Collision detection with energy-based residual for elastic joints robots

Castelmare Mattia
La Sapienza, University of Rome

Physical Human Robot Interaction project

Abstract

The goal of this project is to develop a framework for real-time collision detection based on residual energy in which adopted model-based algorithms use only proprioceptive sensors. The problem is addressed for robots characterized by joint-centered elasticity. To validate the described approach, simulations were performed on a 3R elbow space robot with full joint sensing. In addition, unmodeled sensor friction and noise were added to evaluate the robustness of the algorithm and make the simulations more realistic.

1 Introduction

In the present and the foreseeable future, there will be numerous scenarios where humans and robots must collaborate. These applications may involve designing robots to assist humans in performing complex or physically demanding tasks within industrial environments, or employing them as versatile service aides in households.

Consequently, in such situations, the monitoring of robot collisions becomes crucial to prevent or minimize the risk of human injury resulting from physical contact. The global objective of physical Human-Robot Interaction (**pHRI**) is to prevent upcoming undesired collisions, to handle unavoidable or intentional physical contacts in a safe and robust way and to reactively generate sensor-based motions in situation in which human and robot have to share their workspace.

Achieving this goal necessitates innovative mechanical designs for manipulator links and actuation systems, with a focus on reducing inertia and weight through compliant components.

In addition, it demands extensive utilization of external sensors to swiftly and reliably recognize the proximity between humans and robots, accompanied by the development of motion planning and control strategies that are attuned to human presence. A central and critical issue in this field is the effective management of collisions between robots and humans: the main problem concerns the minimization of potential human injuries resulting from physical contact. Unintended collisions must be avoided through vigilant monitoring of the work area, facilitated by external sensors.

The main challenge concerns the rapid and unpredictable relative movements that can occur between human and robot. In these contexts, reliance on exteroceptive sensors alone may prove insufficient to predict if a collision is going to occur or not.

Furthermore, when direct and intentional interaction between human and robot is required, the ability to distinguish between intentional and unintentional contact becomes critical because the type of contact will change the robot's response after collision. To this end, the acquisition of complete physical information from impact events, including the location and intensity of contact, becomes essential to enable the robot to respond effectively.

To comprehensively address the intricate issue of contact management, a unified framework called "*collision event pipeline*" is introduced here, as can be seen in Figure 1.

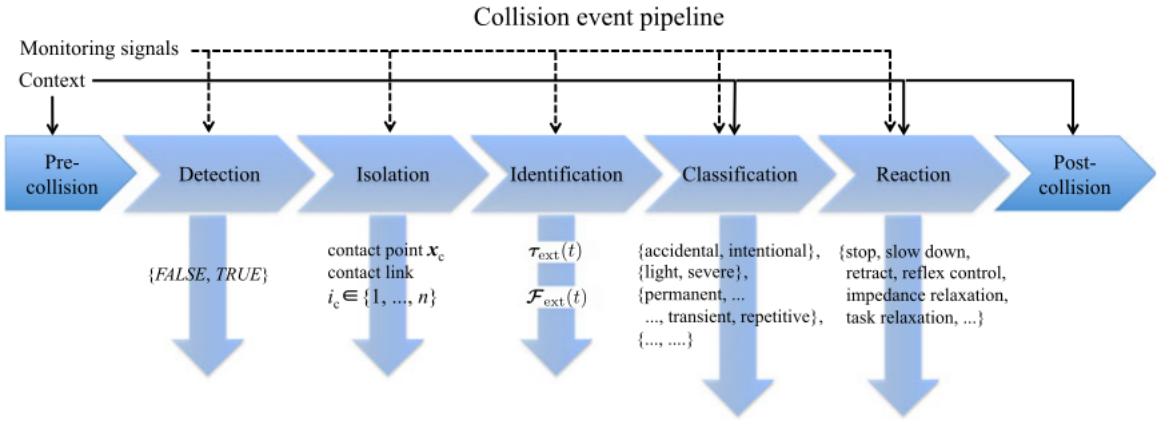

Figure 1: Seven phases of the collision event pipeline and their expected outputs

This framework is designed to encompass all the stages that collisions may go through, offering a systemic approach to their management and classification. Interestingly, the principles outlined in this work extend beyond traditional manipulator robots and find

application in various classes of robots, including mobile robots driven by user-sensed forces, upper-body components of anthropomorphic systems, and even aerial robots.

1.1 Collision event pipeline

In the complete *collision event pipeline* (see Fig. 1) are considered up to seven elementary phases. The stages from detection to identification are context-independent, while the remaining stages depend on internal and external factors, including the state of humans and the environment and ongoing tasks.

Below I will briefly describe the different phases of the pipeline.

1) *Precollision Phase*: in this phase the main goals are collision avoidance and anticipating robot motion to minimize impact. Planning a collision-free path requires knowledge of the geometry of the environment, but efficient conversion of offline motion planning to handle instantaneous changes is still a subject of research, particularly for human-aware scenarios. Collision anticipation relies on external sensors such as on-board vision or RGB-D cameras. Real-time collision-free path generation employs algorithms such as artificial potential and elastic stripes.

However, the rapid speed of human motion, typically an order of magnitude faster than that of high-ratio robots, poses a challenge in ensuring collision-free behavior in advance.

2) *Collision Detection*: this phase involves determining whether a robot has collided, usually during a brief impact event. Detection must be rapid and occur at any point in the robot's structure.

Establishing an appropriate threshold for monitoring signals is a challenge, with the goal of avoiding false positives while maintaining high sensitivity. One approach is to monitor electrical currents in the robot drives, looking for rapid changes caused by collisions. Another method compares actual motor torques or currents with model-based control laws, and deviations indicate a collision. An alternative approach to achieve collision detection and isolation involves the use of sensitive skins.

3) *Collision Isolation*: accurate identification of the specific part of the robot involved in a collision, such as a particular link of a serial manipulator, is critical for effective robot response. Collision isolation aims to identify the point of contact (x_c) or at least determine which link (i_c) of the robot's n-body structure was involved in the collision.

Some methods developed in the past often struggle to reliably achieve collision isolation, even with perfect knowledge of the robot's dynamics. These methods rely on calculations based solely on the desired nominal trajectory or calculate joint accelerations by inverting the mass matrix, distributing the dynamic effects of the collision over multiple joints.
Alternatively, they use acceleration estimates for torque prediction and comparison, introducing noise and inherent delays due to the double numerical differentiation of position data.
The common limitation of these methods is that the impact of a collision on a link can propagate to other link variables or joint controls due to the dynamic couplings of the robot, thus compromising the accuracy of the isolation.
4) *Collision Identification Phase*: in addition to identifying the specific aspects of a collision, such as the links and contact points involved, two other critical factors are directional information and the intensity of the generalized collision force. These factors are typically described in terms of the Cartesian force applied (F_{ext}) at the contact point or the resulting external joint torque (τ_{ext}) during the entire interaction event.
This information plays a substantial role in characterizing the collision event, sometimes providing a complete description. There have been developed methods that simultaneously achieved collision detection, isolation, and identification. This innovative approach treated collisions as anomalies in the behavior of the robot's actuation system. The detector design took advantage of the robot's generalized momentum decoupling property, enabling a more comprehensive understanding and management of collisions.
5) *Collision Classification Phase*: based on the information generated in the previous phases, we can interpret the collision nature in a context-dependent way, such as classifying the collision as accidental or intentional, light or severe, or even labeling its time course as permanent, transient, or repetitive.
6) *Collision Reaction Phase*: in response to a collision event, a robot needs to react purposefully, considering contextual information. Due to the rapid dynamics and high uncertainty of such situations, the robot's response should be integrated into the lowest control level. The simplest reactions, i.e. stopping the robot, might be insufficient and lead to inconvenient scenarios, such as the robot unnecessarily constraining or blocking a human.

To develop more effective reaction strategies, information from the phases of collision isolation, identification and classification should be utilized.

