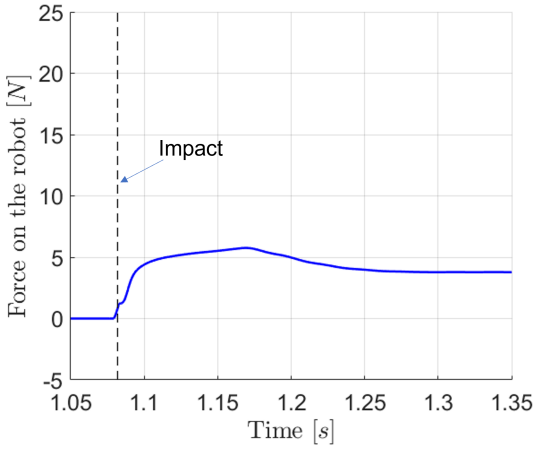
Simulations were carried out with the parameters in Table 12 with the modifications listed in Table 14. The modification are done to match the values used in Chapter 3 to validate the bi-stable mechanism.
The robot moves linearly along a direction following a trapezoidal velocity profile. It hits the obstacle while moving at a constant speed. The robot considered in the simulations is an Omron Adept Viper s650 to keep consistency with the simulations in Chapter 3. The estimation of the parameters of the robot is described in Section 2.2.1.
Figure 78 shows the simulation results of the modified model.

Parameter	*Unit*	*Value*
$\dot{x}_{r_{max}}$	ms^{-1}	0.25
k_c	$Nm^{-1.5}$	$0.3 \cdot 10^5$
χ_c	$Nsm^{1/2}$	$4.5 \cdot 10^4$
k_{be}	$Nm^{-1.5}$	$2.25 \cdot 10^6$
e_{be}	/	0.3
k_{fe}	Nm^{-1}	$2.25 \cdot 10^6$
c_{fe}	Nsm^{-1}	235

Table 14: Modified values of the parameters in Table 12 for the simulation of impacts of the hydraulic system with hard surfaces.



(a)



(b)

Figure 78: Simulation results of impacts with hard surfaces with the hydraulic system and DT = 70 ms showing (a) velocities of robot ($\dot{X}_c$,$\dot{X}_r$) and end-effector ($\dot{X}_t$), and (b) force on the robot

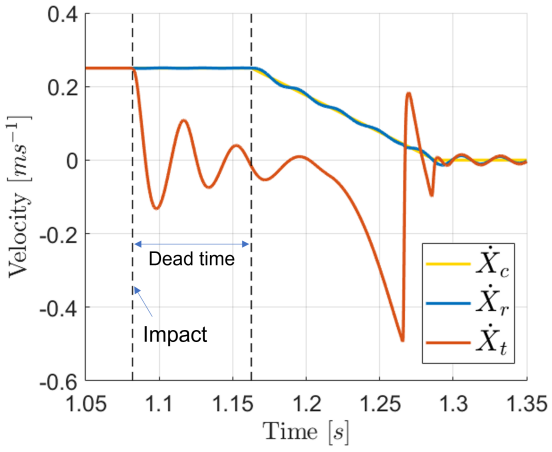
Interestingly, the tool velocity goes smoothly towards 0 ms^{-1} and then oscillates around that velocity. This particularity means that the tool tends to remain in contact with the external surface. Therefore, the system acts as if the robot is actively pushing against the liquid mass. This phenomenon is well shown in Figure 78b. After the impact, the force on the robot increases because the dead-time causes the robot to push against the wall at a constant velocity, compressing the gas. Conversely, when the robot starts to decelerate, the force decreases due to the reduction of the friction force.
These plots are meaningful if compared with the one gathered simulating the bi-stable mechanism. The bi-stable mechanism is simulated using the same values used in Chapter 3 to validate the model. The values are listed in Table 15.

Parameter	Unit	Value
N	/	1
L_0	m	0.062
k_b	Nm^{-1}	490
H	m	0.053
C	m	0.047
C_1	m	0.0235
C_2	m	0.0235
V_c	ms^{-1}	0.25
m_t	kg	0.048
μ	/	0.12
v_0	ms^{-1}	0.001

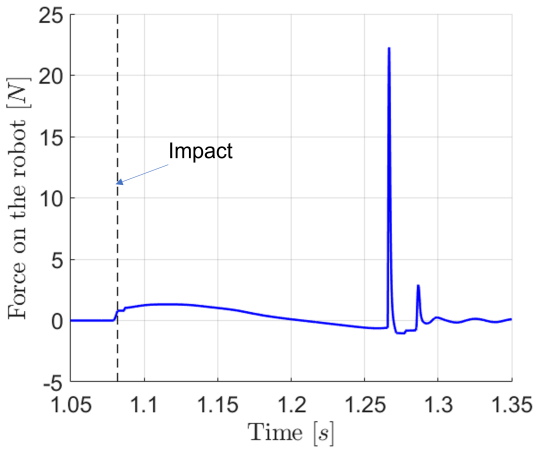
Table 15: Values of the parameters used in Chapter 3 to validate the mathematical model.

The simulated plots are showed in Figure 79.

The hydraulic system is better the bi-stable mechanism also when impacts with hard surfaces are considered. In particular, the tool oscillates close to the surface, whereas the bi-stable mechanism is subjected to a series of strong oscillations. Those oscillations results in a higher force transferred to the robot. However, as seen in the case of impacts with moving objects, reducing the dead-time can be beneficial. Indeed, comparing Figures 78b and 79b, it is clear that the bi-stable mechanism exerts lower forces on the robot than the hydraulic solution during the first instants after the impact.
Figure 80 shows the velocity and force plots for the hydraulic and the bi-stable mechanism in the case of DT $= 10ms$. From Figure 80c, it can be inferred that the external perturbation is insufficient to make the bi-stable mechanism switch. Therefore, in this case, the reduction



(a)



(b)

Figure 79: Simulation results of impacts with hard surfaces with the bi-stable system and DT = 70 ms showing (a) velocities for robot ($\dot{X}_c$,$\dot{X}_r$) and end-effector ($\dot{X}_t$), and (b) force on the robot

in the dead-time eliminates the impacts of the tool against the back end-stop. Consequently, the peaks of Figure 79b are not present anymore. In conclusion, comparing Figure 80b with 80d, it is clear that the bi-stable mechanism is better than the hydraulic one regarding the force transferred to the robot.

4.4 ESTIMATION OF AN EQUIVALENT STIFFNESS

In the previous Sections, a comparison between the bi-stable mechanism and the hydraulic system is proposed. The two systems' characteristics are chosen to have a comparable initial stiffness. In this way, they should react similarly to the first impact, whereas their internal behavior characterizes the dynamics of the components during the following instants. Therefore, this Section proposes the estimation of

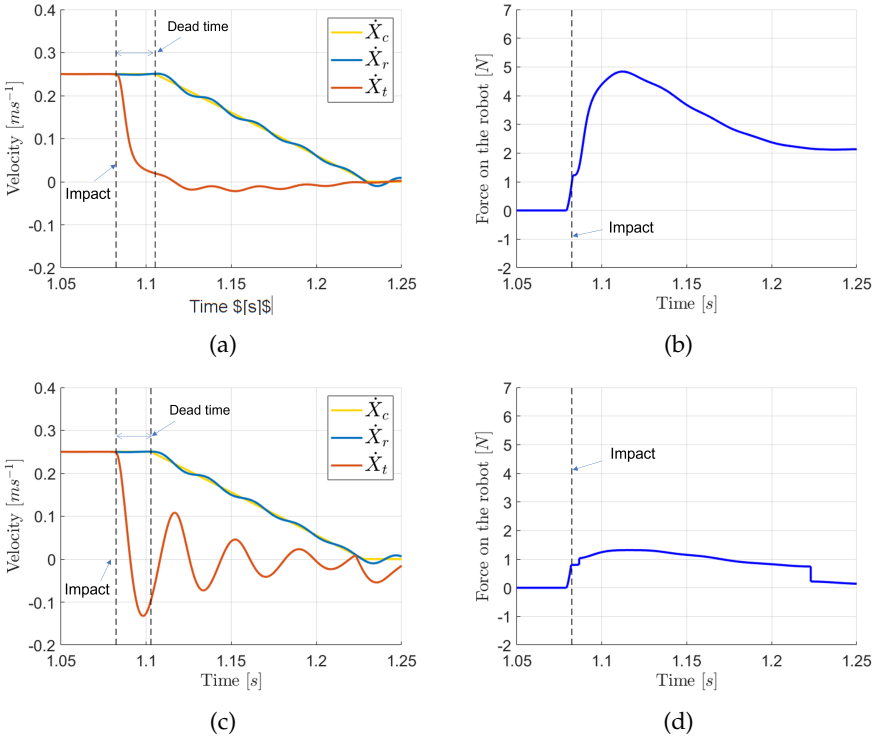


Figure 80: Comparison of velocities and forces during impacts with hard surfaces and DT = 10 ms with: (a,b) hydraulic system, and (c,d) bi-stable mechanism.

an equivalent stiffness for the hydraulic system. This estimation allows the selection of the values of the parameters suitable for the comparison.

The equivalent stiffness is estimated for the hydraulic system at the instant of the impact. Therefore, the hydraulic force in Equation 56 must be expressed with respect to the movement x_t of the tool. Moreover, the terms dependent on the velocities can be neglected. Consequently, the derivation of the equivalent stiffness is reduced to expressing the pressure p_1 of the gas in function of x_t.

Starting from Equation 47, the volume can be expanded using Taylor series:

$$V_g(t) = V_{g,0}\left(1 + \frac{\Delta V}{V_{g,0}}\right) \tag{57}$$

$$= V_{g,0}\left(1 + \frac{A_p(C - x_t)}{V_{g,0}}\right) \tag{58}$$

where ΔV is the variation of volume in the cylinder due to the movement of the tool, and C is the height of the cylinder.

