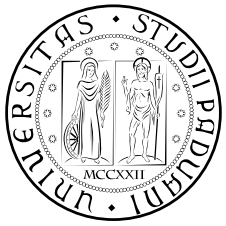


UNIVERSITÀ DEGLI STUDI DI PADOVA

DEPARTMENT OF MANAGMENT AND ENGINEERING
MASTER OF SCIENCE IN MECHATRONIC ENGINEERING

MASTER THESIS

ADVANCED ANTI-WINDUP TECHNIQUES FOR THE LIMITATION OF THE EFFECTS OF THE ACTUATOR SATURATION

Supervisor: Prof. Roberto Oboe

Co-supervisor: Prof. Jan Swevers

External examiner: Prof. Alessandro Beghi

Student: Daniele Ronzani
1132455

ACADEMIC YEAR: 2017-18

ABSTRACT

In this thesis an industrial issue given by a company is analysed. The issue consist on the undesirable effect of actuator sturation. Two approaches are given to solve the issue: a simple and accurate inertia identification algorithm based on the DFT coefficient calculation at one frequency; and advanced anti-windup compensators.

In particular, the principle to understand and to design the so-called modern anti-windup techniques DLAW and MRAW (an LMI-based approach design), and a systematic design procedure for the observer-based anti-windup are given.

In addition the solutions are tested and validated by simulations and tests on the experimental setup.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

During my thesis research I collaborated with Motion Estimation Control and Optimization Research Team (MECO) Research Team [29] at KU Leuven on a research project that aims to solve an industrial problem given by a partner company: LAB Motion Systems[22]. This company is specialized in the development of machines that require advanced motion performance. In particular, they are expert on high-precision air bearing technology and high- performance direct drive.
The issue encountered by LAB company regards a direct drive rotary table. This machine is used for indexing and scanning operations (i.e. point-to-point motion) in several applications such as computer tomography, laser machining, optical inspection, measurement systems and so on. The machine is sold to the customers together with a drivebox motion controller, which implements the current control loop of the direct drive motor and the position and velocity control loop of the positioning system. The motion controller is usually tuned by LAB to satisfy the costumer specifications. Unfortunately, for certain point-to-point operation undesired position oscillation is seen by costumers at the end of the motion.
The workflow of my study has been composed bu the following steps:

- analyse the system;
- identify the source of the undesirable behaviour;
- literature review and study of a possible solution;
- develop a simulator of the system;
- implementation, simulation and test of a solution for the problem.

In the following analysis will emerge that the undesirable behaviour is caused by a nonlinearity in the system, an input saturation of the direct drive motor. In the commercial control software, running in the drivebox motion controller, is already implemented ad hoc anti-windup algorithm to compensate the undesirable effect caused by the saturation. However, that anti-windup compensator is not that effective and the performance are too low for the customers applications.

The aim of this thesis is to analyse and demonstrate the effectiveness of the so-called modern anti-windup techniques compared to the classical anti-windup and the ad hoc solution already implemented

in the system.

The thesis is organized as follow:

- in Chapter 2 the main parts of the system are described;
- in Chapter 3 the system is analysed to find the source of the undesirable oscillations and to propose a possible solution to the problem;
- in Chapter 4 a literature review of the actuator saturation problem is presented in order to seek a solution suitable for this project;
- in Chapter 5 the classical anti-windup, OBSAW, and the modern anti-windup, DLAW and MRAW, are further discussed and studied;
- in Chapter 6 a simulator of the system is developed, an inertia identification algorithm and different anti-windup techniques are implemented and tested to compare the results;
- in Chapter 7 the solutions are validated and compared through tests in the experimental setup;
- in Chapter 8 the conlusions are reported, as well as the possible future developments.

2

SYSTEM DESCRIPTION

The analised system is composed of the following elements:

- power supply: TDK-Lambda $SWS600-48$;
- air supply;
- direct drive motor: LAB RT100S air bearing rotary stage;
- servo drive: LAB Drivebox motion controller;
- PC station: Elmo Application Studio II.

In figure 2.1 the connections between the elements of the system are represented. LAB RT100S is driven by the LAB Drivebox that, at this stage, implements the current, velocity and position control loop. Finally, the PC station, in which the software Elmo Application Studio is running, is connected to the drive with an USB cable. The software allows to configure the control system and to tune the control parameters.

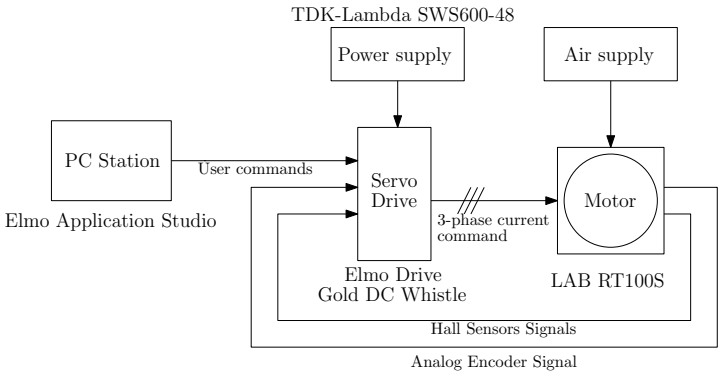


Figure 2.1: Connection representation of the elements of the system.

However, to be able to drive the system with an own design position and velocity control loop other elements are required:

- Real-time Controller board: dSPACE DS1104;
- Interpolator: Tonic Interface x20 TI0020A10A;
- PC station: Matlab and Simulink;

The new system configuration is represented in figure 2.2. In this case, the servo drive is used only as a current driver and it receives current commands by dSPACE DS1104, hall and digital encoder signals, needed for the commutation process. In the controller board

an own designed control algorithm, obtained by code generation of Simulink models, will be implemented.

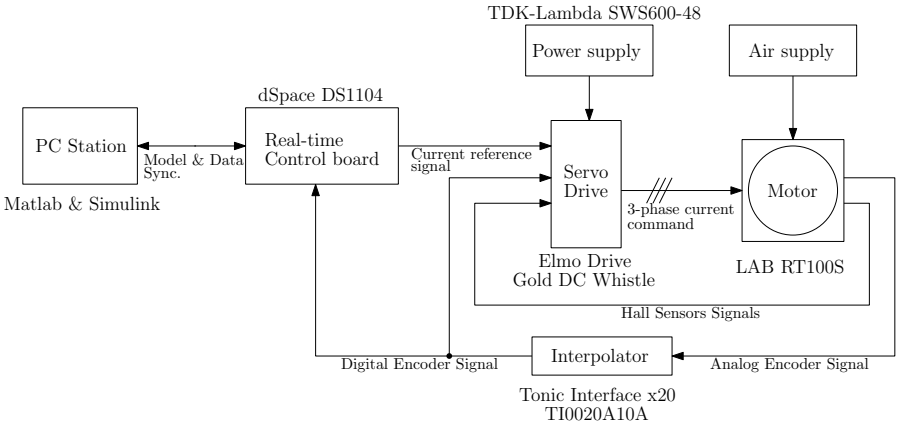


Figure 2.2: Connection representation of the elements of the modified system.

In the following section the main elements of the system are described: LAB RT100S rotary stage, LAB drivebox, the software Elmo Application Studio and dSPACE controller.

2.1 LAB RT100S AIR BEARING ROTARY STAGE

LAB RT100S (see figure 2.3) is a direct drive rotary table that uses air bearing technology. It is mainly used in small measuring devices and as final θ-axis on a multi-axis system. In table 2.1 the main information from the data sheet [21] is reported.

Encoder	15744	periods
Nominal Torque	0.13	Nm
Peak Torque	0.4	Nm
Max. Speed	200	RPM
Total Mass	1.6	kg
Angular Accuracy	±10.3	arcsec

Table 2.1: LAB RT100S main data sheet information.

LAB RT100S is classifiable as a permanent magnet synchronous motor (PMSM) with sinusoidal back electro-motive force (back-EMF). A PMSM needs to be fed by a three-phase inverter managed by a control system of the stator currents.

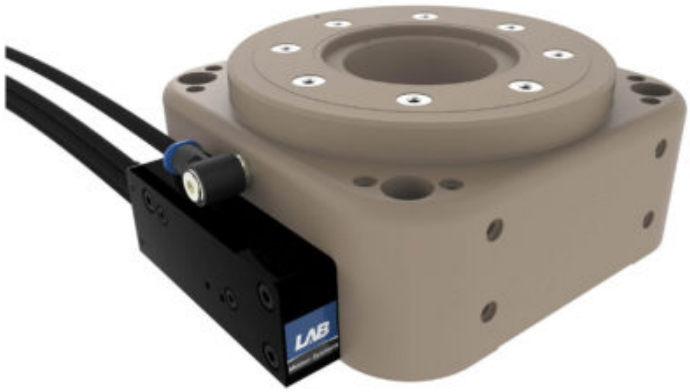


Figure 2.3: LAB RT100S air bearing rotary stage.

2.1.1 *Permanent Magnet Synchronous Motor*

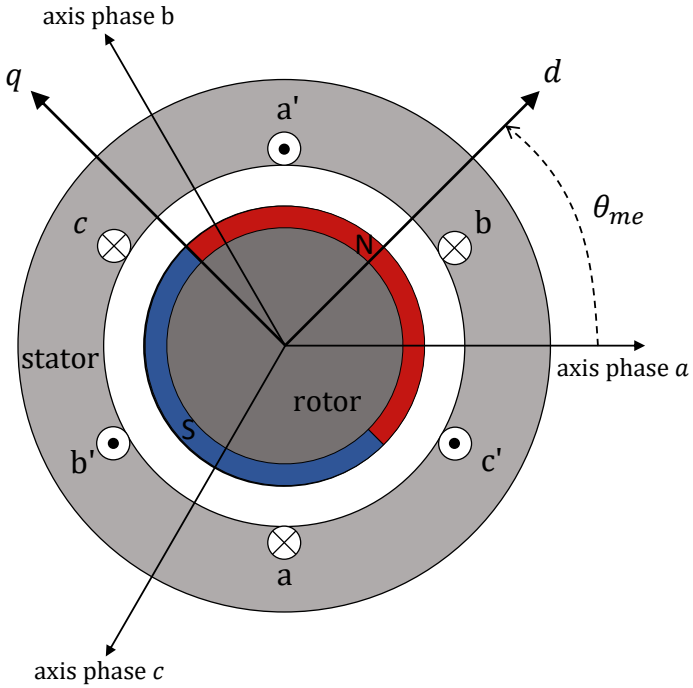


Figure 2.4: Schematic model of PMSM.

A PMSM basically consists of a permanent magnet which acts as rotor surrounded by three equally spaced fixed stator windings, as shown in figure 2.4.

The current flowing in each winding produces a magnetic field vector which sums up to form a resultant stator magnetic field. Torque is generated in the motor by an interaction between the stator magnetic field and the rotor magnetic field, that is produced by the permanent magnet.

Current space vectors are used to model the stator fields in terms of winding current. The mathematical model for a PMSM are in terms of winding current rather then stator magnetic field since they are easier to measure. The current space vector for a given winding has a magnitude that is proportional to the current flowing through the winding and a direction of the field produced by the winding. This allows to represent the stator field as a current space vector that is the vector sum of three current space vector components of the windings. Then the current space vector can be represented into the rotor reference frame by two vector components, the orthogonal (quadrature) component and the parallel (direct) component. The quadrature current component produces a magnetic field that is orthogonal to the rotor magnetic field and therefore it results in torque, while the direct current component produces a field that is aligned with the rotor magnetic field and produces no torque. This concepts can be seen

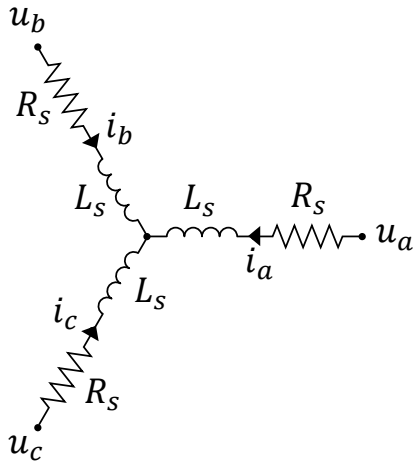


Figure 2.5: Electrical model of PMSM.

from the voltage balance equation of the electrical model of the motor presented in figure 2.5 transformed on the rotor reference frame [23]:

$$\begin{aligned} u_d &= R_s i_d + L_s \frac{di_d}{dt} - \omega_{me} L_s i_q \\ u_q &= R_s i_q + L_s \frac{di_q}{dt} + \omega_{me} L_s i_d + \omega_{re} \Lambda_{mg} \end{aligned} \tag{2.1}$$

where v_d, v_q, i_d and i_q are the voltages and currents in d and q axis respectively. Moreover, R_s, L_s, Λ_{mg}, and ω_{me} are the stator re-

sistance, the phase stator inductance, the permanent magnet flux and the motor electromechanical speed respectively. Expression 2.1 allows to calculate the energy balance on the synchronous reference frame (i.e. rotor reference frame) in order to obtain an expression of the mechanical torque generated by the motor. By multiplying both sides of the equation in 2.1 for $i_d dt$ and $i_q\, dt$ respectively and by adding side by side the two equations it results as follow:

$$\begin{aligned}(u_d i_d + u_q i_q) = (R_s i_d^2 + R_s i_q^2)dt + \\ + L_s i_d di_d + L_s i_q di_q + \\ + \omega_{me} \Lambda_{mg} i_q dt\end{aligned} \tag{2.2}$$

The left side of the equation represents the electric energy fed to the motor in an interval dt. In the right side, the first term represents the heat energy loss through the windings resistance; the second and third consist of magnetic energy stored by the stator inductances L_s. Finally, the last term represents the mechanical energy generated by the motor. However, it must be recalled that the transformation from the stator reference frame to the rotor reference frame is not power invariant, that results decreased by a factor 2/3. Then, recalling that the mechanical power can be expressed as the product of torque τ and the motor mechanical velocity $\omega_m = p\omega_{me}$ it results:

$$\tau = \frac{3}{2} p \Lambda_{mg} i_q \tag{2.3}$$

where p is the number of pole pairs. The expression (2.3) represents the electromagnetic torque generated by the motor by the interaction of rotor magnetic field and stator currents. The expression (2.3) shows clearly that only the quadrature current component, i_q, is useful to generate torque. Thus, to maximize the torque generation through three-phase currents a stator magnetic field synchronous and in quadrature position respect the rotor magnetic field should be induced, such that only the current component i_d is different to zero.

