Treball de Fi de Grau/Màster

Master's Degree in Automatic Control and Robotics

SUPERVISION AND FAULT TOLERANCE FOR ASSITIVE ROBOTICS

MEMÒRIA

Autor: Alberto San Miguel Tello
Director/s: Dr. Guillem Alenyà Ribas
Dr. Vicenç Puig Cayuela

Convocatòria: June 2019

ETSEIB

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

UPC

Abstract

In this Master Thesis, the supervision and control problem of service robots in unknown anthropic domains has been addressed from the Fault Detection and Diagnosis (FDD) framework, presenting a complete Fault-Tolerant scheme able to detect, isolate and compensate the effects of an exogenous force acting on a robotic manipulator. Therefore, a systematic approach has been presented, applied to the TIAGo head subsystem, to obtain a Takagi-Sugeno representation suitable for a Parallel Distributed Controller, with the main advantage of defining the complete behaviour of the system using only its representation at the operational limits. Additionally, the Robust Unknown Input Observer for Takagi-Sugeno Models has been implemented for an incomplete information model scenario, which allows decoupling the given estimation from the effect of exogenous faults, disregarding its behaviour nor eventuality. Finally, a characterization of the real robot actuators has been performed, in order to design the suitable mechanisms for their implementation into the complete Fault-Tolerant scheme.

Acknowledgments

Firstly, I would like to thank my supervisors Vicenç Puig and Guillem Alenyà for their support and for giving me the opportunity to work with them on this project, which represents the beginning of my PhD. Thesis. I am also very grateful for all the support given along this work by the staff at Institut de Robòtica i Informàtica Industrial (IRI), especially to Sergi Hernández and Patrick J. Grosch and my office mates. I want to thank too all my Master colleagues, for being the perfect company along this two-year adventure, in both personal and professional facets.

Finally, I would like to sincerely thank my parents, Ana and Jesús, and my beloved Laura for being my cornerstone up to now and the days to come; and all my friends for all the given encouragement along my engineering career.

This work has been supported by the Spanish State Research Agency through the María de Maeztu Seal of Excellence to IRI MDM-2016-0656.

Contents

ETSEIB

List of Figures

List of Tables

List of Acronyms

AI	Artifical Intelligence
BMI	Bilinear Matrix Inequality
CV	Computer Vision
DH	Denavit Hartenberg
DOF	Degrees Of Freedom
FDD	Fault Detection and Diagnosis
HRC	Human-Robot Collaboration
LME	Linear Matrix Equality
LMI	Linear Matrix Inequality
LPV	Linear Parameter Varying
LQC	Linear Quadratic Control
LTI	Linear Time Invariant
NN	Neural Net
PDC	Parallel Distributed Control

pHRI	Physical Human-Robot Interaction
RUIO-TS	Robust Unknown Input Observer for TS Model
SMC	Sliding Mode Control
TS Model	Takagi-Sugeno Model

Nomenclature

α, β	$\mathbb{D}$-stabilization strip region parameters
$\ddot{\theta}_i$, $\dot{\omega}_i$	Angular acceleration of joint i
$\dot{\theta}_i$, ω_i	Angular velocity of joint i
$\mathcal{B}$	Coriolis terms of a manipulator in configuration-space form
$\mathcal{C}$	Centrifugal coefficients of a manipulator in configuration-space form
$\mathcal{G}$	Gravity effects of a manipulator in configuration-space form
$\mathcal{M}$	Mass matrix of a manipulator in configuration-space form
$\nu(k)$	RUIO-TS internal state at time instant k
$\overline{e}_i$	LQR Maximum admissible value for i-th term
ψ	Apkarian Filter gain
ρ_i	LQR tuning weight for i-th term
σ, r	$\mathbb{D}$-stabilization circular region parameters
τ	Torque vector

τ_s	Time constant of discrete-time pole s
θ_i	Angular position of joint i
$f_i(k)$	Exterted torque at i-th joint by an exogenous force
h_i	*Firing probability* associated to i-th Model rule
K_ω	Angular speed-CEMF voltage constant
K_τ	Torque-current constant
K_i	State-feedback gain associated to i-th Control rule
M_j	*Fuzzy set* associated to premise variable z_j
s	Discrete-time pole
T_s	Sampling time
z	Premise variable set

Chapter 1

Introduction

The concept of robotics has been shaped since the ancient world, with the first mechanical *automatas*, until nowadays, having robots able to optimally perform both simple and repetitive processes and those which require nearly human capabilities. The introduction of electricity allowed the creation of early robotic systems that where massively introduced during the Industrial Revolution, which over the years have been improved and implemented to perform on different environments and under multiple domains, e.g. spatial exploration and medical surgery.

(a) (b)

Figure 1.1: On the left (a), the automatic wine servant maid designed by Philon of Byzantium in 3 B.C., considered as one of the first operating robots of humanity. On the right (b), TIAGo robotic platform for assistive purposes developed by PAL Robotics.

Generally, the use of robots concerns tasks that are too Dangerous, too Dull, too Dirty and too Difficult (*The four D's*) to be done by humans [20], which agrees with the concept that robots are human-controlled mechanisms in a *master-slave* architecture. With the introduction of Artificial Intelligence (AI) techniques in this field, the concept has evolved to delegate humans into a supervisory role, still over the robot.

Recently, an increasing interest has been put into Human-Robot Collaboration (HRC) tasks, where both share the workspace and concur their actions into a common goal. These tasks usually imply a physical Human-Robot Interaction (pHRI), where safety and coexistence features must be guaranteed before collaboration might take place [11]. Thus, one of the main challenges in this field is to reliably detect and identify collisions, also foreseeing those that might occur too, and consequently generate actions that preserve task fulfillment and do not imply any type of risk or harm to the human.

In this work, an initial approach to this challenge has been done from the Fault Diagnosis and Detection (FDD) framework, by developing a Fault-Tolerant control strategy to detect and compensate forces that might be exerted on a robot during a task, which have been considered as faults due to: (1) their *a-priori* unknown effect and eventuality (2) and the assumption that presumably will hinder the fulfillment of tasks.

1.1 Motivation

Over the last few years service robots have been increasingly introduced in our daily lives. According to the International Federation of Robotics (IFR), the total number of professional service robots sold in 2017 rose by 85% with respect to 2016, estimating that in the period from 2019 to 2021 almost 736600 units will be sold [23].

Although service robots have been specifically designed to successfully perform tasks on highly dynamic and unpredictable anthropic domains, as any physical system and also due to their inherent complexity, they are prone to different types of faults. They could arise from the low-level actuators and sensors to the high-level decision layers, but also some of these failures regard interaction tasks e.g. misleading interpretations of the human actions or unexpected scenarios beyond nominal operation. All these factors can lead to a degradation on the performance of the robot or imply critical damage to it, that might even jeopardise its safety.

Thus, their ability to autonomously overcome most of these situations in a safe and efficient manner must play a fundamental role in their implementation, even more when it involves a human on its operation. A supervision and control scheme allows to maintain certain performance standards while assuring a detection, isolation and identification process of the faults that might appear, using the

gathered information on these events to properly generate the most suitable reaction, without neglecting the context of operation.

The presented work focuses on the head subsystem of the TIAGo robotic platform developed by PAL Robotics [1] as a proof of concept, to deal with non-linear dynamics associated to robotic systems. Initially the addressed problem was motivated by an existing issue with the existing position-control strategy (based on a PID), which is not able to reach desired positions when a mass is placed on top of it. Head positioning is crucial to assure the required vision and 3D mapping features of the TIAGo robot in anthropic environments, and so the necessity of overcoming this problem.

1.2 Thesis Objectives

This Master Thesis tackles supervision and control problem of those robots which have to work on complex and highly unpredictable anthropic domains, assuring safety and task fulfillment. Thus, a real scenario has been designed by using the head subsystem of the TIAGo robotic platform, being the main objective to implement an angular position control scheme on this subsystem, which compensates and estimates any external force being exerted on its structure in order to avoid a degradation on the performance of the task under an incomplete information model.

In order to reach the above-mentioned global objective, the following subgoals have been proposed:

- To obtain a linear-like representation of the 2 DOF head subsystem that captures its overall dynamic behaviour.
- To design a control scheme which:
  - tracks desired orientations in terms of angular trajectories according to an optimal performance criterion;
  - obtains a decoupled estimation of the unknown states from the possible exogenous effects.
- To include an isolation and estimation mechanism for external forces and an active fault tolerant technique to compensate its effects within the control scheme.
- To design a realistic simulation and design environment in order to validate the chosen techniques and assumptions on multiple scenarios.

[1] TIAGo by PAL Robotics: `http://tiago.pal-robotics.com/` (Accessed 22 June 2019)

- To characterize the behaviour and limitations of the physical robot actuators, adapting the control scheme according to them with the convenient mechanisms, to be validated on a testing bench platform.

1.3 Thesis Outline

This work has been organized as follows:

Chapter 2: State of the art

In this chapter, an overview of the existing Fault Detection an Diagnosis (FDD) techniques applied to robotic platforms is presented, along with a comparison between the main characteristics of each family of approaches. Furthermore, previous related works for robotic manipulators are also included.

Chapter 3: Takagi-Sugeno Modeling

The dynamic equations for the TIAGo head subsystem are the starting point for this section, where the Takagi-Sugeno modeling framework is presented and applied in order to obtain a linear-like representation of the system that embeds its behaviour.

Chapter 4: Parallel Distributed Control

This chapter focuses on the application and synthesis problem of the State-Feedback Parallel Distributed Control, presenting methods and techniques for the addressed system in this work.

Chapter 5: State and Fault estimation

This chapter firstly introduces the Robust Observer for Unknown Inputs to tackle an incomplete information scenario where not all the states are measurable, under the presence of exogenous forces acting on the TIAGo head subsystem. Also, the Reference Control technique is proposed to estimate and isolate the effects of these forces.

Chapter 6: Fault-tolerant Control Scheme

In this chapter all the methods presented up to here are summarized into the complete Fault-Tolerant Scheme, describing the integration between them.

Chapter 7: Simulation Results

In this chapter, criteria used in the design of the complete Fault-Tolerant scheme is analyzed, and performance results presented under different fault scenarios, comparing the behaviour of various control approaches.

Chapter 8: Parametrization of joint actuators

This chapter acknowledges the implications related with robot's real actuators and their effects, characterizing its behaviour and presenting the appropriate mechanisms to implement them within the complete Fault-Tolerant scheme, also presenting some results in a Test-Bench platform.

Chapter 9: Socioeconomic impact

In this chapter, a brief discussion about service robotics and the impact of this work in society and economy has been included.

Chapter 10: Project Budget

This section has been devoted to include some considerations about the implementation and development of this work from an economic point of view.

Chapter 11: Concluding remarks

Finally, the contributions of this work to the current state of the art are highlighted, along with some remarks on the method that might be further researched.

Chapter 2

State of the art

Fault Detection and Diagnosis (FDD) field has been widely studied for many years using system and control theory [19]. Only on certain generic robotic platforms some of these approaches have been successfully applied, like for wheeled mobile robots [35], being still considered as a relative new field of study for robotic systems.

Current FDD techniques for robots can be classified into three different categories according to their common key characteristics [20]:

1. **Data-driven:** rely on the extraction and processing of data from different parts of the system, in order to detect and determine a faulty behaviour. One example is the work presented in Reference [32], where a Neural Network (NN) is trained and integrated within a Sliding Mode Control (SMC) structure to enhance its robustness.

2. **Model-based:** depend on *a priori* analytical model that depicts the behaviour of a certain fault in the system or the nominal behaviour of the system itself. By comparison of the expected performance given by the model against the current one, faults can be detected and isolated. In Reference [15], a residual is computed using a dynamical model of a 7-DOF robotic arm to detect and obtain information about unexpected collisions to determine suitable reaction strategies.

3. **Knowledge-based:** mimic the behaviour of a human expert, directly associating certain evidences with their corresponding faults. Some of these methods are seen as hybrid techniques that combine Data-Driven and Model-Based approaches. On this line, in Reference [29], a two-layered structure is used, where a model aims at detection and a decision tree (Data-driven) at isolation of different actuator faults.

In the suggested classification, key advantages and weaknesses of the different approaches are worth to be pointed out. It should be mentioned that this classification does not draw clear boundaries between the different groups but determine certain general characteristics that are usually present, and should be considered in the practical application.

The main drawback of Model-based techniques is the use of a model itself. For some robotic platforms is extremely complex to determine analytic expressions that describe their behaviour or establish relationship between their components. This issue is even more significant when trying to describe interactions with the environment or the effect of a fault in the unaffected parts, for example. Here Data-Driven approaches present their main advantage, as no knowledge about the robot is assumed and relies on data of the particular robot where the method is applied to. But most Data-Driven methods require a high computational expense which might make them unfeasible for an on-board and on-line implementation that some robots could require (e.g. an spatial exploration robot). When a learning phase (supervised or unsupervised) is involved, part (or all) of it is carried out offline, as in Reference [18] where high dimensional data of a robotic arm is recorded to obtain dimensional reduction transformations and train a binary Support Vector Machine (SVM) model to be used online. It should be considered that online learning allows to obtain a dynamic method able to capture unexpected behaviours. Computational cost issue can also appear on some Model-based methods but is usually overcome by simplification or reformulation of the model, as in Reference [10], where a modification of the classical Newton-Euler is introduced in order to reduce its computation time for on-line execution purposes. Some Knowledge-based approaches that consist on combining Data and Model-Driven methods are able to establish a trade-off between the discussed issues, and suggest feasible solutions for FDD on robotic systems.

Regarding the characteristics that are required for service robots, autonomy is one of the most relevant ones. Service robots have to usually perform without the intervention of a human that supervises its actions or include a human in its operation loop that is not a robotic expert [26]. Thereby, being able to detect faults, identify them and overcome their effect on the execution of a task is crucial to achieve autonomy. As FDD methods should operate on a supervision level concurrently with all the techniques used for the desired performance, the computational demand on the method plays an important role on its implementation. High demanding processes might interfere with others and be the source of faults themselves.

Thus, according to the discussion above, a Model-based approach has been preferred for this work, as through existing methods analytical equations that define the dynamic behaviour of a n-link manipulator can be obtained, in this case, the 2-DOF head subsystem of the TIAGo robotic platform.

Furthermore, the effect of the exerted torques on the joints by the exogenous force (considered as a fault in this work) can be also defined through the analytical description of its dynamics.

In order to exploit current existing control techniques associated to linear systems, the Takagi-Sugeno modeling technique has been chosen to define a linear-like representation of the system that captures its non-linear behaviour. This method has been already applied to robotic manipulators [28], but within the FDD framework the Linear Parameter Varying (LPV) and SMC methodologies have been intensively applied for fault-tolerant architectures to deal with these non-linearities. Regarding LPV, in [25] the problem is stated for both additive and multiplicative faults using a simplified model, by means of a polytopic fault estimator that online adjusts a state-feedback control law. An application of this method is presented in [24] for the compensation of friction effects associated to mechanical joints. Aforementioned Reference [32] gives a comprehensive review of fault-tolerant control schemes based on SMC to deal with unexpected behaviours due to both uncertainties and faults, and their combination with a NN to estimate them.

Despite the methodology being applied, up to our knowledge, the considered scenarios in related works include a complete availability of the internal states of robotic manipulators, i.e. angular velocity and position. This assumption has not been considered in this work, being one the main objectives to include in the Fault-Tolerant scheme an appropriate estimation mechanism for unknown states under a fault scenario.

Chapter 3

Takagi-Sugeno Modeling

The method presented in this work considers the implementation of a Fault-Tolerant control scheme by means of a Model-based approach. Thereby, an analytic model which describes the behaviour of the system has to be determined which, as aforementioned, is the 2-DOF Head subsystem of the TIAGo robotic platform, presented in Figure 3.1. This type of systems are usually named Pan-Tilt structures, where the Pan movement corresponds to the rotation around a vertical axis and the Tilt one to the rotation around its perpendicular.

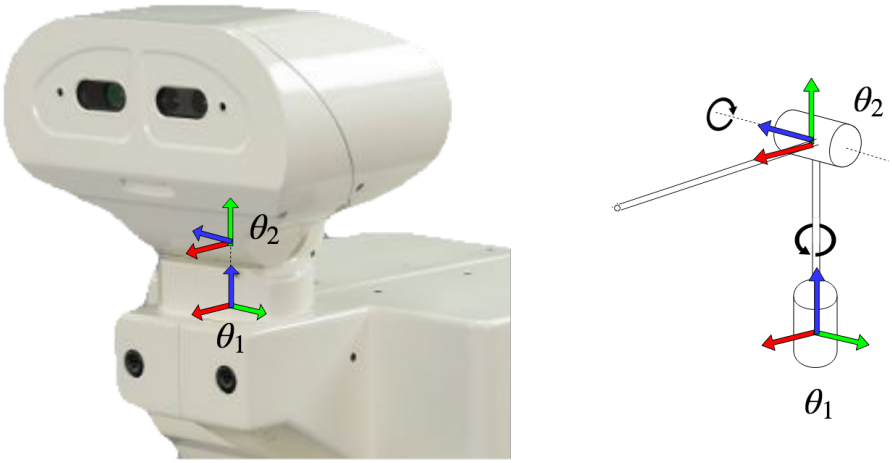


Figure 3.1: TIAGo's head subsystem representation as a two-manipulator link.

3.1 Analytical Model

A two-link representation with two rotational joints (2R manipulator) has been chosen for the sake of simplicity, in order to obtain the associated dynamic behaviour. Firstly, the forward kinematics of the system have been obtained, by characterizing the system through the modified Denavit-Hartenberg (DH) [7] parameters, which describe each of the links and the connection to the neighbouring ones. It should be pointed out that the modified convention of the DH parameters has been chosen in order to provide a straightforward description of the system by considering the i th-joint attached to the i th-frame.

Reference frames have been defined for each of the links according to the already defined ones by PAL Robotics during the development of the robotic platform, which are included in the available simulation model for ROS. These have been also depicted in Figure 3.1, where the rotation axis of the i -th joint corresponds to $\hat{Z}_i$.

Regarding the modified DH convention and the chosen reference frame attachment Table 3.1 includes the parameters, defined as follows:

- $\mathbf{d_i}$ is the distance from $\hat{X}_{i-1}$ to $\hat{Z}_i$ measured along $\hat{Z}_i$
- $\theta_\mathbf{i}$ is the angle from $\hat{X}_{i-1}$ to $\hat{Z}_i$ measured about $\hat{Z}_i$
- $\mathbf{a_{i-1}}$ is the distance from $\hat{Z}_{i-1}$ to $\hat{Z}_i$ measured along $\hat{X}_{i-1}$
- $\alpha_\mathbf{i-1}$ is the angle from $\hat{Z}_{i-1}$ to $\hat{Z}_i$ measured about $\hat{X}_{i-1}$

Link	$\mathbf{d_i}$	$\theta_\mathbf{i}$	$\mathbf{a_{i-1}}$	$\alpha_\mathbf{i-1}$
1	0	θ_1	0	0
2	L	$\theta_2 + \frac{\pi}{2}$	0	π

Table 3.1: Link parameters under Modified DH convention.

In the modified DH convention, the general form of the transformation ${}^{i}_{i-1}T$ matrix (rotation ${}^{i}_{i-1}R$ and translation ${}^{i}_{i-1}t$) between consecutive links is:

$$ {}^{i}_{i-1}T = \begin{bmatrix} {}^{i}_{i-1}R & {}^{i}_{i-1}t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i\, c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i\, s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

obtaining the corresponding ones to the system

$$ {}^{1}_{0}T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad {}^{2}_{1}T = \begin{bmatrix} -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ c_2 & -s_2 & 0 & L \\ 0 & 0 & 0 & 1 \end{bmatrix}, $$

where c_i and s_i are abbrev. for $\cos(\theta_i)$ and $\sin(\theta_i)$, respectively.

Using this representation, the iterative Newton-Euler algorithm [8] has been applied to obtain the dynamic equations, being executed in two steps (considering a link of n joints): an *outward* computation from link 0 to $n-1$ of velocities and accelerations of each link to determine the force and torque acting at each link, and an *outward* force and moment balance from link n to link 1 to determine the acting torque and force in each joint. For a manipulator where all joints are rotational (e.g. the considered head subsystem), Equations (3.1a) to (3.1f) and (3.1h) to (3.1j) represent the formulation of the *outward* and *inward* steps, respectively.

$$ {}^{i+1}\omega_{i+1} =_{i}^{i+1} R^{i}\omega_i + \dot{\theta}_{i+1}^{i+1}\hat{Z}_{i+1} \tag{3.1a} $$

$$ {}^{i+1}\dot{\omega}_{i+1} =_{i}^{i+1} R^{i}\dot{\omega}_i +_{i}^{i+1} R^{i}\omega_i \times \dot{\theta}_{i+1}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1}^{i+1}\hat{Z}_{i+1} \tag{3.1b} $$

$$ {}^{i+1}\dot{v}_{i+1} =_{i}^{i+1} R(\dot{\omega}_i \times^{i} P_{i+1} +^{i} \omega_i \times ({}^{i}\omega_i \times^{i} P_{i+1}) +^{i} \dot{v}_i) \tag{3.1c} $$

$$ {}^{i+1}\dot{v}_{c_{i+1}} =^{i+1} \dot{\omega}_{i+1} \times^{i+1} P_{C_i} +^{i+1} \omega_{i+1} \times ({}^{i+1}\omega_{i+1} \times^{i+1} P_{C_i}) +^{i+1} \dot{v}_{i+1} \tag{3.1d} $$

$$ {}^{i+1}F_{i+1} = m_{i+1}^{i+1}\dot{v}_{c_{i+1}} \tag{3.1e} $$

$$ {}^{i+1}N_{i+1} =_{i+1}^{C} I_{i+1}^{i+1}\dot{\omega}_{i+1} +^{i+1} \omega_{i+1} \times_{i+1}^{C} I_{i+1} \times^{i+1} \omega_{i+1} \tag{3.1f} $$

$$ \tag{3.1g} $$

$$ {}^{i}f_i =_{i+1}^{i} R^{i+1}f_{i+1} +^{i} F_i \tag{3.1h} $$

$$ {}^{i}n_i =^{i} N_i +_{i+1}^{i} R^{i+1}n_{i+1} +^{i} P_{C_i} \times^{i} F_i +^{i} P_{i+1} \times_{i+1}^{i} R_{i+1}^{i}{}^{i+1}f_{i+1} \tag{3.1i} $$

$$ \tau_i =^{i} n_i^{Ti}\hat{Z}_i \tag{3.1j} $$

Some assumptions and considerations have been made in this modeling process:

- Each link is considered as a rigid body, whose mass distribution is characterized by the position of its Center Of Gravity (COG), being its inertia tensor and mass are referred to it.

Parameter	Description	Value
I_1	Inertial tensor of first link	$\text{diag}(0.001192,\ 0.001402,\ 0.000889)\ [kg \cdot m^2]$
I_2	Inertial tensor of second link	$\text{diag}(0.004620,\ 0.004861,\ 0.003132)\ [kg \cdot m^2]$
m_1	Mass of first link	$0.622\ [kg]$
m_2	Mass of second link	$0.661\ [kg]$
L	Dist. of COG_2 from joint axis ${}^2\hat{Z}_2$	$0.098\ [m]$

Table 3.2: Values of inertial and distance parameters of the TIAGo head subsystem.

- Gravity effects are introduced into the computation by defining ${}^0\dot{v}_0 \triangleq g\,\hat{Z}_0$, being g the gravity constant with a value of $-9.8[m/s^2]$.

- Base frame (link 0) is placed at the top part of TIAGo's torso, which features a vertical movement. As it performs at low speeds, effects on the dynamical behaviour of the head subsystem have been neglected, considering the frame as fixed i.e. $\omega_0 = 0$ and $\dot{\omega}_0 = 0$.

- The inertial parameters and relative distances that describe the system have been considered as defined by PAL Robotics in the ROS simulation environment, and listed in Table 3.2. An additional assumption has been made on the inertia tensors for both links, considering null off-diagonal terms.