Equations 47 and 58 can be combined together:

$$p_1(t) = \frac{p_{1,0} \cdot V_{g,0}^k}{V_{g,0}^k \left(1 + \frac{A_p(C - x_t)}{V_{g,0}}\right)^k} \tag{59}$$

$$= \frac{p_{1,0}}{\left(1 + \frac{A_p(C - x_t)}{V_{g,0}}\right)^k} \tag{60}$$

Considering an infinitesimal movement of the tool at the impact, Equation 60 can be expanded using Taylor series as follows:

$$p_1(t) = p_{1,0} \left(1 + \frac{A_p \cdot k \cdot (x_t(t) - C)}{V_{g,0}}\right) \tag{61}$$

Therefore, the hydraulic force becomes:

$$F_{hyd}(t) = p_{1,0} \left(1 + \frac{A_p \cdot k \cdot (x_t(t) - C)}{V_{g,0}}\right) A_p \tag{62}$$

Deriving with respect to tool's movement x_t returns the equivalent stiffness of the system:

$$k_{hyd} = \frac{\delta F_{hyd}}{\delta x_t} = \frac{A_p^2 \cdot k \cdot p_{1,0}}{V_{g,0}} \tag{63}$$

whereas the stiffness of the bi-stable mechanism is given by the equation of the non-linear spring:

$$k_{bm} = k \left(\frac{L_0 - L}{L}\right) \tag{64}$$

The parameters in Table 14 have been chosen to gather a similar stiffness between the two systems. Those values lead to a stiffness $k_{hyd} = 27.28\ Nm^{-1}$ for the hydraulic system, and $k_{bm} = 34\ Nm^{-1}$ for the bi-stable mechanism.

4.5 CONCLUSIONS

The study in this Chapter presents a novel multi-physics approach to collision mitigation. The hydraulic system proposed is a valid alternative to the bi-stable mechanism, and it clearly shows how different tool designs lead to completely different tool behaviors. Moreover, it shows how different tool behaviors can be useful depending on the task and the control system's efficiency.

Simulations demonstrate that the hydraulic solution is able to mitigate the impacts between the robot and both small objects and fixed

obstacles. In particular, simulations show that the hydraulic system can outperform the bi-stable mechanism proposed in the previous Chapters when high dead-times occur. This inference comes from the fact that the hydraulic solution can exert smaller forces on the robot and a smoother movement of the compliant system. In addition, another advantage is the easiness of regulation since the pressure of the gas is the only variable of the system. In contrast, in its actual design, the bi-stable mechanism must be dismounted to modify its stiffness. This advantage must be considered when there is high variability in the environment. However, the bi-stable mechanism should be preferred if more efficient control systems are employed and lower dead times are achieved. Indeed, in those conditions, it provides a more significant reduction in the momentum transferred to objects. Moreover, when impacts with hard surfaces are considered, the bi-stable mechanism can also see a reduction of the forces on the robot if the delay is small enough to prevent the switching of the tool and the consequent impact with the back end-stop.

5

COLLISION AVOIDANCE IN A COLLABORATIVE FRAMEWORK

The work carried out in the previous Chapters aimed to mitigate collisions after they occurred. Their goal was to design mechanical systems that passively react to unexpected impacts reducing the force transferred to the robot and the external environment. In this Chapter, the focus shifts to collision avoidance. The goal is to design solutions to prevent those collisions from happening. In particular, this Chapter presents an active collision avoidance algorithm to prevent collisions in a human-robot collaboration framework.

5.1 COLLISION AVOIDANCE IN HUMAN-ROBOT COLLABORATION

As stated in the introduction of this dissertation, human-robot collaboration (HRC) is an actual research trend [4] that searches to combine human flexibility and dexterity with the robot's repeatability, speed, accuracy, and strength [10]. The critical aspect of this framework is that the collaborative robot is no more constrained inside a fenced cage, safely working at fast speeds far from the human operators. Now, it works actively and responsively with the human operators, sharing the workspace with them.
The modality in which the human operator and the robot share this common workspace defines the type of collaboration. One possible distinction is presented in [86], where the authors defined four categories:

- coexistence, when robot and human operator share the environment without any direct interactions;
- synchronized (or sequential), when they can work on the same area, but sequentially, not at the same time;
- cooperation, when they can work in the same area at the same time but on separate tasks;
- collaboration, when they can work together on the same task and at the same time.

It is clear that, in some conditions, the operator and the robot can interfere with each other during the execution of a task. Therefore, international standards have been defined to determine safety requirements for collaborative robots. These standards can be identified in the technical specification ISO/TS 15066:2016 [26], which extends the Technical Standard UNI EN ISO 10218-1:2011 and UNI EN ISO 10218-2:2011 [25, 87]. Four classes are defined:

- Safety-rated Monitored Stop (SMS), when robot and operators work at different times, with the former completely stopped during the movements of the latter;
- Hand Guiding (HG), when the human operator moves the robot manually;
- Speed and Separation Monitoring (SSM), when the robot's speed and trajectory are dynamically adjusted depending on the human's distance, and the robot is eventually stopped when this distance is too short;
- Power and Force Limiting (PFL), when the robot's control system is designed to limit the contact force to reduce possible damages.

Dedicated robots, called *cobots*, have been designed with particular sensors and procedures to ensure the possibility of operating in one of these four safety classes [32]. These systems are required because the robot is not able to perceive the environment in which it moves. Therefore, increasing its perception is fundamental to exploiting the benefits of collaboration. This property can be achieved through obstacle avoidance, predictive control, and task recognition [33]. For example, in [88], the authors equipped an industrial robot with a 6D force/torque sensor and analyzed its frequency content to delineate the two scenarios of soft contact and hard collision. In the former case, the robot enters an HG mode, whereas, in the latter, it is stopped.

Except for specific tasks, the literature generally offers solutions to the SSM mode. This mode requires the implementation of methodologies to avoid collisions as much as possible, preventing the activation of other safety measures, such as emergency stops. The SSM mode responds to a risky scenario by either reducing the velocities of the robot until eventually stopping it [89] or modifying the robot's trajectory online [90, 91]. Moreover, in an HRC framework, this task must be performed in real-time due to the unpredictability of the human worker.

The approaches generally rely on detecting the human workers with cameras, the embedding of their bodies within simple geometries, and the calculation of their distance with respect to the robot links [92]. The detection of the human operator can be performed using different devices. For example, in [93], the authors used infrared cameras and reflective spheres to detect the human operator and IMU devices attached to the human body to close possible gaps due to occlusions of the vision system. Another example is using relatively simple RGB-D cameras and dedicated algorithms [94] to achieve the same goal. More cameras can be employed to avoid the problem of occlusions of the vision system [91]. Depth cameras are extensively used for their simplicity and reduced cost.

After detection, the human and the robot are generally embedded into simple geometries to speed up the calculation of the distances between the two [92].
Once the distance is calculated, it can be used to reduce the robot's velocity dynamically. For example, a decreasing distance can reduce the robot's velocity depending on the robot's speed or direction [95]. Another possible response is the modification of the robot's trajectory achieved in general by employing Artificial Potential Fields (APF) [91].

In this Chapter, a collision avoidance algorithm is developed to accomplish a cooperative task. Therefore, the human operator and the cobot share the same workspace but work on different tasks. The only assumption in the design is that the human operator and the cobot work face-to-face. Thus, only the wrist of the cobot is expected to collide with the operator, whereas the human operator's collisions with the cobot's links are considered improbable. Nevertheless, the cobot will eventually react to them, thanks to its internal PFL control. The algorithm uses a single depth camera to capture the workspace from above, detect risky scenarios for the robot, and eventually divert the robot's trajectory in real time.
Figure 81 shows a depth image taken during the task. It shows the human operator's hand while reaching something inside the workspace. A pre-processing phase is required to prepare the image for the following operations. Indeed, since the depth image represents the distance of the objects from the camera, the captured image represents the world upside-down. The higher the objects with respect to the plane, the lower their respective points in the image. Therefore, the images must be flipped to represent the actual world. In addition, the actual image can have some pixels with a null value due to external disturbances, such as light or dust. Those pixels have been adjusted by imposing their values equal to the mean value of the surrounding pixels. This simple filter is sufficient for the success of the following image processing described in the next Section.

Furthermore, as can be inferred from Figure 81, the camera captures both the human operator and the cobot. Therefore, they have to be identified. As stated above, most state-of-the-art works rely on human tracking software to detect the human. In contrast, this work adopts an approach similar to the one adopted by [89]. The robot is removed by the image by means of a simple calibration process. In this way, the image presents only obstacles and is easier to analyze.

Whereas implementing traditional algorithms for trajectory generation (APF), the work presented in this Chapter brings two main contributions to the collision avoidance field:

- a method to avoid the estimation of distances by exploiting the interpolation of the environment by means of Radial Basis Functions;

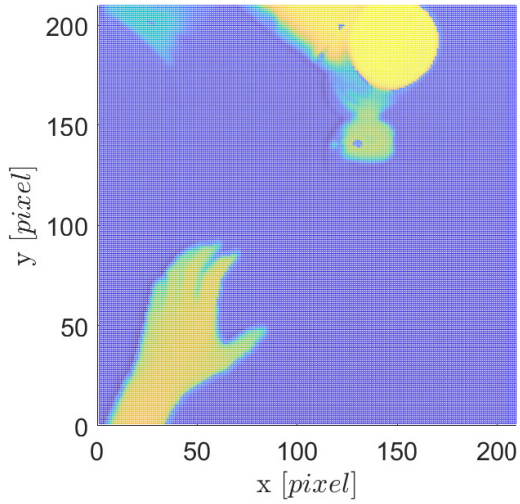


Figure 81: Image captured by the depth camera after a preprocessing phase

- a method to remove the robot from depth images.

The interesting thing about the approach of this work is that it relies entirely on the use of Radial Basis Function, a powerful interpolation tool.