2.1.2 *Field-oriented Control*

The field-oriented control (FOC) technique aims to obtain a desired torque generation minimizing the magnitude of current space vector of the stator currents at any velocity. As seen before, this is ensured by controlling the current i_q to obtain the desired torque and the current i_d to be equal to zero[1].

A block diagram of FOC technique for a PMSM is shown in figure 2.6[23]. In the case represented in figure 2.6, the current reference i_q^* is given by a PI controller that regulates the velocity of the motor. Then

1 For the flux weakening operation this value is modified accordingly, however this case is not considered here

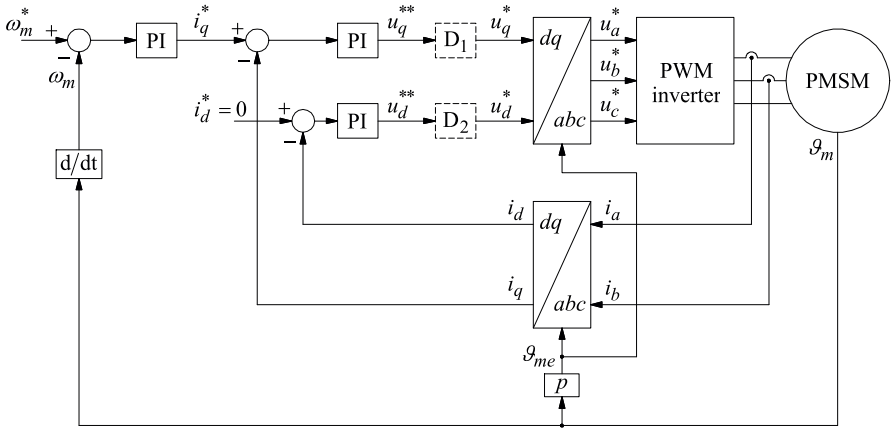


Figure 2.6: Block diagram of FOC of a PMSM.

two loops with PI controller are implemented: one to regulate i_q to the reference i_q^* that represent the desired torque ($\tau = \frac{3}{2}p\Lambda_{mg}i_q$); the second to regulate i_d to zero to obtain the right field orientation. The output of the PI controllers generate the voltage references u_q^** and u_d^**. Then, through decoupling blocks D_1, D_2 and the transformation $\mathbf{T}_{dc/abc}$, from the reference frame abc to dq, is obtained the references u_a^*, u_b^* and u_c^* for the PWM inverter.

In both loops, the current i_d and i_q are calculated from the measured currents (i_a, i_b) through the transformation $\mathbf{T}_{abc/dq}$:

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \mathbf{T}_{abc/dq} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{2.4}$$

where the transformation matrix is

$$\mathbf{T}_{abc/dc} = \frac{2}{3} \begin{bmatrix} \cos(\theta_{me}) & \cos(\theta_{me} - \frac{2\pi}{3}) & \cos(\theta_{me} - \frac{4\pi}{3}) \\ -\sin(\theta_{me}) & -\sin(\theta_{me} - \frac{2\pi}{3}) & -\sin(\theta_{me} - \frac{4\pi}{3}) \end{bmatrix} \tag{2.5}$$

Recalling the voltage balance equations in the reference frame dq:

$$\begin{aligned} u_d &= R_s i_d + L_s \frac{di_d}{dt} - \omega_{me} L_s i_q \\ u_q &= R_s i_q + L_s \frac{di_q}{dt} + \omega_{me} L_s i_d + \omega_{re} \Lambda_{mg} \end{aligned} \tag{2.6}$$

It is clear that the voltage u_d doesn't affect only the current i_d, but also i_q because of the presence of the term $-\omega_{me}L_s i_q$. In a similar way, in the equation of u_q appear both the current i_d an the term related to the flux of permanent magnet, $\omega_{me}\Lambda_{mg}$.
Hence, the two control loops of the currents are not independent and a decoupling action is necessary. The decoupling is obtained, for the loop related to the d axis, subtracting from the reference u_d^** the

term $\omega_{me}L_s i_q$ and, for the loop related to the q axis, adding to the reference $u_q^* *$ the term $\omega_{me}L_s i_d$. In this loop, a feedforward compensation subtraction of the term $\omega_{me}\Lambda_{mg}$ is also realized. The scheme of the decoupling of the axis is presented in figure 2.7.

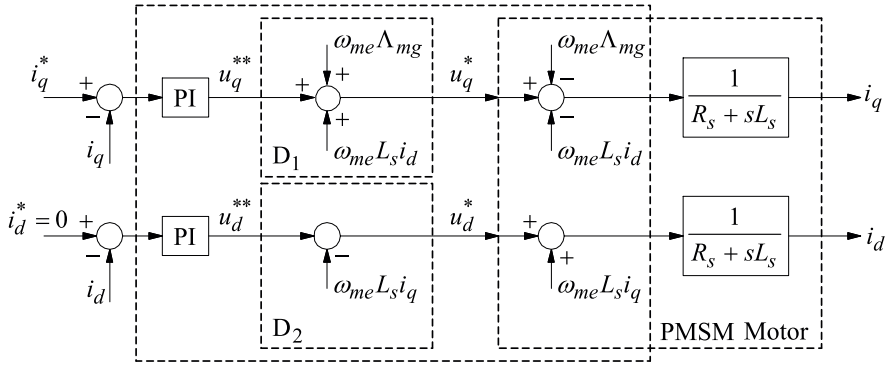


Figure 2.7: Block diagram of axis decoupling in the FOC technique.

Now, it is easy to notice that the added quantities are cancelled by the physical quantities present in the motor. The result of the decoupling is a remarkable simplification of the block diagram which refer to the design of PI controllers for the currents i_d, i_q, as presented in figure 2.8.

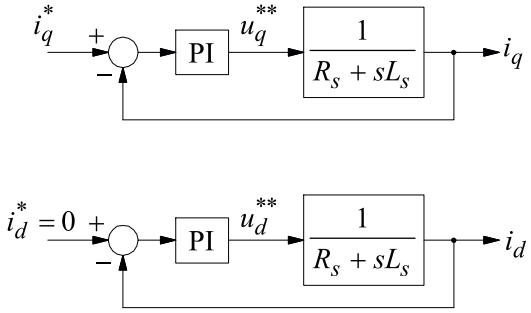


Figure 2.8: Block diagram for the design of PI current controllers.

2.2 LAB DRIVEBOX MOTION CONTROLLER

LAB Drivebox motion controller, represented in figure 2.9, is a motor driver for safe and user-friendly control of brushed and brushless motor (i.e. PMSM with sinusoidal or trapezoidal back-EMF). It can be controlled by a master device or operate in stand-alone mode by running a user program and a convenient set of on-board I/O's. In the following list he main information from the data sheet [21] is reported:

- Supply voltages: 12 V/ 24V/48 V;
- Digital inputs (2): TTL or PLC levels;

- Switch inputs (5): Dry contact;
- Analog input: 1 (differential, $\pm 10\,V$);
- Digital outputs: 4 (2 x TTL + 2 x voltage selectable);
- Hall sensor inputs;
- Analog and digital encoder inputs;
- Motor temperature safety guard input;
- Safe Torque Off (STO) inputs: 2;

Figure 2.9: LAB Drivebox Motion Controller: front panel (above) and back panel (below).

2.3 ELMO APPLICATION STUDIO II

Elmo Application Studio II (EASII) is a software used to set up and command motor drives, as in this case LAB Drivebox, in a user-friendly fashion from a PC station. The main functionalities offered by the software that are interesting for this work are:

- motor and feedback parameter configuration;
- current limit configuration;
- unit modes selection: current and position mode;
- current controller tuning;
- position and velocity controller tuning;
- reference generator;

2.3.1 *Motor and feedback parameter configuration*

The drive is configured by setting the software with some essential information about the motor and the feedback sensor.

Motor type	Rotary Brushless (3 Phase)
Peak Current	3 Arms
Continuos Stall Current	1 Arms
Maximal Motor Speed	200 rpm
Pole Pairs per Revolution	8

Table 2.2: Motor parameter settings.

The motor parameter used for the system under test are reported in table 2.2.

The feedback setting, instead, are different in the two system configuration: the original presented in figure 2.1 and the modified presented in figure 2.2.
In the first case the LAB Drivebox receives directly the analog signal mounted in the motor. In the second case, instead, the dSPACE controller needs a digital feedback signal, obtained by interpolation of the analog signal, hence also LAB Drivebox has to use the same signal.

Sensor type	Analog Sin/Cos
cycles/revolution	15744
counts/revolution	16121856

Table 2.3: Feedback parameter settings: orignal system configuration.

Sensor type	Encoder Quad
cycles/revolution	15744
counts/revolution	314880

Table 2.4: Feedback parameter settings: modified system configuration.

The feedback setting parameter of the two system configuration, original and modified, are reported in table 2.3 and 2.4.

2.3.2 *Limits and protections configuration*

The drive protects the motor from over-current by using a two-stage method. The motor maximum peak (given by the drive parameter PL[1]) is available for the peak duration time (specified by), while for longer periods, the current is limited to its continuos limit (CL[1]).

The current limiting process is dynamic: if the current has been close to its continuous limit, the time allowed for the peak current is reduced. This process is determined as follows: the magnitude value of the measured motor current (in servo drive with FOC, this is $\sqrt{i_q^2 + i_d^2}$) is applied to a first order low-pass filter. The state of the filter is compared with two thresholds. When the state of the filter is higher than the upper threshold, the continuous limit is activated. When the state of the filter is lower than the bottom threshold, the peak limit is activated.
The time constant of the low-filter is $\tau = -PL[2]/log(1 - CL[1]/MC)$, where MC is the maximum servo drive current.

Peak current PL[1]	4.24 A
Continuos current CL[1]	1.41 A
Peak current duration PL[2]	3 s
Max current MC	20 A

Table 2.5: Current limit parameters.

The current limit parameters used for the system under test are reported in table 2.5.

2.3.3 *Unit modes*

An important feature of the drive and its software is the possibility to select different way to controll the motor.

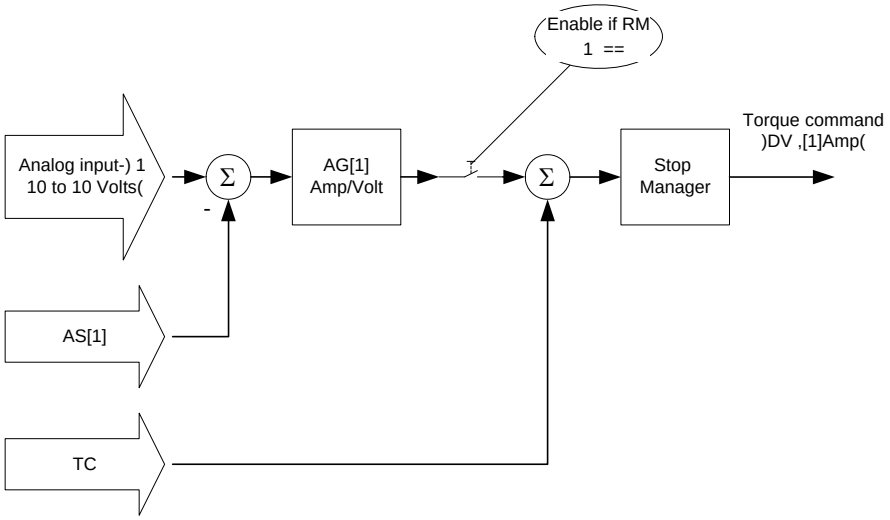


Figure 2.10: Torque control mode: block diagram [6].