- External forces (and torques) acting on the robot head have not been considered into the dynamic characterization of the system, and thus $f_3 = 0$, $n_3 = 0$.

Obtained expressions from Newton-Euler algorithm provide joint torque vector τ as a function of the joint acceleration $\ddot{\theta}$, velocity $\dot{\theta}$ and position θ vectors, that might be arranged into the configuration-space form

$$\tau = \mathcal{M}(\theta)\ddot{\theta} + \mathcal{B}(\theta)[\dot{\theta}\dot{\theta}] + \mathcal{C}(\theta)[\dot{\theta}^2] + \mathcal{G}(\theta)$$

where $\mathcal{M}(\theta) \in \mathbb{R}^{n\times n}$ describes the mass matrix of the manipulator, $\mathcal{B}(\theta) \in \mathbb{R}^{n\times n(n-1)/2}$ the Coriolis terms, $\mathcal{C}(\theta) \in \mathbb{R}^{n\times n}$ the centrifugal coefficients and $\mathcal{G}(\theta) \in \mathbb{R}^n$ the gravity effects; being n the number

of joints. For this system,

$$\mathcal{M}(\theta) = \begin{bmatrix} I_{zz_1} + m_2 d_4^2 + I_{xx_2} c_2^2 + I_{yy_2} s_2^2 - m_2 d_4^2 c(2\theta_2)^2 & 0 \\ 0 & I_{zz_2} + m_2 d_4^2 \end{bmatrix},$$
$$\mathcal{B}(\theta) = \begin{bmatrix} -2c_2 s_2 (I_{xx_2} - I_{yy_2}) + m_2 d_4^2 s(4\theta_2) \\ 0 \end{bmatrix},$$
$$\mathcal{C}(\theta) = \begin{bmatrix} 0 & 0 c_2 s_2 (I_{xx_2} - I_{yy_2}) - \frac{1}{2} m_2 d_4^2 s(4\theta_2) & 0 \end{bmatrix}$$
$$\mathcal{G}(\theta) = \begin{bmatrix} 0 \\ g m_2 d_4 s(2\theta_2) \end{bmatrix}$$

3.2 Takagi-Sugeno formulation

From the configuration space formulation, the expressions for joint accelerations and velocities can be defined as follows

$$\ddot{\theta}_1 = \frac{\mathcal{B}_{1,1}(\theta_2)}{\mathcal{M}_{1,1}(\theta_2)} \dot{\theta}_1 \dot{\theta}_2 + \frac{1}{\mathcal{M}_{1,1}(\theta_2)} \tau_1, \tag{3.2a}$$
$$\ddot{\theta}_2 = \frac{\mathcal{C}_{2,1}(\theta_2)}{\mathcal{M}_{2,2}} \dot{\theta}_1^2 + \frac{1}{\mathcal{M}_{2,2}} \tau_2 + \frac{\mathcal{G}_{2,1}(\theta_2)}{\mathcal{M}_{2,2}}, \tag{3.2b}$$
$$\dot{\theta}_1 = \frac{d}{dt} \theta_1, \tag{3.2c}$$
$$\dot{\theta}_2 = \frac{d}{dt} \theta_2, \tag{3.2d}$$

where

$$\mathcal{M}_{1,1}(\theta_2) = I_{zz_1} + m_2 L^2 + I_{xx_2} c_2^2 + I_{yy_2} s_2^2 - m_2 L^2 c(2\theta_2)^2,$$
$$\mathcal{M}_{2,2} = I_{zz_2} + m_2 L^2,$$
$$\mathcal{B}_{1,1}(\theta_2) = -2c_2 s_2 (I_{xx_2} - I_{yy_2}) + m_2 L^2 s(4\theta_2),$$
$$\mathcal{C}_{2,1}(\theta_2) = c_2 s_2 (I_{xx_2} - I_{yy_2}) - \frac{1}{2} m_2 L^2 s(4\theta_2),$$
$$\mathcal{G}_{2,1}(\theta_2) = g m_2 L s(2\theta_2),$$

recalling that parameters have been already listed and described in Table 3.2.

Further arrangements can be made on the equations to reduce the number of different terms in the

expressions, and so from now on

$$\mathcal{B}_{1,1}(\theta_2) = -2\,\mathcal{C}_{2,1}(\theta_2)$$

Equations from (3.2) are suitable for the following state-space alike representation,

$$\dot{x}(t) = A^c(\theta_2, \dot{\theta}_1)x(t) + B^c(\theta_2)u(t) + g_v^c(\theta_2), \tag{3.3}$$

considering the input torque action vector $u = [\tau_1, \tau_2]^T$, and the state-space vector $x = [\dot{\theta}_1, \dot{\theta}_2, \theta_1, \theta_2]^T$, such that $\dot{x} = [\ddot{\theta}_1, \ddot{\theta}_2, \dot{\theta}_1, \dot{\theta}_2]^T$, being

$$A^c(\theta_2, \dot{\theta}_1) = \begin{bmatrix} 0 & \frac{-2\,\mathcal{C}_{2,1}(\theta_2)}{\mathcal{M}_{1,1}(\theta_2)}\dot{\theta}_1 & 0 & 0 \\ \frac{\mathcal{C}_{2,1}(\theta_2)}{\mathcal{M}_{2,2}}\dot{\theta}_1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$B^c(\theta_2) = \begin{bmatrix} 1/\mathcal{M}_{1,1}(\theta_2) & 0 \\ 0 & 1/\mathcal{M}_{2,2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad g_v^c(\theta_2) = \begin{bmatrix} \frac{\mathcal{G}_{2,1}(\theta_2)}{\mathcal{M}_{2,2}} \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Note that from now on the subscript c will be used to denote that a matrix is defined for the continuous time domain, in case of its absence it corresponds to a discrete time domain definition.

Variable term $g_v^c(\theta_2)$, which corresponds to the gravity effects of the mass in second link, represents a non-linear term within the state-space configuration, being neglected for the following modeling process. Its effect and within the complete control scheme will be addressed again on Chapter 6.

3.2.1 Takagi-Sugeno Model

Fuzzy logic, introduced by L.A. Zadeh [34], in contrast with classical Boolean logic does not define the output of a decision as binary (*Yes*/*No*, $1/0$), but gives a degree of belonging of the output to the extreme values of the decision, according to a set of rules. This definition has been widely used in the AI field as an approximation of the human decision-making process, where in most cases admits a range of possibilities between certain limits.

Takagi-Sugeno Models [31] (henceforth TS Models), named after their designers, apply the concept of fuzzy logic to the description of non-linear system dynamics by a set of rules, used to obtain the overall model by *blending* them together. For state-space representations, Model Rule i is associated

with an i-th Linear Time Invariant (LTI) form as follows:

IF $z_1(t)$ is $M_{1,l}$ and $z_2(t)$ is $M_{2,l}$ and ... $z_p(t)$ is $M_{p,l}$

$$\textbf{THEN} \quad \begin{cases} \dot{x}(t) = A_i^c\, x(t) + B_i^c\, u(t) \\ y(t) = C_i^c\, x(t) + D_i^c\, u(t), \end{cases} \quad \forall i = 1, 2, ..., r$$

Thus, for a certain number of rules r at each time instant k, the belonging of a set of p premise variables $z = \{z_1,\ z_2,\ ...,\ z_j\}$ to the *fuzzy sets* $M_j = \{M_{j,1},\ M_{j,2},\ ...,\ M_{j,l}\}$ defines the system as the i-th linear representation described by $[A_i\ B_i\ C_i\ D_i]$. It should be pointed out that "z_j is $M_{j,l}$" is understood as $M_{j,l}(z_j) = 1$.

In our particular case, TS Model technique is used to give a linear-like description of the considered TIAGo head subsystem, focusing on non-linear terms. According to the Sector Non-linearity approach, an exact fuzzy model construction is assured if sectors where the non-linear behaviour lies are found in the system state-space, and recalling that the model corresponds to a physical system, its variables or implied effects are bounded, and so non-linear terms. Therefore, sectors can be easily determined for local regions around their known bounds.

According to the Sector approach, z_j have been defined to embed the nonlinear terms of the state-space representation in Equation (3.3), such that the following conditions hold:

1. Each $z_j(x_1, x_2, ...)$ has to be bounded in $[\underline{z}_j, \overline{z}_j]$ for the (bounded) set of variables $\{x_1, x_2, ..\}$ and no indeterminate mathematical forms arise within these bounds.

2. Controllability property is assured for the system.

3. The minimal set that fulfill previous conditions is chosen.

Thus, in this case:

$$z_1 \triangleq \mathcal{M}_{1,1}(\theta_2), \qquad z_2 \triangleq \mathcal{C}_{2,1}(\theta_2)\dot{\theta}_1. \tag{3.4}$$

Bounds for z_j can be defined according to the state variables limits, given by the operation conditions of the TIAGo robot, listed in Table 3.3. Non-linear terms imply that an optimization process has to be carried out in order to define these bounds, included in Table 3.4. It should be pointed out that for z_2, variable $\dot{\theta}_1$ affects in terms of amplitude and so it has been set to $\overline{\dot{\theta}}_1$ for computing the limits.

Aforementioned properties are therefore held: (1) z_j are bounded for the bounds on the set of state variables and no indeterminate forms appears within these bounds, (2) controllability property will only depend on the selection of an output model, which will be addressed in Chapter 5, as matrix A^c does

State	Minimum	Maximum
$\dot{\theta}_i\ [rad/s]$	-3	3
$\theta_1\ [^\circ]$	75	75
$\theta_2 [^\circ]$	60	45

Table 3.3: Operational limits for TIAGo head subsystem.

State	Minimum	Maximum
z_1	0.0055	0.0091
z_2	-0.0110	0.0110

Table 3.4: Premise variables limits

not loose rank for any value $z_j \in [\underline{z_j}, \overline{z}_j]$, (3) and the minimal set is chosen as the existing non-linear terms can not be embedded in a lower order set.

In this approach, $M_{i,j}$ are used to define the degree of membership of each z_j to its bounds, and so, giving its linear description:

$$z_j = M_{j,1}(z_j)\bar{z_j} + M_{j,2}(z_j)\underline{z_j} \tag{3.5}$$

where

$$M_{j,1}(z_j) = \frac{\bar{z_j} - z_j}{\bar{z_j} - \underline{z_j}}, \quad M_{j,2}(z_j) = \frac{z_j - \underline{z_j}}{\bar{z_j} - \underline{z_j}}.$$

Figure 3.2 includes the graphical representation of Equation 3.5.

Finally, the number of fuzzy rules r for the TS Model is equal to 2^p, recalling that p is the number of chosen premise variables, as they consider all the possible permutations on the limits of z_j. For the addressed TIAGo head subsystem, four rules have been obtained and, according to the given description for them, their associated linear systems correspond to the *frozen* original system form evaluated at the limit points of operation (embedded in z_i):

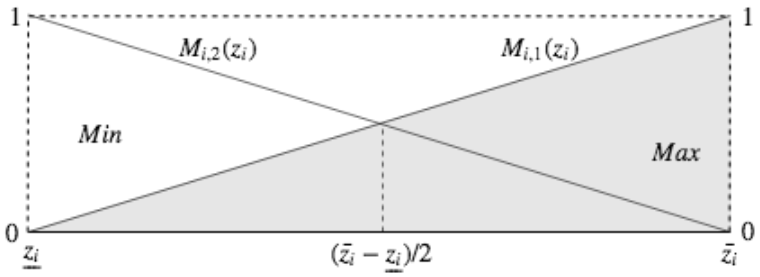


Figure 3.2: Graphical representation of the linear membership function.

Model rule 1

IF $z_1(t)$ is $M_{1,1}$ and $z_2(t)$ is $M_{2,1}$ **THEN** $\dot{x}(t) = A_1^c\, x(t) + B_1^c\, u(t)$

$$\text{where} \quad A_1^c = \begin{bmatrix} 0 & -2\overline{z_2}/\overline{z_1} & 0 & 0 \\ \overline{z_2}/\mathcal{M}_{2,2} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad B_1^c = \begin{bmatrix} 1/\overline{z_1} & 0 \\ 0 & 1/\mathcal{M}_{2,2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \tag{3.6}$$

Model rule 2

IF $z_1(t)$ is $M_{1,2}$ and $z_2(t)$ is $M_{2,1}$ **THEN** $\dot{x}(t) = A_2^c\, x(t) + B_2^c\, u(t)$

$$\text{where} \quad A_2^c = \begin{bmatrix} 0 & -2\overline{z_2}/\underline{z_1} & 0 & 0 \\ \overline{z_2}/\mathcal{M}_{2,2} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad B_2^c = \begin{bmatrix} 1/\underline{z_1} & 0 \\ 0 & 1/\mathcal{M}_{2,2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \tag{3.7}$$

Model rule 3

IF $z_1(t)$ is $M_{1,1}$ and $z_2(t)$ is $M_{2,2}$ **THEN** $\dot{x}(t) = A_3^c\, x(t) + B_3^c\, u(t)$

$$\text{where} \quad A_3^c = \begin{bmatrix} 0 & -2\underline{z_2}/\overline{z_1} & 0 & 0 \\ \underline{z_2}/\mathcal{M}_{2,2} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad B_3^c = \begin{bmatrix} 1/\overline{z_1} & 0 \\ 0 & 1/\mathcal{M}_{2,2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \tag{3.8}$$

Model rule 4

IF $z_1(t)$ is $M_{1,2}$ and $z_2(t)$ is $M_{2,2}$ **THEN** $\dot{x}(t) = A_4^c\, x(t) + B_4^c\, u(t)$

$$\text{where} \quad A_4^c = \begin{bmatrix} 0 & -2\underline{z_2}/\underline{z_1} & 0 & 0 \\ \underline{z_2}/\mathcal{M}_{2,2} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad B_4^c = \begin{bmatrix} 1/\underline{z_1} & 0 \\ 0 & 1/\mathcal{M}_{2,2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.9}$$

Note that $B_1^c = B_3^c$ and $B_2^c = B_4^c$, which will be used in order to simplify conditions based upon

this TS Model along this work. It should be also pointed out that the output model of the system has not been included in the TS rules as it remains invariant with respect to the point or operation, i.e. availability of measurements does not depend on the internal state of the system. Its definition an implications will be further discussed in Chapter 5.

3.2.2 *Defuzzified* representation

All the possible points of operation of the TIAGo head subsystem are confined within the space defined by the obtained TS Model rules, which address only the limit ones. To finally obtain the system response for any point of operation, the *deffuzification* step has to be carried out by *blending* these rules.

A coefficient w_i is associated to the i-th rule for the set z through the joint degree of belong to all the premise variables z_j to the M_j fuzzy sets:

$$w_i(z(k)) = \prod_{j=1}^{p} Mj(z_j(k)) \tag{3.10}$$

such that

$$w_i \geq 0, \qquad \sum_{i=1}^{r} w_i(z(k)) > 0. \tag{3.11}$$

The level of activation h_i, know in the literature as the *firing probability*, of the i-th rule model is defined as:

$$h_i(z(k)) = \frac{w_i(z(k))}{\sum_{i=1}^{r} w_i(z(k))}$$

which implies according to Equations (3.10) and (3.11) that

$$h_i \geq 0, \qquad \sum_{i=1}^{r} h_i(z(k)) = 1.$$

Within the *blending* process, these coefficients h_i determine the output of the TS system for a pair $(x(t),\ u(t))$ as a sum of the weighted subsystems from the rules:

$$\dot{x}(t) = \sum_{i=1}^{r} h_i(z(t))[A_i^c \cdot x(t) + B_i^c u(t)]$$

where for the TIAGo head subsystem $r = 4$:

$$h_1(z_p(t)) = M_{1,1}(z_1) \cdot M_{2,1}(z_2) \qquad h_2(z_p(t)) = M_{1,2}(z_1) \cdot M_{2,1}(z_2)$$

$$h_3(z_p(t)) = M_{1,1}(z_1) \cdot M_{2,2}(z_2) \qquad h_4(z_p(t)) = M_{1,2}(z_1) \cdot M_{2,2}(z_2)$$

To summarize, the *defuzzyfied* representation of a TS model gives a formulation of the system at any operation point according to the *distance* to each of the limit systems within the premise variable space. Therefore, only a limited number of LTI systems is used to confine all the possible ones, meaning an advantage in order to characterize non-linearities and define model-based control techniques.

One of the main objectives in this work has been to consider the implementation phase of all the developed techniques on the real robot, acknowledging in an early phase all the implied issues. Thus, both the system's state space and its TS model have been formulated in the discrete-time domain, which is assumed in the following chapters. Both are equivalent to their continuous-time forms, only with the difference that state-space matrices have been evaluated according to a Zero Order Hold discretization method for a sampling time T_s. Used notation on this domain determines that k stands for a time instant and $k + 1$ for its consecutive one, being T_s the time difference between them.

Chapter 4

Parallel Distributed Control

TS formulation of the TIAGo head subsystem gives a complete description of non-linearities according to linear descriptions of the system on its operational limits. Thereby, its response can be described at a certain instant as a linear combination of them according to their degree of contribution into the overall model. Following this concept, a control strategy for a system can be defined as a set of linear control laws at these limits, being the overall control at a certain point defined as a combination of these limit controllers. This concept was initially presented by Kang and Sugeno in Reference [30] under the name of Parallel Distributed Compensation (PDC from now on).

The PDC offers a procedure to design a control strategy from a given TS model using linear techniques. For each one of the fuzzy rules defined for the model, a control rule can be stated sharing the same premise variables and their corresponding *fuzzy sets* (membership functions). In this work, a state-feedback control has been used as the linear control strategy. From the already presented TS model for the considered system in this work, control rules have been stated as follows:

Control rule 1

IF $z_1(t)$ is $M_{1,1}$ and $z_2(t)$ is $M_{2,1}$ **THEN** $u(k) = -K_1\, x(k)$

Control rule 2

IF $z_1(t)$ is $M_{1,2}$ and $z_2(t)$ is $M_{2,1}$ **THEN** $u(k) = -K_2\, x(k)$

Control rule 3

IF $z_1(t)$ is $M_{1,1}$ and $z_2(t)$ is $M_{2,2}$ **THEN** $u(k) = -K_3\, x(k)$

Control rule 4

IF $z_1(t)$ is $M_{1,2}$ and $z_2(t)$ is $M_{2,2}$ **THEN** $u(k) = -K_4\, x(k)$

As for the TS models, the *defuzzyfied* formulation is applied on the control action vector $u(t)$, using the same procedure and h_i coefficients from the TS model:

$$u(k) = -\sum_{i=1}^{r=4} h_i(z(k))[K_i x(k)] = -K(z(k))x(k) \tag{4.1}$$

This formulation gives a dynamic controller in a gain-scheduling fashion, as a function of the premise variables set z that enclose the operation point of the system.

In this chapter, the synthesis problem for a PDC is presented, giving the conditions to assure stability and other performance criterion that have been considered in this work.

4.1 Lyapunov's stability and Apkarian Filter

The key issue of PDC is to design the feedback control gains K_n assuring stability and certain performance specifications. Although this strategy only implies the definition of the system in its limit operation points, global design conditions have to be considered, i.e. all the operational points and the transitions between them have to be included within the control synthesis problem.

Lyapunov's theory defines global asymptotic stability for an equilibrium point of homogeneous (discrete) linear systems $x(k+1) = Ax(k)$ if there exists a scalar function $V(x)$ such that it simultaneously fulfills a set of conditions. In the control literature [13], the quadratic function $V(x) = xPx^T$ for $P \in \mathbb{R}^{n\times n} \,|\, P = P^T > 0$ is used to arrange those conditions into the following Linear Matrix Inequalities (LMI) to be satisfied:

$$P > 0$$
$$A^T P A + P < 0$$

Note that inequality conditions in the LMI framework regard positive/negative definiteness of the matrices, which is indeed related with their eigenvalues.

On the addressed synthesis problem, these LMIs are reformulated to acknowledge all the possible combinations between limit LTI systems in order to assure stability for any point enclosed by them. Therefore, considering the closed-loop form of the system in each limit point, and according to a state-feedback control:

$$P > 0 \tag{4.2a}$$

$$(A_i - B_i K_i)^T P (A_i - B_i K_i) - P < 0, \qquad \forall\, i = 1, 2, ...r \tag{4.2b}$$

$$G_{ij}^T P G_{ij} - P < 0, \qquad \forall\, i < j \leq r \quad \text{s.t.} \quad h_i \cap h_j \neq \phi \tag{4.2c}$$

where

$$G_{ij} = \frac{(A_i - B_i K_j) + (A_j - B_j K_i)}{2}.$$

These conditions no longer correspond to LMIs, but to Bilinear Matrix Inequalities (BMI), having not jointly convex terms, which implies a Non-deterministic Polynomial time (NP)-hard optimization problem [33] that hinders its feasibility.

Although these conditions could be reformulated in terms of LMIs using variable transformation, the number of conditions will considerably increase up to 9 for the addressed TS Model (recalling that $B_1 \equiv B_2$ and $B_3 \equiv B_4$). Some solutions have been given in the literature to reduce the conservatism of these conditions (into smaller sets and/or with lower range matrices), in this Thesis the application of a filter to $u(t)$ in order to reformulate the model into a form with no input matrix B_i variant on z has been considered. Namely Apkarian filter [3] after its designer, it has the following discrete time form, according to the Euler discretization approach:

$$x_f(k) = (I - T_s \cdot\, A_f)\, x_f(k-1) + T_s \cdot\, B_f\, u_f(k-1) \tag{4.3}$$

where

$$A_f = \begin{bmatrix} -\psi & 0 \\ 0 & -\psi \end{bmatrix}, \qquad x_f(k) = \begin{bmatrix} \tau_1(k) \\ \tau_2(k) \end{bmatrix}, \qquad B_f = \begin{bmatrix} \psi & 0 \\ 0 & \psi \end{bmatrix}, \qquad u_f(k) = \begin{bmatrix} u_{\tau_1}(k) \\ u_{\tau_2}(k) \end{bmatrix}$$

and ψ represents the filter gain, determined such that is fast enough to not present dynamic coupling effects with the controller, which will be further discussed in Chapter 7. Pre-appliance of this filter has been chosen in this case, increasing the system matrices of the system associated to the i-th rule as

follows:

$$\begin{bmatrix} x(k+1) \\ \tilde{x}_f(k) \end{bmatrix} = \begin{bmatrix} A_i & B_i \\ 0 & A_f \end{bmatrix} \begin{bmatrix} x(k) \\ \tilde{x}_f(k-1) \end{bmatrix} + \begin{bmatrix} 0 \\ B_f \end{bmatrix} u_f(k) \tag{4.4}$$

$$y(k) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ \tilde{x}_f(k-1) \end{bmatrix} \tag{4.5}$$

from which the following notation arises:

$$\tilde{A}_i = \begin{bmatrix} A_i & B_i \\ 0 & A_f \end{bmatrix}, \qquad \tilde{B}_i = \begin{bmatrix} 0 \\ B_f \end{bmatrix}, \qquad \tilde{C}_i = \begin{bmatrix} C & 0 \end{bmatrix}, \qquad \tilde{x}(k) = \begin{bmatrix} x(k) \\ \tilde{x}_f(k-1) \end{bmatrix}.$$

As it can be seen, B_i term is allocated on the $\tilde{A}_i$, remaining $\tilde{B}_i$ constant ($\tilde{B}_i \equiv \tilde{B}$) according to ψ for all TS Model rules. TS Model rules for this model are equivalent to the previous ones, as well as z and $M_{i,j}$, with the only difference on the form of A_i and B_i, being the latter one constant in all the four cases.

Under this assumption, Lyapunov's stability LMI conditions from Equations (4.2) can assure global asymptotic stability only regarding A_i variance:

$$P > 0$$
$$(\tilde{A}_i - \tilde{B}K_i)^T P(\tilde{A}_i - \tilde{B}K_i) - P < 0, \qquad \forall\, i = 1, 2, ...r$$

decreasing the number of conditions for the TS Model to five. This reduction in the number of conditions also affects the formulation of additional ones to assure certain properties for the closed-loop system, which will be presented in the following subsections.