7) *Postcollision Phase*: after the robot has responded to a collision and reached a safe state, the next step is to determine for itself how to proceed. This decision-making process involves choices such as whether to continue the original task or abandon it and, if so, how to do so effectively.

For example, if the collision has been classified as intentional, the robot may recognize the human's intention to initiate a specific physical collaboration. However, the challenge of making these decisions remains relatively open. Solving this problem requires not only control-level decision making, but also access to global environmental information and reasoning.

2 Dynamic model

In this section I am going to describe the dynamic model of a robot with elasticity concentrated at the joints and its relevant properties.

2.1 Elastic joints robot

In this section I am going to consider robot involved in the simulations as open kinematic chain having *n+1* rigid bodies, the base and n links interconnected by n (rotary or prismatic) joints undergoing deflection. From a mechanical point of view, each motor is an additional rigid body with its inertial properties.

In order to take into account the elasticity concentrated at the joints the generalized coordinates need to be doubled (motor and link positions). Motor positions are indicated with $\boldsymbol{\theta} \in \boldsymbol{\mathcal{R}^n}$ while link positions are indicated with $\boldsymbol{q} \in \boldsymbol{\mathcal{R}^n}$.

These two quantities are related with the transmitted *elastic torque* expressed as follow:

$$\boldsymbol{\tau_J = K_J(\theta - q)} \tag{1}$$

With joint stiffness matrix $\boldsymbol{K_J = diag\{K_{J,i}\} \in \mathcal{R}^{nxn}}$ is diagonal and positive definite.

To introduce the dynamic model of a robot with elasticity concentrated at joints the following standard assumptions are made:

1) Joint deflections are small, so that flexibility effects are limited to the domain of linear elasticity.

2) The actuators' rotors are modeled as uniform bodies having their center of mass on the rotation axis.

3) Each motor is located on the robot arm in a position preceding the driven link.

For robot with n viscoelastic joints, we consider the so called reduced model of **Spong** (1), which assumes no inertial couplings between the motor and link bodies. In this case the *link dynamics* becomes:

$$\boldsymbol{M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + K_J(q-\theta) = D_J(\dot{\theta}-\dot{q}) - \tau_{F,q} + \tau_{ext}} \tag{2}$$

where $\boldsymbol{M(q) \in \mathcal{R}^{nxn}}$ is the symmetric and positive-definite inertia matrix, $\boldsymbol{C(q,\dot{q})\dot{q} \in \mathcal{R}^n}$ is the centripetal and Coriolis vector, factorized with the matrix C of Christoffels' symbols, and $\boldsymbol{g(q) \in \mathcal{R}^n}$ is the gravity vector. The non-conservative terms on the right-side of Equation 2 are the *active motor torque* $\boldsymbol{\tau_m \in \mathcal{R}^n}$ and the *dissipative torque* $\boldsymbol{\tau_{F,q}}$ that represents the friction terms acting on the link side of the joint (e.g. $\boldsymbol{\tau_{F,q} = f(\dot{q})}$ with $\boldsymbol{f(\dot{q})\dot{q} \geq 0}$).

The joint damping matrix $\boldsymbol{D_J = diag\{D_{J,i}\} \in \mathcal{R}^{nxn}}$ is diagonal and positive semidefinite. All the dissipative terms have been collected on the right-hand side of 2. The value of $\boldsymbol{\tau_J}$ in 1 is also the output of joint torque sensing devices, when available.

For the *motor dynamics* I assume a decoupled second-order system:

$$\boldsymbol{B\ddot{\theta} + K_J(\theta - q) = \tau_m - D_J(\dot{\theta}-\dot{q}) - \tau_{F,\theta}} \tag{3}$$

where $\boldsymbol{B = diag\{B_i\} \in \mathcal{R}^{nxn}}$ is the diagonal positive-definite motor inertia matrix, $\boldsymbol{\tau_{F,\theta}}$ contains friction terms acting on the motor side of the joint.

In many cases, the mechanical design of the transmission/reduction elements is such that one can neglect the joint damping, $\boldsymbol{D_J \approx 0}$ as well as the influence of friction on the link side $\boldsymbol{\tau_{F,\theta} \approx 0}$. On the other hand, the motor friction $\boldsymbol{\tau_{F,\theta}}$ is usually not negligible.

Therefore, when also including the presence of joint torques due to contact forces (acting on the link dynamics), we shall consider the following dynamic model of robots with flexible joints:

$$\boldsymbol{M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau_J + \tau_{ext} = \tau_{tot,J}} \tag{4}$$

$$\boldsymbol{B\ddot{\theta} + \tau_J = \tau_m - \tau_{F,\theta}} \tag{5}$$

Here in 4 the driving torque of the dynamics is the sum of the *elastic torque* $\boldsymbol{\tau_J}$ and the *external torque* $\boldsymbol{\tau_{ext}}$ denoted by $\boldsymbol{\tau_{tot,J}} \in \boldsymbol{\mathcal{R}^n}$ while equation 5 represents the dynamic equation for a motor.

In the case of a flexible coupling robot, the total kinetic energy also includes the part related to motors as follow:

$$\boldsymbol{T = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\dot{\theta}^T B\dot{\theta}} \tag{6}$$

In the total potential energy, the elastic part is also considered as follow:

$$\boldsymbol{U = U_g + U_e = U_g(q) + \frac{1}{2}(\theta - q)^T K_J(\theta - q)} \tag{7}$$

In this way the total mechanical energy becomes:

$$\boldsymbol{E_J = T + U = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\dot{\theta}^T B\dot{\theta} + U_g(q) + \frac{1}{2}(\theta - q)^T K_J(\theta - q)} \tag{8}$$

From 4 and 5, it follows that the power $\boldsymbol{\dot{E}_J}$ exchanged between the environment and the system is given by:

$$\boldsymbol{\dot{E}_J = \dot{q}^T \tau_{ext} + \dot{\theta}^T(\tau_m - \tau_{F,\theta})} \tag{9}$$

The torque $\boldsymbol{\tau_J}$ is only acting between the system and not between system and environment.

3 Collision monitoring method

Solving a collision detection problem means to decide whether a physical collision is occurred or not, limiting as much as possible the occurrence of false positives or false negatives.
The main idea is to introduce a *collision detection* function $cd(.)$ that maps the monitoring signal $\boldsymbol{\sigma(t)}$ into the two disjoint collision classes TRUE (collision) and FALSE (no collision):

$$cd : \sigma(t) \rightarrow \{TRUE, FALSE\} \tag{10}$$

Ideally, this classification is obtained by:

$$cd(\sigma(t)) \begin{cases} \text{TRUE, if } \sigma(t) \neq 0 \\ \text{FALSE, if } \sigma(t) = 0 \end{cases}$$

In practice, the accurate detection of a collision requires appropriate thresholding for robustness because there are many non idealities to be taken in account: for instance, position and velocity sensor noise $\boldsymbol{n_q n_{\dot{q}}}$ or friction torque $\boldsymbol{\tau(q, \dot{q}, \tau_{tot}, \theta, t)}$. Due to these errors in measurements the monitoring signal becomes in practice $\boldsymbol{\sigma(t) \neq 0}$ even when no contact is present.
For this reason there is the need to take into account a small constant robustness margin $\boldsymbol{\epsilon_\sigma}$ such that:

$$cd(\sigma(t)) \begin{cases} \text{TRUE, if } |\sigma(t)| > \epsilon_\sigma \\ \text{FALSE, if } |\sigma(t)| \leq \epsilon_\sigma \end{cases}$$

Since a collision is expected to change the energy level of the robotic system, an intuitive choice of a monitoring function is to resort to an energy argument. For this reason, in (2) it is defined the following scalar quantity as the monitoring signal:

$$\boldsymbol{\sigma_{EJ}(t) = k_\sigma(E_{EJ}(t) - \int_0^t (\dot{\theta}^T(\tau_m - f_m(\dot{\theta})) + \sigma_{EJ})ds - E_{EJ}(0))} \tag{13}$$

with initial value $\boldsymbol{\sigma_{EJ}(0) = 0}$, gain $\boldsymbol{k_\sigma > 0}$, $\boldsymbol{E_{EJ}(t)}$ being the estimate of the total robot energy at time $\boldsymbol{t \geq 0}$ and $\boldsymbol{f_m(q, \dot{q})}$ is the friction torque.
The monitoring signal $\boldsymbol{\sigma_{EJ}}$ can be computed using the commanded torque $\boldsymbol{\tau_m}$, the measured link position $\boldsymbol{q}$, and the velocities $\boldsymbol{\dot{q}}$ and $\boldsymbol{\dot{\theta}}$.