In torque control mode only the current control loop is active and receives torque command (i.e. current reference in Ampere) from user or external device. This unit mode is described by the block diagram in figure 2.10.
The user can set the parameter TC with the user-interface or with the drive programming tool of EASII. An external device, instead, can send a voltage signal (-10 to 10 V) to the analog input port. Then, by setting the gain parameter AG[1] is possible to scale the voltage signal to the desired current reference. It is exactly in this way that LAB Drivebox is set up to be used in combination with dSPACE controller, as represented in figure 2.2.

In position control mode three cascade control loops are configured: position loop, velocity loop and current loop. In this unit mode the position reference can be set by an external device or by the EASII through a reference generator. This unit mode is described by the block diagram in figure 2.11.
The system represented in figure 2.1 is configured in position control mode and the desired position reference is set with the user-interface of EASII.

2.3.4 *Current Controller*

The current controller implemented in LAB Drivebox and configurable with EASII is described in figure 2.12. It is an implementation of FOC technique described in the previous section.

In particular, in figure 2.13 the block diagram of PI controller used for the regulation of the two current components is presented. The

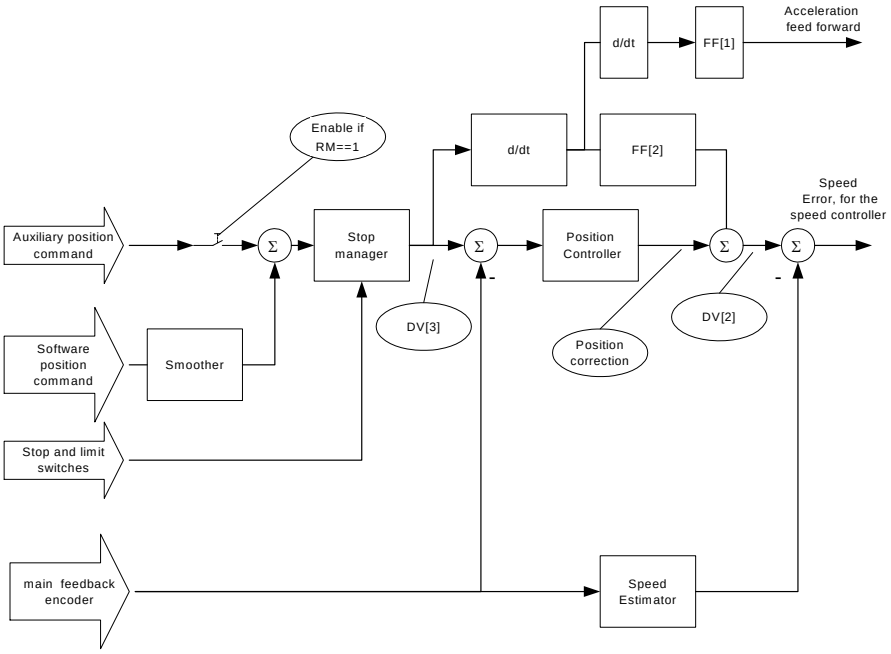


Figure 2.11: Position control mode: block diagram [6].

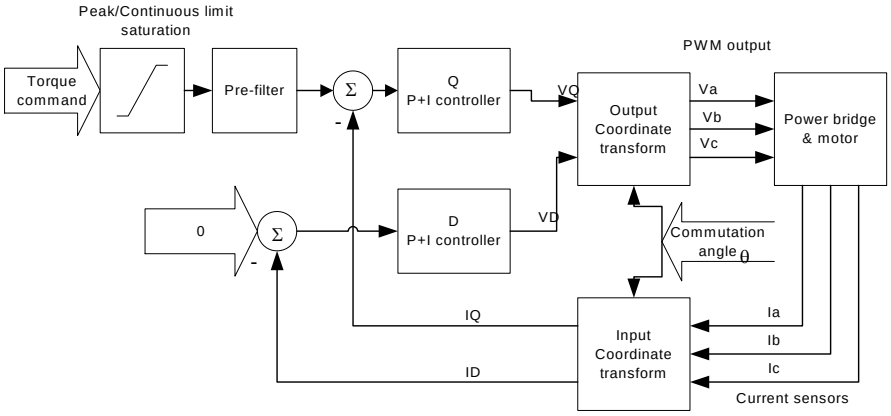


Figure 2.12: Current controller structure: block diagram [6].

only configurable parameters are the proportional gain (KP[1]) and the integral gain (KI[1]). In the figure an anti-windup compensation that modify the action of the integrator whenever saturation occur is also reported, however in the manual [6] no other information are given on how this ad hoc anti-windup compensation is implemented.

2.3.5 *Position and Velocity Controller*

In position mode two outer loops are added to the current loop: velocity and position loop.

The velocity controller, placed in the outer loop of the current controller, is represented in figure 2.14. The structure is fixed and con-

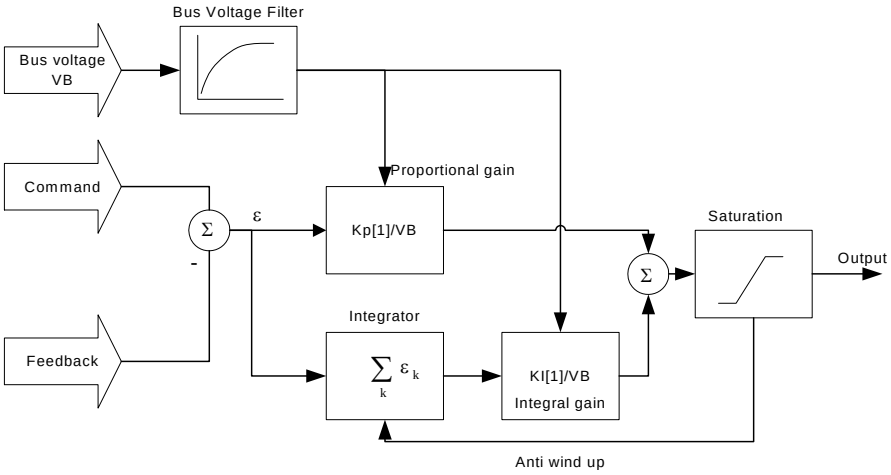


Figure 2.13: PI current controller structure: block diagram [6].

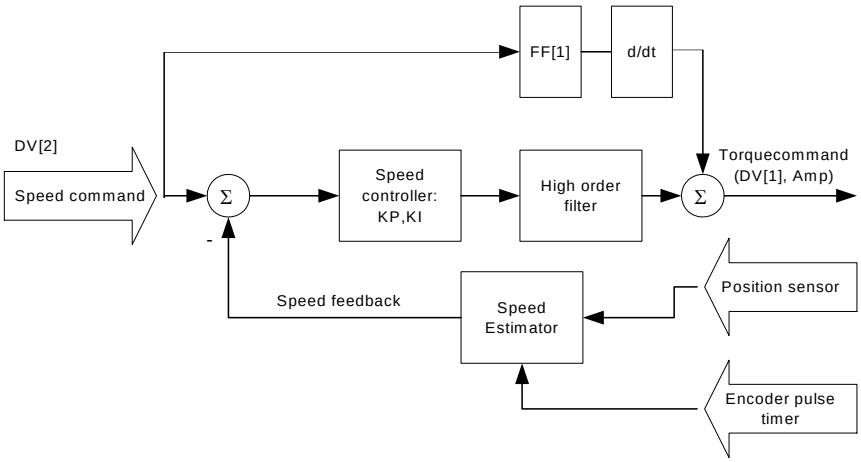


Figure 2.14: Velocity controller structure: block diagram [6].

sists in a PI controller with the possibility to tune the proportional gain (KP[2]) and the integral gain (KI[2]). In addition, it is possible to tune an acceleration feedforward through the gain FF[1].

The position loop, placed in the outer loop of the velocity loop, is described in figure 2.15. The structure is fixed, as in the previous cases, and consist in a proportional controller with a tunable parameter KP[3]. In addition, it is possible to tune a velocity feedforward through the gain FF[2].

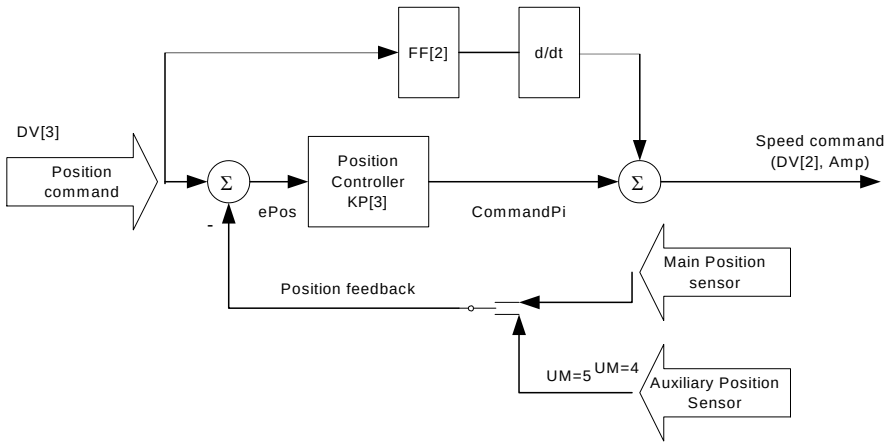


Figure 2.15: Position controller structure: block diagram [6].

2.3.6 *Reference generator*

Parameter	Action
AC	Acceleration, [RPM/s^2]
DC	Deceleration, [RPM/s^2]
SP	Maximum speed, [RPM]
SF	Smooth factor, in millisecond
PR	Relative target position, [deg]
PA	Absolute target position, [deg]

Table 2.6: PTP motion prameters.

When the drive is in position mode the reference generator allows the user to design a position trajectory. The user can specify a point-to-point (PTP) motion subjected to the parameter reported in table 2.6.

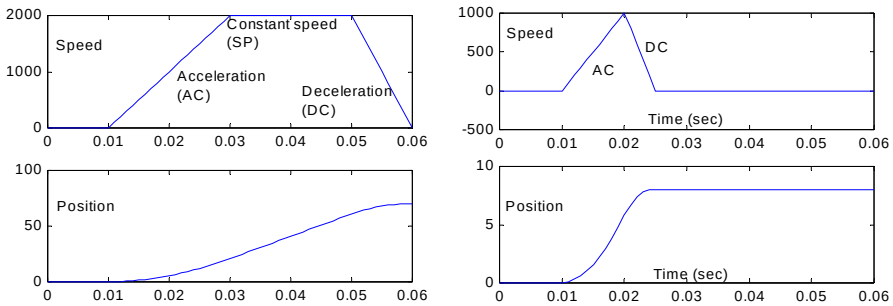


Figure 2.16: PTP trajectory examples[6].

The algorithm generates a trapezoidal motion profile or triangular when the speed limit is not reached. In figure 2.16 two examples, from the manual [6], of the trajectory obtained with the reference generator are illustrated.

2.4 DSPACE DS1104 CONTROLLER BOARD

dSPACE DS1104 Controller Board is a software/hardware platform intended to facilitate interfacing of Simulink models to hardware devices in real-time. With dSPACE, it is possible to develop hardware-in-the-loop (HIL) and rapid control prototyping (RCP) experiments quickly, taking advantage of Matlab and Simulink high-level functions.

The platform allows to implement a real-time control system through four main elements:

- DS1104 PCI board (see figure2.17);
- CP1104 connector panel (see figure2.17);
- ControDesk software;
- DS1104 Real-Time Library for Simulink;

Figure 2.17: DS1104 PCI board and CP1104 connector panel.

Some technical specifications interesting for the application of this work are presented in table 2.7.

Main Processor	MPC8240, PowerPC 603e core, 250 MHz 32 kByte internal cache
Memory	32 MByte synchronous DRAM (SDRAM) 8 MByte boot flash for applications
Analog Output	8 channels, 16 bit, 10 µs max. settling time ± 10 V output voltage range
Incremental Encoder Interface	Two digital inputs, TTL or RS422 24-bit digital incremental encoders Max. 1.65 MHz input frequency 5 V / 0.5 A sensor supply voltage

Table 2.7: dSPACE DS1104 controller board: main technical data.

The dSPACE board DS1104 can be directly programmed from Simulink using the real time interface (RTI) DSPACE blocks in it. The C code generator Simulink Coder previously known as Real Time Workshop is used for automatic building and implementation of simulink models for real time operation on external devices using DS1104.
Encoder input and analog output channels can be initialized and programmed through Simulink RTI modules. The Simulink consists of interface modules for DS1104. The modules are shown in figure 2.18.

The dSPACE ControlDesk software is used as graphical user interface (GUI) to monitor and measure the input and output from the dSPACE controller. It implements the C code generated by Simulink and can manage the parameter of the control system during operation.