5.2 INTERPOLATION USING RADIAL BASIS FUNCTIONS

The collision avoidance algorithm proposed in this Chapter uses Radial Basis Functions [96] to eliminate the robot from the image and interpolate the environment before planning the trajectory. Radial Basis Functions (RBF) were firstly introduced by [97], and they have been applied in many fields over the years [96]. They are widely used to approximate multivariate functions thanks to their ability to handle high-dimensional problems with scattered data [96]. This property made RBFs suitable as neural network models [98]. In particular, their introduction in this field was driven by their simplicity in modeling nonlinear relationships using a linear combination of the weights of the variables. Moreover, they can be used as an interpolation functions. For example, in [99], authors use RBFs to fit point-clouds representing 3D objects. As a result, they obtained smooth surfaces and were also able to repair incomplete meshes. Similarly, in this Chapter, we used RBFs to approximate depth images to obtain a more regular environment where searching for paths.

RBFs are defined as a combination of functions [96]:

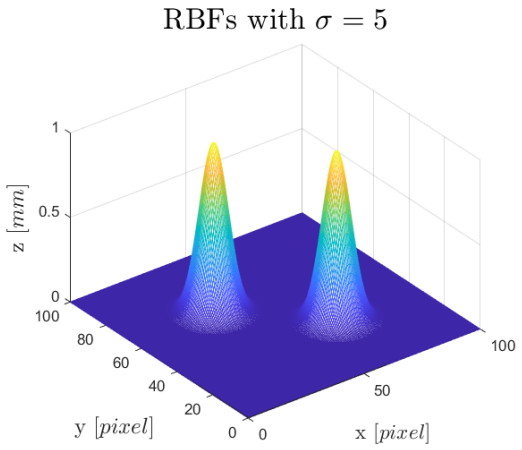
$$s(x) = \sum_{\xi \in \Xi} \lambda_\xi \phi(\|\mathbf{x} - \xi\|) \tag{65}$$

where $\mathbf{x} \in \mathbb{R}^n$, λ_ξ represents the coefficients of the combination, $\phi(\|\mathbf{x} - \xi\|)$ is the basis and Ξ is the set of distinct points used as centers for the RBFs.

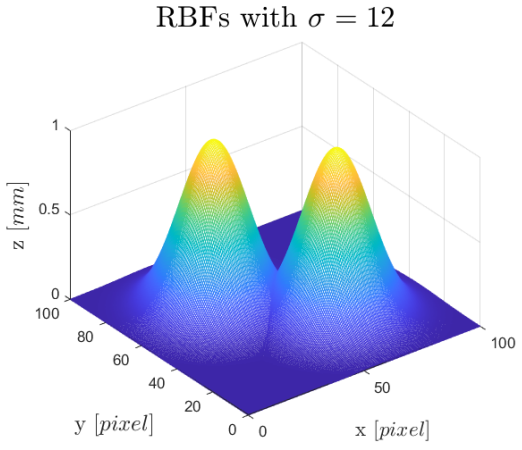
In this work, the RBFs are built using the Gaussian function. Thus, the basis is described by the following equation:

$$\phi(\|\mathbf{x} - \xi\|) = e^{-\frac{\|\mathbf{x}-\xi\|}{2\sigma^2}} \tag{66}$$

where $\mathbf{x}$ are the coordinates (x, y) of the points in which evaluate the function and ξ are the coordinates of the centers of the Gaussian functions. σ is the canonical standard deviation. In this case, σ represents a measure of the influence of a single Gaussian over the others, as can be seen in Figure 82. The higher σ, the wider the spread of the Gaussian function.



(a)



(b)

Figure 82: Mutual influence of RBFs with: (**a**) $\sigma = 5$, (**b**) $\sigma = 12$.

Since the function is constituted of a linear combination of the variables, the weights can be estimated using the least square method. Considering the following simplification:

$$G_i = \phi(\|\mathbf{x} - \xi\|) \tag{67}$$

the weights are obtained with Equation 68:

$$\lambda_\xi = (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T f(x,y) = \mathbf{G}^+ f(x,y) \tag{68}$$

where $\mathbf{G}^+$ is the pseudoinverse of the matrix $\mathbf{G}$, and $f(x,y)$ contains the depth image data. Finally, the interpolated image $f^*(x,y)$ can be estimated using:

$$f^*(x,y) = \mathbf{G} \cdot \lambda_\xi \tag{69}$$

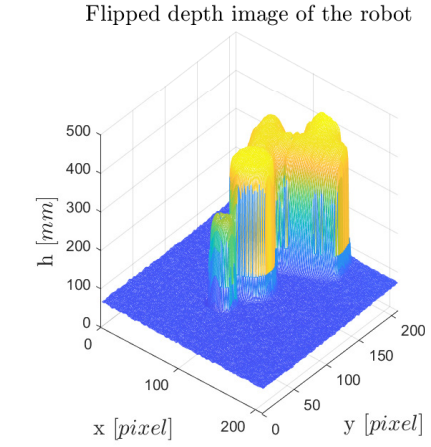
One of the advantages of interpolating using RBFs is that the calculation of $\mathbf{G}^+$ can be performed in advance and stored. Indeed, depending on the dimension of the grid of RBFs, the estimation of $\mathbf{G}^+$ can be time-consuming, making their use for real-time collision avoidance impossible. Therefore, thanks to the pre-calculation, weights estimation is reduced to a simple and fast matrix multiplication.

The value of σ is arbitrary; thus, tests were conducted on fitting different images to find the best value for σ. For an equally spaced grid of RBFs, the best interpolation results have been achieved when:

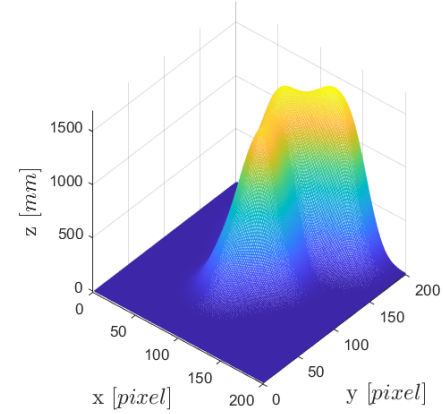
$$\sigma = \frac{D_{\xi_i \xi_j}}{2} \tag{70}$$

where $D_{\xi_i \xi_j}$ is the distance between the Gaussian centers.
Moreover, the number of RBFs is another essential factor. Too many functions result in a more fine interpolation, but they will lengthen the time to interpolate the environment, in particular, the estimation of the pseudoinverse. On the other hand, too few functions result in a coarser interpolation, but they require less time to interpolate the environment. For example, Figure 83a shows a depth image of the robot, while Figures 83b and 83c show the same image interpolated using 15×15 and 30×30 grids of evenly distributed RBFs on the plane, respectively. The image is 210×210 pixels.

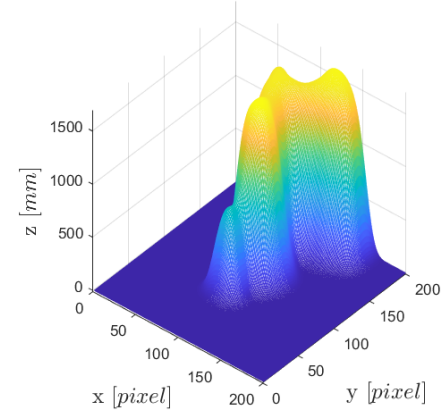
In the case shown in Figure 83, the 30×30 grid of RBFs led to a better representation of the robot than the 15×15 grid. However, the computational time to calculate the pseudoinverse of the 30×30 grid is almost 13 times higher than in the other case. Indeed, the calculation takes around 0.268 ± 0.006 s for the 15×15 grid and 3.440 ± 0.104 s for the 30×30 grid. Again, this estimation can be done offline before the beginning of the actual task, where the stored pseudoinverse is multiplied by the depth image to estimate the weights. This estimation takes around $1.8 \cdot 10^{-3} \pm 2.7 \cdot 10^{-4}$ s for the 15×15



(a)



(b)



(c)

Figure 83: Example of interpolation of the depth image of the robot using RBFs: (**a**) flipped depth image of the robot, (**b**) interpolation result using 15x15 RBFs grid, (**c**) interpolated result using 30x30 RBFs grid.

grid and $1.03 \cdot 10^{-2} \pm 7.08 \cdot 10^{-4}$ s for the 30×30 grid. Therefore, the 15×15 grid speeds up the estimation of the weights by a factor of about 5.7.
Another advantage of using fewer RBFs can be inferred by comparing Figures 83b and 83c. Indeed, cutting the functions around the base returns a bigger area if fewer RBFs are used. This property implies that the interpolation augments the object.Therefore, if the goal is to avoid the object, this extra distance from the actual object increases the safety of the operation.
It is worth noting that the depth image of Figure 83a is raised by about 65 mm. This additional random height improves the interpolation without affecting the final result. Indeed, fitting a function containing a lot of near-zero values results in more oscillations that would affect the gradient direction in the proximity of the peaks. Finally, to clear up the result of possible small oscillations, the weights of the interpolation are filtered by zeroing out the ones under one standard deviation from the mean. This filtering process is described by the system of equations:

$$\lambda_\epsilon = \begin{cases} \lambda_\epsilon & \text{if } \lambda_\epsilon \geqslant \bar{\lambda}_\epsilon + \sigma_{\lambda_\epsilon} \\ 0 & \text{if } \lambda_\epsilon < \bar{\lambda}_\epsilon + \sigma_{\lambda_\epsilon} \end{cases} \tag{71}$$

where $\bar{\lambda}_\epsilon$ is the mean weight, and $\sigma_{\lambda_\epsilon}$ is the standard deviation of the weight vector.
As can be seen in Figures 83b and 83c, the resulting function is regular and continuous. Such a function is easier to work with and can be easily added to other functions.
An additional property is that the calculation of the gradient is straightforward. Hence, it is not necessary to calculate the gradient directly from the final function. Indeed, from Equation 65 and 67, the gradient is given by:

$$\frac{\delta f(x,y)}{\delta x} = \frac{\delta \mathbf{G}}{\delta x}\lambda_\xi \tag{72}$$

$$\frac{\delta f(x,y)}{\delta y} = \frac{\delta \mathbf{G}}{\delta y}\lambda_\xi \tag{73}$$

Thus, it is only necessary to analytically derive Equation 66, calculate its values over the workspace, and store the results.