No acceleration measurement or its estimation is needed.

This scalar quantity has limitations in detecting all types of collisions. These limitations arise from two specific situations:

1) Collisions that do not generate external power occur when the collision force, denoted as $\boldsymbol{F_{ext}}$, acts orthogonally to the velocity of the contact point.

$$\dot{\boldsymbol{q}}^{\boldsymbol{T}}\boldsymbol{\tau_{ext}} = \dot{\boldsymbol{q}}^{\boldsymbol{T}}\boldsymbol{J_c^T}(\boldsymbol{q})\boldsymbol{F_{ext}} = \boldsymbol{V_c^T}\boldsymbol{F_{ext}} = \boldsymbol{0} \leftrightarrow \boldsymbol{V_c} \perp \boldsymbol{F_{ext}} \tag{14}$$

2) When the robot is at rest $\dot{\boldsymbol{q}}^{\boldsymbol{T}}\boldsymbol{\tau_{ext}} = \boldsymbol{0}$ and so the collision is not detected.

In the simulations I am going to consider ideal conditions in which the mechanical energy measured is the real one and not an approximation.

4 Block scheme

In Figure 2 is shown the overall block scheme used in the simulations.

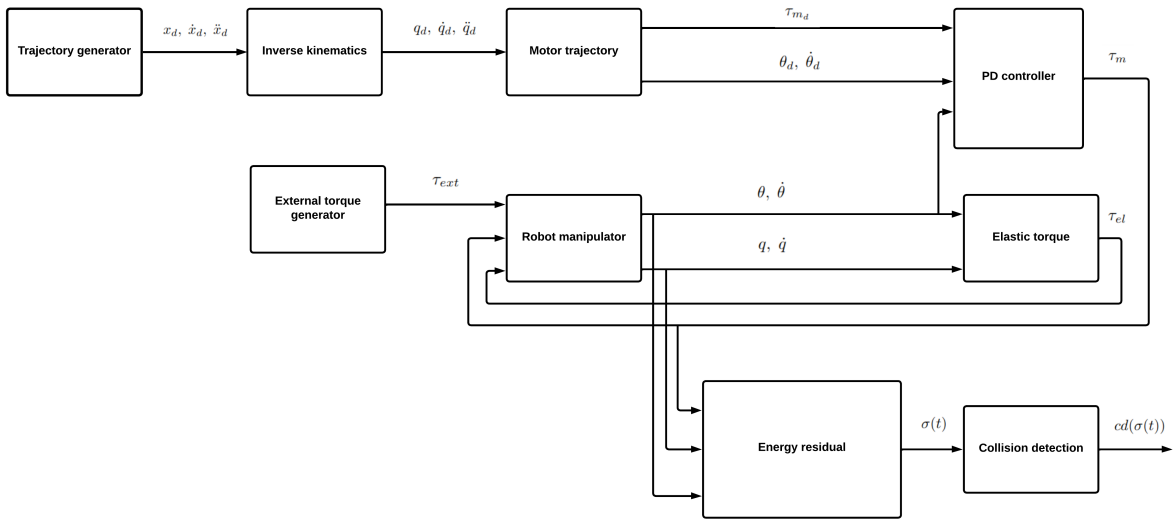

Figure 2: Block scheme

Initially, an ideal trajectory is generated for the robot end-effector, which originates from the first block. Next, the inverse kinematics of the robot is calculated to derive the desired positions, velocities, and accelerations for all joints.

Taking into account the inherent elasticity of the joints, the motor torque and the desired motor position and velocity are then determined based on the above joint parameters.

This process is described in detail in (3).

Next, the *PD controller* comes into play, which generates the actual torque commands applied by the robot. By simulating the behavior of the 3R manipulator, taking into account the external forces resulting from collisions, the actual positions and velocities of the joints and motors can be measured, allowing information on the elastic forces involved to be recovered. With these data (i.e. position and velocity of joints and the motor torque), it is possible to calculate the energy residual, the value of which serves as an indicator for determining the occurrence of a collision.

5 Simulations

To validate the proposed approach, I conducted simulations using Simulink on a 3R spatial elbow-type robot, as depicted in Figure 3 with the dynamic and kinematic parameters of its three links reported in Table 1.

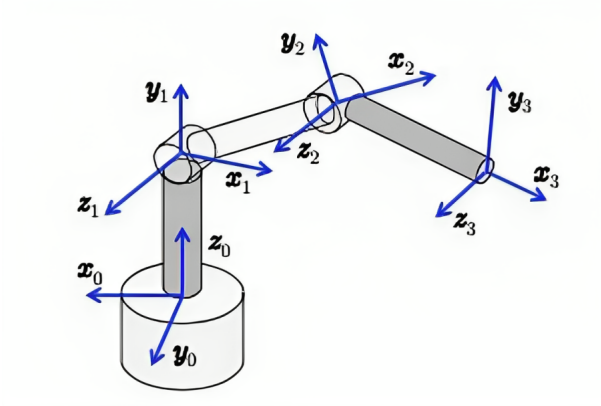

Figure 3: 3R spatial elbow-type robot

This robot features elastic joints and comprehensive joint sensing, including two position encoders and joint torque sensors for each joint.

Link	Mass	Length	Cylinder Radius
1	15 kg	0.5 m	0.2 m
2	10 kg	0.5 m	0.1 m
3	5 kg	0.4 m	0.1 m

Table 1: Robot parameters

The robot is required to execute the trajectory depicted in Figure 4, which includes:
1) a circular arc in the x z plane if $0s \leq t \leq 10s$:

$$\begin{cases} x = x_0 + r \cdot cos\theta \\ y = y_0 \\ z = z_0 + r \cdot sin\theta \end{cases} \tag{15}$$

where $x_0 = -0.3, y_0 = 0.5, z_0 = 0.5, r = 0.3m$ and $\theta = w \cdot t$, where w = $\frac{\pi}{9}$.

2) a stopping point if $10s \leq t \leq 20s$:

$$\begin{cases} x = x_0 + r \cdot cos\theta \\ y = y_0 \\ z = z_0 + r \cdot sin\theta \end{cases} \tag{16}$$

where $x_0 = -0.3, y_0 = 0.5, z_0 = 0.5, r = 0.3m$ and $\theta = w \cdot 10s$, where w = $\frac{\pi}{9}$.

3) a straight line in the xy plane if $20s \leq 30s$:

$$\begin{cases} x = x_0 + r \cdot cos\theta - 0.1 \cdot \dfrac{t - T_2}{T_3 - T_2} \\ y = y_0 - 0.4 \cdot \dfrac{t - T_2}{T_3 - T_2} \\ z = z_0 + r \cdot sin\theta \end{cases} \tag{17}$$

where $x_0 = -0.3, y_0 = 0.5, z_0 = 0.5, r = 0.3m$ and $\theta = w \cdot 10s$, where w = $\frac{\pi}{9}$.