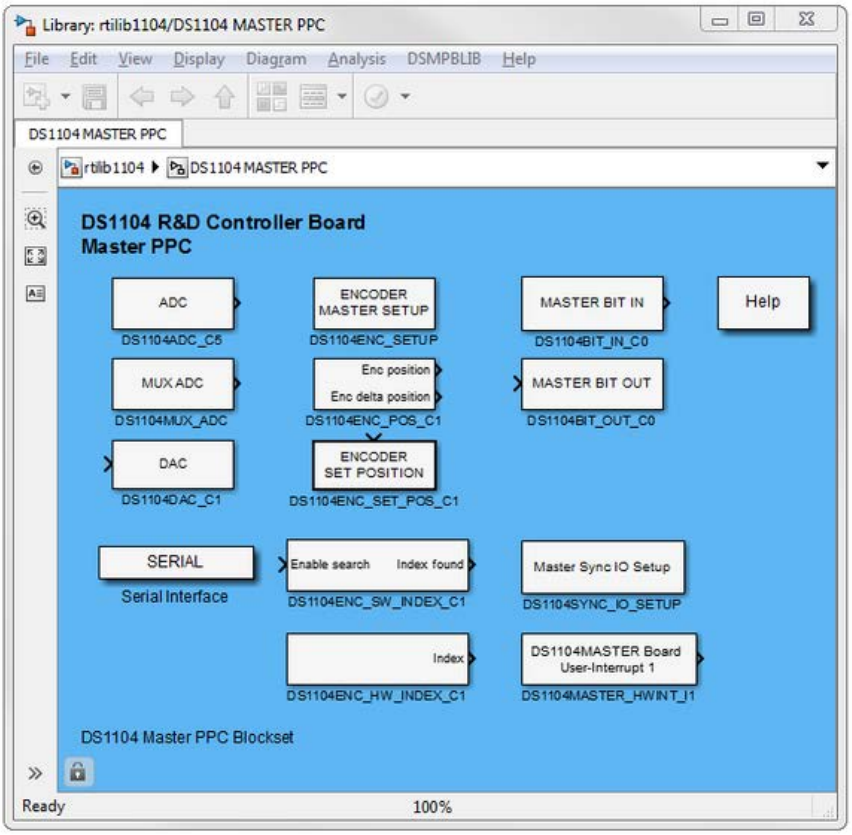


Figure 2.18: Simulink blocks from DS1104 Real-Time Library.

3

SYSTEM ANALYSIS

In this chapter the system will be analyse to find the source of the undesirable behaviour and finally a solution will be proposed to define, at least, the direction of a literature review.

3.1 TASK AND MODEL DEFINITION

The positioning system, in the configuration represented in figure 2.1, is usually used by costumers for scanning operation (i.e. PTP motion) in several applications such as computer tomography, laser machining, optical inspection, measurement systems and so on. A typical operation can be described as follow:

- start from a home position;
- short PTP motion: e.g. 1 deg;
- stand still to complete the task;
- repeat the short PTP motion until a final position: e.g 180 deg;
- homing.

To test the system a short PTP motion of 1 deg will be performed, which represents the scanning operation, and a PTP motion of 180 deg, which represents instead the homing operation.

First the model of the control system is defined. Considering: the equation of electromagnetic torque and mechanical load

$$\tau = \frac{3}{2} p \Lambda_{mg} i_q = K_t i_q = B\omega_m + J\frac{d\omega_m}{dt}, \tag{3.1}$$

(where B and J are the viscous friction and the total inertia of the motor respectively), the electric model of the motor 2.1, the FOC technique with axis decoupling, the velocity and position control loop, the resultant simplified model of the system is represented in figure 3.1.

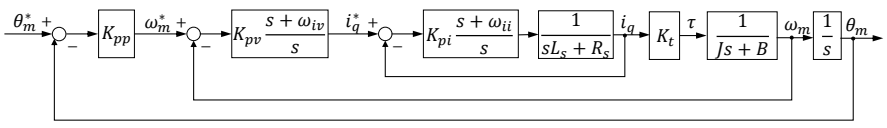


Figure 3.1: Block diagram of the system.

Then, the control system is tuned to satisfy the costumer specifications reported in table 3.1.

Total load inertia, J	0.009 Kg m^2
Current closed loop bandwidth	2500 Hz
Velocity closed loop damping	1
Position closed loop bandwidth	25 Hz
Position closed loop damping	1

Table 3.1: Costumer control system specifications.

The resultant controllers parameters, calculated as the state of the arts [24], are presented in table 3.2.

K_{pi}	5.207 V/A
ω_{ii}	1131.29 Hz
K_{pv}	30.72 A/(rad/s)
ω_{iv}	12.52 Hz
K_{pv}	78.54 1/s

Table 3.2: Controllers parameters.

3.2 TEST IN IDEAL CONDITION

The system will be tested in the ideal condition where the nominal parameters, necessary to tune the control system and to define the reference, are consistent with the physical system. In particular, the value of the inertia J, reported in table 3.1, correspond to total inertia seen by the motor (i.e. motor inertia plus load inertia). Hence, the control system is tuned with the parameters reported in table 3.2.

To command a PTP motion through the EASII software it is necessary to indicate to the reference generator the velocity limit, the acceleration limit and of course the desired absolute or relative position. The velocity limit is 200 rpm, as indicated in the datasheet (see table 2.1). The acceleration, instead, is calculated as the ratio between the peak torque (see table 2.1) and the total inertia of the motor times a safety factor of 0.95:

$$\alpha_{max} = 0.95\frac{\tau_{peak}}{J} = 42.22\,rad/s^2. \tag{3.2}$$

The system response of 1 deg and 180 deg PTP motion is shown in figures 3.2 and 3.3 respectively where is reported the plots of measured position, velocity and current.

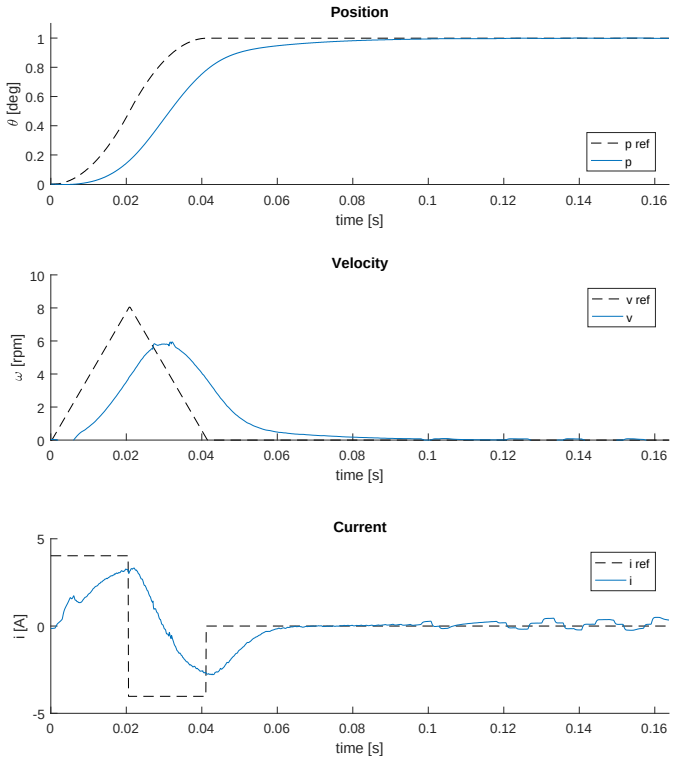


Figure 3.2: 1 deg PTP motion: position, velocity and current plot.

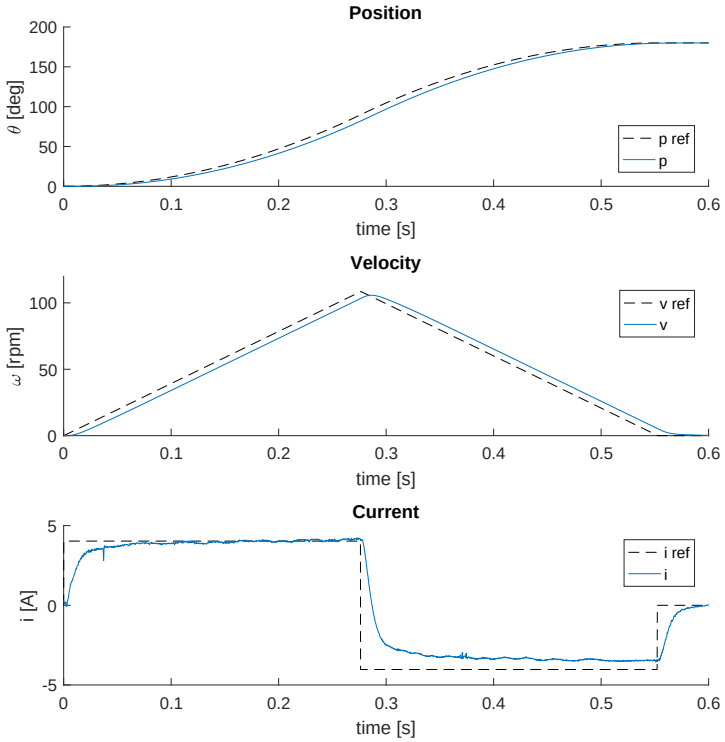


Figure 3.3: 180 deg PTP motion: position, velocity and current plot.

The systems response doesn't exhibit oscillation in both the cases,

1 deg and 180 deg. However, the rise time of the scanning motion is slow compared to the reference, instead it is acceptable in the homing motion. This is due to the position closed loop bandwidth that is not high enough. This is easily solvable with feedforward compensation of the reference signal.

3.3 TEST IN NOT IDEAL CONDITIONS

The system will be tested here in the non ideal conditions where some nominal parameter are not consistent with the physical system. In particular, the case where the installed inertia on the motor is bigger than what the system is designed for is considered. This could happen when, for instance, the inertia is not known a priori and the identification algorithm is not accurate enough. Or, another example, when the costumer adds more inertia without re-tuning the control system.

This case is examined by designing the control system and the references for an inertia decreased by 20% of its original value, $J_{design} = 0.8\,J$. The new parameters of the control system and the new limits for the reference generator are reported in table 3.3.

J_{design}	$0.0072\ Kg\,m^2$
K_{pi}	$5.207\ V/A$
ω_{ii}	$1131.29\ Hz$
K_{pv}	$24.57\ A/(rad/s)$
ω_{iv}	$12.52\ Hz$
K_{pv}	$78.54\ 1/s$
Velocity limit	$200\,rpm$
Acceleration limit	$52.78\ rad/s^2$

Table 3.3: Control system parameters with $J_{design} = 0.8\,J$.

The critical point is that the reference is planned with a higher acceleration. Thus, the torque effort requested to the system is above the maximum torque deliverable by the motor or, equivalently, the current command is above the maximum current limit:

$$\begin{aligned} \tau_{requested} &= \alpha_{max} J = 0.475\,Nm > \tau_{peak} = 0.40\,Nm \\ i_{q,requested} &= \frac{\alpha_{max} J}{K_t} = 5.17\,A > i_{q,max} = 4.24\,A \end{aligned} \tag{3.3}$$

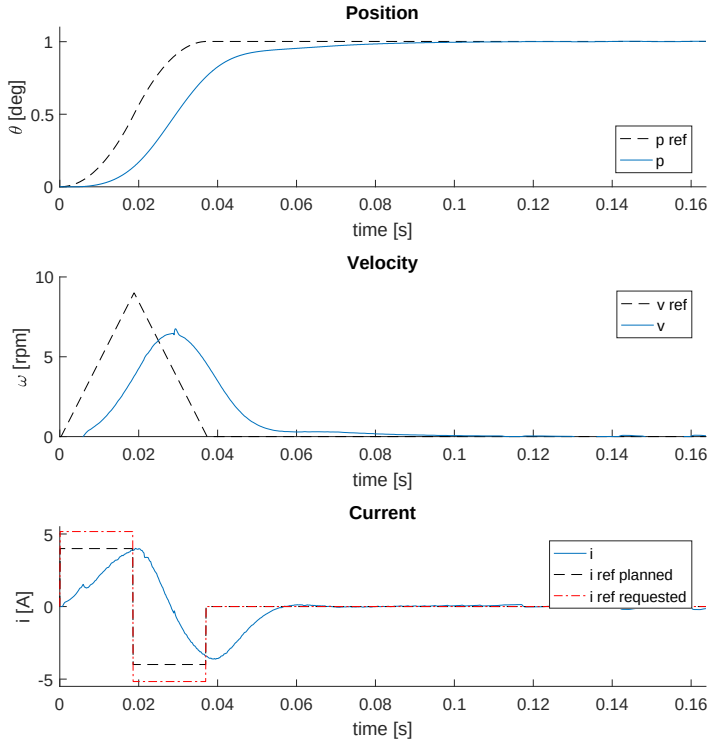


Figure 3.4: 1 deg PTP motion with $J_{design} = 0.8\,J$: position, velocity and current plot.