5.2.1 *Robot removal from depth images*

Independently of the vision system adopted, the robot will always appear in the images taken during the task.
Most of the works track the robot or use its kinematics to understand where its joints are to calculate its distance from the operator [92, 94].

Another approach is shown in [89]. The robot is removed from the scene using its augmented CAD model to eliminate possible disturbances or false alarms. As a result, the processed depth image data contains only obstacles.

Following this last work, this Section presents another method to remove the robot from the images. This process is performed by means of a simple calibration method. First, the robot moves on the plane following a zig-zag pattern, as shown in Figure 84. Meanwhile, the camera takes snapshots of its movements, and the robot's position (x_r, y_r) at every snapshot is read from the robot controller. After this

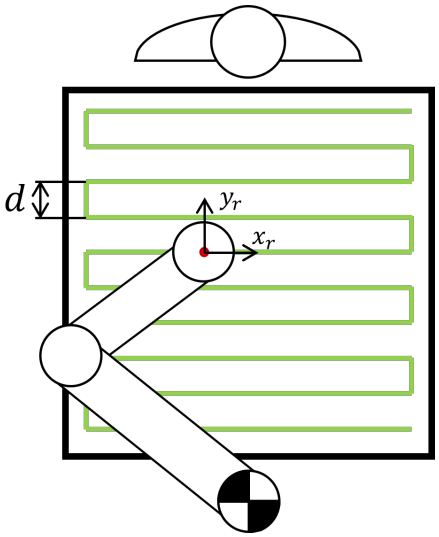
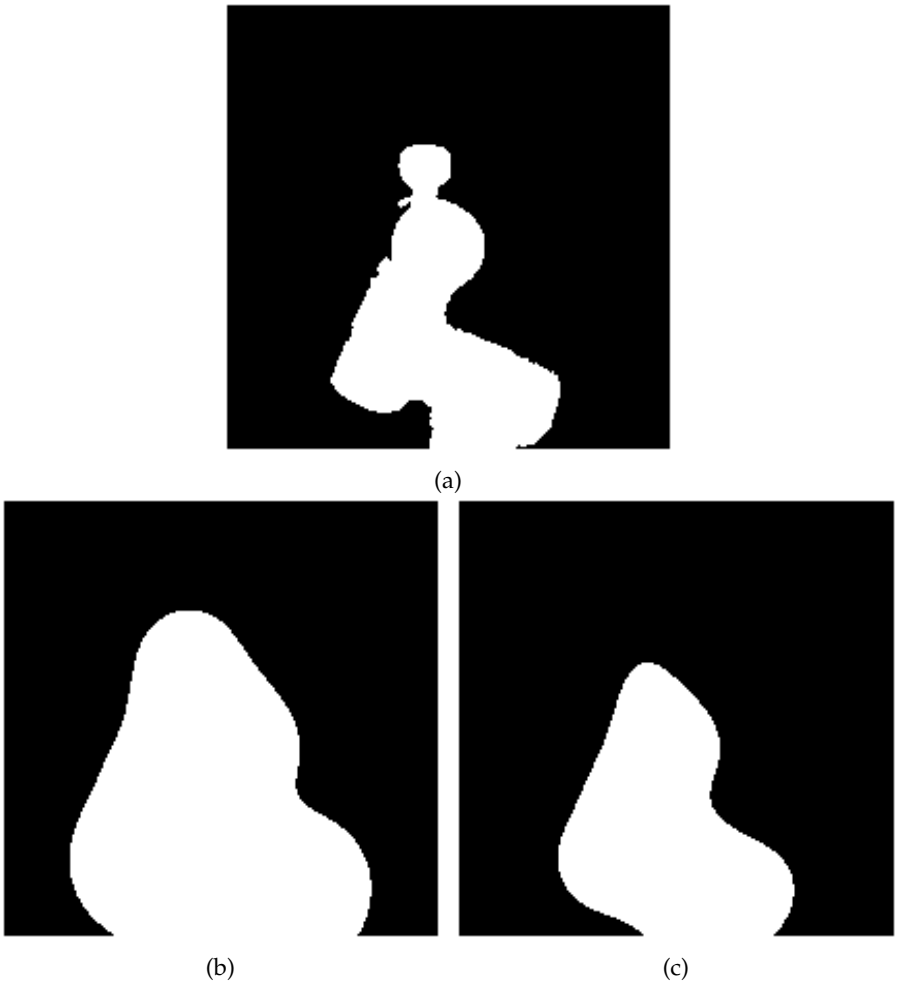


Figure 84: Trajectory followed by the robot during the calibration process

process, the images are interpolated using RBFs, as shown in Figure 83b and 83c. The interpolation generates an augmented representation of the robot that is useful to be more conservative during the elimination phase. Then, the images are binarized, and the MATLAB function *regionprops* is used to find the pixels occupied by the robot. Finally, a matrix is populated with those pixels and the corresponding position of the robot in the collaboration space. This calibration method is easy to implement and convenient when a reliable CAD model of the robot is unavailable.

Figure 85 shows the binarized images. The threshold for the binarization is selected experimentally, searching for a good approximation of the robot. The threshold corresponds to the height at which the interpolated function is cut. Therefore, higher thresholds mean greater heights for the cut and a closer match with the original image. This iterative process is simplified since the threshold selected for one position has shown good consistency for all the other positions of the robot. Comparing Figures 85a and 85b clearly shows that the interpolation augments the robot's encumbrance. On the other hand, if the threshold is too high, as in Figure 85c, the augmentation becomes



(a)

(b) (c)

Figure 85: Example of binarization: (**a**) original depth image, (**b**) interpolated image with a threshold of 100 mm, (**b**) interpolated image with a threshold of 500 mm.

negligible and insufficient.
The augmentation simplifies the calibration process since it is unnecessary to have a snapshot of the robot at every point of the plane. Consequently, the parameter d in Figure 84 must be modified accordingly to the threshold.
During the actual task, the algorithm searches the database for the closest point to the actual robot position and retrieves the matrix of pixels occupied by the interpolated robot. Finally, it clears all those pixels in the actual image, setting their value to 0 mm.

In Figure 86, the result of the filtering is proposed. Figure 86a shows the human operator's hand approaching the collaboration space and the robot performing a task. In Figure 86b, the robot is removed from the scene, and the human operator is interpolated. The image shown in 86b is the one used for future operations.

The final function can be employed in different ways. For example, the height of the function can be used to define different scenarios. In-

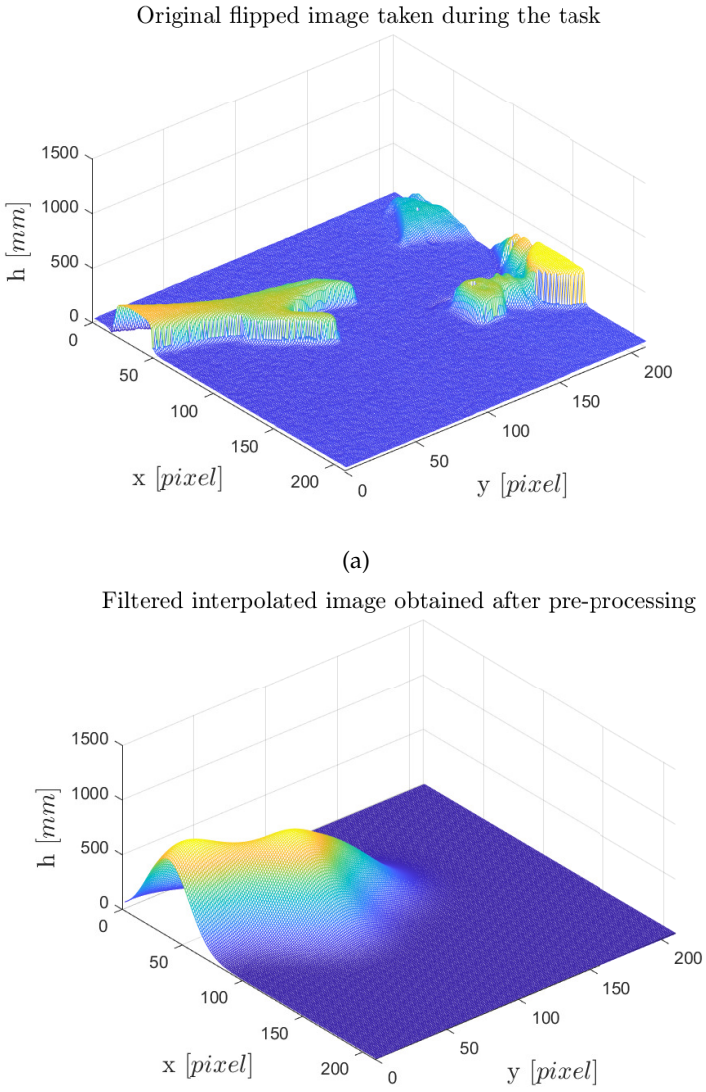


Figure 86: Example of filtering: (**a**) original depth image, with the operator hand on the left and the robot on the right; (**b**) filtered and interpolated image, where the robot has been completely removed.

deed, the closer the human operator to the cobot, the higher the function is. The cobot's speed can be adjusted depending on this height. Another possibility is to use the function as a potential function. For example, from Figure 86b, it can be inferred that a gradient descent method can be used to move the cobot away from the human operator, represented as a hill. The following Section presents this possible use. Again, the interpolated environment does not require estimating the distance.