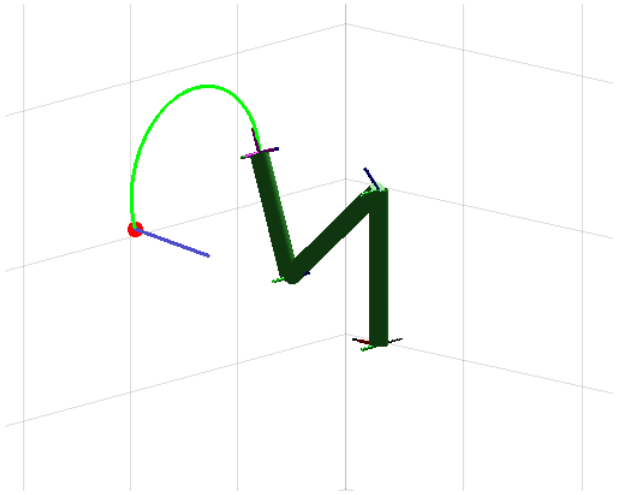

Figure 4: Ideal trajectory

While executing the trajectory, I introduced simulated external contacts by applying forces to different parts of the robot as:

$$\begin{cases} \boldsymbol{F_{ext}} = [\mathbf{-30; -20; 0}] \text{ if } 3s \leq t \leq 6s \text{ applied to the end-effector.} \\ \boldsymbol{F_{ext}} = [\mathbf{0; 0; 15}] \text{ if } 12s \leq t \leq 16s \text{ applied at } 0.2\ m \text{ from the second joint to the third link.} \\ \boldsymbol{F} = [\mathbf{10; -20; 0}] \rightarrow \boldsymbol{F_{ext}} = \dfrac{\mathbf{1}}{\mathbf{2}}\boldsymbol{F} \cdot \mathbf{(t - 18)} \text{ if } 18s \leq t \leq 23s \text{ applied to the end-effector.} \\ \boldsymbol{F_{ext}} = [\mathbf{0; 0; -35}] \text{ if } 26s \leq t \leq 29s \text{ applied at } 0.35\ m \text{ from the second joint to the second link.} \end{cases} \tag{18}$$

Therefore, to express the torques perceived at joints, I used the following formula with the corresponding Jacobian:

$$\boldsymbol{\tau_{ext}} = \boldsymbol{J^T_{position}} \cdot \boldsymbol{F_{ext}} \tag{19}$$

After validating the presented approach under ideal conditions, I introduced some non-idealities, such as viscous friction on the joints and sensor noise on the encoders and joint torque sensors.

I modeled the viscous friction in the following way:

$$\boldsymbol{f(\dot{\theta})} = \boldsymbol{-D_v \cdot \dot{\theta}} \tag{20}$$

where $D_v = 0.8\frac{Nms}{rad}$.

I opted to represent sensor noise as white Gaussian noise, formulated as follows:

$$\boldsymbol{i} = \boldsymbol{i} + \boldsymbol{i \cdot \gamma \cdot \mathcal{N}(0, 1)} \tag{21}$$

In this context, the symbol γ denotes the error percentage that delineates the attributes of our sensor, distinguishing between high, medium, or low quality. I conducted a comparative analysis between two scenarios: one where the noise sensor is solely applied to the position encoders and another where it is exclusively integrated into the joint torque sensors.

In consideration of uncertainties rather than relying on idealized conditions, I introduced a constant robust margin, which value is $\epsilon_\sigma = 2[W]$, within the framework of the collision monitoring method.

I evaluated the approach's robustness by contrasting the outcomes with and without accounting for friction in the residual term, essentially determining whether the model's friction expression was known in advance or not.

5.1 Ideal case

According to (3), the nominal static torque in the case of a fixed desired position is denoted as $\boldsymbol{\tau_d = g(q_d)}$, and the corresponding desired motor position is $\boldsymbol{\theta_d = q_d + K_J^{-1} g(q_d)}$. Where $\boldsymbol{g(q_d)}$ is the gravity vector associated to the 3R elbow type robot $\boldsymbol{g(q) = (\frac{\partial U_g(q)}{\partial q})^T}$. The control law employed to guide the robot from the desired trajectory previously described is:

$$\boldsymbol{\tau = \tau_d + K_P(\theta_d - \theta) - K_D\dot{\theta}} \tag{22}$$

with symmetric and diagonal $\boldsymbol{K_P > 0}$ (as a minimum) and $\boldsymbol{K_D > 0}$.

During all the simulations I chose the following values for the gains and the springs stiffness:

K_J	K_P	K_D	K_0
$9k\frac{Nm}{rad}$	$150\frac{Nm}{rad}$	$50\frac{Nms}{rad}$	$900\frac{1}{s}$

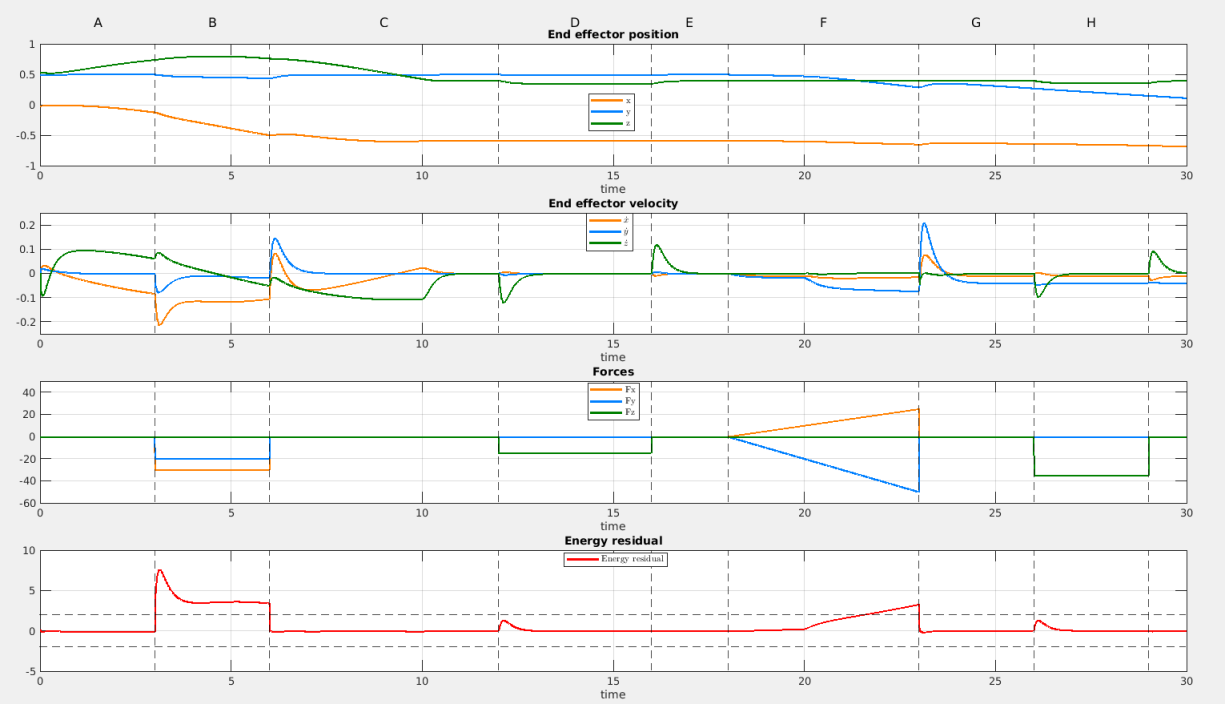

Figure 5: Simulation in ideal conditions

As can be seen from Figure 5 the simulation is divided in 7 phases:

1) **Phase A**: in this phase, no external forces act on the robot, allowing it to precisely follow the ideal trajectory (a circular arc on the xz plane). Throughout this phase, the energy residual remains consistently at 0 since no collision occurs.

2) **Phase B**: in this phase, an external force is applied to the end-effector, causing the robot to deviate from its ideal trajectory. The energy residual deviates from 0 as soon as contact with external force occurs. This shows the effectiveness of the energy residual to correctly detect an external contact.

3) **Phase C**: during this phase, no external forces are exerted on the robot. The robot accurately executes the initial part of the trajectory, which includes the arc of the circle, and then stops at a predefined stopping point.

4) **Phase D**: in this phase, an external force is applied along the z-axis to the third link at 0.2 m from the third joint . However, since the robot is at rest with $\dot{\boldsymbol{q}} = \mathbf{0}$, the energy residual remains below the threshold. Consequently, the collision is not correctly recognized.