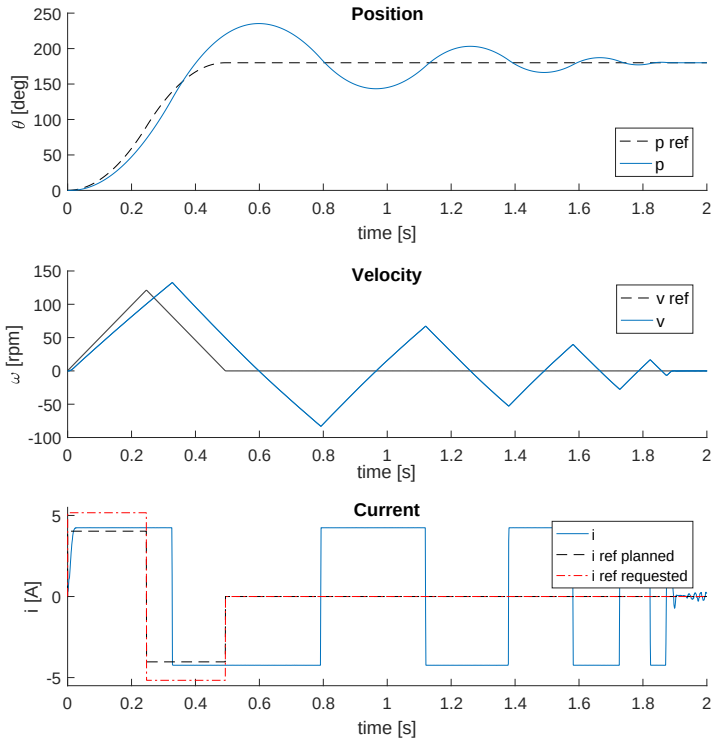


Figure 3.5: 180 deg PTP motion with $J_{design} = 0.8\,J$: position, velocity and current plot.

The new system response of 1 deg and 180 deg PTP motion is shown in figures 3.4 and 3.5 respectively where is reported the plots of measured position, velocity and current.
Even if the reference current requested is higher than planned and above the current limit, as is shown by the red dot-dashed line in the current plot, the scanning motion is well accomplished as in the previous case. This is due to the dynamic response limitation imposed by the position bandwidth for this short motion.
The system response for the homing motion, instead, exhibits oscillations. In this case, the current command is saturated by the current limitation imposed by LAB Drivebox. The oscillations are caused by the action of the anti-windup, implemented for the PI current controller as shown in figure 2.12, that tries to recover the linear response of the system.

3.4 ATTEMPT OF A SOLUTION

The oscillation seen in the last PTP motion response, figure 3.5, is probably what the costumers usually see. To make a recap of the analysis in the following list the possible causes of the undesired oscillations are presented:

- inertia added to the rotary table without re-tuning the control system;
- the acceleration limit for the reference generator is imposed too high: torque requests undeliverable;
- slow response recovery by the anti-windup when current saturation occur;

What came out is that: the source of the issue is a saturation nonlinearity, not taken into account during the design process, and that undesirable oscillations are seen when saturation occurs due to several reason, as listed before.

In order to prevent that saturation takes place it is necessary to have an accurate knowledge or estimation of the total inertia seen by the motor. In this manner it is possible to design a reference with an acceleration limit achievable by the motor without reaching the current limit, i.e. torque limit.
Hence, an accurate identification algorithm should be implemented to allow the user to estimate the inertia whenever a new load is mounted to the motor or oscillations are seen.
In addition, to improve the performance of the system when saturation occur, in other words, to avoid undesirable oscillation and

recover the unconstrained response as soon as possible, in the following chapter a better solution for the actuator saturation problem rather then the ad hoc anti-windup for a PI controller implemented in LAB Drivebox will be sought. However, it is not possible to modify or improve the anti-windup of LAB Drivebox because the control structure is fixed and doesn't allow modification. Thus, a solution that can be implemented in the outer loop, position and velocity, through the dSPACE controller board will be investigated.

4 LITERATURE REVIEW

In chapter 3 a saturation nonlinearity in input of the plant, in short actuator saturation, has been recognize as the source of the undesirable oscillations seen by costumers during operations with the positioning system under test. In this chapter a literature review of the actuator saturation problem will be presented in order to seek for a suitable solution for this project.

4.1 INTRODUCTION

Actuator saturation occurs when a controller demands a signal which is larger than the one the actuator is capable of delivering. This strongly nonlinear phenomenon can cause undesirable behaviours such as degradation of performance or even instability in control loops which otherwise would exhibit a satisfactory behaviour. In particular, actuator saturation have been implicated in various aircraft crashes and the meltdown of the Chernobyl nuclear power station [26].

In spite of that, it is a common practice for control engineers to design controller taking into account actuator constrains often only implicitly. A straightforward way to bypass the problem is to limit the magnitude of control signal, for instance, restricting bandwidth of linear controller or by making the actuator large enough so that, during operation, the input effort commanded is well below the saturation levels. However, this implicit way of handling actuator constraints clearly decrease the overall achievable performance of the system or increase the cost, which are both unacceptable in many applications.

In literature, a more proper way to handle the problem is widely discussed from the earliest solution in 1967 [7]. What emerges are two approaches which can be adopted to avoid undesirable saturation effects.
The first, is referred as *one-step* approach which consists in designing a controller, also nonlinear, tanking into account actuator constraints from the beginning. An example of this approach is the Model Predictive Control (MPC), a model-based design which can systematically handle hard input constraint solving an online optimal control problem. For this last reason, this method is applicable in relatively slow sampled processes. Although, in the last few years, big effort was made to find more efficient solution and extend the field of applicability.

The second, is referred as *anti-windup* augmentation. In this approach a controller is designed without taking into account explicitly the actuator constraints, rather then, by ensuring desirable performance. Then, after the controller has been designed, a so-called Anti-Windup Compensator (AWC) is designed to handle saturation constraints and to ensure stability properties. Such an approach is consider attractive in practice because the AWC becomes active only when saturation is encountered and otherwise the nominal performance ensured by the controller is preserved.

The actuator saturation problem considered in this work will not be the ordinary behaviour of the system, rather than it will be an exception that occurs in some particular cases as point out in the previous analysis (see section3.3). The system will mainly work in the linear region where a traditional controller, that ensure desirable performance, already exists. This fact makes the AWC approach the most suitable solution for this work. Hence, the literature review will focus in this direction.

4.2 ANTI-WINDUP TECHNIQUES

It has already been pointed out that actuator saturation can cause undesirable effects such as performance degradation or even instability. The term *windup* is a phenomenon associated with saturation in systems with integral controllers and, originally, it referred to the build-up of the integrator's capacitor during saturation. The following dissipation of this charge would then cause long settling times and excessive overshoot. Actually, for more complicated controllers, this windup phenomenon is more difficult to analyse, however the word *windup* still represents the undesirable behaviour that arises when saturation occurs and the controller ignores the mismatch between his output and the plant input. *Anti-windup* refers to the augmentation of a controller in a feedback loop to avoid windup or to recover the unconstrained response after saturation occurred.
In literature the anti-windup is widely studied from the early days of analog control where the windup problem was first identified [18]. From that time till the latest study it is possible to classify different strategies of AWC [27, 2]:

- ad hoc methods;
- classical anti-windup:
  - observer-based anti-windup;
  - conditioning technique;
  - generic frameworks;
- modern anti-wndup:

- direct-linear anti-windup;
- model recovery anti-windup.

The aim of this literature review is not to give a detailed explanation of all anti-windup solutions, rather than to give an overview of the most relevant methods. In this way it is possible to seek for anti-windup which can improve the ad hoc methods available in commercial control software, and to give a systematic way of design with guarantees of stability and performance.

4.2.1 *Ad hoc methods*

From the first anti-windup solution many ad hoc methods were developed specifically for PID controllers. For their easy implementability these AWCs are widely used by practitioners and it is possible to find them still nowadays in commercial software for control systems [6, 13, 25]. However, no guarantees of stability and recovery performance are ensured.

A first method to avoid integrator windup is to turn off the integrator action as soon as the actuator saturates and update the integrator if the control signal de-saturate [32].
A second method is the *incremental algorithm* [1] in which the structure of PID is modified moving the integrator outside the controller. In this way the controller produces only the increment of the control signal and then it is fed to an integrator. Windup is avoided by stopping the integration whenever the output saturates.
Another method is the *conditional integration* [1]. The idea is to apply integral action when the tracking error is within certain specific bound. Intuitively, the bounds can be set such that the actuator does not saturate given the instantaneous or predicted value of the system output. Basically, the limits on the actuator are translated into limits on the tracking error.

4.2.2 *Classical anti-windup*

Classical anti-windup, referred also as *back-calculation* or *tracking*, was first describe in Fertik (1967)[7] followed by Åström and Rundqwist (1989) [1] as an anti-windup solution for PID controllers. To avoid windup, an extra feedback loop is added by feeding the difference between the control output and the actuator output back to the integrator through an adjustable gain. In this way, that signal is zero when there is no saturation, otherwise the feedback signal prevents the integrator from winding up. There where no theoretic guidelines to choose the value of the gain. As a result this method does not en-

sure any guarantees of stability and recovery performance.

As a natural extension of the classical anti-windup in [1] the authors came up with the so-called Observer-based anti-windup (OBSAW) that follows the same idea of the back-calulation applied, however, to a general state space controller. In this case to avoid windup, the feedback signal of the AWC is fed to the state of the controller through a static gain. In Kapoor et al. (1998) [14] a design procedure for the selection of the anti-windup gain is given to ensure stability of the constrained closed-loop system.

Another extension of the back-calculation anti-windup is the *Conditioning Technique* formulated by Hanus et al. (1989) [11] and his later modification by Walgama et al. (1992) [33]. The principal behind the method is to modify the reference to the controller whenever the controller output is different from the plant input. The reference is modified such that if the new reference had been applied to the controller, the actuator would not saturate. The modification consists in feeding the difference of controller output and plant input to the reference through a gain. No guidelines are given for the choice of the gain.

After the development of many AWC a big effort was made to provide unification schemes, see Walgama and Sternby (1990) [32], Kothare et al. (1994) [16] and Edwards and Postlethwaite (1998) [5]. The benefits from such a unification was to show that each AWC scheme was a special case of the others. Moreover, the generic frameworks opened for establishing a general synthesis and stability analysis for AWC schemes.

4.2.3 *Modern anti-windup*

Even though classical anti-windup represents an improvement from the early ad hoc solution for PID controllers, the lack of formal stability guarantees motivated many researcher to tackle the anti-windup problem from a constrained stabilization perspective. As a result, from the late 1990's onwards, researcher developed systematic methods to design anti-windup that ensure stability guarantees and optimal performance properties, the so-called *modern anti-windup* techniques. These methods can be categorize into two main architectures: Direct Linear Anti-Windup (DLAW) and Model Recovery Anti-Windup (MRAW).

DLAW belongs, historically, to the AWC summarized in Kothare et al. (1994)[16] but was first proposed by Mulder et al. (2001) [20], followed by [10, 4, 12]. MRAW follows a completely different paradigm

and was first proposed by Teel and Kapoor (1997) in [30], followed by [35, 34]. A comprehensive description of these methods and their references can be found in Galeani et al. (2009) [9], Tarbouriech et al. (2011) [28] and Zaccarian and Teel (2011) [34].

4.2.4 *Overview*

The aim of this literature review is to seek a solution that might improves the ad hoc anti-windup methods, implemented in commercial control software, and to give guarantees of stability and recovery performance. With this objective in mind, what emerges is:

- OBSAW is the only classical anti-windup with a described method of designing to ensure stability properties [14];
- DLAW and MRAW have been developed to be design with systematic methods to ensure stability guarantees and optimal performance properties [9, 28, 34];

Thus, the modern anti-windup techniques, DLAW and MRAW, and the OBSAW will be studied, implemented and tested to perform a comparison and to verify the effectiveness of the methods.

5 ANTI-WINDUP

In this chapter the anti-windup techniques selected in chapter 4 that improve the ad hoc anti-windup methods, already implemented in LAB Drivebox, and gives guarantees of stability and recovery performance will be described and studied.

5.1 PRELIMINARIES

5.1.1 *The unconstrained closed-loop system*

The *unconstrained closed-loop system* is a plant-controller pair that, when connected in feedback without input saturation, behaves in a satisfying manner. The unconstrained system behaviour is something to emulate when trying to specify and solve the anti-windup problem.

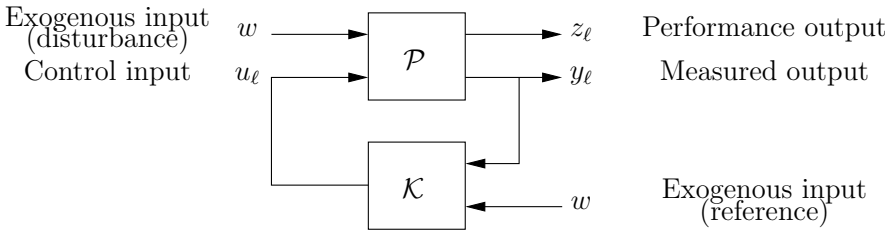


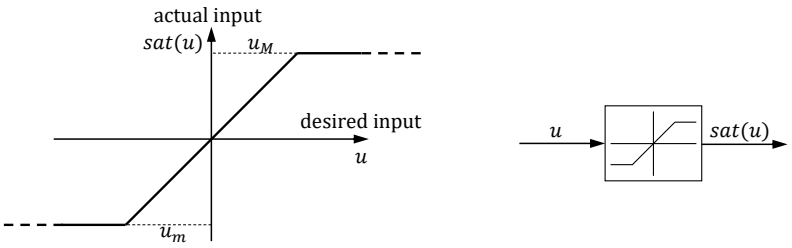
Figure 5.1: The unconstrained closed-loop system.