4.2 Optimal Control Design

Robotic applications quite often require optimal performance in order to minimize costs in terms of resources and time consumption, as well as present some characteristics on their output(s) related with their application. On this work through the formulation of an optimal controller design has been oriented towards assuring that performance maintains the required operational characteristics.

Therefore, the synthesis problem has been stated as a Linear Quadratic Control (LQC from now on) in the desired discrete time domain. The state-feedback gains K_i of the control law from Equation (4.1) are found such that the following quadratic performance criterion is minimized, assuming zero

initial state:

$$J = \sum_{0}^{\infty} \left[x(k)^T Q x(k) + u(k)^T R u(k) \right], \tag{4.6}$$

which considering Lyapunov's stability and LMI framework is subject to the following constraints for the state-space system from Equation (4.4):

$$\begin{bmatrix} \gamma I & 0 \\ 0 & Y \end{bmatrix} > 0$$

$$\begin{bmatrix} -Y & Y\tilde{A}_i^T - W_i\tilde{B}_i^T & Y(Q^{1/2})^T & W_i^T \\ \tilde{A}_i Y - \tilde{B}_i W_i & -Y & 0 & 0 \\ Q^{1/2}Y & 0 & -I & 0 \\ W_i & 0 & 0 & -R^-1 \end{bmatrix} < 0$$

where $Y \equiv P^{-1}$ and $W_i = K_i Y$, stating the problem to minimise γ, which represents the upper bound for the LQC criterion from (4.6).

The optimality goal for the system response is achieved by accordingly define the form of $J(k)$ through matrices Q and R. The first one allows to control convergence speed of the states towards their references, while the latter one is selected to limit the control effort required. Multiple tuning methods can be used for this purpose, being chosen for this work the so-called Bryson's rule [17]. Originally the reciprocal of the maximum admissible magnitude $\overline{e}$ for each variable x_i or input u_i was used for tuning. But in this work, in order to conveniently bias the solution towards certain objectives, a user-defined weight $\rho_i \in [0, 1]$ has been also included for each term:

$$Q = \begin{bmatrix} \rho_1/\overline{e}_1 & 0 & 0 & 0 \\ 0 & \rho_2/\overline{e}_2 & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \rho_n/\overline{e}_n \end{bmatrix}, \tag{4.7}$$

$$R = \begin{bmatrix} \rho_{n+1}/\overline{e}_{n+1} & 0 & 0 & 0 \\ 0 & \rho_{n+2}/\overline{e}_{n+2} & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \rho_{n+m}/\overline{e}_{n+m} \end{bmatrix}. \tag{4.8}$$

Through this method, the operative characteristics of the TIAGo robot can be embedded within

the controller design. Diagonal form of Q and R is chosen in order to simplify the tuning process, neglecting cross effects between variables. Further details on the tuning method will be given in Chapter 7 according to the desired objectives.

4.3 $\mathbb{D}$-stabilization

Besides of the LQC synthesis conditions stated in the previous section, the system's closed-loop response can be regulated by the well-known pole-placement technique in the Automatic Control field, which consists in constraining the system eigenvalues to a region in the (discrete-time) complex plane, which has an associated behaviour of the system that defines its transient response. In the context of this work, $\mathbb{D}$ is a convex and symmetric LMI region on the complex plane $\mathbb{C}$ such that

$$\mathbb{D} \triangleq \{s \,|\, s \in \mathbb{C},\ F_{\mathbb{D}}(s) < 0\},$$

being $F_{\mathbb{D}}(s)$ the so-called characteristic function of the $\mathbb{D}$ formulated as

$$F_{\mathbb{D}}(s) = L + sM + \overline{s}M^T,$$

where $\overline{s}$ denotes the conjugate of pole s, and matrices L and M define the characteristics of the LMI region.

Therefore, closed-loop system is said to be $\mathbb{D}$-stable if $\lambda_i(\tilde{A}_i - \tilde{B}K_i) \in \mathbb{D}$. Including Lyapunov's stability, following LMI conditions assure both:

$$P > 0$$
$$\mathcal{R}_{\mathbb{D},i} \triangleq L \otimes P + M \otimes (\tilde{A}_i - \tilde{B}K_i) + M^T \otimes (\tilde{A}_i - \tilde{B}K_i)^T, \qquad \forall\, i = 1, 2, ...r$$

where $\otimes$ represents the Kronecker product.

For this work, $\mathbb{D}$-stabilization has been considered to define a *strip* region characterized by α and β such that poles s:

$$\beta < s < \alpha \tag{4.9}$$

This particular kind of region has been chosen attending to the design of the complete Fault-Tolerant scheme, such that there do not exist dynamic couplings between the PDC control and the observer, by controlling time constant τ_s for s to assure that observer dynamics are n times faster than PDC ones. In Chapter 7, α and β will be defined and the implications of this issue discussed.

Regarding the definition of $F_{\mathbb{D}}(s)$, for this region

$$L = -2 \cdot \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}, \qquad M = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

In this work, both $\mathbb{D}$-stabilization and LQC conditions have been simultaneously used to formulate the controller synthesis problem for the TIAGo head susbsystem, such that they complement each other: a *strip* LMI region in the complex plane is defined to constraint the LQC conditions, in order to obtain a solution that lies on it, according to the chosen optimal criterion.

4.4 Integral Control by State Augmentation

Presented PDC synthesis methods have been described according to the TS Model, directly defined from the analytical equations that characterize the dynamic behaviour of the TIAGo head subsystem. Using this approach, obtained State-Feedback control law (globally and asymptotically) stabilizes the system to an equilibrium point in the joint position space. Thus, desired poses will be injected in the complete Fault-Tolerant Scheme as these equilibrium points. Under a fault scenario, this behaviour might not be completely achieved even using a fault estimation and compensation mechanism due to (1) the discrete-time approach, (2) the sample delay induced by the Apkarian filter, (3) and an inaccurate estimation of the fault. First one refers to the possible time inconsistencies that might arise when (presumably) estimating a magnitude using the values from previous time instants, which becomes more apparent as the discretization time T_s increases, and that could be also related to the latter reasons. In addition, physical actuators present non-linear effects that have not been considered in the dynamical model of the TIAGo head (which will be further analyzed in Chapter 89, that might hinder reaching the desired poses.

Therefore, an state augmentation of the system has been integrated to include integral terms in control actions, assuring null steady-state error on the joint position states. It should be pointed out that, in this work, the method is aimed to compensate the inaccuracies mentioned above, but not as a complete fault rejection mechanism.

Integral states x_I to include correspond to the integral of the error e, which results in the difference equation:

$$x_I(k+1) = x_I(k) + e(k) = x_I(k) + C_{des}x(k) - y_{des}(k)$$

being the reference $y_{des}(k) = [\theta_{1,des}(k)\ \ \theta_{2,des}(k)]^T$ and C_{ref} the output matrix of the appropriate

dimensions such that

$$C_{des}x(k) = [\theta_1(k)\ \theta_2(k)]^T$$

Thus, the augmented state-space form for the system associated to the i-th rule,

$$\begin{bmatrix} x_I(k+1) \\ \tilde{x}(k+1) \end{bmatrix} = \begin{bmatrix} I & C_{des} \\ 0 & \tilde{A}_i \end{bmatrix} \begin{bmatrix} x_I(k) \\ \tilde{x}(k) \end{bmatrix} + \begin{bmatrix} 0 \\ \tilde{B} \end{bmatrix} u_f(k) - \begin{bmatrix} I \\ 0 \end{bmatrix} y_{des}(k) \tag{4.10}$$

Note that the term y_{ref} will not be included in the PDC controller synthesis, and that the integral state augmentation is performed over the already augmented system that includes the Apkarian Filter. All the synthesis methods developed in this section are equivalently applicable to this form and augmented states will be used in the LQR tuning process described before.

Chapter 5

State and Fault Estimation

The design of a service robot has to consider the entropy and variability associated to those environments where the robot shares its workspace with humans or even has to collaborate with them, where usually there does not exist a complete information model, or its partially observable. Furthermore, own's robot internal state is sometimes unknown or inaccurate due to a lack of measurements or a low-reliability estimation of them based on measured ones. These issues are addressed in this chapter, where a fault estimation mechanism for any force that might be exerted on the TIAGo head subsystem is presented along with an observer formulation robust to these fault.

5.1 Robust Unknown Input Observer for Takagi-Sugeno Models

Observer models have been a classic topic in the Automatic Control field, playing an estimation role of certain systems' states that are *a priori* unknown or unmeasurable, (generally) considering the analytical model of the system to control. An additional effort has been put in their design to assure a robust performance disregarding exogenous events (namely disturbances in the literature) that might affect on its estimation, and therefore ill-posing the overall control approach. Disturbance rejection [14] presents an observer structure that includes the estimation and compensation of the disturbance, although based on *a priori* model of its behaviour. This conservatism has been overcame through Unknown Input Observers (UIO) [9], which decouples the state estimation from the effect of a disturbance f, only considering as known its input characteristics, determined by the matrix E_f:

$$x(k+1) = Ax(k) + Bu(k) + E_f f(k)$$
$$y(k) = Cx(k)$$

For the addressed TIAGo head system, torques exerted by external faults have a straightforward interpretation as these disturbances within the analytical equations, being UIO a suitable method to apply. But, with no regards about the required formulation for the TS framework, it assures decoupling only iff:

$$\text{rank}(CE_f) = \text{rank}(E_f), \tag{5.1a}$$

$$(C, A_I) \text{ is detectable, being } A_I = A - E_f[(CE_f)^T CE_f]^{-1}(CE_f)^T CA \tag{5.1b}$$

Condition from Equation (5.1a) only establishes that disturbances should affect those states included in the output model (measurable), but the one from Equation (5.1b) assures that all the non-observable states are observable for A_I, which is not fulfilled for all the A_i matrices of the TS model.

M. Chadli and H.R. Karimi have studied the UIO structure directly applied to TS Models, assuring stability and decoupling properties in the already mentioned Lyapunov and LMI frameworks, giving the formulation of a general Robust Unknown Input Observer for TS Models (RUIO-TS) [5] for a system with the form:

$$x(k+1) = \sum_{i=1}^{r} h_i(z(k))[A_i x(k) + B_i u(k) + R_i f(k)] \tag{5.2a}$$

$$y(k) = Cx(k) + Ff(k) \tag{5.2b}$$

where $f(k) \in \mathbb{R}^{n_f}$ stands for the unknown input (disturbance) vector being n_f the number of unknown inputs considered, and $R_i \in \mathbb{R}^{n \times n_f}$ and $F \in \mathbb{R}^{m \times n_f}$ represent the influence of $f(k)$ in the system behaviour. The existence of a solution is provided such that the following condition holds:

$$rank \begin{bmatrix} CR_h & F \\ -F_f & 0 \end{bmatrix} = rank \begin{bmatrix} F_f \\ R_h \end{bmatrix} + rank[F], \tag{5.3}$$

where F_f is the diagonal matrix F as terms, and R_h the horizontal concatenation of all the R_i.

Recalling the applicability issue of the UIO for the TIAGo head TS Model, the existence condition for RUIO-TS does not include a rank limitation on the form of the state-space matrices, overcoming the issue presented before. Also, condition from Equation (5.3) presents a remarkable relaxation with respect to previous analogous methods [4], as it mostly implies that no restriction is imposed between the matrices that determine the input and output influence of an unknown input (as e.g. disturbance or fault). Considering any kind of system, this determines that RUIO-TS considers a wider set of scenarios by including those where the same input has a different effect on the input than on the output.

The structure for the RUIO-TS gives an estimation of the system states $\hat{x}$:

$$\nu(k+1) = \sum_{i=1}^{r} h_i(z(k))[N_i\nu(k) + G_i u(k) + L_i y(k)] \tag{5.4a}$$

$$\hat{x}(k) = \nu(k) - Ey(k) \tag{5.4b}$$

where $o(k)$ corresponds to RUIO's state vector, with the same dimensions as $x(k)$. Matrices N_i, G_i, L_i and E are the observer gains to be synthesized such that asymptotic stability of the state estimation error dynamics are assured. Figure 5.1 includes an schematic representation of the RUIO-TS structure.

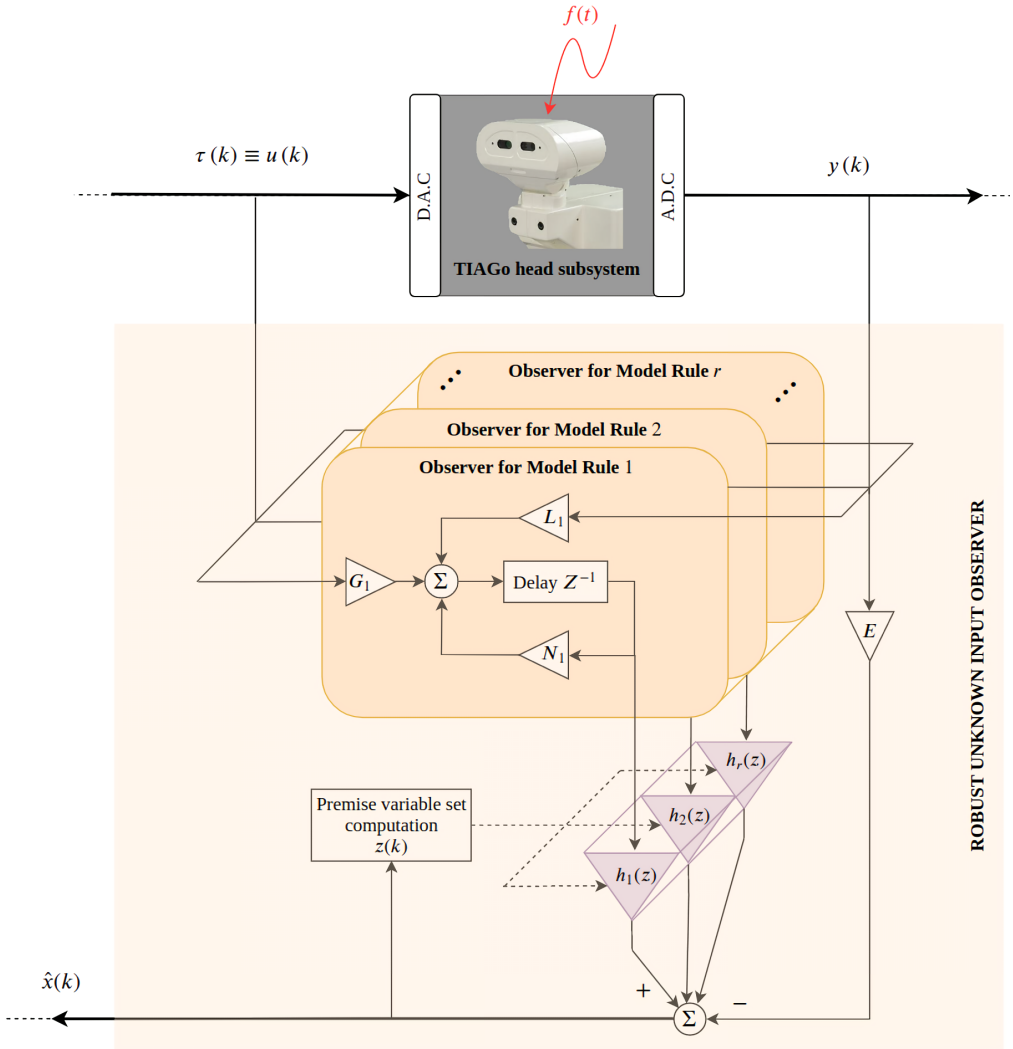


Figure 5.1: Schematic representation of the Robust Unknown Input Observer for Takagi-Sugeno Models in discrete-time.

Defining the estimation error $e(k)$ as

$$e(k) = x(k) - \hat{x}(k) = (I + EC)x(k) - z(k) + EFf(k),$$

and being its dynamics given by its derivative,

$$\begin{aligned} e(k+1) = &\sum_{i=1}^{r} h_i(z(k))(N_i e(k) + (TA_i - K_i C - N_i)x(k) + (TB_i - G_i)u(k) + \\ &+ (TR_i - K_i F)f(k) + EFf(k+1) \end{aligned}$$

where

$$T = I + EC, \tag{5.5a}$$

$$K_i = N_i E + L_i, \tag{5.5b}$$

To assure that estimation error converges to zero as time tends to infinite, disregarding the effect of the disturbance:

$$e(k+1) = \sum_{i=1}^{r} h_i(z(k)) N_i e(k)$$

which is obtained if the following constraints hold:

$$N_i = TA_i - K_i C \tag{5.6a}$$

$$TB_i - G_i = 0 \tag{5.6b}$$

$$TR_i - K_i F = 0 \tag{5.6c}$$

$$EF = 0 \tag{5.6d}$$

If E is determined, T can be obtained from the definition in Equation (5.5a) and G_i deduced from Equation (5.6b). Equations (5.6c) and (5.6d) come from the disturbance rejection in $e(k+1)$, which assures decoupling capabilities.

As with the PDC, LMI framework allows to include these conditions into a Lyapunov's stability problem that gives the definition for all the observer gain matrices fulfilling desired requirements. But the conditions originally presented for the RUIO-TS include the inverse of an intermediate variable in order to find a solution for E, which has been proved in this work to lead to an ill-posed the solution such that the observer does not completely decouple the unknown input effect in the estimation of the states. As no other method has been found to overcome this issue, the problem has been directly

stated in terms of stability for N_i, which implies a BMI formulation and therefore a NP-hard problem. Tractability issue for this problem will be addressed in Chapter 7.

An additional stability requirement for N_i has been included into the BMI formulation to avoid couplings between the dynamics of the PDC control and the RUIO-TS, as aforementioned. Following the $\mathbb{D}$-stabilization procedure already presented in Section 4.3 and the work presented in Reference [22], RUIO-TS poles have been placed in a circular region centered in σ with a radius r, such that poles are at least n times slower than PDC ones.

Therefore, the complete synthesis problem for the RUIO-TS is given by the following conditions,

$$\begin{bmatrix} r(M_i + M_i' - S_i) & M_j^T N_i^T - \sigma M_j^T \\ (*) & rS_j \end{bmatrix} < 0, \forall\,(i,j) \in I_r^2, \text{ being } I_r = 1, 2, ..., r$$

$$(I + EC)R_i = K_i F, \qquad \forall\, i = 1, 2, ...r$$

$$EF = 0.$$

where $S_i, M_i \in \mathbb{R}^{n \times n}$ are auxiliary intermediate symmetric matrices, and N_i is defined as in Equation (5.6a). Remaining RUIO-TS gain matrices L_i and G_i are defined as aforementioned:

$$G_i = (I + EC)B_i$$
$$L_i = K_i - N_i E$$

Note that for this synthesis problem, the discrete-time domain TS Model is used without applying the Apkarian pre-filtering nor the Integral State Augmentation methods described in Chapter 4, as the synthesis problem formulation already includes variant B_i terms and only regards on convergence in the state estimation.

Finally, the selection of R_i and F has been done according to the concept of fault considered in this Thesis. The desired Fault-Tolerant Control scheme is focused on those forces exerted on a manipulator during its operation to contacts, which might be produced from a physical Human-Robot Interaction. These forces will exert certain torques in the rotational joints of the head subsystem, i.e. the unknown input term $f(k)$ from Eqs. (5.2). Thus, each R_i has to define this relation for each f_i in the state-space form, i.e. defined according to its point of operation with the same form as B_i as they also are input torques to the system:

$$R_i \triangleq B_i(z) \tag{5.8}$$

Matrix F will be null independently on the measured state variables, as R_i matrices already describe

the complete effect of the faults, and therefore including the same relation on the output states will be redundant.

To check if the existence condition of the RUIO-TS observer from Equation (5.3) is satisfied, the output model of the system must be also determined. Output matrix C has been defined such that:

$$y(k) = Cx(k) = [\dot{\theta}_2(k)\ \theta_1(k)\ \theta_2(k)]^T \tag{5.9}$$

which determines a scenario with an incomplete information model where $\dot{\theta}_1$ is not measurable. Considering this assumption, condition from Eq. (5.3) does not hold for given R_i and F, as only one of the affected states is measurable. Thus, remaining F null,

$$R_i(z(k)) \triangleq [B_{i,(1,1)}\ B_{i,(2,2)}\ 0\ 0]^T, \tag{5.10}$$

where $B_{i,(n,m)}$ the term in the (n, m) position of matrix B_i, fulfills the existence condition for the measurement model, and in terms of the fault scenario it assumes that the same torque component affects both joints equally. Although this assumption differs with the given description of the fault effect, it has been proven to assure the desired decoupling requirements for an scenario with different torque components affecting each joint, as it will be seen on simulation results presented in Chapter 7.

5.2 Fault estimation

The RUIO-TS structure is able to decouple the state estimation from the effect of an unknown torques that affect the system. This behaviour is desired for scenarios where with a partially complete or inaccurate information model, but also some insights about the fault characteristics could be useful in order to evaluate further measures. If we extrapolate the TIAGo head example to a robotic manipulator of n joints that has a direct collision with a human, information about the associated force could be crucial: in the case of an undesired contact, to avoid injuring the person; otherwise, to gather data that specifies the intentionality of the human action.

Therefore, a mechanism to estimate the magnitude and isolate the fault has been implemented by means of a Reference Control structure, using the analytical expressions from Eqs. (3.2), without any fault term being injected. The reference signals for the head subsystem will be also given to the Reference Control, and the same PDC design will be used, but the RUIO-TS as its states are available. The magnitude of the fault $\hat{f}(k)$ will be obtained as the difference between the Reference Control states

$x_{ref}(k+1)$ and the one obtained from the decoupled estimation $\hat{x}(k+1)$:

$$\hat{f}(k) = \Theta_f[x^a_{ref}(k+1) - \hat{x}^a(k+1)], \tag{5.11}$$

where Θ_d matrix is defined so $\hat{f}$ has the components of the disturbance in both axis $\hat{f}_{\theta_1}$ and $\hat{f}_{\theta_2}$, corresponding to the differences on the first and second components of $x(t+1)$, and

$$\begin{aligned} \hat{x}^a(k+1) &= A(\hat{x}(k))\ \hat{x}(k) + B(\hat{x}(k))u(k), \\ x^a_{ref}(k+1) &= A(x_{ref}(k))x_{ref}(k) + B(x_{ref}(k))u_{ref}(k), \end{aligned} \tag{5.12}$$

standing 'ref' subindexes for variables of the reference structure. The superindex 'a' points out that the values at $k+1$ are obtained by means of the analytic model of the TIAGo head subsystem, which is considered as a close enough approximation of the evolution of the states.

The role of this fault estimation technique within the overall Fault-Tolerant scheme will be further analyzed in Chapter 6.

Chapter 6

Fault-tolerant Control Scheme

In this Chapter, the integration of the described methods along this work into the general Fault-Tolerant scheme is presented. Figure 6.1 depicts an schematic representation of all the involved elements and the relationship between them in this Fault-Tolerant control architecture.

At each time instant, the desired positions for both axis y_{des} are externally given (e.g. by a trajectory planner) to both the Reference and Main Control structures. They are directly introduced into the state-feedback law for the augmented system including the integral control states, as described in Equation (4.10).

State-feedback PDC computes the corresponding gain $K(z(k))$, and the feedback control action $u_{\text{PDC}}(k)$ is obtained considering for the augmented state space after the application of the Apkarian Filter and Integral Control techniques. It should be noted that for the main control structure, the decoupled estimation of the states $\hat{x}(k)$ given by the RUIO-TS has been used on the augmented states. The control action is computed as

$$u_{\text{PDC}}(k) = -K(z(k)) \cdot [x_I(k) \quad \tilde{x}(k)]^T, \tag{6.1}$$

and the control torque $\tau(k)$ is obtained by applying the discrete time formulation of the Apkarian filter from Equation (4.3).