5) **Phase E**: an external force in the z direction is no longer applied, and the robot remains at rest.

6) **Phase F**: the robot is stationary, and an increasing force to the end-effector is applied. Initially, the energy residual remains below the threshold as the robot doesn't move. Nonetheless, when the robot starts its motion along a straight line in the xy plane at t = 18s, the collision is not promptly identified, taking approximately 1.5 seconds to do so. Beyond this time frame, the energy residual surpasses the predefined safe threshold $\boldsymbol{\epsilon_\sigma}$ and correctly registers the collision event.
This occurs due to the fact that the robot is moving with a really low link velocity $\dot{\boldsymbol{q}}$.

7) **Phase G**: while the robot continues to move along the xy plane, a force is applied in the z direction. Since this force is orthogonal to the end-effector's velocity, it isn't recognized, as indicated by the monitoring signal not exceeding the safe threshold.

Figure 6 clearly illustrates the position and velocity profiles of the end-effector with respect to ideal motion. This visual representation emphasizes that, in the absence of collisions, the end-effector consistently adheres to the ideal motion.

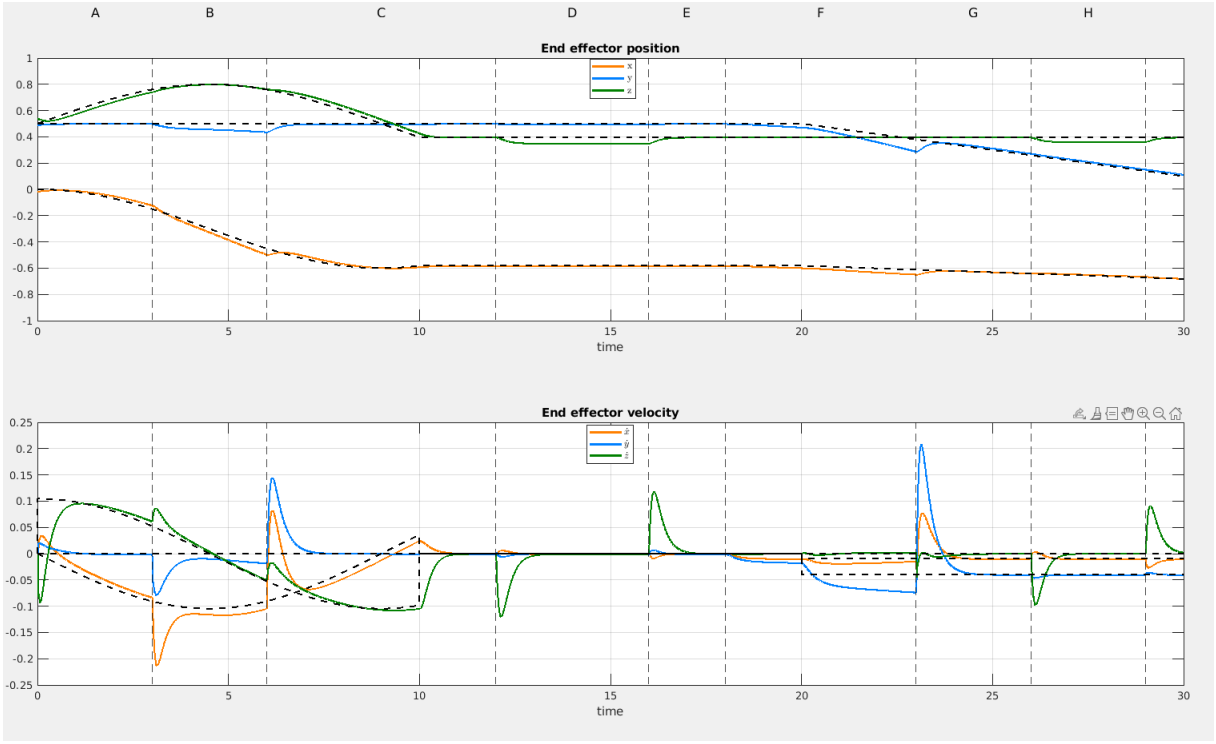

Figure 6: Ideal position and velocity profiles of the end-effector.

5.2 Real case

After evaluating the approach under ideal conditions, I proceeded to examine its robustness when subjected to non-ideal factors. Specifically, I introduced viscous friction torque to the motor side and added sensor noise to the position encoders.

As previously mentioned, the expression for the friction is as follows:

$$\boldsymbol{f(q, \dot{q}) = -Dv \cdot \dot{\theta}} \tag{23}$$

where $D_v = 0.8 \frac{Nms}{rad}$.

I then conducted a comparison between two distinct scenarios. In the first case, only the position encoders were subject to Gaussian noise, while in the second case, only the joint torque sensors were affected by this noise.

The expression for the Gaussian noise is as follows:

$$\boldsymbol{i = i + i \cdot \gamma \cdot \mathcal{N}(0, 1)} \tag{24}$$

where i is the original signal to which I applied the noise.

Given the presence of uncertainties, I anticipate that the monitoring signal may take non-zero values even during intervals when no collisions are taking place and the magnitude of these value can also exceeds the safe robust margin $\boldsymbol{\epsilon_\sigma}$ previously introduced.

5.2.1 Noise at the position encoders

In this instance, to ensure reasonable results, I configured the error percentage, denoted as γ, to be 0.001. This value means that the sensor is introducing an error of 0.1% and so I am considering an high quality sensor.
I maintained the same configuration as previously described and conducted simulations in order to see the behavior of the describe framework.

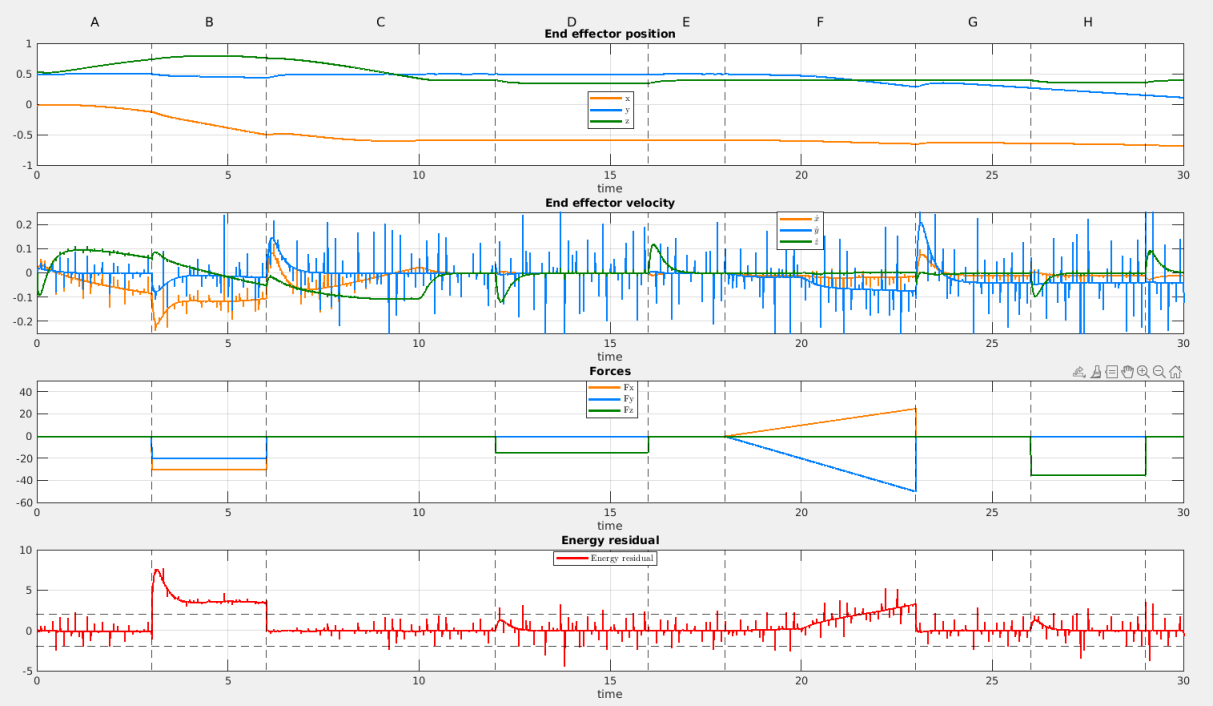

Figure 7: Simulation in the presence of noise on the position encoders.