The previous Sections presented a novel approach to detecting human interference in a cooperative framework, providing a solution that does not require the calculation of distances. The interpolation of the human operator generates a continuous function that sees them (or any other obstacles) as hills. Therefore, if it is coupled with a convex function that represents the target position for the robot as a minimum, a traditional gradient descent method can be employed to drive the robot versus the target and, simultaneously, divert it away from the operators. In such a way, the robot can avoid the operator without feature recognition. This method can be formalized as Artificial Potential Field (APF). This method is used extensively in the field of path planning [100] to modify the trajectory of the robot online. Traditionally, the method requires the generation of a repulsive function depending on the distance of the moving agent from the obstacles [58]. On the contrary, in this work, the interpolated environment is used directly as a potential.

After removing the robot, the filtered image can be analyzed to find the optimal path for the robot. An attractive force drives the robot to the target, and a repulsive one moves it away from obstacles.

The attractive force is imposed constant and is formulated as [58]:

$$\mathbf{F}_a = k_a \frac{\mathbf{e}(\mathbf{p})}{\|\mathbf{e}(\mathbf{p})\|} \tag{74}$$

where k_a is a constant, and $\mathbf{e}(\mathbf{p}) = (\mathbf{p}_{goal} - \mathbf{p})$ is the distance between the current Cartesian position of the robot ($\mathbf{p}$) and the target ($\mathbf{p}_{goal}$).

The repulsive force is derived from the gradient of the interpolated image. It is worth remembering that this operation is a simple multiplication between the gradient of the RBFs stored in the database and their estimated weights.
The repulsive force is defined as:

$$\mathbf{F}_r = -k_r \begin{bmatrix} \frac{\delta f^*}{\delta x} \\ \frac{\delta f^*}{\delta y} \end{bmatrix} \tag{75}$$

where k_r is a constant, and f^* is the result of the depth image interpolation and filtering to remove the robot.
The repulsive force generated in this way sees the gradient pointing out of the hill radially, as in Figure 87.

However, using this modality can lead to the problem of local minima. For example, if the robot is asked to move horizontally along $y = 50$ *pixels*, it will find itself stuck in the local minima on the left of the obstacle. This minimum point is generated by the equality of the attractive force pointing at the target and the repulsive force pointing in the opposite direction. In a cooperative context, this problem

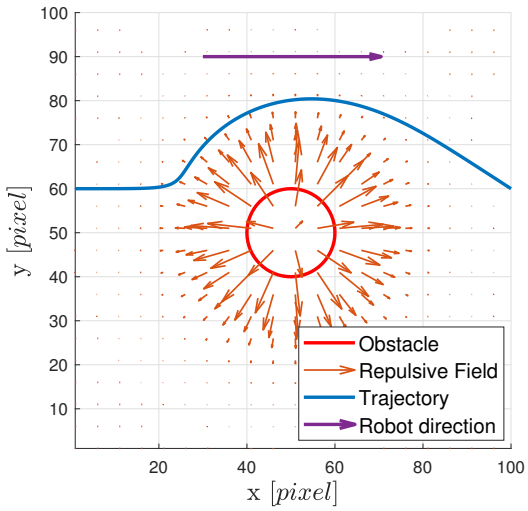


Figure 87: Trajectory generated using APF that drives the robot from $P_{start} = [0; 60]$ to $P_{end} = [100; 60]$ and avoids the red obstacle

will rarely appear due to the continuous movements of the operator. However, in a cooperative framework, the gradient can be exploited in another way to generate a more practical trajectory for the robot. Indeed, using the classic gradient of Equation 75 generates a trajectory along the fastest route. In other words, the robot is attracted to the areas where the repulsive force is weakest. Anyhow, Figure 86b shows that if the robot is moving along x, it has just one possible road to avoid the operator, which is by increasing its y coordinate. To optimize the trajectory and avoid the local minima problem, the repulsive force can be defined as follows:

$$\mathbf{F}_r = \mp k_r \begin{bmatrix} \frac{\delta \mathbf{f}^*}{\delta y} \\ -\frac{\delta \mathbf{f}^*}{\delta x} \end{bmatrix} \tag{76}$$

Inverting the gradient along the two directions creates a vortex field, as shown in Figure 88. The sign in Equation 76 decides the direction of rotation.

Since the operator comes from outside the workspace, only one of the two directions of the vortex (clockwise or counterclockwise) can be used. The right direction depends strictly on the direction of motion of the robot. For example, the vortex field in Figure 88 is clockwise, following the robot's direction.

Finally, the total force acting on the robot is :

$$\mathbf{F}_t(\mathbf{p}) = \mathbf{F}_a(\mathbf{p}) + \mathbf{F}_r(\mathbf{p}) \tag{77}$$

This force can be used in different ways to modify the trajectory [58]. In this work, it is used as Cartesian velocity:

$$\dot{\mathbf{p}} = \mathbf{F}_t(\mathbf{p}) \tag{78}$$

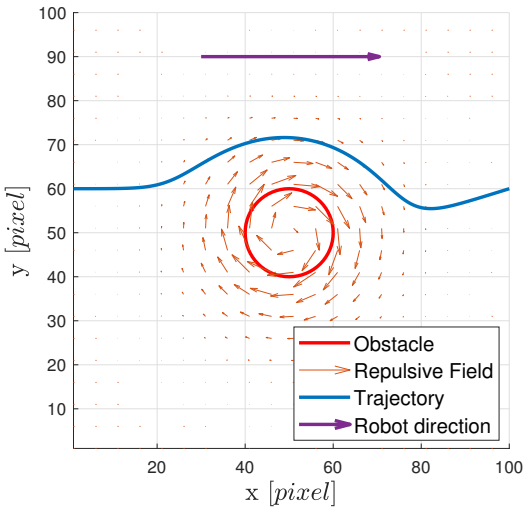


Figure 88: Trajectory generated using a vortex repulsive field around the obstacle.

Then, the path is generated using the gradient descent method [58]:

$$\mathbf{p}_{i+1} = \mathbf{p}_i + \dot{\mathbf{p}}\Delta t \tag{79}$$

where Δt is the time elapsed between two path computations.

In Figure 89, a typical path is shown. For display purposes, the operator is static. The robot successfully diverts its trajectory from the straight one to compensate for the human interference in the scene.

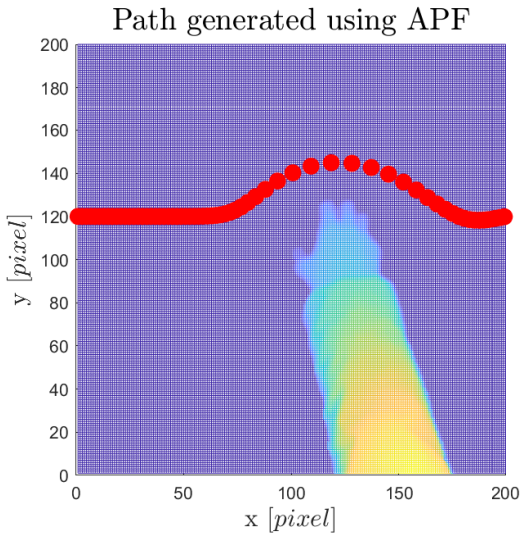


Figure 89: Path generated with APF method showing the deviation in correspondence to human interference. The red dots represent the robot tool center point.

It can be stated that the path in Figure 89 seems too close to the operator's hand. The maximum distance from the operator can be adjusted in two ways:

1. tuning the coefficients of the potentials;
2. increasing the σ of the RBFs (without modifying the weights).

The second way shows another advantage of the use of RBFs. Once the weights have been calculated, they can be assigned to another grid of RBFs to generate a function that is always an approximation of the system but can have different σ. The final result is tighter or broader than the original, as shown in Figure 90.

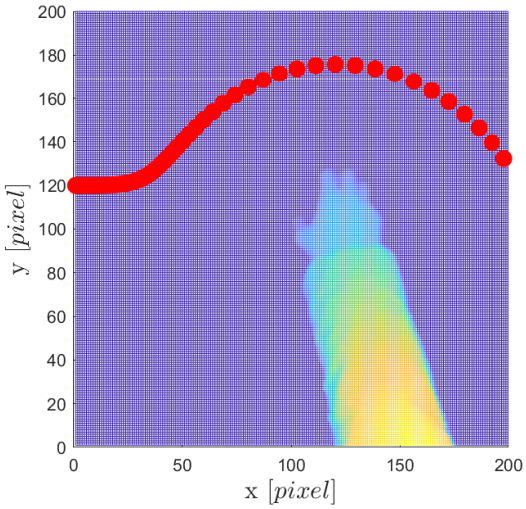


Figure 90: Path generated with APF method using an augmented value for σ ($\sigma' = 1.5\sigma$)

Finally, a fundamental parameter analyzed to ensure that the algorithm has adequate capabilities for online control is the time necessary to compute a new command for the robot. This parameter was measured experimentally. The algorithm was tested using a Techman TM5-700 and a Kinect depth camera; both controlled from MATLAB. The images were interpolated using a 15×15 Gaussian grid. The tests returned a mean total time of 0.081 ± 0.008 s to capture a new image and generate the successive point. This time comprises the time necessary to gather the robot's position from its controller and the image from the camera. Therefore, it can be influenced by the efficiency of those acquisitions. The time related to only algorithm processing was estimated at 0.022 ± 0.004 s. As can be seen, about 70% of the time is independent of the algorithm, that alone can perform at about 45 Hz, confirming its online capabilities.

5.4 CONCLUSIONS

This Chapter presents a novel approach to human detection and avoidance in cooperative tasks.
The algorithm can detect the presence of the human operator in the

workspace and modify the robot trajectory accordingly, avoiding potential collisions. This algorithm can be used online at a frequency dependent mainly on the robot and the camera.