Simultaneously, the estimation of the torques exerted by the exogenous force affecting the system $\hat{f}_i(k)$ are obtained according to Equations (5.11) and (5.12) from $\hat{x}(k)$ and $x_{ref}(k)$. Additionally, an estimation of the gravity effects vector $\hat{g}_v(k)$ is obtained using the analytic form given in Equation (4.3), and both are injected using an Active Compensation mechanism, obtaining the compensated control torque:

$$\tau'(k) = \tau(k) - [\hat{g}_v(k) + \hat{f}(k)] \tag{6.2}$$

This mechanism implies that the State-Feedback plus RUIO control scheme will regard only for the part of control torques related with the *nominal* operation (i.e. moving to different poses), as the fault and gravity effects will be compensated using their estimation.

Finally, both the TIAGo head subsystem and its analytical model from the Reference Control structure update their outputs on the respective injected control torques for the next time instant $k+1$. As aforementioned, the real subsystem only provides the value of $\dot{\theta}_2$, θ_1 and θ_2, recalling that for the Reference Control structure all states are known.

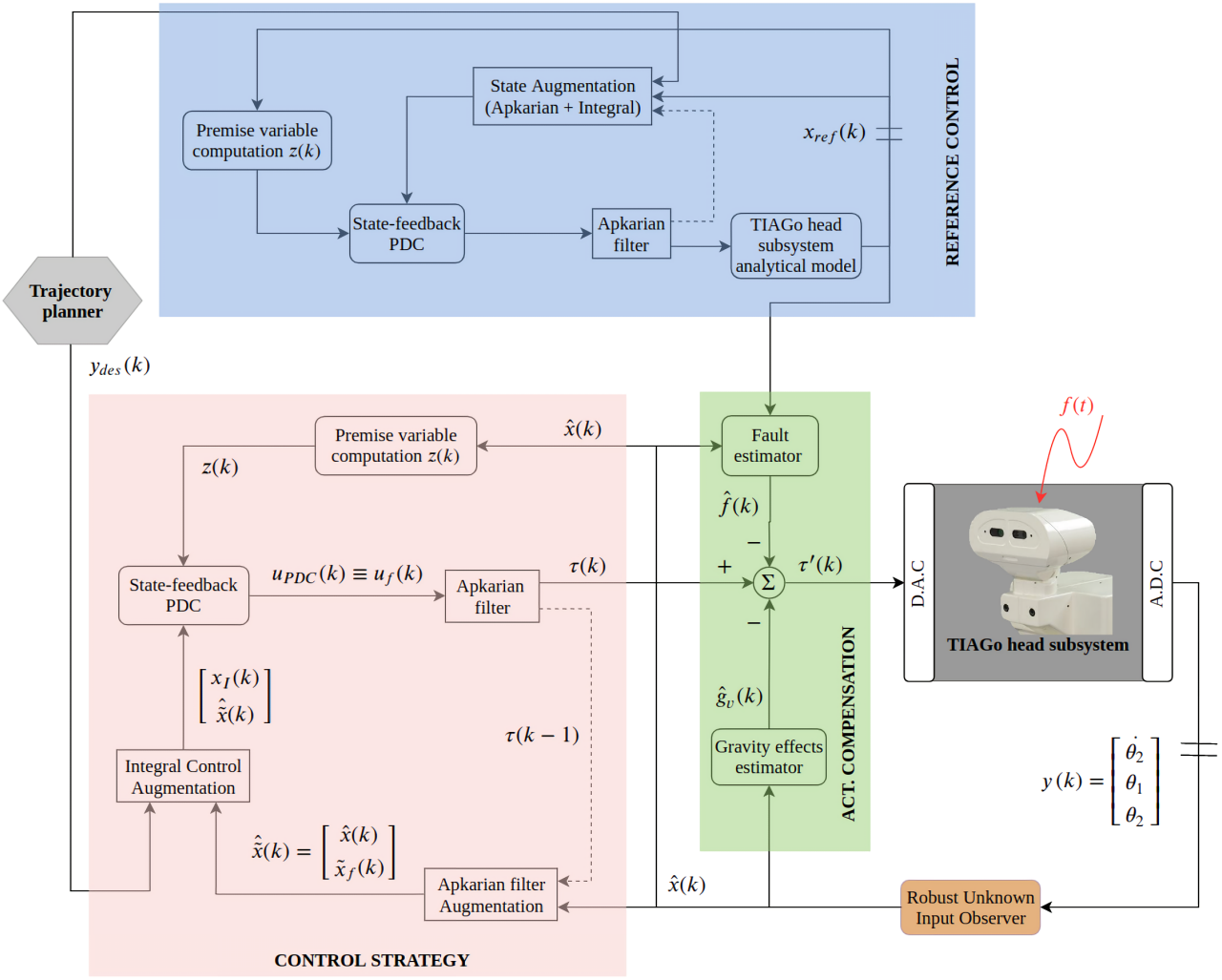


Figure 6.1: Schematic representation of the complete fault-tolerant control approach on discrete time.

Chapter 7

TIAGo Head Simulation Results

This chapter is devoted to present results for the different key aspects discussed in previous chapters, and to demonstrate the behaviour of the overall control scheme in a realistic simulation environment. Three main sections have been considered:

1. **Optimal Control**, to discuss the effect of the LQR tuning method and the results regarding fulfillment of the desired performance.

2. **Scheduling mechanism**, which only presents graphical descriptions about the behaviour of the TS Model and PDC according to the operation point of the system.

3. **Fault Scenarios**, to conclude this chapter with an analysis of the complete Fault-Tolerant control scheme under different exogenous forces.

All the simulations have been performed using MATLAB[2] programming environment. TIAGo head subsystem has been included into the implementation as a continuous-time model, using the already presented analytic expressions, physical parameters and limits of the real system. In the used discrete-time implementation, Zero Order Hold (ZoH) discretization method has been applied for a sampling time $T_s = 10\ [ms]$ ($f_s = 100\ [Hz]$). The selection of the discretization time has been chosen regarding execution times on the implementation in the real platform, and will be further analyzed on Chapter 8. Apkarian filter gain Ψ has been determined to be less or equal to $1/T_s$ in order to avoid consistency related problems between samples from consecutive time instants on its application.

Regarding the design phase, YALMIP toolbox [3] has been used as it offers the possibility to work

[2]MATLAB programming environment by MathWorks: `https://www.mathworks.com/products/matlab.html`(Accesed 22 June 2019)

[3]YALMIP optimisation and modeling toolbox for MATLAB: `https://yalmip.github.io/` (Accesed 22 June 2019)

with multiple optimization software under a common formulation in MATLAB. In the PDC synthesis problems, Semi-Definite Programming Algorithms (SDPA) provided by SEDUMI[4] and MOSEK[5] have been used. Recalling tractability issues for the BMI formulation on the RUIO-TS problem, PENBMI [6] solver has been used due to the number of available features to adjust the global optimization algorithm that have been proved to overcame them.

7.1 Optimal Control

The PDC synthesis problem has been stated for this work by combining both $\mathbb{D}$-Stabilization and LQR conditions, in order to fulfill optimality assuring a bound over τ_s in a *strip* region defined by α and β such that there does not exist coupling effects with the RUIO-TS that might alter the closed-loop dynamics.

In terms of the system's response to an step input, settling time T_{sett} depends on the value for τ_s, and so bounding $\tau_s \in [\alpha, \beta]$ will imply the equivalent for T_{sett}. But, recalling that the LQR problem will be already tuned according to an optimal criterion over the performance characteristics, this specification has been only used to constraint this problem. According to these guidelines, the following values have been selected:

$$\alpha = 0.993, \quad \beta = 0.9 \tag{7.1}$$

being the magnitude of a pole in the discrete time domain $|s| = e^{-\tau_s T_s}$ and defining $T_{sett}(s, \%) = -\ln(\%)/\tau_s$, for an error band of 2% around the amplitude of an step input, and with the already set $T_s = 10[ms]$:

$$T_{sett}(\beta, 2\%) \leq T_{sett}(s, 2\%) \leq T_{sett}(\alpha, 2\%) \tag{7.2}$$

$$0.37[s] \leq T_{sett}(s, 2\%) \leq 5.57[s] \tag{7.3}$$

At this point it should mentioned that in the discrete time complex plane, constant τ_s is associated with circular trajectories around the origin, instead of the mentioned *strip* region. This applies certain conservatism by only considering an smaller area where the poles with the same τ_s can lie, but it indeed benefits the application of LQR conditions, as most of the neglected areas correspond to poles with larger phase, which is related with greater oscillatory behaviours.

[4] SEDUMI optimisation over symmetric cones software: `http://sedumi.ie.lehigh.edu/` (Accesed 22 June 2019)

[5] MOSEK large scale optimization software: `https://www.mosek.com/` (Accesed 22 June 2019)

[6] PENBMI modeling an optimization software for MATLAB by TOMLAB: `https://tomopt.com/tomlab/` (Accesed 22 June 2019)

Regarding the LQR problem, weighted Bryson's rule has been applied to embed the performance specifications of the TIAGo head into the tuning of matrices Q and R as the maximum admissible magnitudes for state variables and input actions:

- For angular positions $\theta(k)$ and velocities $\dot{\theta}(k)$, these values correspond to the upper bound of their limits obtained from the given operational conditions of TIAGo, already listed in Table 3.3.
- Same maximum admissible values have been set for Apkarian filter states $x_f(k)$ and input action $u_f(k)$, as both refer to the input torque for the system. For both angles, the limit has been set to $1[N \cdot m]$, which will be considered along the simulations as an upper bound for the maximum torque associated to a force in either joints, and therefore the maximum torque to be exerted by the state-feedback control mechanism without the existence of an Active Compensation.
- For integral states $x_I(k)$, admissible values will determine the maximum allowed error on joint angles, and so, lower values will correspond to a faster response, i.e. greater control actions associated to these states. As this might hinder fulfillment of other operational conditions like speed and torque, which are also subject to the physical system limits, their values have been set to the maximum movement amplitude of each joint.

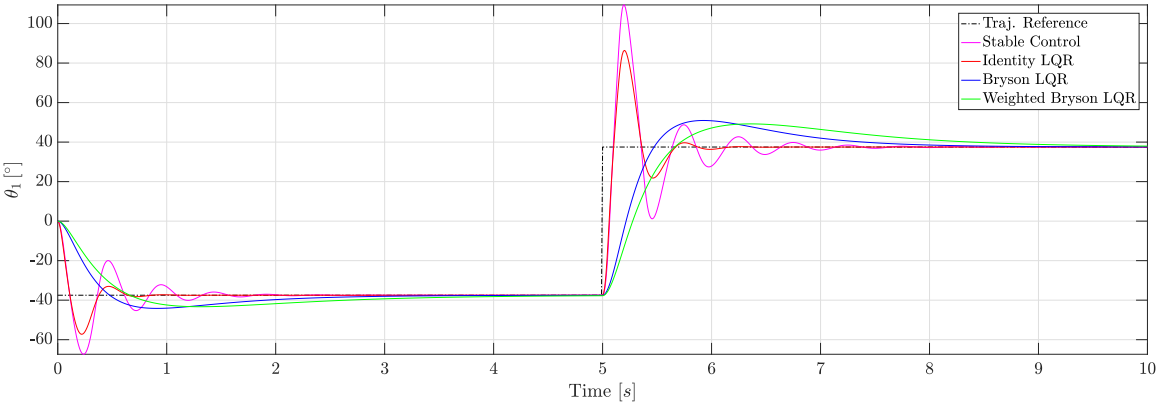
These maximum admissible values have been included in Table 7.1 according to the assumptions from above. Note that those terms which have common maximum admissible values for both joints have been assigned with the same weight, and have been denoted with the subscript i.

Parameter	**Max. admissible value** $\overline{e}_i$	**Tuning weight** $\_i$
x_{I,θ_1}	150 [°]	0.000001
x_{I,θ_2}	105 [°]	0.000001
$\dot{\theta}_i$	3 $[rad/s]$	0.999
θ_1	75 [°]	0.01
θ_2	60 [°]	0.01
x_{f,τ_i}	1 $[N \cdot m]$	0.05
u_{f,τ_i}	1 $[N \cdot m]$	0.05

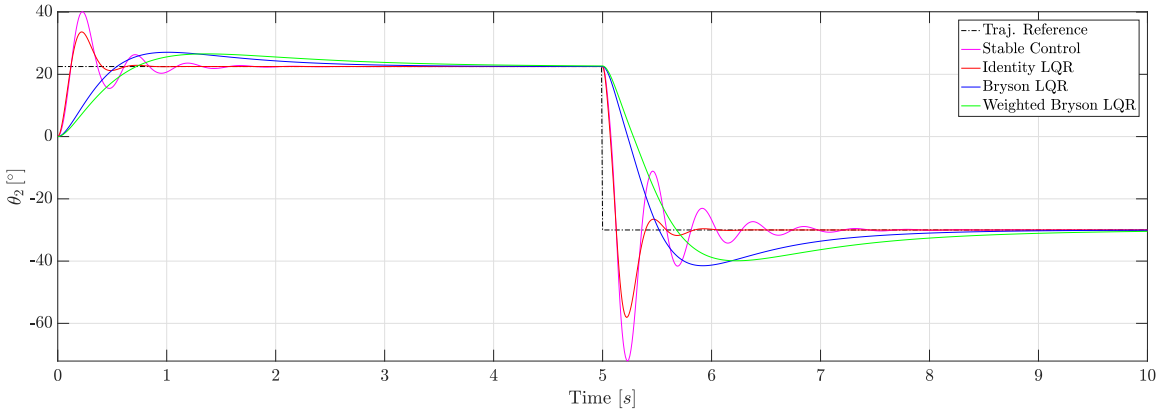
Table 7.1: Maximum admissible errors and solution tuning weights used on LQR synthesis problem.

The main objective of this procedure is to find a solution that maintains the operational conditions for the transient response of the system, which will be determinant on the presented behaviour under the effect of an exogenous fault force. To verify this, step references from the half-lower to half-upper limit of the joint angles have been given to the complete Fault-Tolerant Scheme, without any fault being exerted, only compensating for the gravity effects. Over these results, weights ρ_i have been iteratively

tuned towards a solution that fulfills the conditions, which has been also included in Table 7.1. Figure 7.1 shows the evolution of angular positions for this solution, along with the obtained responses for solutions (1) regarding only stability conditions, (2) for identity Q and R matrices, and (3) applying the non-weighted Bryson's rule LQR. The maximum values for each state variable and input in the simulation of all these solutions have been listed in Table 7.1, where those that fulfill the maximum admissible errors from Table 7.1 have been highlighted.



(a)



(b)

Figure 7.1: Closed-loop system response for step input reference under different PDC solutions, for Pan (a) and Tilt (b) angles.

Results show that, as expected, the solution regarding only stability nor the identity tuning of the LQR matrices fulfill all the design conditions, also presenting an undesired oscillatory behaviour. Obtained solution by applying Bryson's rule tuning presents an improvement with respect to them, but still the angular velocity surpasses the maximum admissible value. Applying over this solution a

Solution	$\overline{x}_{I,\theta_1}$ [°]	$\overline{x}_{I,\theta_2}$ [°]	$\overline{\dot{\theta}}_i$ $[rad/s]$	$\overline{\theta}_1$ [°]	$\overline{\theta}_2$ [°]	$\overline{x}_{f,\tau_i}$ $[N \cdot m]$	$\overline{u}_{f,\tau_i}$ $[N \cdot m]$
Stable Control	10.37	7.73	20.22	109.42	72.11	3.31	3.31
Identity LQR	8.45	7.33	18.04	86.42	58.09	2.75	2.75
Bryson's LQR	30.24	24.68	3.97	50.93	41.48	0.48	0.48
W. Bryson's LQR	39.12	30.59	2.98	49.21	39.90	0.46	0.46

Table 7.2: Maximum values for simulation on step reference signal.

set of weights designed towards this objective led to a PDC that assures achieving all the objectives simultaneously. It should be pointed out that the overshooting effect on the response of these two latter solutions is associated to the presence of dominant zeros in the closed-loop system, and should be further addressed in the future.

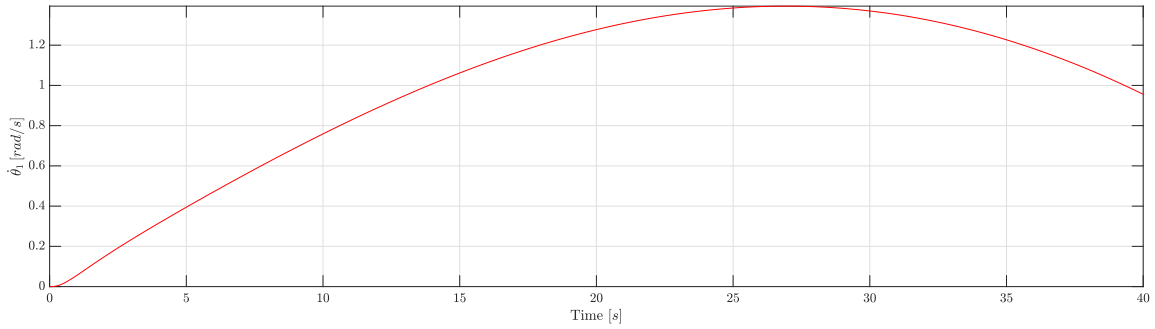
Finally it should be pointed out that in the Fault-Tolerant general scheme, angle joint references are considered to describe continuous trajectories between their limits, without any step in their values. But to avoid making assumptions on the trajectory planner, which is a completely external element associated, for example, with an upper task planner layer, this step reference *worst-case scenario* has been used to tune the LQR.

7.2 Scheduling mechanism

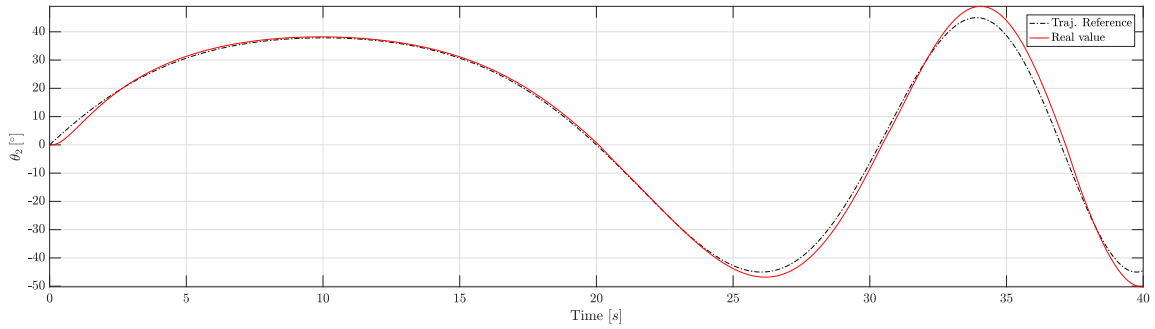
Using the optimal solution from previous section, a scenario is presented for a set of reference trajectories to track by the TIAGo head subsystem, in order to visualize the behaviour of the complete control architecture under the TS Model formulation and also to prove that the system maintains stability for all the operational points. The chosen scenario for this simulation does not include (yet) any exogenous force nor the RUIO-TS estimation (complete information model).

Figure 7.2 presents the evolution of joint velocity $\dot{\theta}_1$ and position θ_2, which determine the value of the premise variables set $z(k)$, according to which the activation levels h_i associated to each i rule are determined. This behaviour corresponds with the *defuzzyfication* step that represents the system as a linear-like combination of the limit ones, and regarding the control action it resembles a gain-scheduling fashion dependent on $z(k)$.

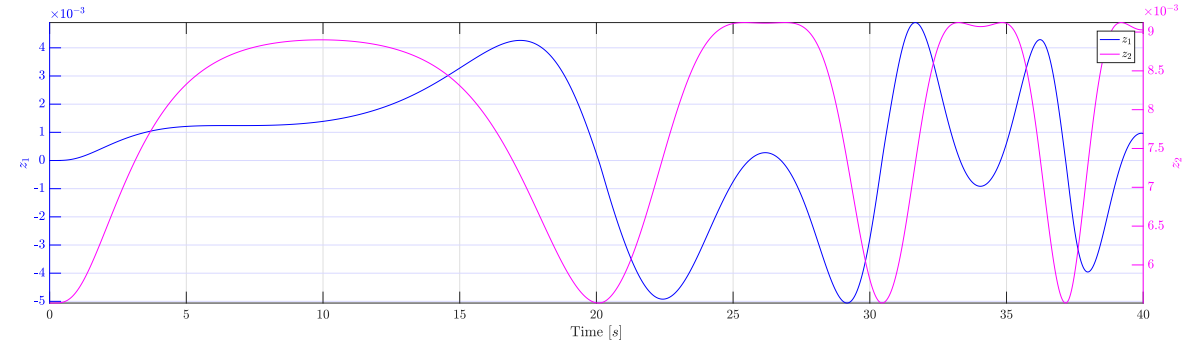
For the same scenario and reference, Figures 7.3(a) and 7.3(b) represent the trajectory of the TIAGo head matrices A and B, respectively, in terms of z with respect to time. The corresponding values for matrices A_i and B_i have been included as constant combinations of z_1 and z_2, which confine the trajectory of the system, being generated a *bounding box* that encloses all the possible behaviours.