As illustrated in Figure 7, there are frequent occurrences where the monitoring signal deviates from the safe threshold, even though the robot is not encountering any physical force collisions. This discrepancy is primarily attributed to the noise present in the position encoders (both at link and motor side).

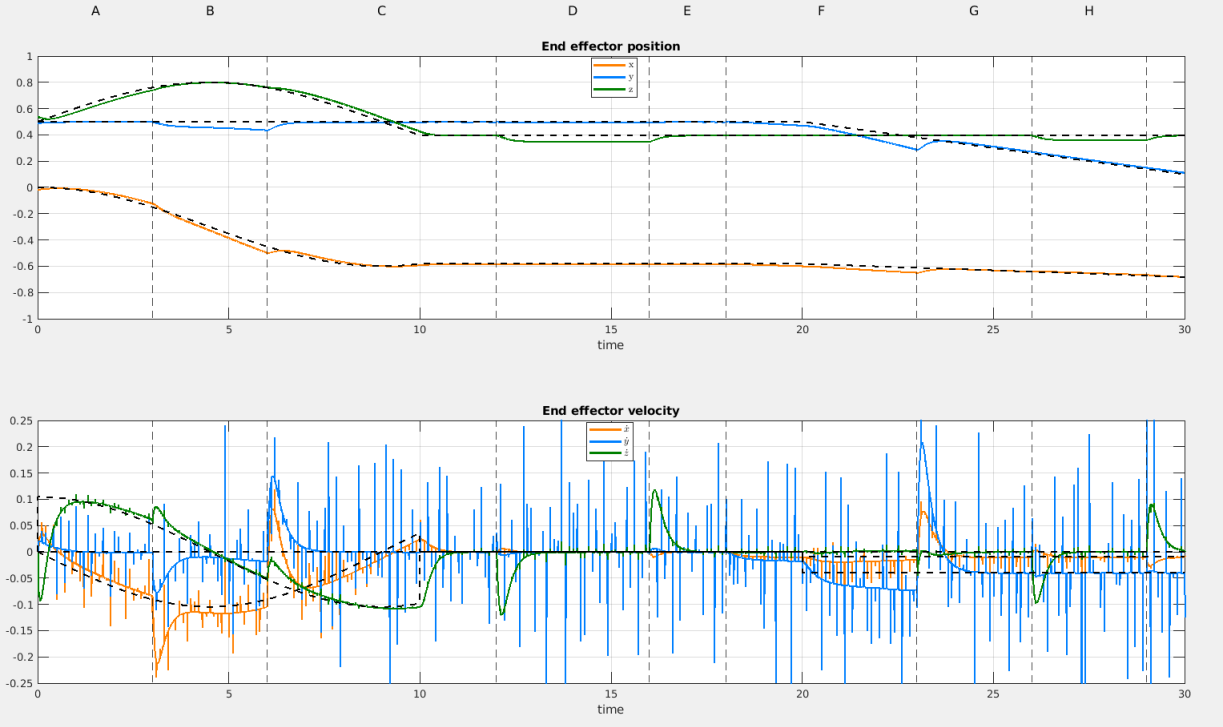

Figure 8: Position and velocity of the end-effector in the presence of noise on the position encoders.

This issue arises because the velocity profile of the end-effector, as shown in Figure 8, exhibits multiple spikes, leading to a corresponding fluctuation in the power exchange. Despite our selection of a high-quality sensor (as indicated by the low percentage error $\gamma = 0.001$), further improvements seem unattainable.

5.2.2 Noise at the joint torque sensors

I conducted the same simulation, but I introduced an error measurement on the joint torque sensors instead of the position encoders. After several trials, I decided to establish $\gamma = 0.01$, which means that the sensor introduces an error of 1% (medium quality sensors).
Figure 9 illustrates the collision scenario with joint torque sensor noise, where the energy residual profile remains consistent with the ideal case.
In Figure 10, it can be observed that the elastic torque exhibits some oscillation, but it does not introduce significant fluctuations in the residual that is able to correctly recognizes the external collision.

Based on these findings, it can be asserted that when constructing a real robot, it is more advantageous to equip it with lower quality joint torque sensors as opposed to high-quality position encoder sensors. This approach not only leads to cost savings but also yields superior performance in the collision detection framework.