A general representation of the unconstrained closed-loop system is illustrated in figure 5.1. The *unconstrained plant* is represented by $\mathcal{P}$ and the *unconstrained controller* is represented by $\mathcal{K}$. The plant in the unconstrained closed-loop has a control input u_l, an exogenous input w, a measured output y_l, and a performance output z_l. One controller input is equal to the measured plant output while the other is coming from an exogenous signal, perhaps a reference command. The exogenous inputs affecting the plant and the controller have been grouped into the common symbol w for convenience. The internal state trajectory of the unconstrained closed-loop system, denoted (x_{pl}, x_{cl}), is called the *unconstrained response* and the signals u_l, z_l, y_l are called, respectively, the *unconstrained control input response*, *unconstrained performance output response*, and *unconstrained measured output response*.

As already pointed out, the unconstrained closed-loop represent the desirable behaviour of the system. Hence, the origin of the unconstrained closed-loop system is assumed to be asymptotically stable

when the exogenous inputs are set to zero, if not the anti-windup problem does not make sense. This automatically implies that the internal state of the plant can be stabilized by the control input.

5.1.2 *Actuator saturation*



(a) (b)

Figure 5.2: The saturation nonlinearity: (a) graph, (b) block diagram.

When a control system gives a request or command to an actuator, the actuator typically produces an output (force, torque, displacement, or other physical quantity), within its operating range, that is closest to the requested value. Values outside of the actuator's amplitude limits are mapped into the range of capabilities to the *saturation function* or *nonlinearity* described mathematically by the following equation:

$$\mathrm{sat}(u) := \begin{cases} u_M, & \text{if } u \geqslant u_M \\ u, & \text{if } u_m \leqslant u \leqslant u_M \\ u_m, & \text{if } u \leqslant u_m \end{cases} \tag{5.1}$$

where u_M and u_m correspond to the maximal and minimal attainable actuator value.
The input u is said to be in the *linear region* where the saturation nonlinearitu is equal to the identity when $u_m \leqslant u \leqslant u_M$. The saturation function is called a *simmetric saturation* if $u_M = -u_m$.

For the case of multi-input control system, the symplest vector-valued saturation function corresponds to the *decentralized saturation function* which consist of a vector of scalar saturation function, the i^{th} function depending only on the i^{th} component of the input vector. In other words, the vector-valued decentralized function has the form:

$$\sigma(u) := \begin{bmatrix} \mathrm{sat}_1(u_1) \\ \mathrm{sat}_2(u_2) \\ \vdots \\ \mathrm{sat}_{n_u}(u_{n_u}) \end{bmatrix}, \tag{5.2}$$

where $\mathrm{sat}_i(\cdot)$ is defined as in (5.1) for all i.

5.1.3 *The saturated closed-loop system*

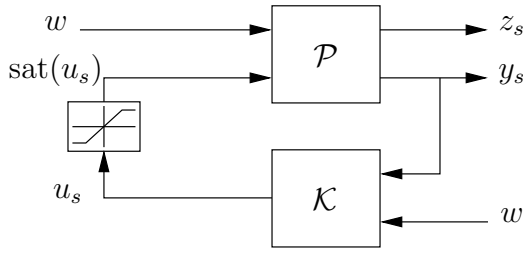


Figure 5.3: The saturated closed-loop system.

As pointed out in the previous chapter, the windup phenomenon can occur when plant input saturation is introduced into the unconstrained closed-loop system. A general representation of the *saturated closed-loop system* is reported in figure 5.3. The internal state trajectory of the saturated closed-loop system is called the *saturated response*; the trajectories u_s, z_s and y_s are called, respectively, the *saturated control input response*, *saturated performance output response*, and *saturated measured output response*.

5.1.4 *Qualitative Objective*

The anti-windup augmentation should achieve the following qualitative objectives:

- *Small signal preservation*: to make the response of the anti-windup augmented closed-loop system mach the response of the unconstrained closed loop whenever this is possible;
- *Asymptotic stability*: to make, in absence of exogenous inputs, a desired constant operating point asymptotically stable with a basin of attraction at least as big as the set of the states over which the system is expected to operated;
- *Input-output stability*: to induce a bounded response for initial states and exogenous inputs that are expected during operation;
- *Unconstrained response recovery*: to recover the unconstrained closed-loop response asymptotically whenever this is possible.

5.1.5 *The augmented closed-loop system*

The small signal preservation property suggests that the output of the anti-windup augmentation should be identically equal to zero in the case where the output of the unconstrained controller is always within the limits of the actuator and the state of the anti-windup augmentation, if it exist, is initialized to zero. The most common way to

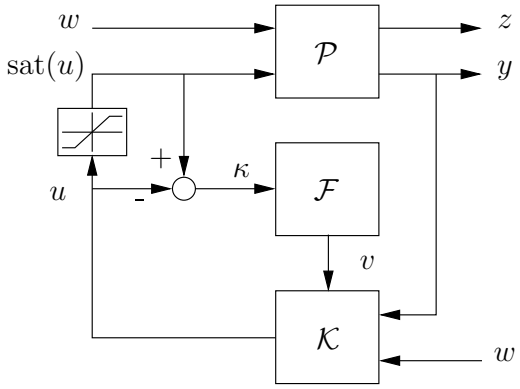


Figure 5.4: The anti-windup augmented closed-loop system.

achieve this feature is to make the input of the anti-windup augmentation equal to the difference between the input commanded to the actuator and the input supplied by the actuator to the plant. The resultant architecture is shown in figure 5.4. The corresponding closed-loop system is called the *anti-windup augmented closed-loop system*. The following terminology is introduced for this system. The internal state trajectory of the closed loop is the *anti-windup augmented response*. The signals u, z, and y are called, respectively, the *anti-windup augmented control input response*, the *anti-windup augmented performance output response*, and the *anti-windup augmented measured output response*.

The block $\mathcal{F}$, which is called the *anti-windup augmentation* or *filter*, represents a system that is possibly dynamic. In its state-space representation, it has the general form

$$\begin{aligned} \dot{x}_{aw} &= A_{aw}x_{aw} + B_{aw}\kappa \\ v &= C_{aw}x_{aw} + D_{aw}\kappa. \end{aligned} \tag{5.3}$$

Figure 5.4 shows the anti-windup augmentation block $\mathcal{F}$ affecting the control system by injecting its output signals into the unconstrained controller block. The way in which the output signal is injected determines two different scenarios illustrated in figure 5.5. The diagram on the left side of figure 5.5 shows the case where the anti-windup augmentation injects signals only at the input and output of the unconstrained controller. This is referred to as *external anti-windup augmentation*. The diagram on the right side of figure 5.5 shows the case where the anti-windup augmentation affects the state equations of the unconstrained controller. This is referred to as *full-authority anti-windup augmentation*.

To describe the effect of the signal v on the unconstrained controller, let the unconstrained controller have state-space realization (A_c, B_c, C_c, D_c). Then the external anti-windup augmentation in the

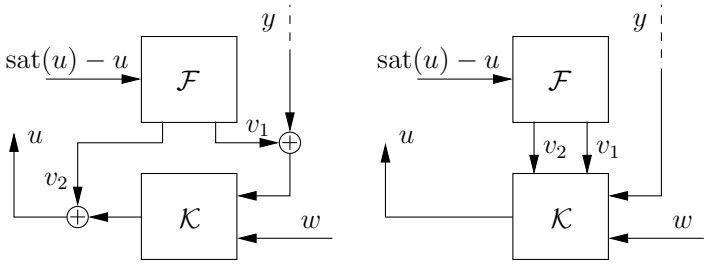


Figure 5.5: External anti-windup augmentation (left) and full-authority augmentation (right).

left-hand diagram of figure 5.5 affects the controller by injecting its output $v = (v_1, v_2)$ as follows:

$$\mathcal{K} \quad \begin{cases} \dot{x}_c &= A_c x_c + B_c \begin{bmatrix} y + v_1 \\ w \end{bmatrix} \\ u &= C_c x_c + D_c \begin{bmatrix} y + v_1 \\ w \end{bmatrix} + v_2 \end{cases} \tag{5.4}$$

Alternatively, the full-authority anti-windup augmentation in the right-hand diagram of figure 5.5 affects the controller by injecting its outputs v_1 and v_2 as follows:

$$\mathcal{K} \quad \begin{cases} \dot{x}_c &= A_c x_c + B_c \begin{bmatrix} y \\ w \end{bmatrix} + v_1 \\ u &= C_c x_c + D_c \begin{bmatrix} y \\ w \end{bmatrix} + v_2 \end{cases} \tag{5.5}$$

These equation are also represented grapically in the block diagram of figure 5.6.

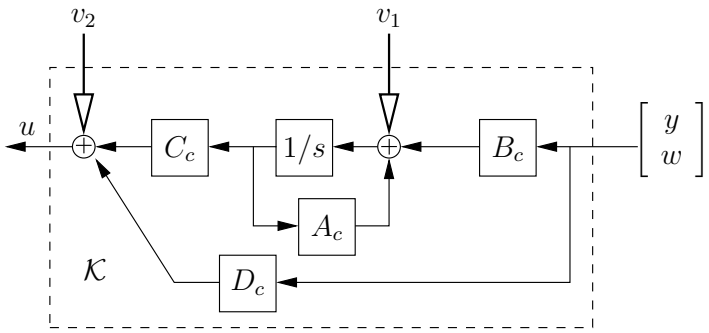


Figure 5.6: Anti-windup signals in full-authority anti-windup augmentation.

Any full-authority anti-windup compensator can be implemented as an external anti-windup compensator when the matrix multiplying y in the $\dot{x}_c$ equation has full row rank.

5.1.5.1 *Algebraic loops introduced by proper anti-windup filters*

The general form of the anti-windup filter $\mathcal{F}$ is given in equation (5.3). When the condition $D_{aw} \neq 0$ is satisfied the anti-windup filter is said to be *strictly proper*. Otherwise, the anti-windup filter is said to be *proper*.

Unless the anti-windup filter is strictly proper, it is possible that the anti-windup augmentation in figure 5.4 introduces an algebraic loop into the anti-windup augmented control system. An *algebraic loop* is a closed path that does not involve passing through any dynamic element while traversing the path. When it exists, the algebraic loop introduced into the anti-windup augmented controller in figure 5.4 occurs in the loop from the input of the saturation element through the anti-windup filter to the unconstrained controller and back to the input of the saturation element.

An algebraic loop generates an implicit equation that may or may not have a unique solution. An algebraic loop is said to be *well-posed* if the implicit equation that it creates has a unique solution. Otherwise, the algebraic loop is said to be *ill-posed*.

The anti-windup augmentation algorithms designed with the modern anti-windup techniques, described in the following chapter, will guarantee that the algebraic loops are well-posed. However, such guarantees do not give insight into how to solve the implicit equation in order to generate an explicit commanded input to send to the plant's actuators.

To solve this problem, occurring in the implementation process on a real-time environment, will be considered the result of the following analysis by relying the use of a lookup table to generate the explicit control value.

The algebraic loops in the anti-windup augmented controller have the form

$$u = \zeta + \Lambda(u - sat(u)) \tag{5.6}$$

where ζ is some function of the plant output and the unconstrained controller states. When well-posed, the solution to the equation (5.6) has a piecewise affine form with respect to the variable ξ with 3^n possibly distinct regions, where n is the number of components of u. This is because equation (5.6) can be rewritten as $\Xi(u) = \zeta$, where the function Ξ is piecewise affine with 3^n regions, corresponding to all of the different combinations of the three conditions i) $ui < (u_m)_i$, ii) $(u_m)_i \leqslant u_i \leqslant (u_M)_i$, and iii) $(u_M)_i < u_i$ for i ranging from 1 to n. Let the matrix M_i and the vector b_i be such that $Xi(u) = M_i u + b_i$ for u values in the i^{th} region. Since the algebraic loop is well-posed, the matrix M_i is invertible for each i. The solution to the equation $\Xi(u) = \zeta$ is then given by $M_j^{-1}(\zeta - b_j)$, where j is such that $M_j^{-1}(\zeta - b_j)$ is a vector belonging to region j. Due to well-posedness, there will always

exist such an index j and, when there is more than one index, each index returns the same answer for the solution u.

5.2 OBSERVER-BASED ANTI-WINDUP

The observer-based anti-windup (OBSAW) was first described by Åström and Rundqwist (1989) in [1]. The idea of this AWC is based on a back-calculation approach by feeding back the difference between the control output and the actuator output (measured or emulated) through a static gain. However, no indication on how to design the static gain was given in [1]. However, Kapoor et al. (1998) [14] developed a systematic procedure to design the OBSAW to ensure global stability. In addition, through numerous classical anti-windup augmentations have been proposed in literature, it has been shown in Walgama and Sternby (1990) [32] that a majority of them can be cast into the form of an OBSAW. This two facts motivated the study of this method.

The OBSAW can be seen as full-authority anti-windup filter with the following static representation:

$$\nu = \begin{bmatrix} L \\ 0 \end{bmatrix} (sat(u) - u). \tag{5.7}$$

where L is the static gain of the OBSAW.