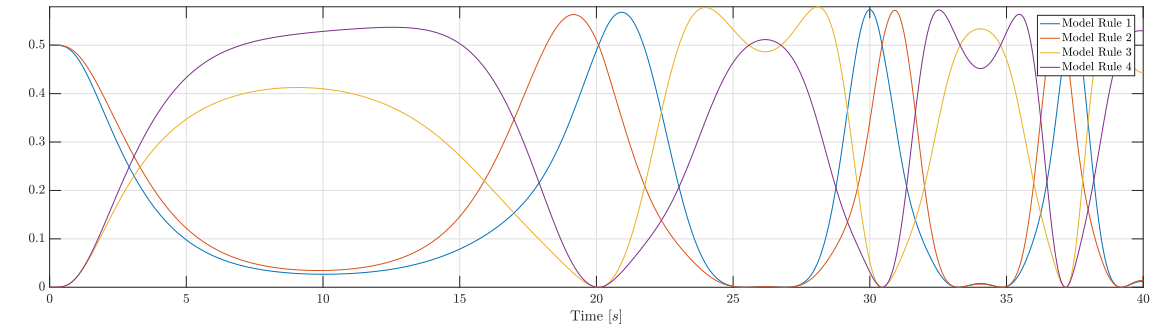
(a)



(b)

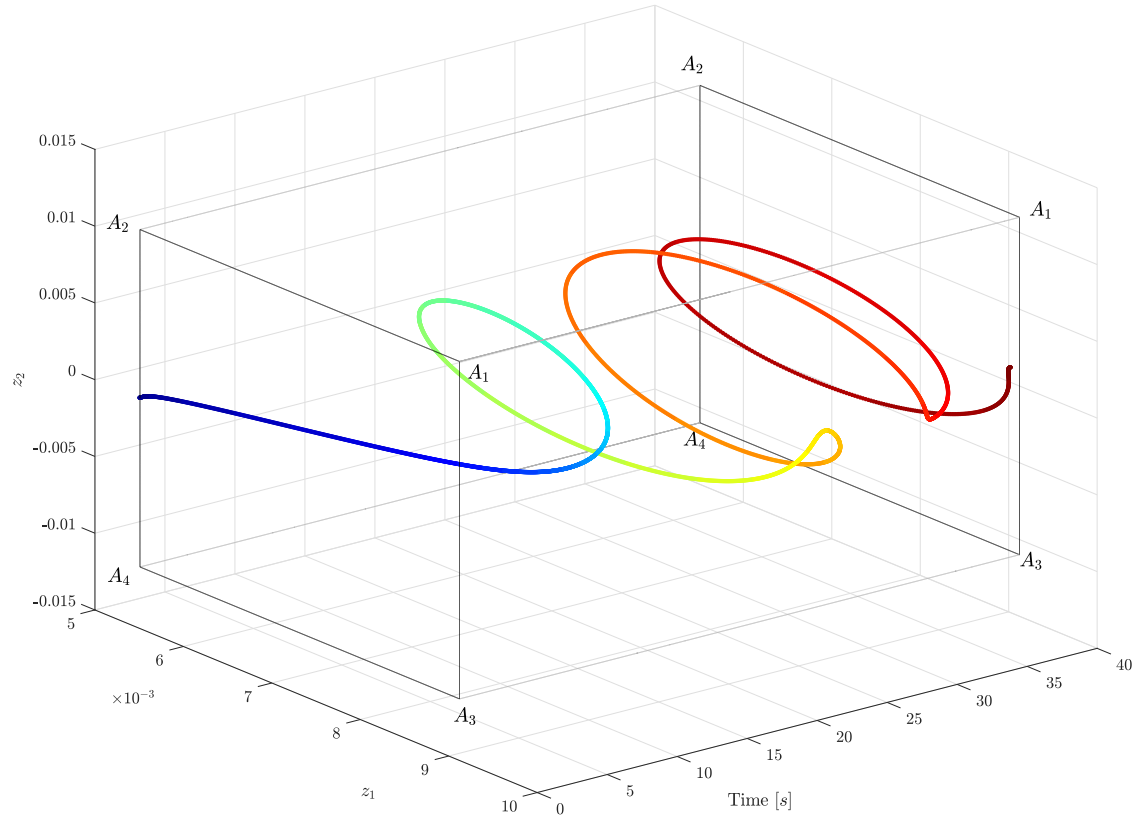


(c)

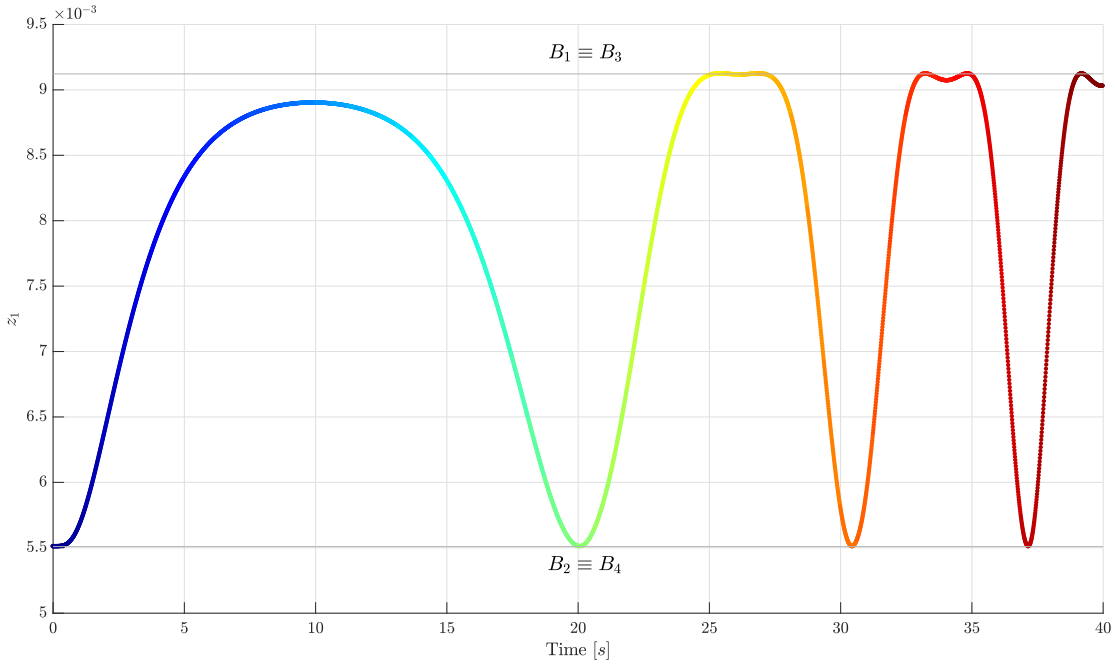


(d)

Figure 7.2: Evolution of $\dot{\theta}_1$ (a), θ_2 (b), premise variables z (c) and activation levels h_i (d).



(a)



(b)

Figure 7.3: Visualization of state-space $A(z)$ (a) and $B(z)$ matrix form evolution in terms of premise variable set z.

7.3 Fault Scenarios

Finally, the complete Fault-Tolerant scheme has been evaluated using the Active Compensation mechanism and the RUIO-TS under the incomplete information model formulated in Chapter 5.

First of all, the issue that has been being mentioned along this work on avoiding dynamic couplings between RUIO-TS and the PDC has to be recalled. For this purpose, $\mathbb{D}$-stabilization conditions have been imposed for both synthesis problems, being determined the dynamics of RUIO-TS as 10 times faster than the ones of PDC. Therefore, the value of β defined on Section 7.1 has been used to define the LMI circular region (σ, r) for RUIO-TS poles, also considering that all the poles have been placed in the Left Half Plane of the circular complex plane region that defines stability:

$$\sigma = r = 0.2391 \tag{7.4}$$

Figure 7.4 shows placement of the poles of all the limit LTI subsystems in the complex plane, where the red region corresponds to PDC and blue one to RUIO-TS.

The results of two different fault scenarios have been included in this work, comparing in each one the obtained outcomes for three control strategies: (1) the complete Fault-Tolerant control scheme including the Active Compensation Mechanism, denoted as "Complete Scheme" (2) the same scheme without the integral control state augmentation of the system, "No Integral States", and (3) the currently existing Propotional Integral and Derivative (PID) position control on the TIAGo head subsystem, "Position PID". The gain values for the latter one (which is, indeed, a PD control) have been obtained from the given specifications of the robotic platform, and are included in Table 7.3, and its discrete-time implementation has been depicted on the scheme included in Figure 7.5.

Faults consist on forces exerted on the COG of the second link, which is displaced from the second axis by L (defined in Table 3.2), with a time-dependant magnitude $|\mathcal{F}|$ and orientation $\mathcal{F}^{\phi}$ around the vertical axis of the head base.

Joint	P Gain	I Gain	D Gain
θ_1	1	0	32
θ_2	5	0	34

Table 7.3: Actuator PID gains for the TIAGO head subsystem.

In every scenario, the performance of each one method is presented separately in terms of the evolution of states $\dot{\theta}_i$, θ_i and control τ and fault active compensation $\hat{f}$ actions. The comparison is made in terms of the absolute error and its cumulative, normalized by the total number of time steps, and the Mean Squared Error (MSE) for the whole simulation. Desired trajectories for the joint angles

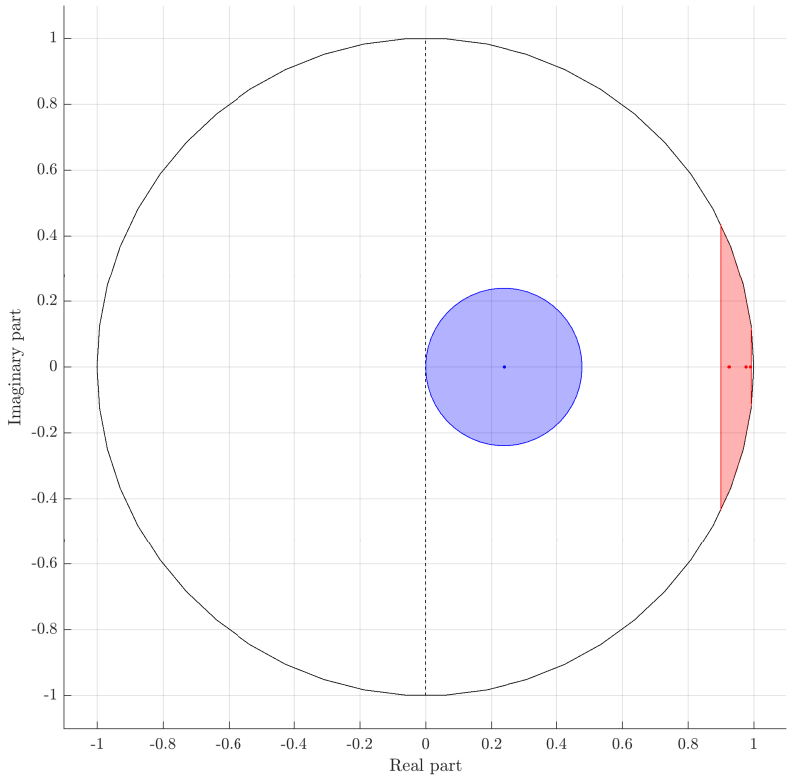


Figure 7.4: Placement of poles and defined LMI regions for closed-loop system with state-feedback PDC (red) and RUIO-TS (blue) solutions in the complex plane.

for all the cases are described by the following equations:

$$\theta_{1,des}(t) = \overline{\theta}_1 \sin(0.1t)$$
$$\theta_{2,des}(t) = \overline{\theta}_2 \sin(0.2t - 6.6 \cdot 10^{-3}\, t^2)$$

Disregarding the control method applied, the estimation of $\dot{\theta}_1$ given by the RUIO-TS is used, and it will be also analyzed. Finally, the evolution of exogenous force in each scenario is presented along with the estimation obtained by the complete Fault-Tolerant Scheme, in terms of the exerted joint torques, and its magnitude and orientation.

Previous to the presentation of both scenarios, Table 7.4 includes the MSE of each control strategy for the common trajectories under no-faults, to be used as a baseline values.

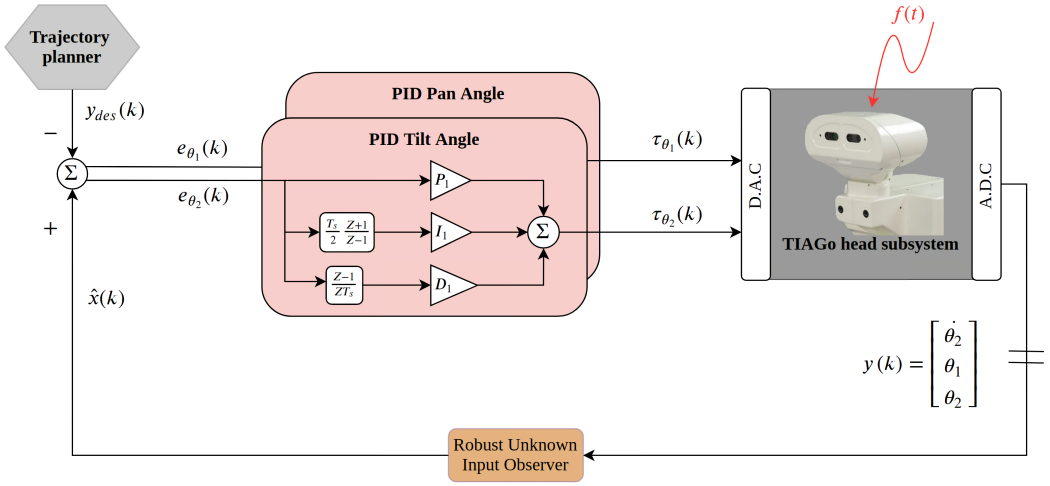


Figure 7.5: Schematic representation of the discrete-time PID position control implemented.

Joint	Complete Scheme	No Integral States	Position PID
θ_1	0.2672	22.4914	0.1588
θ_2	0.7585	13.9740	0.8299

Table 7.4: Mean Squared Error (MSE) over desired trajectory under no-faults for the considered control strategies.

7.3.1 Scenario I: Mass on top

Recalling the initial motivation behind the development of the Fault-Tolerant control scheme proposed in this work, in this initial scenario the exogenous force represents a mass placed at the top part of the second link. Therefore, the orientation of the force is constant and equal to zero, and the magnitude has been introduced as a *rounded-edge* trapezoidal signal with a rise time of $1\,[s]$, in order to emulate the mass placement effect.

Joint	Complete Scheme	No Integral States	Position PID
θ_1	0.2672	22.5578	0.1584
θ_2	3.3504	56.0422	2789.7548

Table 7.5: Mean Squared Error (MSE) over desired trajectory under Fault Scenario I for the considered control strategies.

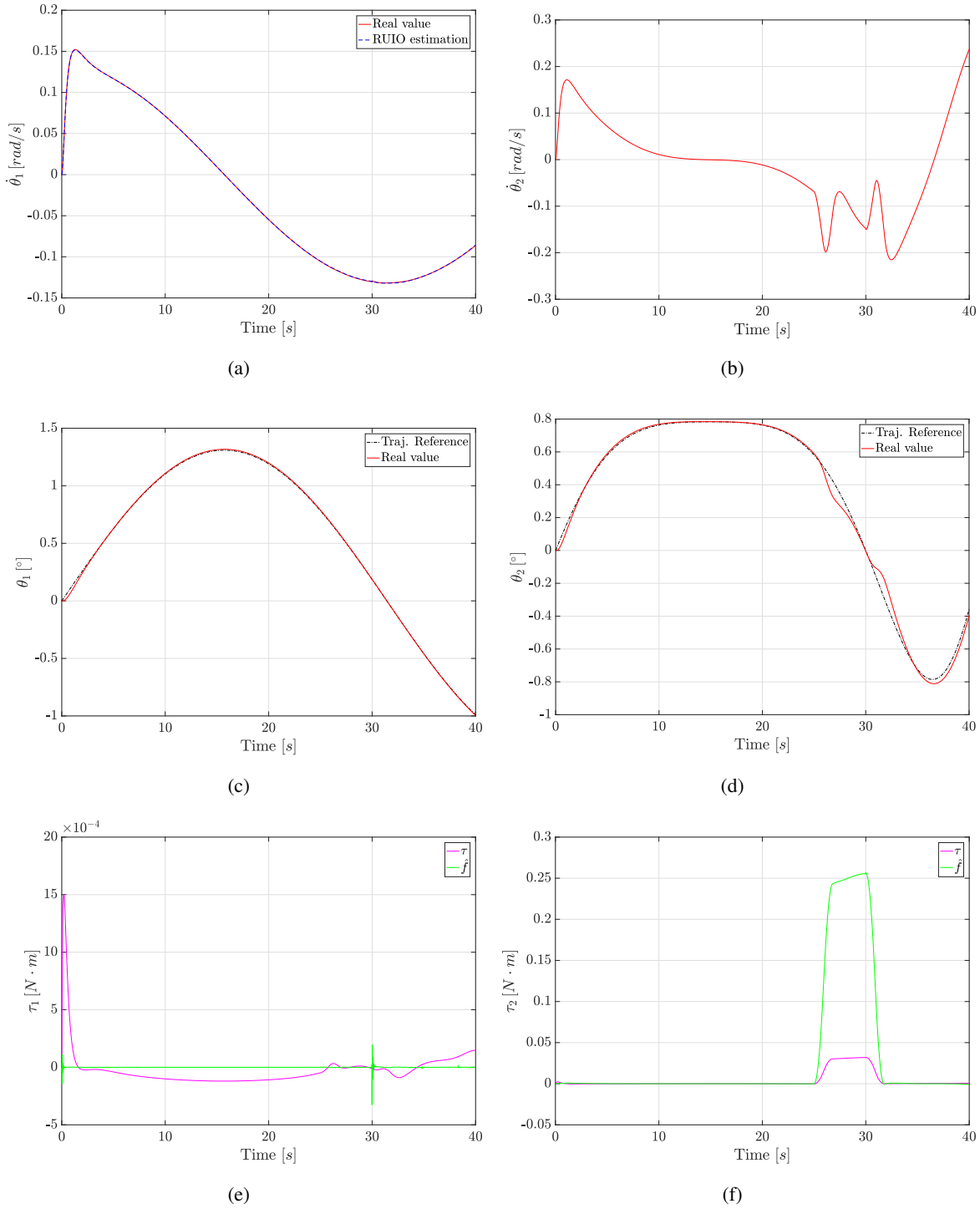


Figure 7.6: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

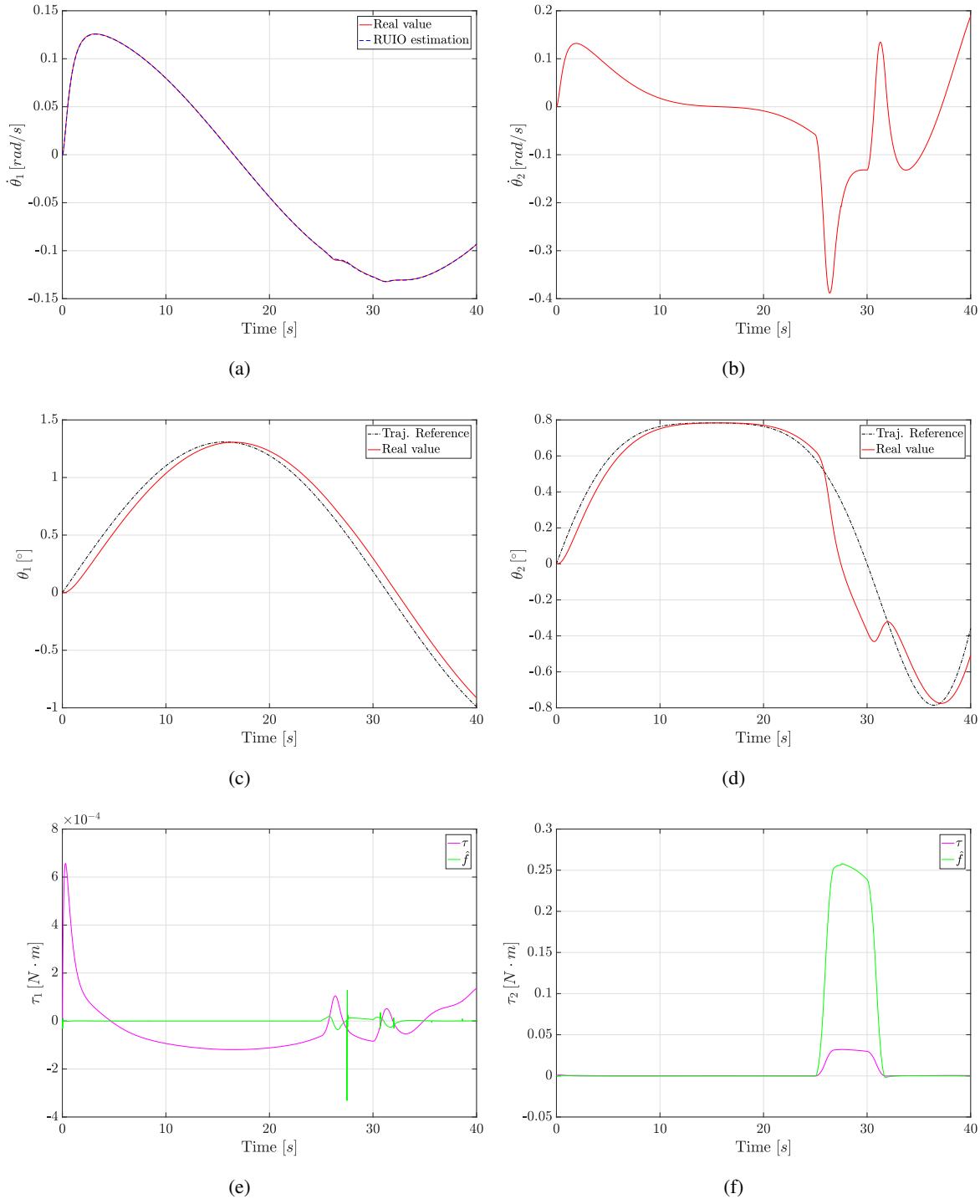


Figure 7.7: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

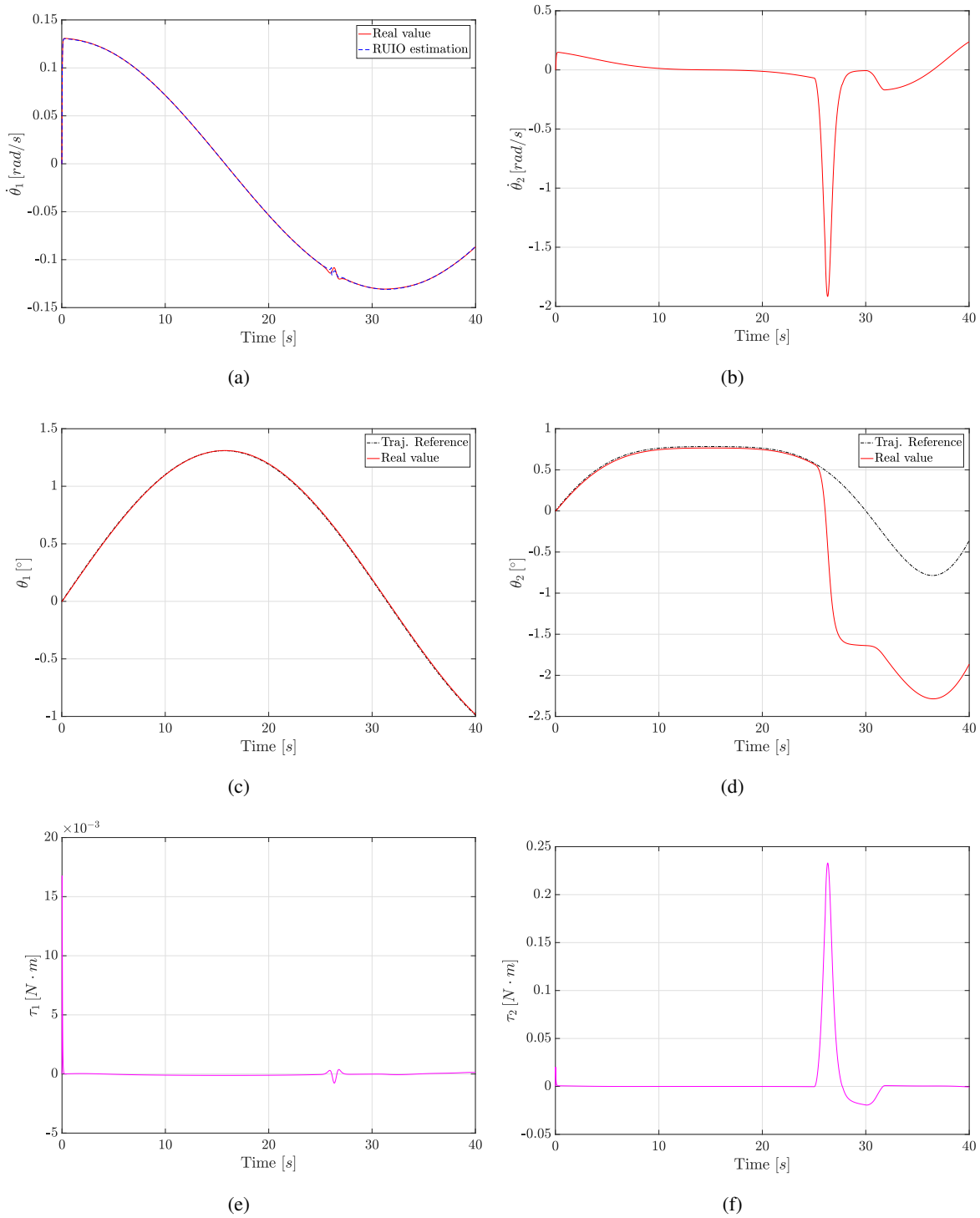
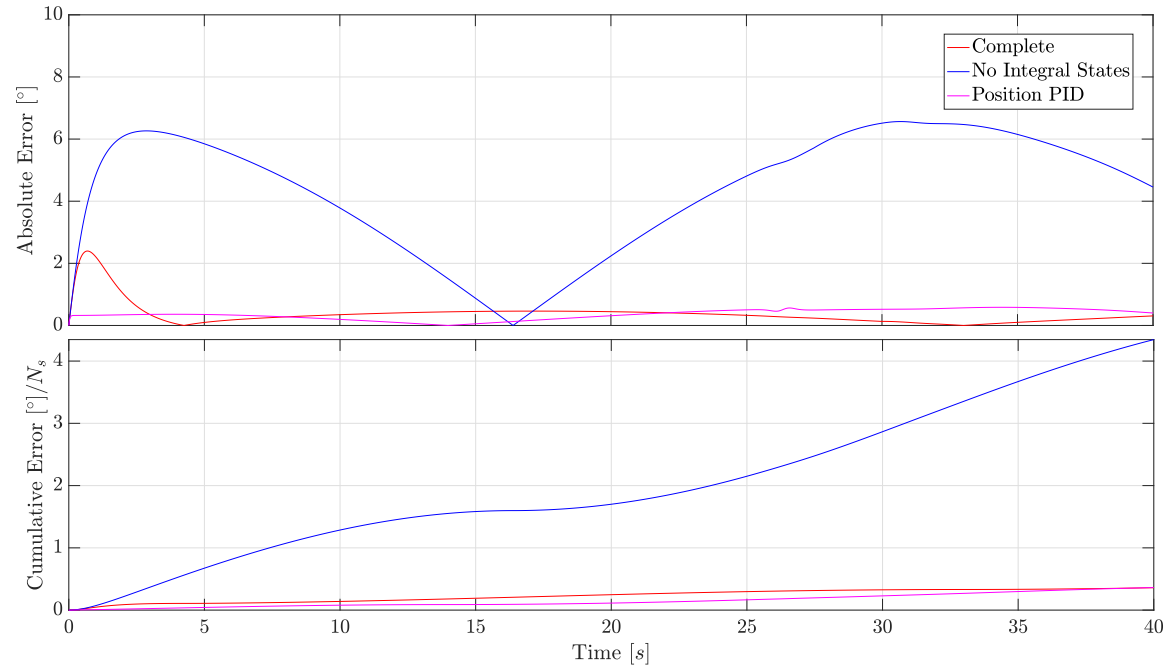
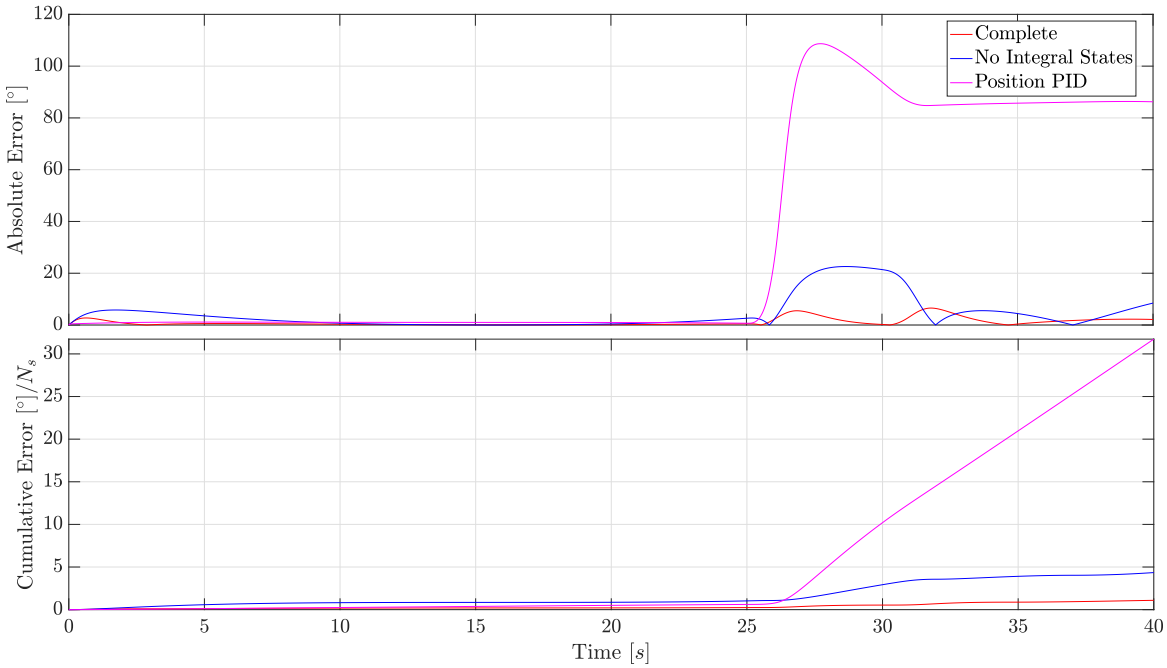