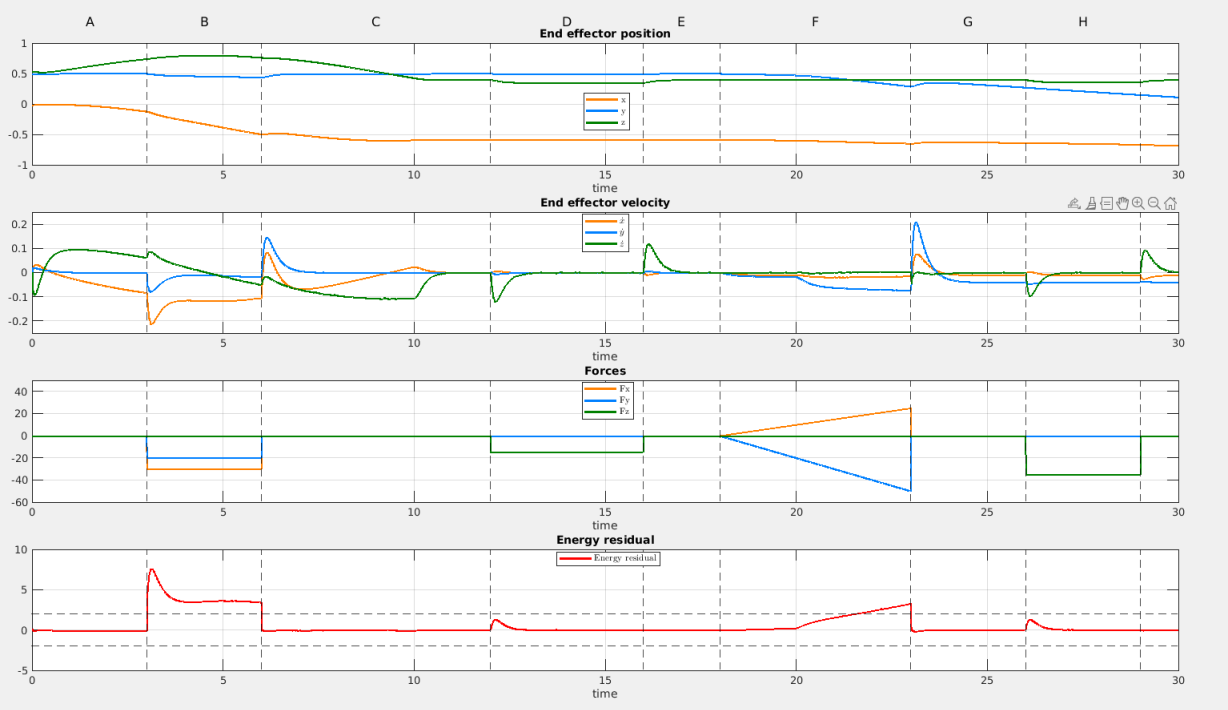

Figure 9: Simulation in the presence of noise on the joint torque sensors

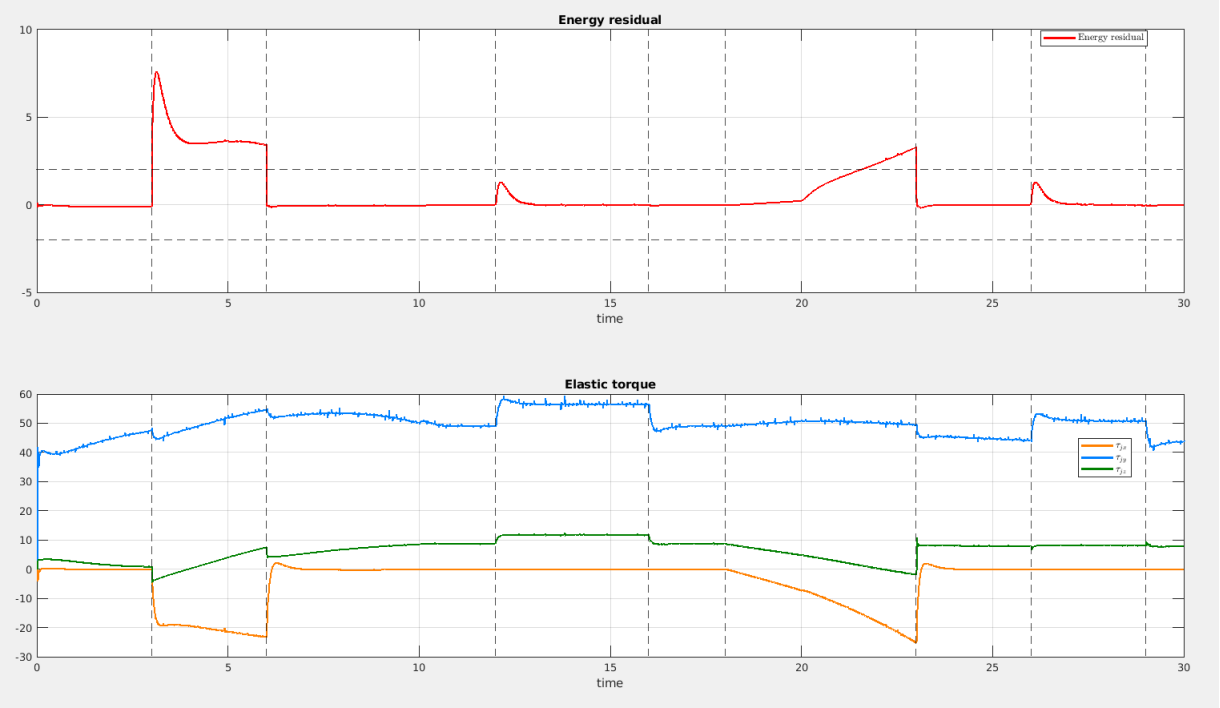

Figure 10: elastic torque and energy residual in the presence of noise on the joint torque sensors

5.2.3 No friction term in the residual

To test the robustness of the energy residual method I decided to eliminate the friction term in the residual as follow:

$$\boldsymbol{\sigma_{EJ}(t) = k_\sigma(E_{EJ}(t) - \int_0^t (\dot{\theta}^T(\tau_m - f_m(\dot{\theta})) + \sigma_{EJ})ds - E_{EJ}(0)} \tag{25}$$

$$\boldsymbol{\sigma_{EJ}(t) = k_\sigma(E_{EJ}(t) - \int_0^t (\dot{\theta}^T\tau_m + \sigma_{EJ})ds - E_{EJ}(0)} \tag{26}$$

I repeated the same simulation and, as shown in Figure 11, it is evident that the residual profile is able to identify collisions. This implies that, even in the absence of viscous friction modeling, the demonstrated approach remains robust in identifying potential collision forces.

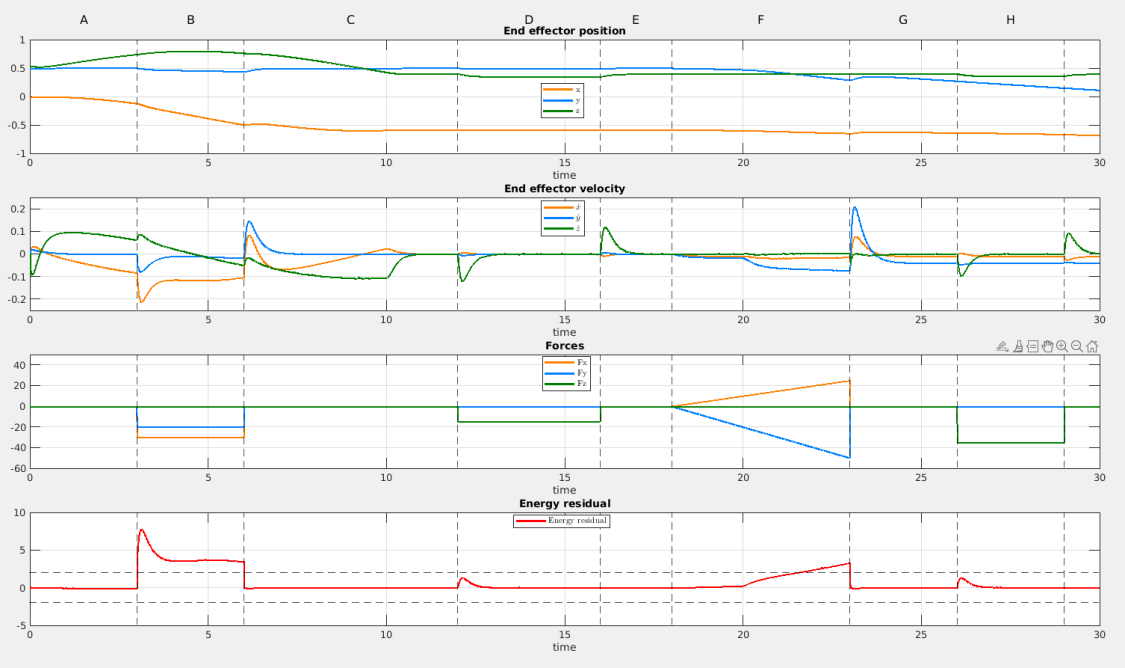

Figure 11: simulation in the case of no modeling of the viscous friction.

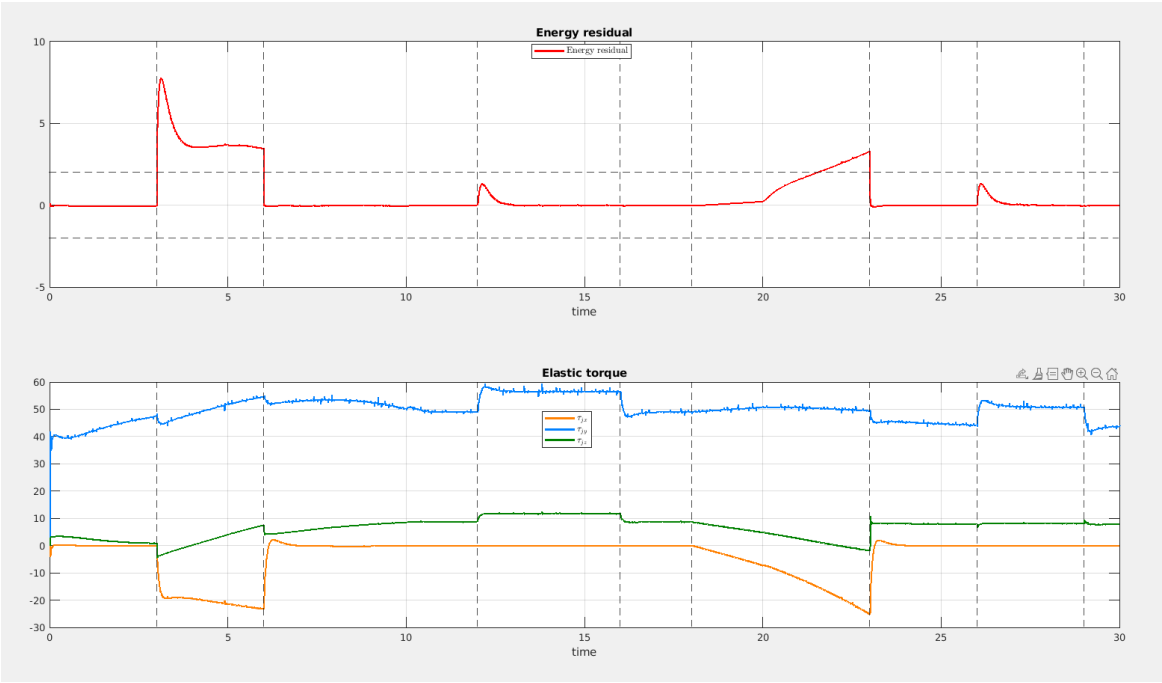

Figure 12: elastic torque and energy residual in the case of no modeling of the viscous friction

5.2.4 Different value of the viscous coefficient

In this section I want to analyze the effect of the different viscous friction coefficient D_v on the proposed energy-based framework. I want to compare 4 cases:

1) $D_v = 0.8 \ \frac{Nms}{rad}$ see Figure 13.