The exogenous input affecting the controller is assumed to be a reference signal r, the controller input $e = r - y$, while the exogenous input affecting the plant a distubance d. The effect on the uncostrained controller is described as follow:

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c(r-y) + \nu_1 \\ &= A_c x_c + B_c(r-y) + L(sat(u) - u) \\ &= (A_c - LC_c)x_c + (B_c - LD_c)(r-y) + Lsat(u) \\ u &= C_c x_c + D_c(r-y) + \nu_2 \\ &= C_c x_c + D_c(r-y) \end{aligned} \tag{5.8}$$

where $A_c - LC_c$ resemble the dynamic of an observer.

The architecture of the controller automatically satisfies the small signal preservation objective, while the others qualitative objectives of the anti-windup augmentation are pursued following the algorithm for synthesizing the OBSAW matrix L proposed in [14]. The algorithm aims to induce an invariant subspace for the dynamic behaviour of the mismatch between the constrained and saturated closed-loops. The dynamics in this invariant subspace are identical to the behaviour of the plant with input saturation starting at the origin, stabilized by linear state feedback and driven by the mismatch between the unconstrained input and this input passed through a saturation function.

The state feedback gains are determined by the nominal controller and the properties of the invariant subspace. In other words, the main idea behind the synthesis of the gain L is to ensure that, with the linear plant given by $(A, B, C, 0)$ the mismatch between the constrained and saturated plant input and output can be characterized by a system of the form

$$\begin{aligned} \dot{X} &= AX + B[sat(K_1X + u_l) - u_l] \quad X(0) = 0 \\ Y &= CX \\ U &= K_1X \end{aligned} \tag{5.9}$$

where Y is the mismatch between the outputs, U is the mismatch between the inputs and u_l is the plant input of the unconstrained closed-loop. The gains K_1 are completely determined by the nominal controller and by the choice of invariant subspace if it is not unique.

5.2.1 *Observer anti-windup synthesis*

Consider a system of the form:

$$\dot{x} = Ax + B_u sat(u) + B_w d \quad y = Cx \tag{5.10}$$

($D = 0$ is taken for simplicity) under a linear compensator of the form (5.8). Let $x \in \mathbb{R}^n$ and $x_c \in \mathbb{R}^{n_c}$. To see the effect of L on the anti-windup problem, the error between the anti-windup augmented closed-loop and the unconstrained closed-loop is considered. The unconstrained closed-loop with state $\xi_l = [x_l, x_{c,l}]^T$ is given by:

$$\begin{aligned} \dot{\xi}_l &= \begin{bmatrix} A + B_u D_c C & B_u C_c \\ B_c C & A_c \end{bmatrix} \xi_l + \begin{bmatrix} -BD_c \\ -B_c \end{bmatrix} r + B_d d \\ &= A_{cl}\xi_l + B_{cl} r + B_d d \\ y &= \begin{bmatrix} C & 0 \end{bmatrix} \xi_l = C_{cl}\xi_l \\ u_l &= \begin{bmatrix} -D_c C & C_c \end{bmatrix} \xi_l + D_c r = K_{cl}\xi_l + D_c r \end{aligned} \tag{5.11}$$

The anti-windup augmented closed-loop with state $\xi = [x, x_c]^T$ yields:

$$\begin{aligned} \dot{\xi} &= \begin{bmatrix} A + B_u D_c C & B_u C_c \\ B_c C & A_c \end{bmatrix} \xi + \begin{bmatrix} -BD_c \\ -B_c \end{bmatrix} r + B_d d + \begin{bmatrix} B_u \\ L \end{bmatrix} (sat(u) - u) \\ &= A_{cl}\xi_l + B_{cl} r + B_d d + L_{cl}(sat(u) - u) \\ y &= C_{cl}\xi_l \\ u_l &= K_{cl}\xi_l + D_c r \end{aligned} \tag{5.12}$$

Then, by introducing the mismatch between the anti-windup augmented and the unconstrained closed-loop system such as:

$$\bar{X} = \xi - \xi_l, \quad Y = y - y_l, \quad U = u - u_l, \tag{5.13}$$

and subtracting (5.11) from (5.12) gives the system

$$\begin{aligned} \dot{\bar{X}} &= A_{cl}\bar{X} + L_{cl}(sat(u) - u) \quad \bar{X}(0) = 0 \\ &= (A_{cl} - L_{cl}K_{cl})\bar{X} + L_{cl}[sat(u_l + K_{cl}\bar{X}) - u_l] Y \quad = C_{cl}\bar{X} \\ U &= K_{cl}\bar{X} \end{aligned} \tag{5.14}$$

It was shown in [14] that if A_{cl} is Hurwitz and if there exist matrices $T \in \mathbb{R}^{n_c} \times \mathbb{R}^{n+n_c}$ and $H \in \mathbb{R}^{n_c} \times \mathbb{R}^{n_c}$ such that

$$TA_{cl} = HT \tag{5.15}$$

and such that the last n_c columns of T have full rank (denoting these columns T_2 and the remaining columns T_1), then the choice:

$$L = -T_2^{-1}T_1B_u \tag{5.16}$$

ensures that the mismatch between the anti-windup augmented and unconstrained closed-loop inputs and outputs can be described by a system of the form (5.9). Then, through certain assumptions [14] X is guaranteed to decay to zero, implying that the unconstrained response recovery and internal and external stability are ensured.

To demonstrate this, consider that $TA_{cl} = HT$, from (5.15), and $TL_{cl} = 0$, from(5.16), then it can be seen that (5.14), in the coordinates (Z, X) where $Z = T\bar{X}$, takes the form

$$\begin{aligned} \dot{X} &= A_{cl}\bar{X} + B_u[sat(K_1X + K_2Z + u_l) - u_l] \\ Z &= HZ \end{aligned} \tag{5.17}$$

where $K_1 = D_cC - C_cT_2^{-1}T_1$ and $K_2 = C_cT_2^{-1}$. Note that the invertibility of T_2 makes the coordinate transformation from $(\bar{X})$ to (X, Z) invertible. Necessarly, $A + BK_1$ and H are Hurwitz. Note that $Z(0) = 0$ and clearly the subspace $(X, Z) : Z = 0$ is invariant. So the mismatch between y and y_l and between u and u_l is governed by the dynamical system

$$\begin{aligned} \dot{X} &= AX + B[sat(K_1X + u_l) - u_l] \quad X(0) = 0 \\ Y &= CX \\ U &= K_1X \end{aligned} \tag{5.18}$$

In [14], a procedure for constructing (whenever possible) the transformation matrix T was presented. The procedure is based on the decomposition of the matrix A_{cl} as:

$$A_{cl} = U\Lambda U^{-1} \tag{5.19}$$

where $U, \Lambda \in \mathbb{R}^{n+n_c} \times \mathbb{R}^{n+n_c}$, and Λ has the following block triangular structure:

$$\lambda = \begin{bmatrix} \Lambda_1 & * \\ 0 & \Lambda_2 \end{bmatrix} \tag{5.20}$$

with $\Lambda_1 \in \mathbb{R}^n \times \mathbb{R}^n$, $\Lambda_2 \in \mathbb{R}^{n_c} \times \mathbb{R}^{n_c}$. Now, whenever the matrix formed by the last n_c rows and columns of U^{-1} is invertible, then T can be chosen as the last n_c rows of U^{-1}, i.e., $T = [0 \quad I]U^{-1}$, where I is an $n_c \times n_c$ identity matrix. It is straightforward to see that this choice of T satisfy (5.15) with $H = \Lambda_2$.

5.3 MODERN ANTI-WINDUP TECHNIQUES

5.3.1 *Modern anti-windup objectives*

Modern anti-windup techniques have been developed with the aim to give a systematic design procedure that ensures stability guarantees and optimal performance recovery. These modern anti-windup can be categorized into two different approaches: Direct Linear Anti-Windup (DLAW) and Model Recovery Anti-Windup (MRAW). Both of them can be seen as full-authority or external anti-windup thus, as seen in section 5.1, the small signal preservation objective is naturally enforced by the anti-windup structure. The remaining three objectives (internal stability, external stability and unconstrained response recovery) can be enforced by quantitative performance indexes, on which modern techniques relies, to define the anti-windup design algorithm [34].

The idea is, instead of pursuing asymptotic convergence directly for these objectives, to consider a different measure of signals that implies asymptotic convergence in many situations. That measure is the $\mathcal{L}_2$ norm: whenever the square of the norm of a signal can be integrated locally, the square root of the corresponding infinite integral is called the $\mathcal{L}_2$ norm of the signal. For a signal $t \mapsto x(t)$ the symbol $\|x\|_2$ will be used to denote the $\mathcal{L}_2$ norm. In particolar,

$$\|x\|_2 := \sqrt{\int_0^t |x(\tau)|^2 d\tau}. \tag{5.21}$$

When the $\mathcal{L}_2$ norm of a signal $t \mapsto x(t)$ is finite, the signal is said to belong to $\mathcal{L}_2$, written $x(\dot{)} \in \mathcal{L}_2$, and its size is said to be square integrable. The choice of $\mathcal{L}_2$ norm is motivated, according to Barbalat's lemma[15], to the following fact:

> For a scalar, non-negative, uniformly continuous signal $t \mapsto f(t)$, if the infinite integral of the square of the function is bounded, then necessarily the function values converge to zero as t becomes arbitrarily large.

This fact and the qualitative objectives of anti-windup augmentation motivate pursuing a solution that guarantees the size of the state of the anti-windup augmented closed-loop system is square integrable, and also that the size of the difference between this state and the state of the unconstrained closed-loop system is square integrable.

It is acknowledged here that square integrability is pursued also because it is a mathematically tractable problem that admits convenient synthesis algorithms. In particular, that algorithms involve solving a set of linear matrix inequality (LMI) in order to certify internal stability or quantify external performance.

5.3.1.1 *Internal and external stability via the standard $\mathcal{L}_2$ gain*

It is reasonable to assume that the size of the performance plant output z is related to the size of the state of the plant and, based on the discussion above, that if z belongs to $\mathcal{L}_2$, then this state will converge to zero. This condition is a type of detectability assumption. Therefore, one way to pursue internal and $\mathcal{L}_2$ external stability is the following: guarantee, by means of anti-windup augmentation, that there exist $\gamma > 0$ and $\beta > 0$ satisfying

$$\|z\|_2 \leqslant \beta |x_{cl}(0)| + \gamma \|w\|_2 \tag{5.22}$$

for initial conditions and disturbances that are expected during operation. This is a standard performance characterization for closed-loop control systems with exogenous inputs. The smallest possible value of γ that can be used is typically called the $\mathcal{L}_2$ gain from w to z. The anti-windup synthesis objective can then be formulated as to minimize the $\mathcal{L}_2$ gain γ from w to z. Unless the plant is exponentially stable, it is impossible to ensure a global finite $\mathcal{L}_2$ whatever the anti-windup augmentation is. However, it is always possible to enforce a finite gain over a local region, the aim in this case is to maximize the finite gain region while still minimizing γ.

5.3.1.2 *Internal and external stability and URR via the URR gain*

As before, it is reasonable to suppose that the size of the difference between the performance plant output z and the performance plant output in the unconstrained closed-loop system z_l is related to the size of the plant state mismatch $x_p - x_{p,l}$ and that if $z - z_l$ belongs to $\mathcal{L}_2$, then this state mismatch will converge to zero. Therefore, one way to pursue unconstrained response recovery (and, from the properties of the unconstrained closed-loop system, also internal and external stability for exogenous inputs that cause $u_l(t)$ to converge in an $\mathcal{L}_2$ sense to the region where $sat(\cdot)$ is linear) is the following: guarantee,

by means of anti-windup augmentation, that there exist $\gamma > 0$ and $\beta > 0$ satisfying

$$\|z - z_l\|_2 \leqslant \beta |x_{aw}(0)| + \gamma \|u_l - sat(u_l)\|_2 \tag{5.23}$$

for initial conditions and unconstrained controller signals that are expected during operation. Also in this case the anti-windup design objective is to minimize γ, and the optimal valued is commonly called unconstrained response recovery (URR). It will not be possible, even with anti-windup augmentation, to induce a finite URR gain globally unless the plant is exponentially stable. On the other hand, it will always be possible to induce a regional finite URR gain. As is the case for standard $\mathcal{L}_2$ gain, there is always a tradeoff between the guaranteed region of successful operation and the URR gain over this region.

5.3.1.3 *Sector characterization of nonlinearities*

In order to arrive at LMIs when checking the internal stability and $\mathcal{L}_2$ external stability for feedback loops with saturations or deadzones, one typically inscribes the saturation or deadzone into a conic region and applies the S-procedure (see appendix A). To understand the idea behind inscribing a nonlinearity into a conic region, consider a scalar saturation function. Figure 5.7 contains, on the left, the block diagram of the saturation function and, on the right, the graph of the saturation function. The figure emphasizes that the graph of the saturation function is contained in a conic sector delimited by the line passing through the origin with slope zero and the line passing through the origin with slope one. The deadzone nonlinearity, as shown in figure 5.8, is also contained in this sector. In fact, the sector contains any scalar nonlinearity with the property that its output y always has the same sign as its input u and y has a magnitude that is never bigger than that of u. This condition can be expressed mathematically using the quadratic inequality $y(u - y) \geqslant 0$, which, when focusing on the deadzone nonlinearity where $y = q$, becomes

$$q(u - q) \geqslant 0. \tag{5.24}$$

This condition says that $qu \geqslant q^2$, which captures the sign and magnitude information described above.