Figure 7.8: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

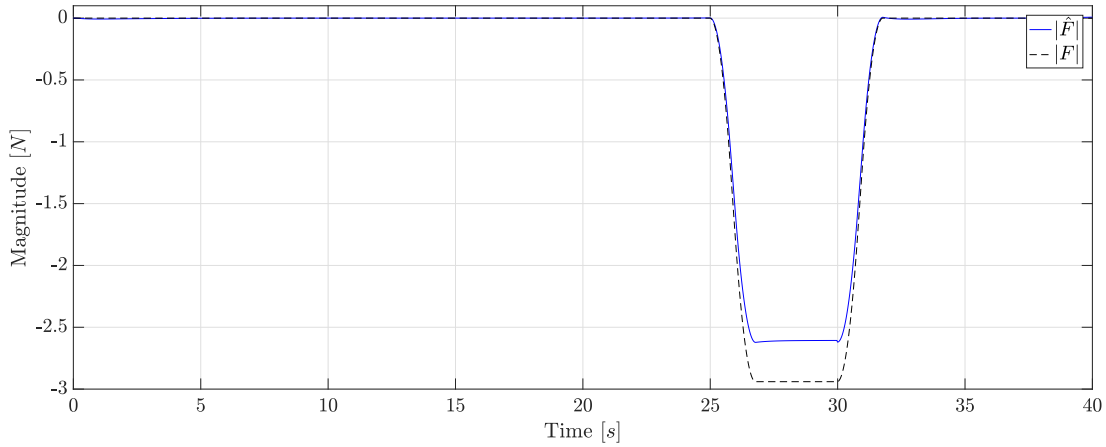


(a)

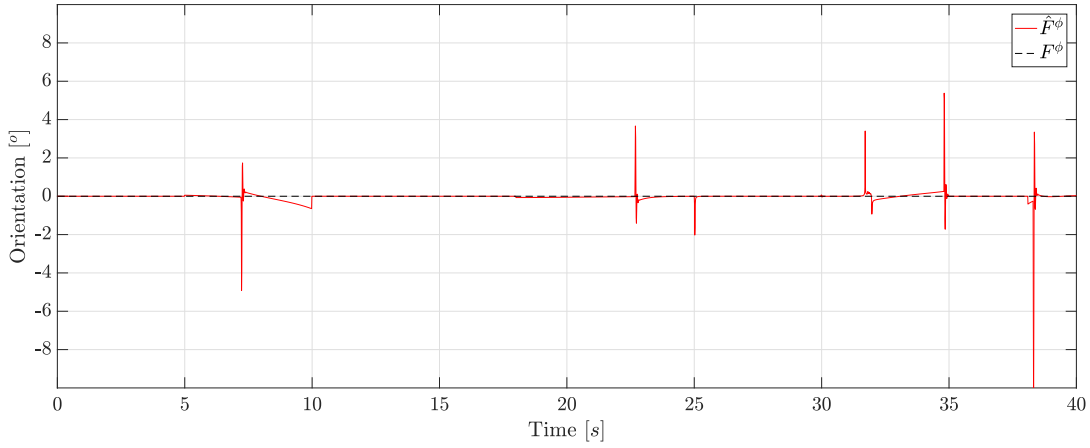


(b)

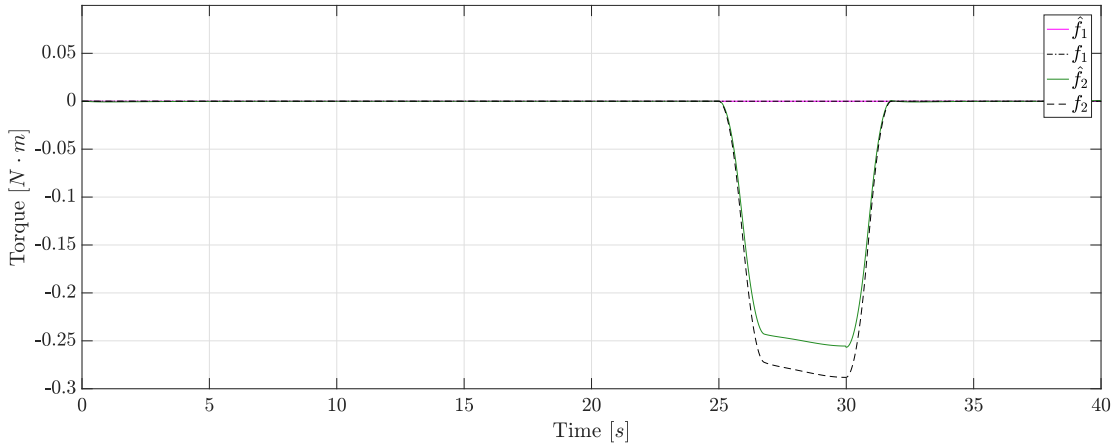
Figure 7.9: Pan (a) and Tilt (b) angles absolute error and its normalized cumulative evolution for the considered control schemes under Fault Scenario I.



(a)



(b)



(c)

Figure 7.10: Evolution of applied and estimated force (a) magnitude, (b) orientation and (c) exerted joint torques given by the complete Fault-Tolerant scheme under Fault Scenario I.

7.3.2 Scenario II: Constant time-dependent force

This scenario considers a force with both magnitude and orientation variable with time:

$$|\mathcal{F}_{II}|(t) = 5 \cdot 10^{-2}(t-20) + 3\sin(t-20)/(t-20)$$
$$\mathcal{F}_{II}^{\phi}(t) = 45\sin(0.17t)$$

Contrary to the previous case, this situation is not directly associated with a real one, and has been designed only to test the robustness of presented methods under a high demanding scenario in terms of fault rejection.

Joint	Complete Scheme	No Integral States	Position PID
θ_1	0.5037	34.9081	6982.1631
θ_2	2.7949	71.5806	10276.7359

Table 7.6: Mean Squared Error (MSE) over desired trajectory under Fault Scenario II for the considered control strategies.

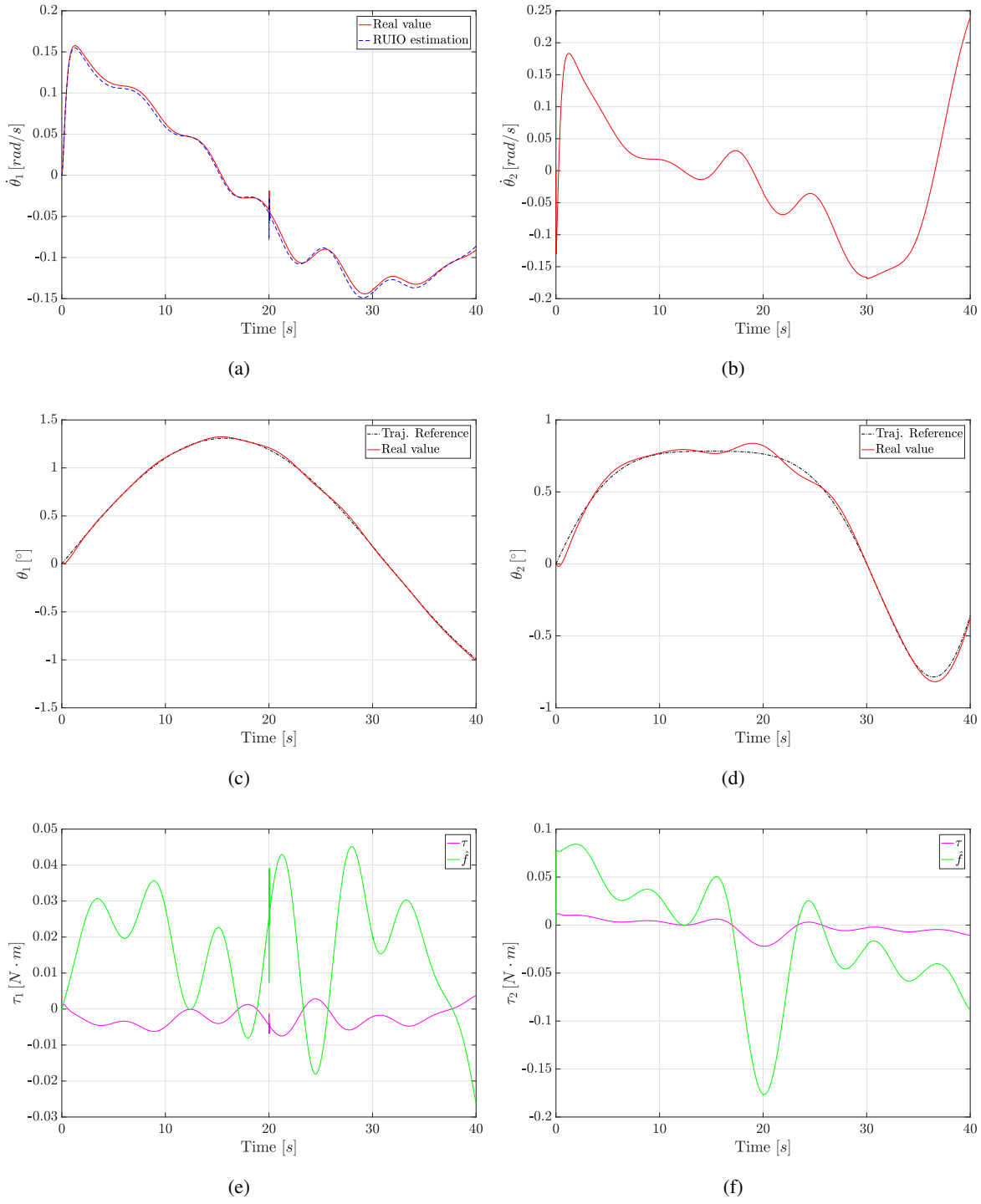


Figure 7.11: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

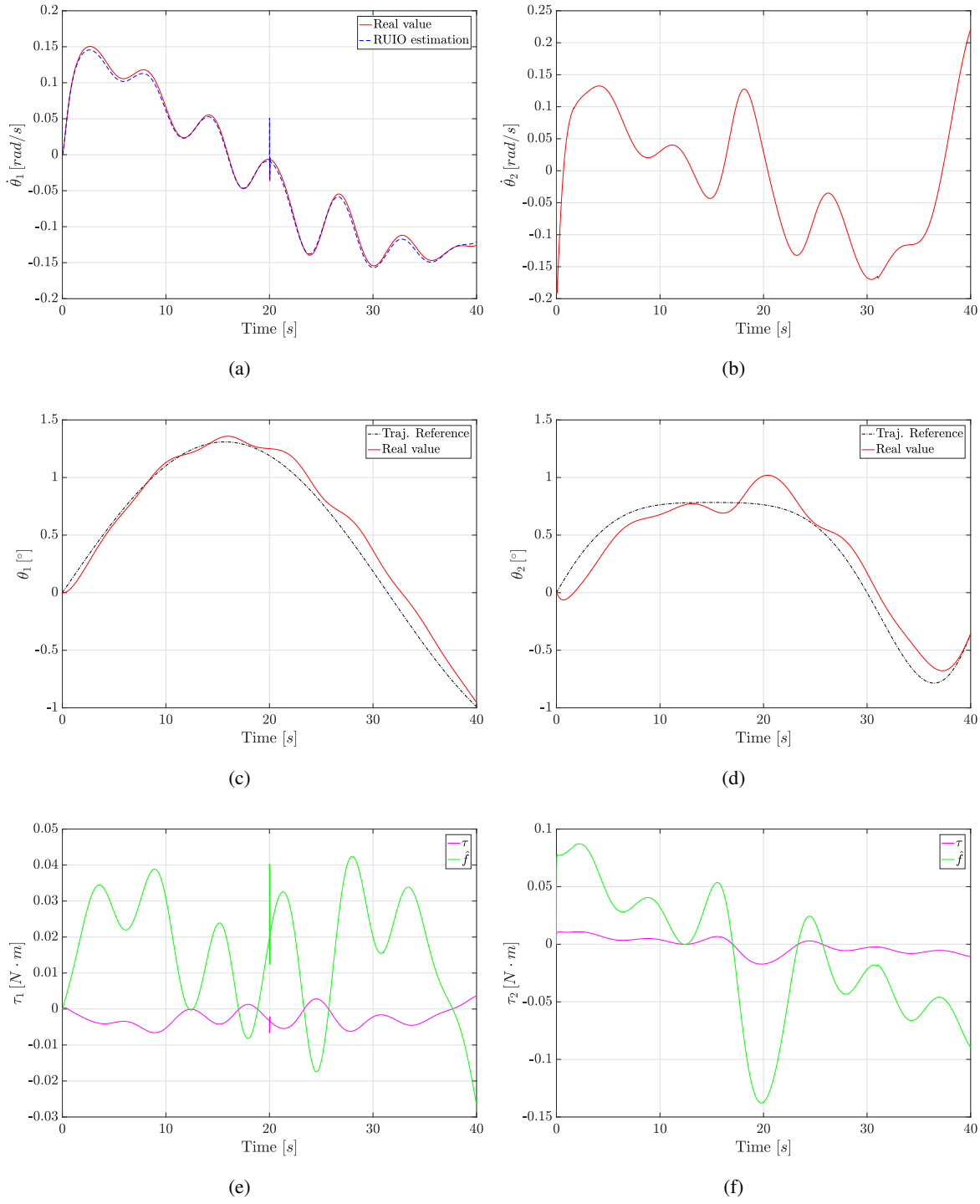


Figure 7.12: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

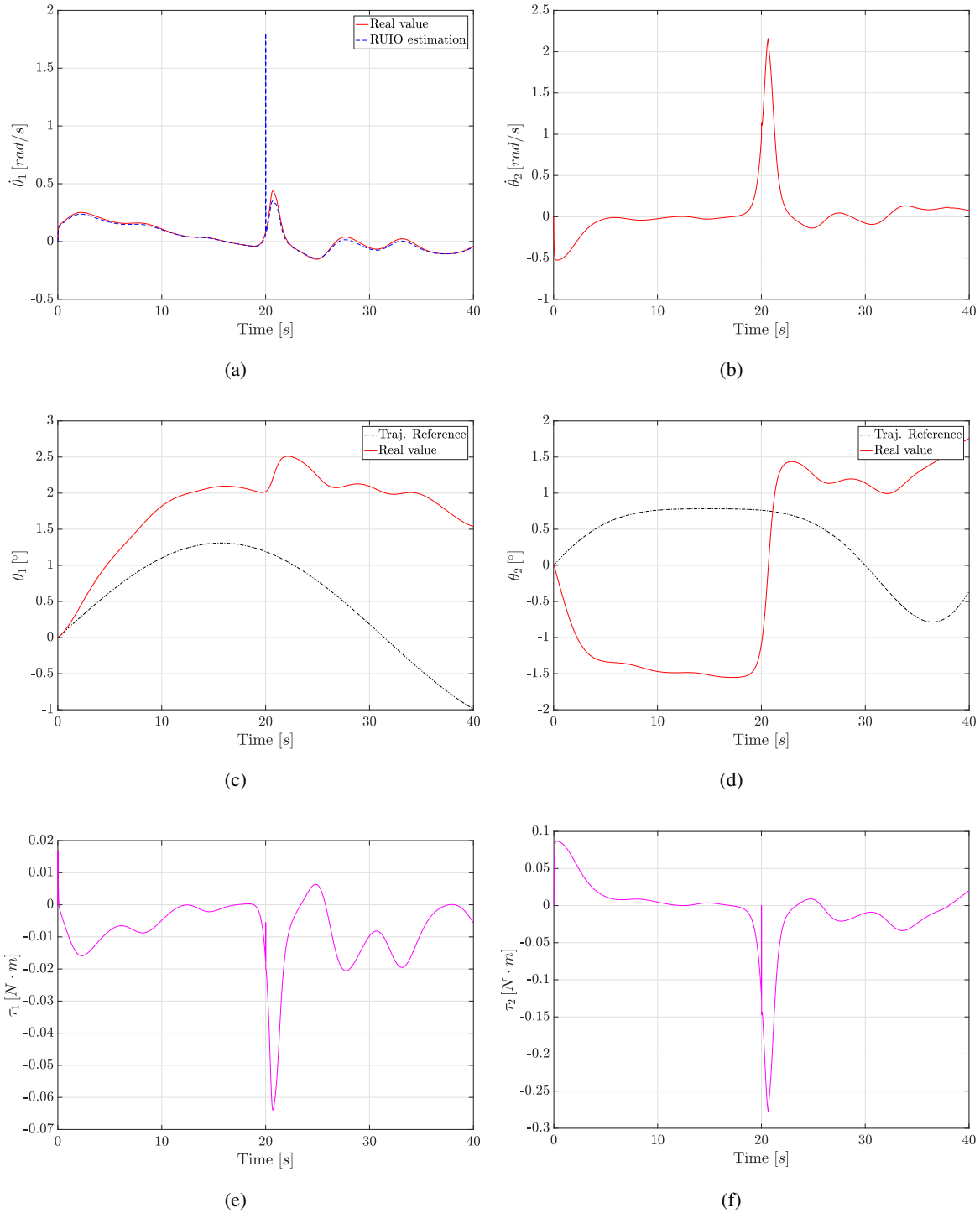
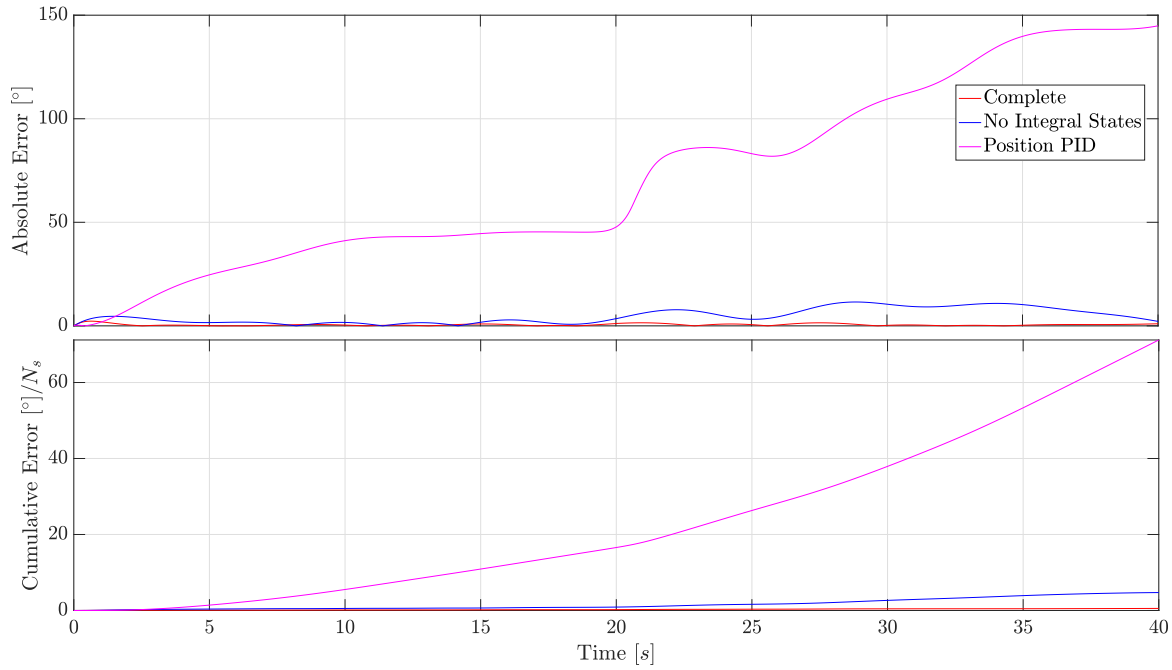
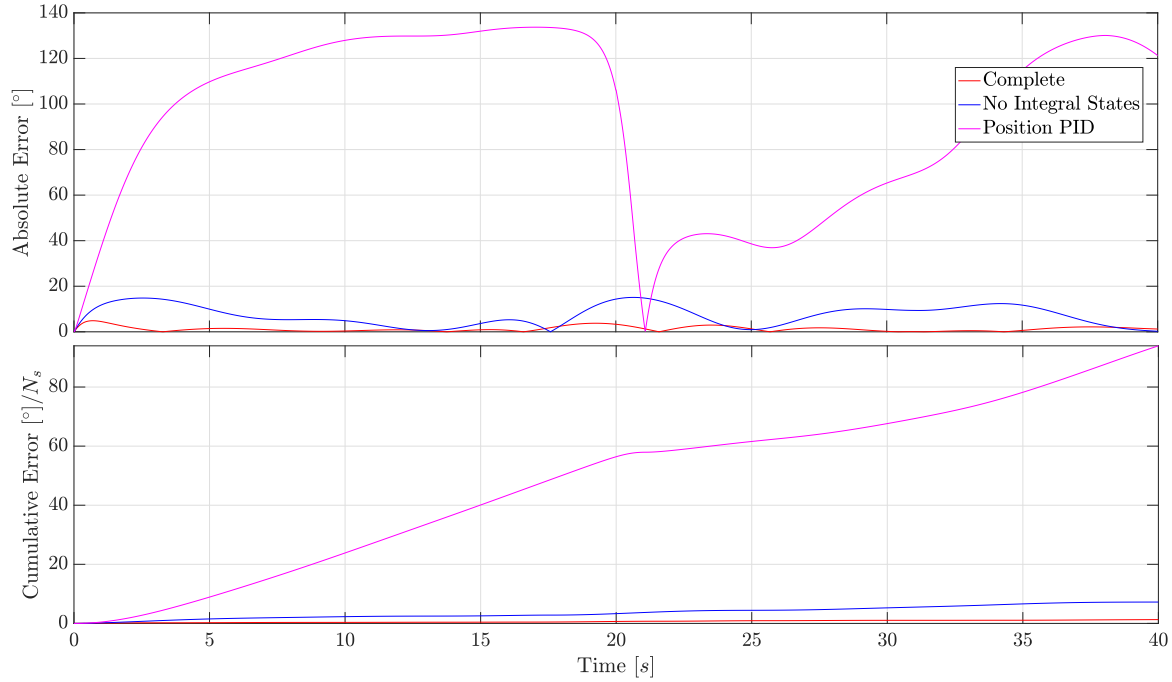


Figure 7.13: System states evolution for the complete Fault-Tolerant control scheme under Fault Scenario I.