2) $D_v = 2 \ \frac{Nms}{rad}$ see Figure 14.

3) $D_v = 5 \ \frac{Nms}{rad}$ see Figure 15.

4) $D_v = 10 \ \frac{Nms}{rad}$ see Figure 16.

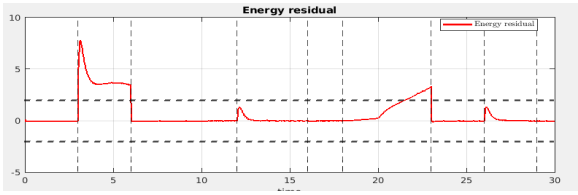

Figure 13: $D_v = 0.8$

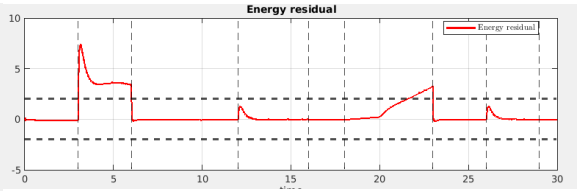

Figure 14: $D_v = 2$

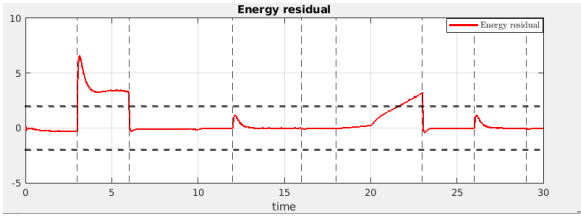

Figure 15: $D_v = 5$

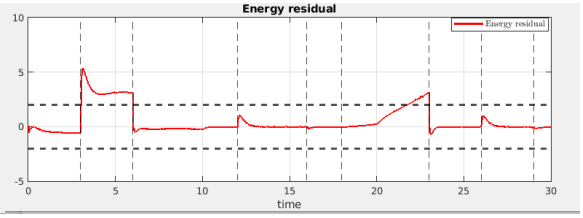

Figure 16: $D_v = 10$

Figure 17: Comparison of Images

The images clearly demonstrate the robustness of the energy-based framework, even when subjected to high viscous friction torque, as shown in Figure 16. In particular, even in the case of the highest value of the viscous friction coefficient ($D_v = 10$), the energy residual accurately detects external collisions.

However, the increase in the friction term presents a drawback at the end of the application of an external force. In this cases, the residual energy shows some fluctuation but it remains within the acceptable constant margin. ϵ_σ and so it does not produce false positives.

This detail can be clearly seen in the Figure 18.

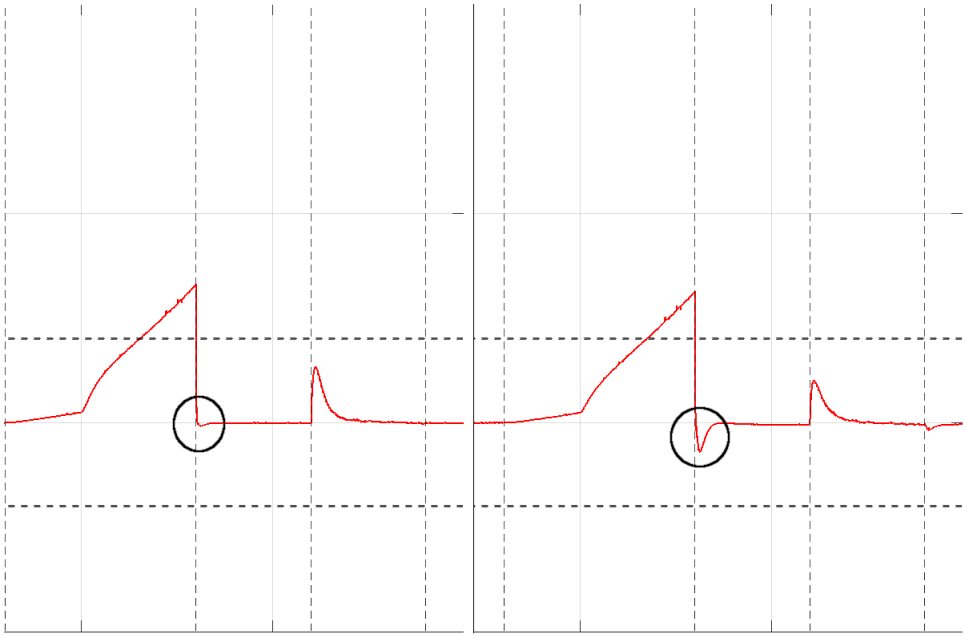

Figure 18: Effect of an higher viscous friction coefficient between $D_v = 0.8$ (left image) and $D_v = 10$ (right image).

6 Conclusion

This report takes an in-depth look at the effectiveness of the energy-based residual method in the field of collision detection. The exploration begins with an introductory overview of the collision event pipeline, providing insights into its main stages.

Since the simulations were carried out on a 3R elbow type robot equipped with elastic joints, the report delves into the dynamic model of the robot, taking into account the influence of elasticity.

Next, the report introduces a monitoring signal based on the energy characteristics of the robots used in the simulations. This signal has the crucial purpose of collision detection.

Initially, the methodology was validated under ideal conditions. Four separate forces were applied during specific time intervals and on different robot links. In this case the monitoring signal showed accurate collision detection capability, recording values that exceeded the established safety threshold exclusively during periods of external contact force application.

The report then proceeds to evaluate the robustness of the method by introducing Gaussian noise into both the position encoders and the coupling torque sensors, along with viscous friction on the motor side.

The results show a preference for equipping the robot with lower quality joint torque sensors than with high quality position encoders. In the latter case, the energy residual shows many false positives: it has many peaks that exceed the robust constant margin.

In addition, the report evaluated the strength of the method in the presence of unmodeled viscous friction, without incorporating it into the residual energy expression. Even in this scenario, the method effectively recognized collisions without causing significant alterations in the monitoring signal profile, reaffirming its reliability.

To conclude the series of experiments, I chose to increase the viscous coefficients and present four separate cases. In the scenario with the highest value of the friction coefficient, the method successfully identified all collisions, with the only discernible differences in the transient phases and the cessation of applied forces. Remarkably, the shifts in the monitoring signal remained within acceptable limits.

However, it is essential to recognize the limitations of this method. In particular, when external contact forces act perpendicular to the velocity of the contact point, the energy expended becomes practically zero, leading to a zero monitoring signal, even in the presence of a collision. The same situation occurs when the link velocity is zero or nearly so.

A potential remedy is to collaboratively introduce an additional monitoring signal based on the external torque, thereby improving collision detection capabilities in such situations.

References

[1] M. W. Spong, "Modeling and control of elastic joint robots," *Journal of Dynamic Systems Measurement and Control-transactions of The Asme*, vol. 109, pp. 310–319, 1987. [Online]. Available: https://api.semanticscholar.org/CorpusID:119935441

[2] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[3] A. De Luca and W. Book, *Robots with Flexible Elements*. Berlin, Heidelberg:

Springer Berlin Heidelberg, 2008, pp. 287–319. [Online]. Available: https://doi.org/10.1007/978-3-540-30301-5_14