The algorithm mixes Radial Basis Functions to interpolate the scene and Artificial Potential Fields to generate the path. In particular, RBFs create a continuous function on which the gradient is calculated. Then, such a gradient is used as a repulsive potential for APFs that automatically generate robot trajectory. The algorithm requires only a single depth camera. Therefore, it does not require complex elaborations on multiple images or complex setups. Instead, it requires just the classical calibration of the camera and a second simple calibration in which the robot moves over the workspace. The second calibration is necessary to remove the robot from future images so that only the obstacle (i.e., the human operator) is present in the scene.

It is worth stating that using a single depth camera presents some limitations. For example, the algorithm is very conservative since it can not consider the third dimension. However, the only safe case not considered is when the operator executes actions above the robot. Indeed, possible worst-case scenarios can take place when the operator is working beneath the robot, such as the one in which the robot is handling a sharp object. Unfortunately, it is impossible to overcome this limitation using a single depth camera without restricting other views. For example, the operator can move one arm over the robot and the other beneath it at the same time. Thus, the safety of the operator is not guaranteed in any case.

Finally, it is worth noticing that the algorithm is proposed to avoid dynamic obstacles (human operators). However, the simplicity of the method makes it suitable to be implemented also for static obstacles.

6

CONCLUSIONS

This thesis addressed the problem of collision management in industrial robotics, a trending topic in the past years due to the increased interest in autonomous robots and collaborative robotics. These two trends increase the flexibility of the robotic cells, but they also increase the unpredictability of the robot's movements and of the environment in which the robot moves.
The main contribution of this thesis consists in a mechanical approach to mitigating unexpected collisions between a robotic end-effector and the environment in which the robot moves. The literature offers many ways to cope with these collisions that work on the control system. However, in the era of collaborative robotics and autonomous robots, an additional safety layer constituted of a passive mechanical system can improve the overall safety of the tasks. This thesis achieves this by proposing a compliant mechanism that decouples the tool from the robot, increasing the possible response time to the collision and reducing the force transferred to the environment and to the robot. Moreover, having a system to mitigate the collisions can increase the operative velocities, increasing the overall performance of the task.
In the first Chapter, an introduction is outlined, defining the actual trends in industrial robotics and the risks consequent to their adoption.
The second Chapter presents a mono-dimensional study of the impacts between the robot end-effector and objects disposed on a plane and free to move. First, the study analyzes the effects of the robot dynamics, the end-effector compliance, and the impact characteristics. Then, a new end-effector is designed that reduces the momentum transferred to the robot and to small objects with which the robot impacts. This property is achieved by employing a bi-stable compliant mechanism in the form of a non-linear spring. In this form, the system is able to impose an initial resistance to the collision, necessary to the execution of the task, and yield when the impact force is too high. In particular, the tool is decoupled from the robot. Therefore, the two systems can move separately during a hard collision. Based on this solution, a novel end-effector is proposed for a sorting task, and experimental validation confirms the validity of the predictions of the mathematical model. Moreover, the compliant system was compared to a rigid tool, showing a significant reduction in the momentum transferred to the objects.
In the third Chapter, the study of impacts has been integrated by

considering impacts with hard surfaces. These impacts can lead to more damage to the overall system and must be mitigated as much as possible. A new prototype is proposed. It is constituted of a module (containing the bi-stable mechanism) to be interposed between the robot flange and the tool. This module generalizes the bi-stable mitigating solution to any 1D tool. Moreover, it was used to validate the updated model. The Chapter concludes with an extension of the bi-stable mechanism in 2D. The new prototype introduces a rotational module that orients the 1D bi-stable mechanism along the direction of the impact. The prototype was tested and validated experimentally, successfully mitigating hard collisions. However, due to the force necessary for the rotation, the system is not suitable for impacts with small objects when they require high rotations. Instead, it can be helpful when the object orientation is inaccurate. In this case, the 2D tool can compensate for the misalignment.

In the fourth Chapter, a different solution is sought for collision mitigation. In particular, a multi-physics approach is adopted that brought to the design of a hydraulic solution. This study is practical to analyze the effectiveness of the bi-stable mechanism against other possible mitigation approaches and have more insights on how to mitigate collisions. The new solution is constituted of a main hydraulic cylinder connected to an auxiliary one, partially filled with pressurized gas. During an impact, the tool pushes the fluid from one cylinder to the other, compressing the gas. Therefore, unlike the mechanical bi-stable mechanism, the hydraulic solution exerts an increasing resistance to the movement of the tool. The solution has demonstrated not only to be able to mitigate collisions but also to be better than the bi-stable mechanism when the control system is inefficient (i.e., when the reaction comes with a high delay from the instant of the impact). However, with a lower dead-time, the bi-stable solution is more efficient and leads to lower forces and velocities of the impacted objects than the hydraulic solution.

Finally, in the fifth Chapter, the focus is shifted to collision avoidance algorithms. The Chapter proposes an approach to divert the robot trajectory to avoid human operators. This method uses a single depth camera and does not require estimating the distance between the robot and the human operator. The algorithm is based on the interpolation of the environment and the removal of the robot from the images using Radial Basis Functions (RBF) and the calculation of the robot's trajectory in real-time using Artificial Potential Fields (APF). The interpolated environment is used as a repulsive potential to actively divert the robot's trajectory when the operator is too close to the robot. The algorithm was tested experimentally and successfully modified the robot's trajectory online to avoid the operator.

Future works regard the experimental validation of the hydraulic solution, the extension of the study of collisions in two and three

dimensions and the design of alternative solutions to collision mitigation. Furthermore, from the collision avoidance point of view, improvements can be made to enhance the overall performance of the algorithm. For example, analyzing the variation in the weights of the RBFs caused by the movements of the human operator may help predict those movements and the consequent intentions of the operator. These predictions would lead to better trajectories and reactions.

BIBLIOGRAPHY

[1] L. Barbazza, M. Faccio, F. Oscari, and G. Rosati, "Agility in assembly systems: a comparison model," *Assembly Automation*, vol. 37, no. 4, pp. 411–421, 2017.

[2] C. Finetto, M. Faccio, G. Rosati, and A. Rossi, "Mixed-model sequencing optimization for an automated single-station fully flexible assembly system (f-fas)," *The International Journal of Advanced Manufacturing Technology*, vol. 70, no. 5, pp. 797–812, 2014.

[3] G. Rosati, M. Faccio, L. Barbazza, and A. Rossi, "Hybrid fexible assembly systems (h-fas): bridging the gap between traditional and fully flexible assembly systems," *The International Journal of Advanced Manufacturing Technology*, vol. 81, no. 5, pp. 1289–1301, 2015.

[4] IFR, "Top trends robotics 2020." https://ifr.org/ifr-press-releases/news/top-trends-robotics-2020, 2020.

[5] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[6] G. Cipriani, M. Bottin, and G. Rosati, "Applications of learning algorithms to industrial robotics," in *The International Conference of IFToMM ITALY*, pp. 260–268, Springer, 2020.

[7] IFR, "Ai in robotics." https://ifr.org/post/ai-in-robotics-blog, 2022.

[8] M. Faccio, M. Bottin, and G. Rosati, "Collaborative and traditional robotic assembly: a comparison model," *The International Journal of Advanced Manufacturing Technology*, vol. 102, no. 5, pp. 1355–1372, 2019.

[9] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[10] E. Matheson, R. Minto, E. G. Zampieri, M. Faccio, and G. Rosati, "Human-robot collaboration in manufacturing applications: A review," *Robotics*, vol. 8, no. 4, p. 100, 2019.

[11] G. Rosati, G. Boschetti, A. Biondi, and A. Rossi, "On-line dimensional measurement of small components on the eyeglasses

assembly line," *Optics and Lasers in Engineering*, vol. 47, no. 3-4, pp. 320–328, 2009.

[12] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 769–776, IEEE, 2017.

[13] L. Bergamini, M. Sposato, M. Pellicciari, M. Peruzzini, S. Calderara, and J. Schmidt, "Deep learning-based method for vision-guided robotic grasping of unknown objects," *Advanced Engineering Informatics*, vol. 44, p. 101052, 2020.

[14] P. Jiang, Y. Ishihara, N. Sugiyama, J. Oaki, S. Tokura, A. Sugahara, and A. Ogawa, "Depth image-based deep learning of grasp planning for textureless planar-faced objects in vision-guided robotic bin-picking," *Sensors*, vol. 20, no. 3, p. 706, 2020.

[15] Z. Zhu and H. Hu, "Robot learning from demonstration in robotic assembly: A survey," *Robotics*, vol. 7, no. 2, p. 17, 2018.

[16] E. De Coninck, T. Verbelen, P. Van Molle, P. Simoens, and B. Dhoedt, "Learning robots to grasp by demonstration," *Robotics and Autonomous Systems*, vol. 127, p. 103474, 2020.

[17] M. Ragaglia *et al.*, "Robot learning from demonstrations: Emulation learning in environments with moving obstacles," *Robotics and autonomous systems*, vol. 101, pp. 45–56, 2018.

[18] G. Solak and L. Jamone, "Learning by demonstration and robust control of dexterous in-hand robotic manipulation skills," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8246–8251, IEEE, 2019.

[19] Z. Cao, H. Hu, X. Yang, and Y. Lou, "A robot 3c assembly skill learning method by intuitive human assembly demonstration," in *2019 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, pp. 13–18, IEEE, 2019.

[20] A. Lobbezoo, Y. Qian, and H.-J. Kwon, "Reinforcement learning for pick and place operations in robotics: A survey," *Robotics*, vol. 10, no. 3, p. 105, 2021.

[21] F. Li, Q. Jiang, W. Quan, R. Song, and Y. Li, "Manipulation skill acquisition for robotic assembly using deep reinforcement learning," in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 13–18, IEEE, 2019.