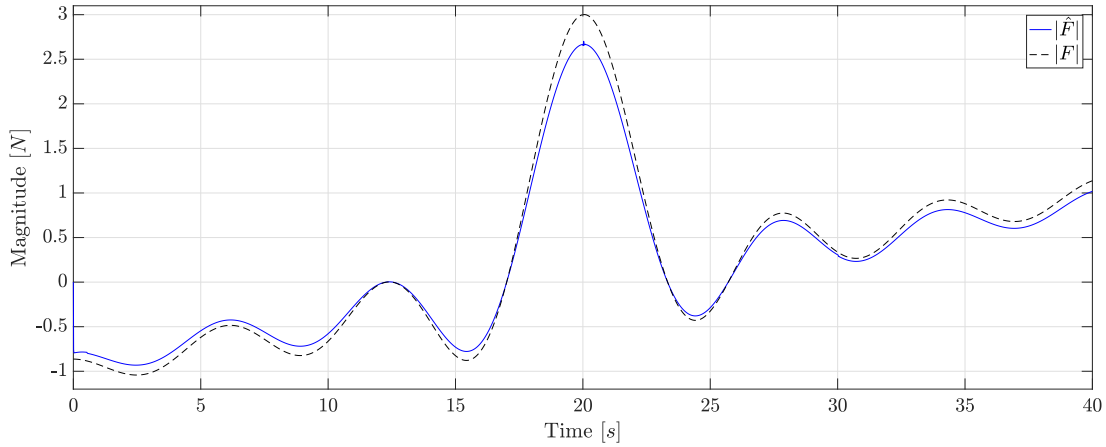


(a)

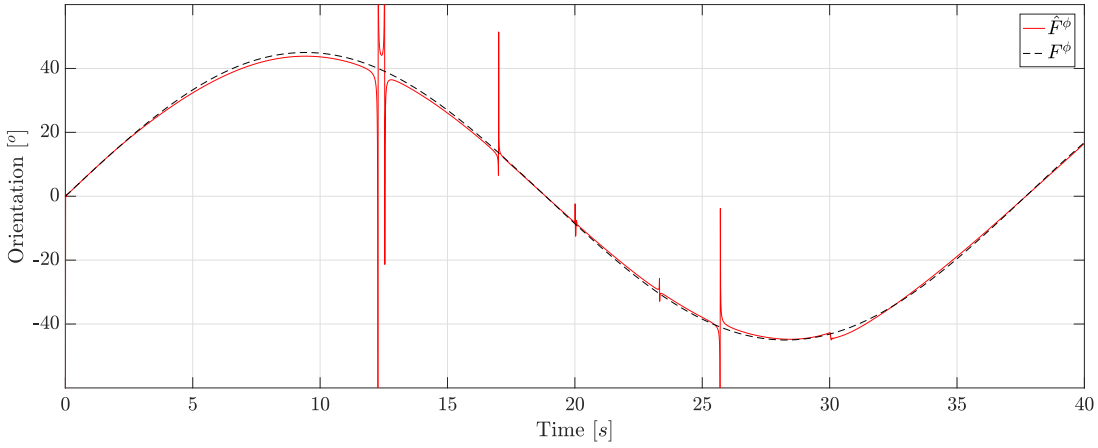


(b)

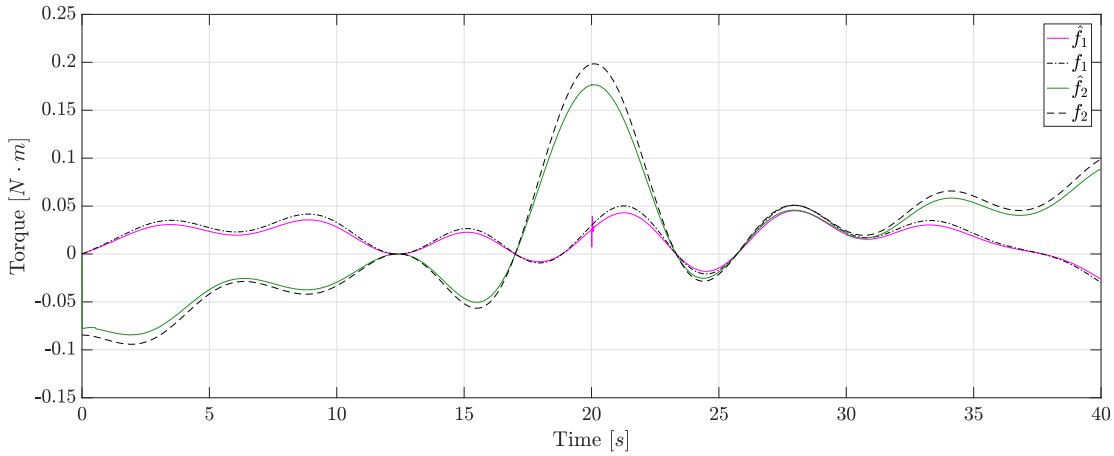
Figure 7.14: Pan (a) and Tilt (b) angles absolute error and its normalized cumulative evolution for the considered control schemes under Fault Scenario II.



(a)



(b)



(c)

Figure 7.15: Evolution of applied and estimated force (a) magnitude, (b) orientation and (c) exerted joint torques given by the complete Fault-Tolerant scheme under Fault Scenario II.

7.3.3 Discussion

On both fault scenarios, presented Fault-Tolerant Scheme outperforms the currently existing position based PID control, which present a behaviour completely driven by the actuating force on the TIAGo head subsystem. The effect of integral states helps tracking the trajectory, but does not present complete a fault rejection action, as it can be seen by comparing the control actions τ given by the PDC on the complete scheme and the one without integral augmentation.

The fault estimation captures the trend of the exerted joint torques by the exogneous force, although it does present a narrow estimation error in the magnitude of joints torques. Table 7.7 includes the maximum and mean estimation errors for both scenarios. This error mainly affects on estimating the total force magnitude, with a less evident error on th estimation of the orientation. It should be pointed out that this latter one, due to the use of inverse tangent function on the estimation, presents an asymptotic behaviour on regions close to singularities, that have not been considered as relevant, being relatively straightforward to filter them out.

Estimated torque		**Fault Scenario I**	**Fault Scenario II**
$\hat{f}_1$	Max.	–	18.08 %
	Mean.	–	9.89 %
$\hat{f}_2$	Max.	11.68 %	15.08 %
	Mean	11.11 %	10.11 %

Table 7.7: Maximum and mean estimation error (in % with respect to the true value) on exogenous exerted torques on the joints for both Fault Scenario I and II.

Regarding the RUIO-TS observer, which has been applied for all the three presented methods, it has successfully estimated the unknown velocity $\dot{\theta}_1$ disregarding the acting force in the TIAGo head subsystem. On the second fault scenario, where an exogenous torque was being exerted in both joints, although it does present a minor decoupling effect, its overall influence on the system is negligible.

Chapter 8

Parametrization of joint actuators

One of the main objectives of this work has been to bring proposed methods together with the implementation on the real platform by considering all the issues that might arise due to the physical limitations imposed by the real components. Therefore, the complete Fault-Tolerant scheme has been designed to work on a range of operational limits and according to a discrete-time approach suitable for sequential programming. To completely cover this topic, this chapter is mainly devoted to physical actuators of the TIAGo head subsystem and the introduced mechanisms to use them into the complete control scheme.

Figure 8.1: Dynamixel servomotors from ROBOTIS.

8.1 Dynamixel Servomotor Actuator

Both Pan and Tilt joints of the considered TIAGo Head subsystem use Dynamixel servomotors from ROBOTIS[7], specifically, model MX-64. This actuator series have been widely used in many robotic platforms for both industrial and research applications, due to their reconfigurability and performance, mainly in those which require a precise control over the position, e.g. small-scale humanoid robots.

As aforementioned, the provided static PID position-based control by Dynamixel servomotors

[7]Dynamixel actuator series from ROBOTIS: `http://en.robotis.com/shop_en/category.php?ca_id=20` (Accesed 22 June 2019)

has been adjusted for nominal operation conditions of the TIAGo head subsystem, and under the effect of exogenous forces does not achieve desired poses. This issue initially motivated this work, which has been built over the premise that all the methods have to be finally implemented using the current Dynamixel servomotors, to reduce costs and attempting to propose a cross solution over similar platforms that perform under *a priori* unknown exogenous (or endogenous) forces that hinder desired control.

Dynamixel MX-64 only features a "Torque Control Mode" based on the current provided to the motor, not assuring any correspondence with the exerted torque on the axis. Indeed, ROBOTIS does not give any information about the internal mechanism on this mode, and has been assumed to be a feed-forward action according to the observed behaviour. To assure a suitable reliable torque-based control over this mode, a set of mechanisms have been designed to consider the main physical effects associated to servomotors: friction, efficiency and magnetic behaviour. Firstly, in this section, a mathematical description of the servomotor and these effects is given, to be used in the empirical determination of the main parameters in the next section, that will finally determine the appropriate compensation mechanisms in the last section of this chapter.

8.1.1 Mathematical model

Dynamixel MX-64 uses MAXON DC EC-Max (ID 283840) [8] brushed motor, which can be modeled through a LR series electric circuit that includes the input voltage at the terminals of the motor u. The Counter Electro-Motive Force (CEMF) , denoted as e, opposes the current flow i as a consequence of DC motor shaft coils moving relative to the magnetic field, and can be related to its speed $\dot{\theta}_m$ through K_ω, obtaining the following description of the complete electric behaviour:

$$\begin{aligned} u &= R\,i + L\,\frac{d\,i}{dt} + e \\ &= R\,i + L\,\frac{d\,i}{dt} + \frac{\dot{\theta}_m}{K_\omega} \end{aligned}$$

The motor output torque τ_m is proportional to the current according to a constant K_τ, which can be related to K_ω assuming that the electrical power in the circuit must be equal to the mechanical power plus electrical losses:

$$K_\tau = K_\omega^{-1} \tag{8.1}$$

[8] MAXON brushless motors catalog: `https://www.maxonmotor.es/maxon/view/category/motor` (Accesed 22 June 2019)

Assuming an ideal power conservation, output acceleration of the DC motor shaft $\ddot{\theta}_m$ can be defined according to its moment of inertia J_m, the output torque of the complete servomotor τ_l at the end of the gearbox defined by N_{gear}, and the associated friction torques $\tau_{fric,m}$:

$$J_m\ddot{\theta}_m = \tau_m - \frac{\tau_l}{N} - \tau_{fric,m}$$

Equivalently, output acceleration at the gearbox end axis $\ddot{\theta}_l$ can be defined:

$$J_l\ddot{\theta}_l = \tau_l - \tau_{fric,l}$$

The described servomotor electric circuit and elements have been summarized in Figure 8.2.

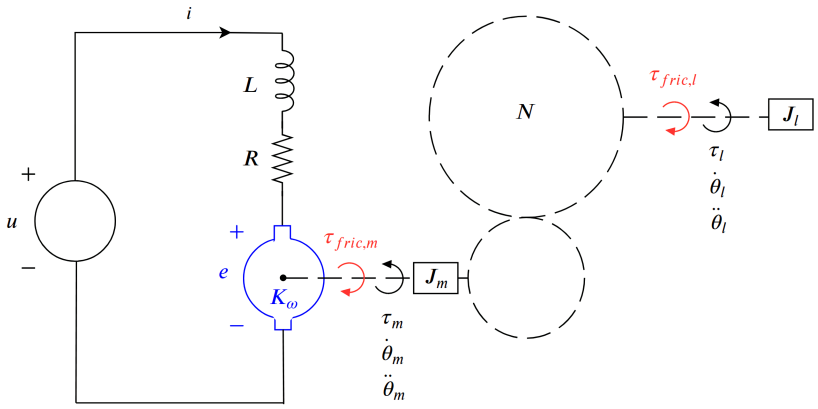


Figure 8.2: Schematic representation of Dynamixel servomotor internal behaviour.

As this work is not aimed at obtaining a complete description of the model, presented mathematical model have been used to determine a simplified version where the main effects are enclosed under global parameters. Considering that "Torque Control Mode" allows to give current set-points for the motor i_{set} with a resolution of $4.5\,[mA]$, it can be directly associated to the output torque fo the servomotor through a global K_τ^*:

$$\tau_l = K_\tau^*\, i_{set} \tag{8.2}$$

All the friction terms are enclosed under τ_{fric}, and K_ω^* is obtained as in Equation (8.1). Inductance effect can be neglected as according to the specifications $L = 0.488\,[mH]$ with respect to $R = 12.4\,[\Omega]$, which allows to obtain the associated current to the CEMF:

$$i_{CEMF} = \frac{\dot{\theta}_l}{K_\omega^*\, R}$$

Therefore, the following equation defines the output acceleration for the simplified model including the current command:

$$J_l\ddot{\theta}_l = \tau_l - \tau_{CEMF} - \tau_{fric} \tag{8.3a}$$

$$= K_\tau^* \left(i_{set} - \frac{\dot{\theta}_l}{R} \right) - \tau_{fric} \tag{8.3b}$$

The benefits of this simplified description of the overall servomotor behaviour are twofold: (1) defines the output torque in terms of the desired current set-point, and (2) allows to embed efficiency and performance discontinuities by defining K_τ^* dependent on the operation point. On DC Motors, efficiency is non-linearly defined according to the operation range, and also the linear relation from Equation (8.2) can be an strong assumption, primarily at low torques. Thus, by empirically determine a mapping for K_τ^* to relate output torque of the servomotor and current set-points allows to overcome both issues at once. The experimental determination of this parameter will be described in the following section

8.1.2 Friction effects

Friction phenomena are widely encountered in most electro-mechanical systems, and are the result of a combination of multiple complex physical effects. For robotic joints the associated control problems have been widely studied and generally, compensation of their effects is based on a estimation given by a empirically determined model, e.g. as presented in Reference [6]. In the literature, a frequently-used model to describe joint friction was given by Stribeck [2] from tribology field of study, whose parameters are obtained according to experimental data. This model combines observed behaviours into a single description, which defines three differentiated ranges of friction: (1) an initial force constraint equal to the Coulomb friction $\tau_{fric,c}$, associated to the contact between surfaces, the Static Friction $\tau_{fric,s}$, associated to an initial rest state, (2) an exponential decreasing of the total friction defined according to the so-called Stribeck-velocity $\dot{\theta}_s$, and (3) friction region where total value corresponds to the Coulomb component and the Viscous one $\tau_{fric,v}$, which linearly depends on angular velocity. Due to its generalized used and the existing literature about the Stribeck model it has been chosen as an initial approach to characterize the friction phenomenon in the Dynamixel servomotor. Experimental determination of the required parameter has been described in the following subsection.

Regarding gearbox transmissions, literature reflects a change on the friction behaviour according to the load applied at its end by modifying the Coulomb component [12]. In the case of the Dynamixel MX-64 servo motor, a value of $N = 200$ suggests that this effect might not be negligible, and therefore

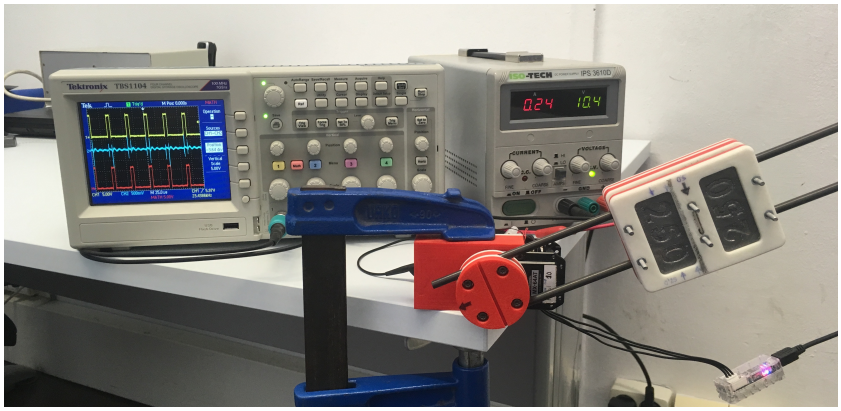


Figure 8.3: Designed Test-Bench platform, consisting on a pendulum-mass structure attached to a Dynamixel servomotor.

will be addressed on the following experimental section.

8.2 Behaviour characterization and identification of parameters

To perform the experiments, a pendulum test-bench platform (Figure 8.3) has been designed, using Dynamixel MX-64, which will be also used to validate the applicability of the complete Fault-Tolerant control scheme with these actuators. This allowed to test both the behaviours of Pan and Tilt joints for different masses and distances w.r.t the joint axis.

The first experiment aimed at obtaining the aforementioned mapping between the current set-points and the exerted torque, defined under K_τ^*. Therefore, using the pendulum structure, different loads where applied at certain distances to obtain multiple samples for the same torque, considering the minimum required set-point to generate the corresponding torque that stabilizes the mass. This experiment mapping has been performed for torques within $\pm 3\,[N \cdot m]$, which has been considered as the maximum torque (considering control action and Active Compensation of exogenous forces and gravity) that the servomotor will exert. Figure 8.4 shows obtained samples, which have been fitted to a 5-th degree polynomial function, also depicted.
As it can be seen, a linear relation can not be assumed in this case for low torques. This result is quite relevant for the PDC, which in this work has been designed to perform on low control actions in order to fulfill the imposed optimality objectives.

Also as a consequence of the obtained design for the PDC, the range of reached maximum angular speeds has been observed to be relatively low ($\in \pm 0.1\,[rad/s]$) for given reference trajectories. Thus, as a first approximation to the friction problem, Viscous term has been neglected, focusing on both

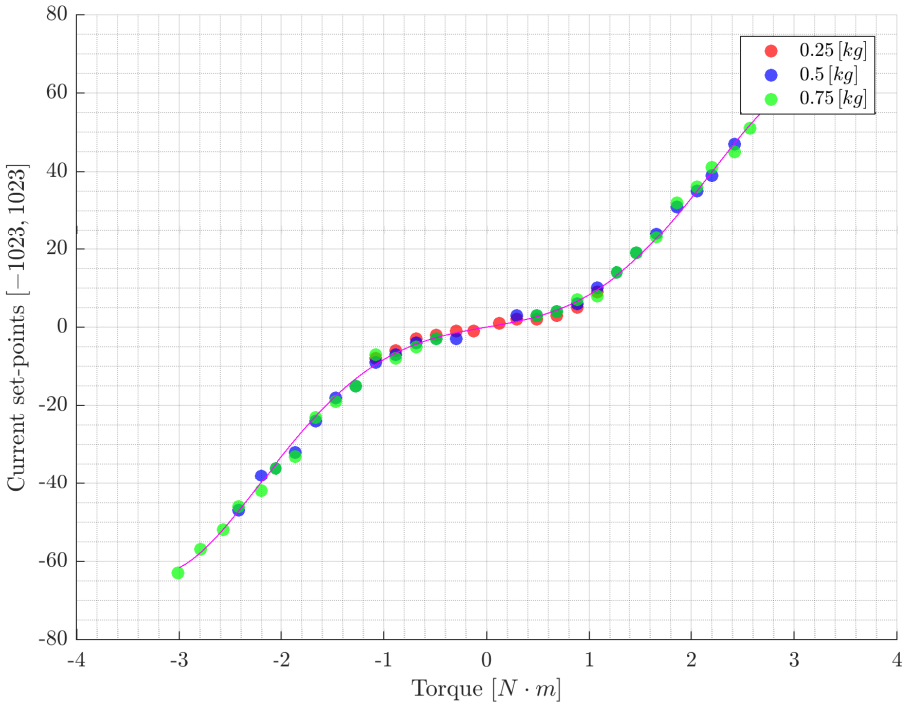


Figure 8.4: Experimental mapping samples between torque values given by the servomotor and correspondent current set-points, under different masses, to determine K_τ^*.

Static and Coulomb effects.

Static Friction, namely *stiction*, is defined as the existing force between two non-moving surfaces that opposes to the motion, additionally to the Coulomb one which depends on the normal component and a coefficient of friction defined for the surfaces in contact. For a joint, this implies that an *extra* initial torque has to be exerted in order to initiate the angular movement on the axis. Therefore, it has been considered as the associated current set-point for the axis without any load being applied. Using K_τ^*:

$$\tau_{fric,s} \simeq 0.2\ [N \cdot m]$$

Regarding Coulomb Friction, the magnitude for Pan and Tilt angles have been considered be different, due to the effect of the gearbox transmission. In the Pan joint, no load is applied on the axis during its movement except for the forces that might appear due to the mass rotation around the axis, that are already included by the model-based control structure. But for the Tilt joint, gravitational forces from the mass are continuously being applied on the axis, mainly affecting when the control

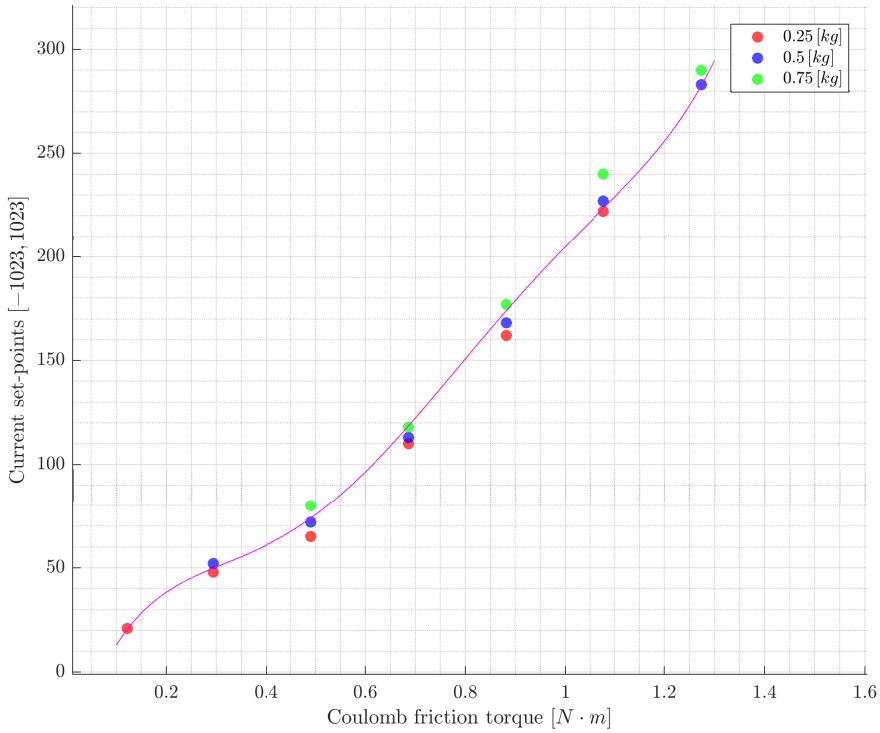


Figure 8.5: Experimental mapping samples between torques and correspondent current set-points, under different masses, to determine Coulomb friction effects .

action is opposite to it. Also, as the mass will modify its position during operation, and thus changing its effective distance to the axis of rotation, this effect will be dependent on the gravity torque exerted at each position. Therefore, a second experiment has been performed to determine a mapping between current set-points and applied torques similarly to the one for determining K_τ^*, but in this case considering the minimum value that initiates the movement over different loads, as the value of $\tau_{fric,s}$ is known. Figure 8.5 includes the obtained samples, which have been also approximated by a 5-th degree polynomial. In this case only a mapping for the range $[0, 1.3]$ has been considered, recalling that the maximum nominal torque exerted by the mass in the TIAGo head subsystem is 0.64 $[N \cdot m]$, and also to avoid damaging the motor by inducing high currents at stall.