[22] M. Vecerik, T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, T. Rothörl, T. Lampe, and M. Riedmiller, "Leveraging demonstrations for deep reinforcement learning on robotics

problems with sparse rewards," *arXiv preprint arXiv:1707.08817*, 2017.

[23] J. Luo, O. Sushkov, R. Pevceviciute, W. Lian, C. Su, M. Vecerik, N. Ye, S. Schaal, and J. Scholz, "Robust multi-modal policies for industrial assembly via reinforcement learning and demonstrations: A large-scale study," *arXiv preprint arXiv:2103.11512*, 2021.

[24] M. Mavsar, M. Deniša, B. Nemec, and A. Ude, "Intention recognition with recurrent neural networks for dynamic human-robot collaboration," in *2021 20th International Conference on Advanced Robotics (ICAR)*, pp. 208–215, IEEE, 2021.

[25] ISO, "Robots and robotic devices - safety requirements for industrial robots – Part 1: robots," Tech. Rep. ISO 10218-1:2011, 2011.

[26] ISO, "Robots and robotic devices - collaborative robots," Tech. Rep. ISO/TS 15066:2016, 2016.

[27] S. Seriani, P. Gallina, L. Scalera, and V. Lughi, "Development of n-dof preloaded structures for impact mitigation in cobots," *Journal of Mechanisms and Robotics*, vol. 10, no. 5, p. 051009, 2018.

[28] Y. Cohen, S. Shoval, M. Faccio, and R. Minto, "Deploying cobots in collaborative systems: major considerations and productivity analysis," *International Journal of Production Research*, vol. 60, no. 6, pp. 1815–1831, 2022.

[29] M. Askarpour, D. Mandrioli, M. Rossi, and F. Vicentini, "Formal model of human erroneous behavior for safety analysis in collaborative robotics," *Robotics and computer-integrated Manufacturing*, vol. 57, pp. 465–476, 2019.

[30] Z. Liu, X. Wang, Y. Cai, W. Xu, Q. Liu, Z. Zhou, and D. T. Pham, "Dynamic risk assessment and active response strategy for industrial human-robot collaboration," *Computers & Industrial Engineering*, vol. 141, p. 106302, 2020.

[31] S. Robla-Gómez, V. M. Becerra, J. R. Llata, E. Gonzalez-Sarabia, C. Torre-Ferrero, and J. Perez-Oria, "Working together: A review on safe human-robot collaboration in industrial environments," *IEEE Access*, vol. 5, pp. 26754–26773, 2017.

[32] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[33] L. Antão, J. Reis, and G. Gonçalves, "Voxel-based space monitoring in human-robot collaboration environments," in *2019*

24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 552–559, IEEE, 2019.

[34] G. Boschetti, M. Faccio, and R. Minto, "Control model for collaborative manufacturing: An integrated opened framework for human-robot collaboration," in *The International Conference of IFToMM ITALY*, pp. 403–413, Springer, 2020.

[35] A. Bonci, P. D. Cen Cheng, M. Indri, G. Nabissi, and F. Sibona, "Human-robot perception in industrial environments: A survey," *Sensors*, vol. 21, no. 5, p. 1571, 2021.

[36] R.-J. Halme, M. Lanz, J. Kämäräinen, R. Pieters, J. Latokartano, and A. Hietanen, "Review of vision-based safety systems for human-robot collaboration," *Procedia CIRP*, vol. 72, pp. 111–116, 2018.

[37] P. A. Lasota, T. Fong, J. A. Shah, *et al.*, "A survey of methods for safe human-robot interaction," *Foundations and Trends® in Robotics*, vol. 5, no. 4, pp. 261–349, 2017.

[38] D. Tommasino, G. Cipriani, A. Doria, and G. Rosati, "Effect of end-effector compliance on collisions in robotic teleoperation," *Applied Sciences*, vol. 10, no. 24, p. 9077, 2020.

[39] D. Tommasino and et al., "Development of an end-effector for mitigation of collisions," *Volume 7: 17th IEEE/ASME Int. Conf. Mechatron. Embed. Syst. Appl. MESA*, 08 2021.

[40] D. Tommasino, M. Bottin, G. Cipriani, A. Doria, and G. Rosati, "Development and validation of an end-effector for mitigation of collisions," *J. of Mechanical Design*, pp. 1–33, 09 2021.

[41] G. Cipriani, M. Bottin, G. Rosati, and M. Faccio, "A radial basis functions approach to collision avoidance in collaborative tasks," *IFAC-PapersOnLine*, vol. 55, no. 2, pp. 307–312, 2022.

[42] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.

[43] K. Hashtrudi-Zaad and S. E. Salcudean, "Analysis of control architectures for teleoperation systems with impedance/admittance master and slave manipulators," *The International Journal of Robotics Research*, vol. 20, no. 6, pp. 419–445, 2001.

[44] D. Sun, F. Naghdy, and H. Du, "Application of wave-variable control to bilateral teleoperation systems: A survey," *Annual Reviews in Control*, vol. 38, no. 1, pp. 12–31, 2014.

[45] L. Chan, F. Naghdy, and D. Stirling, "Application of adaptive controllers in teleoperation systems: A survey," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 3, pp. 337–352, 2014.

[46] R. Uddin and J. Ryu, "Predictive control approaches for bilateral teleoperation," *Annual Reviews in Control*, vol. 42, pp. 82–99, 2016.

[47] M. Bottin, S. Cocuzza, and M. Massaro, "Variable stiffness mechanism for the reduction of cutting forces in robotic deburring," *Applied Sciences*, vol. 11, no. 6, p. 2883, 2021.

[48] S. D. Bhargav, S. Chakravarthy, and G. Ananthasuresh, "A compliant end-effector to passively limit the force in tele-operated tissue-cutting," *Journal of Medical Devices*, vol. 6, no. 4, 2012.

[49] R. M. Brach, *Mechanical impact dynamics: rigid body collisions*. Brach Engineering, LLC, 2007.

[50] L. Skrinjar, J. Slavič, and M. Boltežar, "A review of continuous contact-force models in multibody dynamics," *International Journal of Mechanical Sciences*, vol. 145, pp. 171–187, 2018.

[51] K. H. Hunt and F. R. E. Crossley, "Coefficient of restitution interpreted as damping in vibroimpact," *Journal of Applied Mechanics*, vol. 42, pp. 440–445, 06 1975.

[52] D. W. Marhefka and D. E. Orin, "Simulation of contact using a nonlinear damping model," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1662–1668 vol.2, 1996.

[53] A. Schwab, J. Meijaard, and P. Meijers, "A comparison of revolute joint clearance models in the dynamic analysis of rigid and elastic mechanical systems," *Mechanism and machine theory*, vol. 37, no. 9, pp. 895–913, 2002.

[54] S. Hu and X. Guo, "A dissipative contact force model for impact analysis in multibody dynamics," *Multibody System Dynamics*, vol. 35, no. 2, pp. 131–151, 2015.

[55] H. Zhang, J. Wang, G. Zhang, Z. Gan, Z. Pan, H. Cui, and Z. Zhu, "Machining with flexible manipulator: toward improving robotic machining performance," in *Proceedings, 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*, pp. 1127–1132, IEEE, 2005.

[56] M. Bottin, S. Cocuzza, N. Comand, and A. Doria, "Modeling and identification of an industrial robot with a selective modal approach," *Applied Sciences*, vol. 10, no. 13, p. 4619, 2020.

[57] A. Doria, S. Cocuzza, N. Comand, M. Bottin, and A. Rossi, "Analysis of the compliance properties of an industrial robot with the mozzi axis approach," *Robotics*, vol. 8, no. 3, p. 80, 2019.

[58] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer Science & Business Media, 2010.

[59] OMRON, *Omron Adept Viper s650/s850 Robot with MB-60R/eMB-60R User's Guide*.

[60] W. J. Bottega, *Engineering vibrations*. CRC Press, 2014.

[61] A. Borboni and R. Faglia, *Parasitic phenomena in the dynamics of industrial devices*. CRC Press, 2017.

[62] A. Borboni, M. Lancini, and R. Faglia, "Residual vibration reduction with commanded motion optimization," vol. Volume 2: Dynamics, Vibration and Control; Energy; Fluids Engineering; Micro and Nano Manufacturing of *Engineering Systems Design and Analysis*, 07 2014.

[63] J. Fu, J. Zhang, Z. She, S. E. Ovur, W. Li, W. Qi, H. Su, G. Ferrigno, and E. De Momi, "Whole-body spatial teleoperation control of a hexapod robot in unstructured environment," in *2021 6th IEEE International Conference on Advanced Robotics and Mechatronics (ICARM)*, pp. 93–98, 2021.

[64] M. Bottin, G. Rosati, and G. Cipriani, "Iterative path planning of a serial manipulator in a cluttered known environment," *Mechanisms and Machine Science*, vol. 91, pp. 237–244, 2021.

[65] D. Reyes-Uquillas and T. Hsiao, "Safe and intuitive manual guidance of a robot manipulator using adaptive admittance control towards robot agility," *Robotics and Computer-Integrated Manufacturing*, vol. 70, p. 102127, 2021.

[66] B. Hannaford and R. Anderson, "Experimental and simulation studies of hard contact in force reflecting teleoperation," in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pp. 584–589, IEEE, 1988.

[67] D. A. Lawrence, "Stability and transparency in bilateral teleoperation," *IEEE transactions on robotics and automation*, vol. 9, no. 5, pp. 624–637, 1993.

[68] A. Bicchi and G. Tonietti, "Fast and "soft-arm" tactics [robot arm design]," *IEEE Robotics Automation Magazine*, vol. 11, no. 2, pp. 22–33, 2004.

[69] R. Weitschat, J. Vogel, S. Lantermann, and H. HÃ¶ppner, "End-effector airbags to accelerate human-robot collaboration," pp. 2279–2284, 2017.

[70] SCHUNK, "Compensating." https://schunk.com/br_en/gripping-systems/category/gripping-systems/handling/compensating/, 2020.

[71] J. Shintake, V. Cacucciolo, D. Floreano, and H. Shea, "Soft robotic grippers," *Advanced materials*, vol. 30, no. 29, p. 1707035, 2018.

[72] K. Chandrasekaran, A. Somayaji, and A. Thondiyath, "A novel design for a compliant mechanism based variable stiffness grasper through structure modulation," *Journal of Medical Devices*, vol. 15, no. 1, 2021.

[73] M. Mohammadi, D. Bicego, A. Franchi, D. Barcelli, and D. Prattichizzo, "Aerial tele-manipulation with passive tool via parallel position/force control," *Applied Sciences*, vol. 11, no. 19, p. 8955, 2021.

[74] R. v. Ham, T. Sugar, B. Vanderborght, K. Hollander, and D. Lefeber, "Compliant actuator designs," *IEEE Robotics & Automation Magazine*, vol. 3, no. 16, pp. 81–94, 2009.

[75] S. S. Jujjavarapu, A. H. Memar, M. A. Karami, and E. T. Esfahani, "Variable stiffness mechanism for suppressing unintended forces in physical human–robot interaction," *Journal of Mechanisms and Robotics*, vol. 11, no. 2, 2019.

[76] H. N. Huynh, H. Assadi, E. Riviere-Lorphevre, O. Verlinden, and K. Ahmadi, "Modelling the dynamics of industrial robots for milling operations," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101852, 2020.

[77] I.-H. Hwang, Y.-S. Shim, and J.-H. Lee, "Modeling and experimental characterization of the chevron-type bi-stable microactuator," *Journal of Micromechanics and Microengineering*, vol. 13, pp. 948–954, aug 2003.

[78] S. A. Zirbel, K. A. Tolman, B. P. Trease, and L. L. Howell, "Bistable mechanisms for space applications," *PLOS ONE*, vol. 11, pp. 1–18, 12 2016.

[79] S. P. Pellegrini, N. Tolou, M. Schenk, and J. L. Herder, "Bistable vibration energy harvesters: A review," *Journal of Intelligent Material Systems and Structures*, vol. 24, no. 11, pp. 1303–1312, 2013.

[80] M. Tonan, M. Bottin, A. Doria, and G. Rosati, "A modal approach for the identification of joint and link compliance of

an industrial manipulator," in *The International Conference of IFToMM ITALY*, pp. 628–636, Springer, 2022.

[81] Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, and T. Hirai, "Fast graspability evaluation on single depth maps for bin picking with general grippers," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1997–2004, IEEE, 2014.

[82] M. Danielczuk, J. Mahler, C. Correa, and K. Goldberg, "Linear push policies to increase grasp access for robot bin picking," in *2018 IEEE 14th international conference on automation science and engineering (CASE)*, pp. 1249–1256, IEEE, 2018.

[83] K. Xu, H. Yu, Q. Lai, Y. Wang, and R. Xiong, "Efficient learning of goal-oriented push-grasping synergy in clutter," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6337–6344, 2021.

[84] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martín-Martín, A. Garg, S. Savarese, and K. Goldberg, "Mechanical search: Multi-step retrieval of a target object occluded by clutter," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1614–1621, IEEE, 2019.

[85] V. Cossalter and et al., "On the non-linear behaviour of motorcycle shock absorbers," *Proc. of the Institution of Mech. Eng., Part D: J. of Automobile Engineering*, vol. 224, no. 1, pp. 15–27, 2010.

[86] R. Müller, M. Vette, and A. Geenen, "Skill-based dynamic task allocation in human-robot-cooperation with the example of welding application," *Procedia Manufacturing*, vol. 11, pp. 13–21, 2017.

[87] ISO, "Robots and robotic devices - safety requirements for industrial robots – Part 2: robot systems and integration," Tech. Rep. ISO 10218-2:2011, 2011.

[88] E. Mariotti, E. Magrini, and A. D. Luca, "Admittance control for human-robot interaction using an industrial robot equipped with a f/t sensor," in *2019 Int. Conf. on Robotics and Automation (ICRA)*, pp. 6130–6136, 2019.

[89] E. Magrini, F. Ferraguti, A. J. Ronga, F. Pini, A. De Luca, and F. Leali, "Human-robot coexistence and interaction in open industrial cells," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101846, 2020.

[90] F. Flacco, T. KrÃ¶ger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *2012 IEEE Int. Conf. on Robotics and Automation*, pp. 338–345, May 2012.

[91] L. S. Scimmi, M. Melchiorre, M. Troise, S. Mauro, and S. Pastorelli, "A practical and effective layout for a safe human-robot collaborative assembly task," *Applied Sciences*, vol. 11, no. 4, 2021.

[92] S. Secil and M. Ozkan, "Minimum distance calculation using skeletal tracking for safe human-robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102253, 2022.

[93] A. Amorim, D. Guimares, T. Mendona, P. Neto, P. Costa, and A. P. Moreira, "Robust human position estimation in cooperative robotic cells," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 102035, 2021.

[94] S. Mauro, L. S. Scimmi, and S. Pastorelli, "Collision avoidance system for collaborative robotics," in *Advances in Service and Industrial Robotics* (C. Ferraresi and G. Quaglia, eds.), (Cham), pp. 344–352, Springer International Publishing, 2018.

[95] C. Byner, B. Matthias, and H. Ding, "Dynamic speed and separation monitoring for collaborative robot applications–concepts and performance," *Robotics and Computer-Integrated Manufacturing*, vol. 58, pp. 239–252, 2019.

[96] M. D. Buhmann, "Radial basis functions," *Acta Numerica*, vol. 9, pp. 1–38, 2000.

[97] O. Strack and I. Jankovic, "A multi-quadric area-sink for analytic element modeling of groundwater flow," *J. of Hydrology*, vol. 226, no. 3-4, pp. 188–196, 1999.

[98] D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks," *Complex Syst.*, vol. 2, 1988.

[99] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans, "Reconstruction and representation of 3d objects with radial basis functions," in *Computer Graphics (SIGGRAPH '01 Conf. Proc.), pages 67-76. ACM SIGGRAPH*, pp. 67–76, Springer, 2001.

[100] C. Warren, "Global path planning using artificial potential fields," in *Proceedings, 1989 Int. Conf. on Robotics and Automation*, pp. 316–321 vol.1, 1989.

Rimanete ambiziosi nei vostri obiettivi, perché rassegnarsi a una vita mediocre non vale mai la pena.

— Sergio Marchionne

RINGRAZIAMENTI

Così si conclude questa tesi e questo breve capitolo della mia vita. Portare a termine il dottorato è stata un'impresa impegnativa, altamente formante a livello umano, soprattutto considerato che è stata sommersa dall'arrivo del COVID dopo soli 5 mesi dall'inizio.
Un'impresa che non sarebbe stata possibile se non fosse stato per il supporto delle persone che mi hanno circondato in questi 3 anni.
Innanzitutto, ringrazio il Prof. Giulio Rosati per avermi dato la possibilità di intraprendere questo percorso. Il dottorato era una di quelle possibilità post-laurea che mai avrei preso in considerazione non fosse stato per lui.
Ringrazio di cuore anche il Prof. Alberto Doria, fonte inesauribile di conoscenza e di ispirazione e che mi ha insegnato molto col suo rigore metodologico e la sua curiosità, che lo portano a interrogarsi anche sulle minime virgole di ogni lavoro.
Ringrazio i miei colleghi, in particolare Nicola, Riccardo, Matteo e Mimmo, non solo per le nostre "partitine", fonte di motivazione e rilassamento anche nei giorni più intensi, ma soprattutto per esser stati per me modelli da seguire e da cui imparare.
Ringrazio i miei tesisti, Mimmo, Guido, Vittorio, Umberto, Alessandro, Daniel, Nicola, Davide, Diego, Nicola, Tiziano, Sante, Federico e Giacomo. Ci tengo a nominarvi uno a uno perché tutti mi avete accompagnato in questo percorso giorno dopo giorno. Grazie a voi, il dottorato è stato fonte di crescita personale, oltre che professionale.
A special thank goes to Prof. Sunil K. Agrawal and all the people of the ROaR Lab at Columbia University. Thanks for welcoming me so open-heartedly into your group. Even if it was just for 6 months, I learned a lot from you all, so that my time there profoundly changed my vision on how to do research.
Grazie a tutti i miei amici, soprattutto per le piccole cose: un messaggio, un aperitivo, un sushi organizzato all'ultimo momento, serate a giochi in scatola finite in tragedia. Mi avete aiutato a metter da parte stress e ansie e godermi il tempo passato insieme. Ringrazio in particolare Ste per le nostre chiamate infinite e gli aperitivi interminabili e per esser sempre stato pronto ad ascoltarmi e consigliarmi.
GRAZIE ai miei genitori, a mio fratello e mia sorella per avermi supportato e sopportato in questa esperienza. Vi ringrazio perché siete i miei più grandi fan, lo siete sempre stati e lo sarete sempre. Ringrazio in particolare mio fratello, la prima persona a cui chiedo un parere su qualsiasi cosa "tecnica" e colui che mi motiva costantemente a migliorare e fare sempre quell'1% in più ogni giorno.
Ringrazio i suoceri, Sabrina e Giovanni, nonchè Eleonora, Sebastiano

e Paola. Mi avete accolto come parte della vostra famiglia e il vostro costante supporto è per me inestimabile.
Voglio chiudere ringraziando colei che ogni giorno mi strappa un sorriso e mi fa sentire la vita più leggera, Beatrice. Lei che mi accompagna da anni e che spero vorrà farlo ancora per molto tempo. Lei che continua a spronarmi a dare di più, a essere felice e trovare la mia strada nel mondo. Non sarei arrivato a scrivere queste ultime righe se non fossi stata al mio fianco.