8.3 Torque-based control mechanism

On the assumptions made regarding the servomotor behaviour, defined in Equations (8.3), and characterizing CEMF and friction effects, an output torque can be generated by actively compensating these effects and defining a current set-point associated to a desired torque given by the control strategy. Figure 8.6 summarizes this mechanism and its integration within the complete Fault-Tolerant control (from which it receives the compensated control torque τ'), depicted in Figure 6.1.

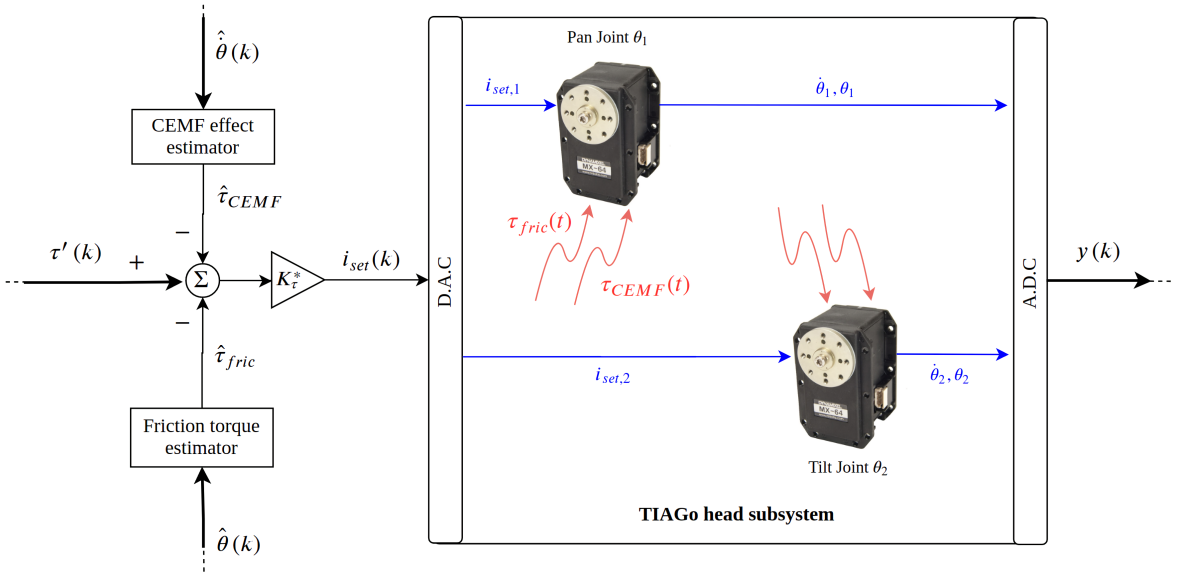


Figure 8.6: Integration of the servomotor effects estimation and compensation mechanisms.

8.4 Test-Bench Results

The dynamical model for the pendulum test-bench can be directly obtained for both joints from the already complete description given in Equations (3.2), where the Coriolis and centrifugal terms are null, being the only difference between each joint the gravity term, that is only present for Tilt angle. In order to obtain results closer to the TIAGo head subsystem, a mass of $0.5\,[kg]$ has been placed at the corresponding distance such that the torque for each joint is equivalent to the one produced by the second link mass.

The PDC has been designed using the LQR method for the same weighted Bryson's rule and $\mathbb{D}$-stabilization criterion. The RUIO-TS structure not been implemented for this cases, as solution existence condition from Equation (5.3) does not hold for an single-joint scenario where angular

velocity is not measurable.

Communication with Dynamixel MX-64 motor has been also done through the MATLAB programming environment, using the available DYNAMIXEL SDK library. The servomotor is connected to the computer using U2D2 device by TTL serial communication, also provided by ROBOTIS. The discretization time $T_s = 10\,[ms]$ has been determined according to the code execution and communication times required, such that is greater than the total consumed time in the *worst-case scenario*. Larger execution times are associated with reading and writing operations with the Dynamixel, being the total time of execution per cycle of approx. $0.5\,[ms]$, using MATLAB 2016B and the library communication functions for Protocol 1.0. Idle time has been added on purpose in order to cope with other executions to be included in the final TIAGo head subsystem implementation, e.g. the RUIO-TS computation.

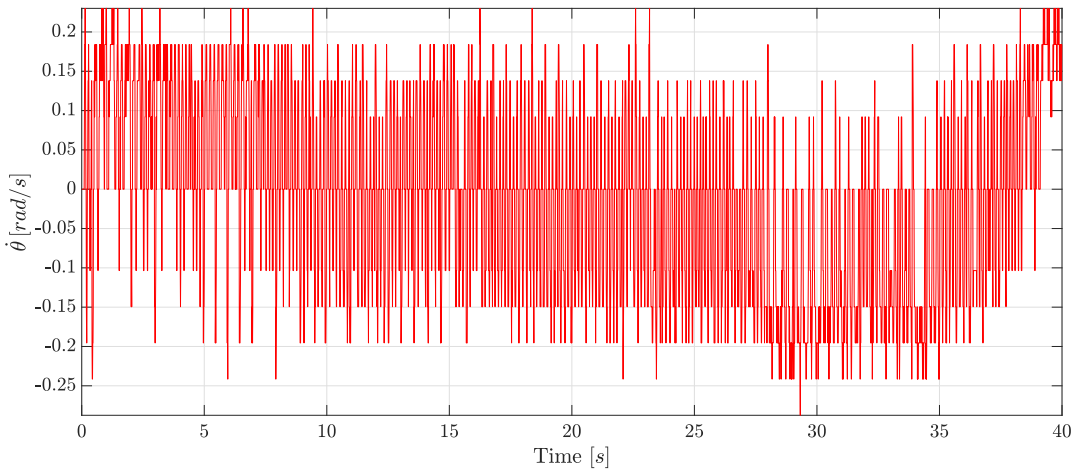
8.4.1 Pan joint

Figure 8.7 shows the evolution of the states and given current set-points of the pendulum test-bench in a planar configuration, in order to assemble the Pan joint. The reference trajectory for θ_2 presented in section 7.3 has been used.

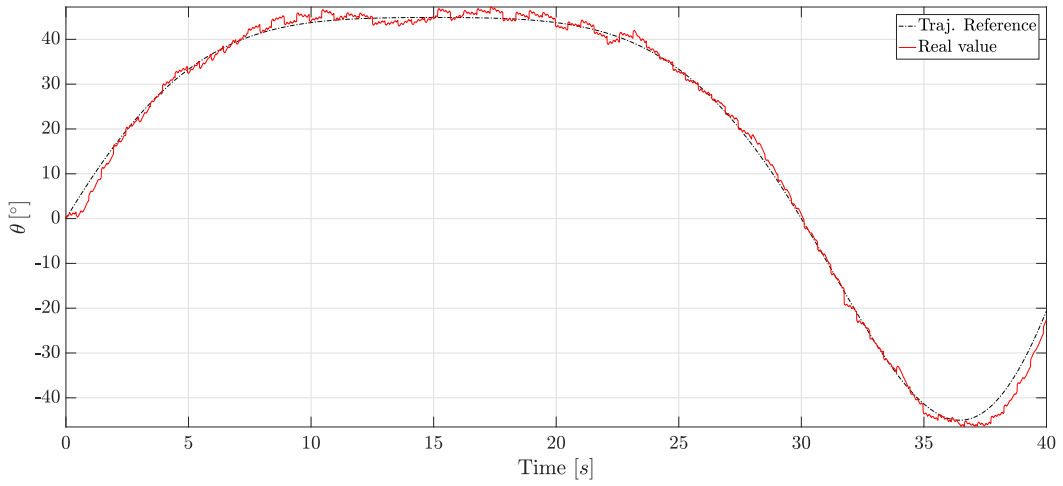
These results show that the overall tracking of the desired trajectory is achieved, although there exists a continuous *chattering* effect that approx. corresponds to a band of $\pm 2\,[^\circ]$ w.r.t. reference positions. This phenomenon might be caused by the noisy velocity measurement, misleading the control actions given by the PDC. A frequential analysis of this signal has been performed to design an appropiate low-pass filter which highly reduced the noise, but this reduction of noise implied a trade-off on the measurement accuracy, increasing *chattering* error band. Another reason behind this effect might be that the accuracy given by the "Torque Control Mode" at the chosen T_s does not offer a precise enough control over the torque to track trajectories, leading to an oscillation at low current set-point values.

8.4.2 Tilt joint

For this second case, no successful results have been achieved, obtaining high oscillating behaviours that surpass the maximum admissible position values. Disregarding all the aforementioned reasons presented for the Pan Joint case, (that can be also associated to this situation), in this case the active compensation mechanism of the Coulomb friction might be the main reason behind. Having an asymmetric behaviour according to the direction of motion might hinder the compensation on tracking a reference, which requires a precise control at low switching torques around zero.



(a)



(b)

Figure 8.7: Angular velocity (a) and position (b) evolution for the complete Fault-Tolerant control scheme applied to a the pendulum Test-Bench in a Pan Joint disposition.

Chapter 9

Socioeconomic impact

The purpose of a service robot can be defined as "either aiding or performing actions that contribute towards improvement of the quality of life of an individual" [16]. The development of robots for domestic domains is mainly focused on delivering a reduction on fatigue and stress in order to enhance daily welfare. This is particularly significant for disable and elder people that usually require a high-cost human expert attention.

For robots intended to perform physical interactions with people, like the TIAGo robotic platform, assuring safety and dependability are key issues in order to introduce them into environments shared with humans. The use of information from cognitive interactions (e.g. Natural Language Processing) and task context might play an important role on it, but main concerns are related with collision avoidance, detection and reaction, as anthropic domains are assumed to be dynamic, unstructured and partially unknown. This work proposes a control structure that fulfills a set of necessary operational constraints to assure its safe response in the eventuality of a-priori unknown exogenous force under a lack of the necessary measurements. Although direct contacts with a human have not been specifically addressed, the presented approach might be implemented within a complete physical Human-Robot Interaction architecture that deals with collisions. Some research initiatives directly aim at this challenge, e.g. european projects PHRIENDS and PRHIDOM [1] study and develop key components of the next generation of robots, including industrial ones, to share the environment and interact with people.

Recent advances in Artificial Intelligence (AI) and Computer Vision (CV), have put the spotlight on robotics and the possible impact that their application will imply in future's society and economy. With respect to everyday life environments, assistive robotics must face a set of challenges depending on the task they are intended to perform, requiring a high acceptance degree of this technology by the users. Nowadays, robots are being introduced at all education stages such that future generations are

aware of their capabilities, which will highly enhance those tasks where both human and robot have to achieve a common goal.

Regarding the realm of industrial applications, there exist a generalized fear that the increasing introduction of robotics might result on a downward pressure on the wages, mainly of low-skilled workers, and an increasing return to owners of capital [27], leading to a spiral of augmenting inequality. Automation does indeed substitutes jobs, as is intended to do, but there is an ample evidence that it allows a re-allocation of manpower on both jobs and tasks that require high levels of empathy, creativity and decision-making that are unlikely to be automatable in the near future. Also, robots can be used to complement and augment human labour on repetitive and hazardous tasks, and their implementation will require related jobs to supervise and manage them. Considering a wider scenario, the associated increase of productivity will enable some firms to restructure their supply change, bringing back parts of the manufacturing processes to their country of origin, known as *reshoring*, which will generate advantages at the regional level, improving national competitiveness. McKinsey Global Institute predicts that up to half the total productivity growth needed to ensure a constant rise in GDP over the next 50 years will be driven by automation [21].

Chapter 10

Project Budget

This chapter includes a basic cost analysis associated to the development of this work. The implementation in the TIAGo robot is assumed to have null cost (disregarding required technical support), as it will use currently existing hardware and software on the platform. Total cost is composed of:

- Computer hardware and related software, including academic license for MATLAB programming environment, LMI and BMI solvers: $C_{\text{Dev.}} = 4000$ €
- Test-bench platform, including its design, materials, and use of equipment, and the Dynamixel MX-64 servomotor and U2D2 device: $C_{\text{Test}} = 600$ €
- Manpower and technical support, considering an average price of 25 [€/hour]: $C_{\text{Test}} = 21250$ €
- General costs, such as water, electricity and others: $C_{\text{Gen.}} = 1000$ €

Therefore, the estimated total cost of the project, obtained as a sum of all these terms, ascends to **twenty-six thousand eight hundred and fifty euros (26850 €)**. The TIAGo robot is already available in this case, having a cost of around 60000 € for the complete model. If we consider the implementation of the developed work in 100 units, the price per unit will only rise the overall price by a 0.4%, which does not mean a significant impact.

Lastly, it is important underlying that a much precise cost estimation should be carried out in the future, also considering the possible implementation of the developed techniques in other systems.

Chapter 11

Concluding remarks

The designed complete Fault-Tolerant scheme has been proved to successfully tackle the supervision and control problem associated to a robotic manipulator with *a priori* unknown exogenous forces acting on it, presenting a significant improvement in the addressed TIAGo head subsystem with respect to the previous static PID-based control.

A complete description of the system has been obtained using the Takagi-Sugeno framework, consisting on embedding the non-linear behaviour into a representation based on its operational limits, confining all the possible behaviour between them. From this representation, a state-feedback Parallel Distributed Controller has been designed using the same concept, which represents a clear advantage in order to assure global conditions for the system in terms of stability and transient response, which has been done using the Linear (and Bilinear) Matrix Inequalities formulation.

Up to our knowledge, related previous works do not consider an incomplete information model along with the fault occurrence, being addressed in this Master Thesis for the first time. Therefore, the Robust Unknown Input observer has been implemented within the complete Fault-Tolerant scheme, such that a decoupled estimation is obtained for the unknown velocity of the Pan Joint of the TIAGo head subsystem, disregarding the exogenous force behaviour nor its eventuality.

Regarding fault detection and isolation, a Reference Control structure is implemented in parallel to the main scheme, which allows to estimate the magnitude of exerted torques by the exogenous force in each joint. Using this estimation, an Active Compensation mechanism allows to counter act its effect in order to assure that the desired poses are achieved.

Finally, a characterization of the main phenomena in the currently used joint actuators has been performed for both Joint and Tilt angles, introducing the compensation mechanism that make them suitable for the torque-based implementation.

11.1 Future work

After the obtained results and development of this work, some ideas have arose that might lead to future research opportunities:

- Perform an in-depth characterization of the all parts within the servomotor, evaluating and overcoming the physical limitations that might imply a torque-based control, and/or explore other approaches or similar actuators trying to reduce as much as possible the cost impact on the overall project budget.
- Regarding the variant B matrix issue, explore different approaches to relax design conditions without implying the inclusion of additional structures to the model, e.g. using an approach based on Polya's Theorem .
- To bring even more closer design and implementation on the real platform by incorporating in the dynamical model some effects seen on the real joint actuators, e.g. friction, such that are also under the overall design criteria.
- Include within the TS Model context variables into the premise set z, given by upper planning layers, obtaining a control approach that relates to the context of the task to be performed.
- In case of extending this approach to scenarios where the force exerted on the manipulator represents a collision with a human: (1) analyze how the gathered information might be used to generate suitable reactions, (2) include within the optimality criterion objectives that assure minimal injuries to the human (e.g. according to the Head Injury Criterion) in the event of an unintended contact, and (3) evaluate the possible use of information from exteroceptive mechanisms able to foresee collisions (e.g. human and robot pose tracking).

11.2 Publications

During the development of this Master Thesis, the following publications related to the presented work have been made:

- Alberto San Miguel, Vicenç Puig and Guillem Alenyà. *Chapter 2: Fault-tolerant Control of a Service Robot* (Submitted), a contribution for the book entitled *Fault Diagnosis and Fault-tolerant Control of Robotic Systems*, Institute of Engineering and Technology (IET) .

- Alberto San Miguel, Vicenç Puig and Guillem Alenyà. *Fault-tolerant Control of a Service Robot using a LPV Robust Unknown Input Observer* (Accepted). 4^{th} IEEE International Conference on Control and Fault-Tolerant Systems, September 2019, Casablanca, Morocco.

Bibliography

[1] ALAMI, R., ALBU-SCHÄFFER, A., BICCHI, A., BISCHOFF, R., CHATILA, R., DE LUCA, A., DE SANTIS, A., GIRALT, G., GUIOCHET, J., HIRZINGER, G., ET AL. Safe and dependable physical human-robot interaction in anthropic domains: State of the art and challenges. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2006), IEEE, pp. 1–16.

[2] ANDERSSON, S., SÖDERBERG, A., AND BJÖRKLUND, S. Friction models for sliding dry, boundary and mixed lubricated contacts. *Tribology international 40*, 4 (2007), 580–587.

[3] APKARIAN, P., GAHINET, P., AND BECKER, G. Self-scheduled h-∞ control of linear parameter-varying systems: a design example. *Automatica 31*, 9 (1995), 1251–1261.

[4] CHADLI, M. An lmi approach to design observer for unknown inputs takagi-sugeno fuzzy models. *Asian Journal of Control 12*, 4 (2010), 524–530.

[5] CHADLI, M., AND KARIMI, H. R. Robust observer design for unknown inputs takagi–sugeno models. *IEEE Transactions on Fuzzy Systems 21*, 1 (2012), 158–164.

[6] COLOMÉ, A., PLANELLS, A., AND TORRAS, C. A friction-model-based framework for reinforcement learning of robotic tasks in non-rigid environments. In *2015 IEEE international conference on robotics and automation (ICRA)* (2015), IEEE, pp. 5649–5654.

[7] CORKE, P. I. A simple and systematic approach to assigning denavit–hartenberg parameters. *IEEE transactions on robotics 23*, 3 (2007), 590–594.

[8] CRAIG, J. J. *Introduction to robotics: mechanics and control, 3/E*. Pearson Education India, 2009.

[9] DAROUACH, M., ZASADZINSKI, M., AND XU, S. J. Full-order observers for linear systems with unknown inputs. *IEEE transactions on automatic control 39*, 3 (1994), 606–609.

[10] DE LUCA, A., AND FERRAJOLI, L. A modified newton-euler method for dynamic computations in robot fault detection and control. In *2009 IEEE International Conference on Robotics and Automation* (2009), IEEE, pp. 3359–3364.

[11] DE LUCA, A., AND FLACCO, F. Integrated control for phri: Collision avoidance, detection, reaction and collaboration. In *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)* (2012), IEEE, pp. 288–295.

[12] DOHRING, M. E., LEE, E., AND NEWMAN, W. S. A load-dependent transmission friction model: theory and experiments. In *[1993] Proceedings IEEE International Conference on Robotics and Automation* (1993), IEEE, pp. 430–436.

[13] DUAN, G.-R., AND YU, H.-H. *LMIs in control systems: analysis, design and applications*. CRC press, 2013.

[14] FRANKLIN, G. F., POWELL, J. D., WORKMAN, M. L., ET AL. *Digital control of dynamic systems*, vol. 3. Addison-wesley Menlo Park, CA, 1998.

[15] HADDADIN, S., ALBU-SCHAFFER, A., DE LUCA, A., AND HIRZINGER, G. Collision detection and reaction: A contribution to safe physical human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2008), IEEE, pp. 3356–3363.

[16] HARPER, C., AND VIRK, G. Towards the development of international safety standards for human robot interaction. *International Journal of Social Robotics 2*, 3 (2010), 229–234.

[17] HESPANHA, J. P. *Linear systems theory*. Princeton university press, 2018.

[18] HORNUNG, R., URBANEK, H., KLODMANN, J., OSENDORFER, C., AND VAN DER SMAGT, P. Model-free robot anomaly detection. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2014), IEEE, pp. 3676–3683.

[19] J., G. *Fault detection and diagnosis in engineering systems*. 1998.

[20] KHALASTCHI, E., AND KALECH, M. On fault detection and diagnosis in robotic systems. *ACM Computing Surveys (CSUR) 51*, 1 (2018), 9.

[21] MANYIKA, J. A future that works: Ai, automation, employment, and productivity. *McKinsey Global Institute Research, Tech. Rep* (2017).

[22] MONTAGNER, V. F., LEITE, V. J., AND PERES, P. L. Discrete-time switched systems: Pole location and structural constrained control. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)* (2003), vol. 6, IEEE, pp. 6242–6247.

[23] OF ROBOTICS, I. F., Ed. *Executive Summary World Robotics Service Robot*. 2018.

[24] PATTON, R., CHEN, L., AND KLINKHIEO, S. An lpv pole-placement approach to friction compensation as an ftc problem. *International Journal of Applied Mathematics and Computer Science 22*, 1 (2012), 149–160.

[25] PATTON, R. J., AND KLINKHIEO, S. Lpv fault estimation and ftc of a two-link manipulator. In *Proceedings of the 2010 American Control Conference* (2010), IEEE, pp. 4647–4652.

[26] PRASSLER, E., AND KOSUGE, K. Domestic robotics. *Springer handbook of robotics* (2008), 1253–1281.

[27] SACHS, J. D., AND KOTLIKOFF, L. J. Smart machines and long-term misery. Tech. rep., National Bureau of Economic Research, 2012.

[28] SCHULTE, H., AND GUELTON, K. Modelling and simulation of two-link robot manipulators based on takagi sugeno fuzzy descriptor systems. In *2006 IEEE International Conference on Industrial Technology* (2006), IEEE, pp. 2692–2697.

[29] STAVROU, D., ELIADES, D. G., PANAYIOTOU, C. G., AND POLYCARPOU, M. M. Fault detection for service mobile robots using model-based method. *Autonomous Robots 40*, 2 (2016), 383–394.

[30] SUGENO, M., AND KANG, G. Fuzzy modelling and control of multilayer incinerator. *Fuzzy sets and systems 18*, 3 (1986), 329–345.

[31] TAKAGI, T., AND SUGENO, M. Fuzzy identification of systems and its applications to modeling and control. In *Readings in Fuzzy Sets for Intelligent Systems*. Elsevier, 1993, pp. 387–403.

[32] VAN, M., AND KANG, H.-J. Robust fault-tolerant control for uncertain robot manipulators based on adaptive quasi-continuous high-order sliding mode and neural network. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 229*, 8 (2015), 1425–1446.

[33] VANANTWERP, J. G., AND BRAATZ, R. D. A tutorial on linear and bilinear matrix inequalities. *Journal of process control 10*, 4 (2000), 363–385.

[34] ZADEH, L. A., KLIR, G. J., AND YUAN, B. *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers*, vol. 6. World Scientific, 1996.

[35] ZHUO-HUA, D., ZI-XING, C., AND JIN-XIA, Y. Fault diagnosis and fault tolerant control for wheeled mobile robots under unknown environments: A survey. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* (2005), IEEE, pp. 3428–3433.