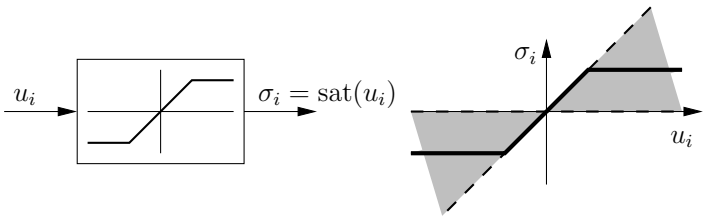


Figure 5.7: The scalar saturation function and its sector properties.

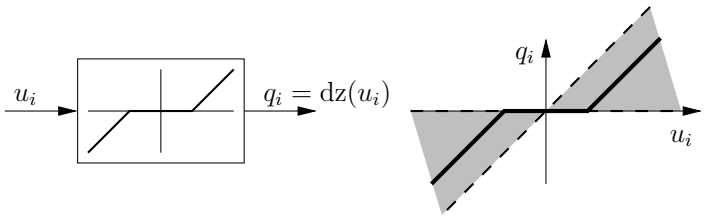


Figure 5.8: The scalar deadzone function and its sector properties.

For a decentralized nonlinearity where each component of the nonlinearity is inscribed in the sector described above, the quadratic condition

$$\sum_{i=1}^{n_u} w_i q_i (u_i - q_i) \geqslant 0. \tag{5.25}$$

holds, where q_i are the components of the output vector y, u_i are the components of the input vector u, and w_i are arbitrary positive weightings. This can also be written as

$$q^T W(u - q) \geqslant 0, \tag{5.26}$$

where W is a diagonal matrix consisting of the values w_i.

A sector characterization of nonlinearities introduces some conservativeness since the analysis using sectors will apply to any nonlinearity inscribed in the sector. The payoff in using sector characterizations is that the mathematical description, in terms of quadratic inequalities, is compatible with the analysis of feedback systems using quadratic functions. Indeed, the S-procedur permits, combining the quadratic inequality describing the sector with the quadratic inequalities involved in the directional derivative, to arrive at LMIs for the analysis of feedback loops with sector nonlinearities.

5.3.2 *Direct linear anti-windup*

The direct linear anti-windup (DLAW) was the first systematic method relying on LMI-based optimization problem (see appendix A) to design the anti-windup augmentation in order to formally ensure stability and performance properties. A first static DLAW was proposed by Mulder et al.(2001) [20], more recently, extensions of the approach involving dynamic compensators have been proposed in [35, 34]. In this section the basic results will be presented considering dynamic anti-windup schemes, since the static schemes can be derived as particular cases.

5.3.2.1 *DLAW analysis and synthesis*

The target for this class of solutions, are linear saturated plants in the form

$$\begin{aligned} \dot{x}_p &= A_p x_p + B_{pu} sat(u) + B_{pw} w \\ y &= C_p x_p + D_{pu} sat(u) + D_{pw} w \\ z &= C_z x_p + D_{zu} sat(u) + D_{zw} w \end{aligned} \tag{5.27}$$

where $x_p \in \mathbb{R}^{n_p}$, $u \in \mathbb{R}^{n_u}$, $w \in \mathbb{R}^{n_w}$, $y_p \in \mathbb{R}^{n_p}$ are the state, the input, the exogenous input and the measured output vectors of the plant, respectively. $z \in \mathbb{R}^{n_z}$ is the regulated output used for performance purposes. Considering the plant (5.27), it is assumed that an n_c-order dynamic output stabilizing controller

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c u_c + B_{cw} w + v_1 \\ y_c &= C_c x_c + D_{cu} u_c + D_{cw} w + v_2 \end{aligned} \tag{5.28}$$

has been designed, where $u_c \in \mathbb{R}^{n_p}$ is the controller input, $y_c \in \mathbb{R}^{n_u}$ the output and v_1, v_2 are the two anti-windup output signals. A natural assumption is that the above controller provides stability and the required performances of the closed-loop system when no saturation occurs, i.e. the unconstrained closed-loop system (see figure 5.1) with interconnections: $sat(u) = u = y_c$, $u_c = y$, $v_1 = v_2 = 0$ is globally asymptotically stable. In other words, assuming also well-posedness of the interconnections, i.e. the matrix $\Delta = I - D_c D_{pu}$ is non singular, the controller is required to make the unconstrained system closed-loop state matrix

$$\hat{A} = \begin{bmatrix} A_p + B_{pu}\Delta^{-1} D_c C_p & B_{pu}\Delta^{-1} C_c \\ B_c(I_p + D_{pu}\Delta^{-1} D_c) C_p & A_c + B_c D_{pu}\Delta^{-1} C_c \end{bmatrix} \tag{5.29}$$

necessarily Hurwitz.

The DLAW strategy consists in selecting the anti-windup filter $\mathcal{F}$ in figure 5.4 as a full-authority anti-windup augmentation that produce the signal $v = [v_1 \; v_2]^T$

$$\begin{aligned} \dot{x}_{aw} &= A_{aw} x_{aw} + B_{aw}(sat(u) - u) \\ v_1 &= C_{aw,1} x_{aw} + D_{aw,1}(sat(u) - u) \\ v_2 &= C_{aw,2} x_{aw} + D_{aw,2}(sat(u) - u) \end{aligned} \tag{5.30}$$

where $x_{aw} \in \mathbb{R}^{n_{aw}}$ is the anti-windup state, $u_{aw} = sat(u) - u$ is the anti-windup input and $[v_1 \; v_2]^T \in \mathbb{R}^{n_c + n_u}$ is the anti-windup output. The goal of DLAW design is to compute suitable matrices A_{aw}, B_{aw}, C_{aw}, D_{aw} so that the so-called anti-windup augmented closed-loop system (5.27), (5.28), (5.30) satisfies desirable stability and performance property. For the sake of brevity, here only full order schemes ($n_{aw} = n_p + n_c$) with some reference to static version ($n_{aw} = 0$) as

particular cases, are recalled. The most common performance measure, optimized by DLAW strategies, is the input-output gain from w to z that, by (5.22), can be expressed as the inequality $\|z\|_2 \leqslant \gamma\|w\|_2$.

Considering the usual interconnections $u = y_c$, $u_c = y$, and by expressing the saturation nonlinearity in terms of the dead-zone nonlinearity $dz(u) = sat(u) - u = q$ the systems (5.27), (5.28), (5.30) can be combined to obtain the following augmented closed-loop system

$$\begin{aligned} \dot{x}_{cl} &= A_{cl}x_{cl} + B_{1cl}q + B_{2cl}w \\ y_{cl} &= C_{1cl}x_{cl} + D_{11cl}q + D_{11cl}w \\ z &= C_{2cl}x_{cl} + D_{21cl}q + D_{22cl}w \end{aligned} \tag{5.31}$$

where

$$\begin{aligned} A_{cl} &= \begin{bmatrix} \hat{A} & B_v C_{aw} \\ 0 & A_{aw} \end{bmatrix}, \quad B_{1cl} = \begin{bmatrix} B_q + B_v B_{aw} \\ B_{aw} \end{bmatrix}, \quad B_{1cl} = \begin{bmatrix} B_2 \\ 0 \end{bmatrix}, \\ C_{1cl} &= \begin{bmatrix} C_1 & C_{v1} D_{aw} \end{bmatrix}, \quad D_{11cl} = D_1 + C_{v1} D_{aw}, \\ C_{2cl} &= \begin{bmatrix} C_2 & C_{v2} C_{aw} \end{bmatrix}, \quad D_{21cl} = D_2 + C_{v2} D_{aw}, \\ D_{12cl} &= \Delta^{-1}(D_{cw} + D_c D_{pw}) \quad D_{22cl} = D_{zw} + D_{zu}\Delta^{-1}(D_{cw} + D_c D_{pw}) \end{aligned} \tag{5.32}$$

and

$$\begin{aligned} B_V &= \begin{bmatrix} B_{pu}\Delta^{-1}[0\ I_{n_u}] \\ B_c D_{pu}\Delta^{-1}[0\ I_{n_u}] + [I_{n_c}\ 0] \end{bmatrix}, \quad B_q = \begin{bmatrix} B_{pu}(I_{n_u} + \Delta^{-1} D_c D_{pu}) \\ B_c D_{pu}(I_{n_u} + \Delta^{-1} D_c D_{pu}) \end{bmatrix}, \\ B_2 &= \begin{bmatrix} B_{pu}\Delta^{-1}(D_{cw} + D_c D_{pw}) + B_{pw} \\ B_c B_{pu}\Delta^{-1}(D_{cw} + D_c D_{pw}) + B_{cw} + B_c D_{pw} \end{bmatrix}, \\ C_1 &= \begin{bmatrix} \Delta^{-1} D_c C_p & \Delta^{-1} C_c \end{bmatrix}, \quad C_2 = \begin{bmatrix} C_z + D_{zu}\Delta^{-1} D_c C_p & D_{zu}\Delta^{-1} C_c \end{bmatrix}, \\ C_{v1} &= \Delta^{-1} \begin{bmatrix} 0 & I_{n_u} \end{bmatrix} \quad C_{v2} = D_{zu}\Delta^{-1} \begin{bmatrix} 0 & I_{n_u} \end{bmatrix}, \\ D_1 &= \Delta^{-1} D_c D_{pu} \quad D_2 = D_{zu}(I_{n_u} + \Delta^{-1} D_c D_{pu}) \end{aligned} \tag{5.33}$$

In order to develop LMI conditions for the compensator synthesis, as already pointed out in section 5.3.1.3, the following generalized sector characterization of the dead-zone function, first introduced in [4], is commonly exploited: define the set $S(u_{sat}) := \{u \in \mathbb{R}^{n_u}, w \in \mathbb{R}^{n_u} : -u_{sat} \leqslant u - w \leqslant u_{sat}\}$ then the following holds

Lemma 5.1. *If u and w belongs to the set $S(u_{sat})$, then the nonlinearity $q(u) = sat(u) - u$ satisfy the following inequality*

$$q(u)^T S^{-1}(q(u) + w) \leqslant 0 \tag{5.34}$$

for any diagonal positive definite matrix S

Based on this characterization, in [3] the following sufficient condition for the global stability of the anti-windup augmented closed-loop system (5.31) is provided

Proposition 5.1. *If there exist a symmetric positive matrix* $Q \in \mathbb{R}^{n\times n}$, *where* $n = n_{aw} + n_c + n_p$, *a diagonal matrix* $S \in \mathbb{R}^{n_u \times n_u}$ *and a positive scalar* gamma *such that the following condition hold:*

$$\begin{bmatrix} QA_{cl}^T + A_{cl}Q & B_{1cl}S - QC_{1cl}^T & B_{2cl} & QC_{2cl}^T \\ * & -2(S + 2D_{11cl}S) & -D_{12cl} & SD_{21cl}^T \\ * & * & -I & D_{22cl}^T \\ * & * & * & -\gamma^2 I \end{bmatrix} < 0 \quad (5.35)$$

then

- *If* $w = 0$, *the origin of system (5.31) is globally asymptotically stable;*
- *The closed loop system trajectories are bounded for any initial condition and any* $w(t) \in \mathcal{L}_2$;
- *The system is externally* $\mathcal{L}_2$ *stable with*

$$\int_0^T z(t)^T z(t)dt \leqslant \gamma^2 \int_0^T w(t)^T w(t)dt + \gamma^2 x_{cl}(0)^T Q^{-1} x_{cl}(0),\ T \geqslant 0. \quad (5.36)$$

Proof. Consider the quadratic Lyapunov function $V(x_{cl}) = x_{cl}^T Q x_{cl}$. Then both internal and external stability are ensured for any $x_{cl}(0)$ if

$$\dot{V}(x_{cl}) < \gamma^2 w^T w - z^T z \quad (5.37)$$

Lemma 5.1 hold globally if $w = u$, hence results

$$q^T S^{-1}(q + u) \leqslant 0. \quad (5.38)$$

Therefore, applying the S-procedure to the two inequalities above (5.37) and (5.38) results

$$\dot{V}(x_{cl}) + \frac{1}{\gamma^2} z^T z - w^T w - 2q^T S^{-1}(q + u) < 0 \quad (5.39)$$

which, by Schur's complement, is equivalent to (5.35). Then it is easy to verify that when $w = 0$, $\dot{V}(x_{cl}) < 0$, hence global asymptotic stability trivially follows. The condition (5.36) is obtained by integrating (5.39), then Lemma 5.1 and positive definiteness of V ($\dot{V}(x_{cl}(T)) > 0$ for $T > 0$) yields

$$\begin{aligned} \int_0^T z^T z dt &\leqslant \gamma^2 \dot{V}(x_{cl}(0)) + \gamma^2 \int_0^T w^T w dt \\ &\leqslant \gamma^2 x_{cl}(0)^T Q^{-1} x_{cl}(0) + \gamma^2 \int_0^T w^T w dt. \end{aligned} \quad (5.40)$$

□

As long as the synthesis problem is faced, the condition (5.36) become not convex, in particular bilinear matrix inequality (BMI) arises, since the product between the matrix variables A_{aw}, B_{aw}, C_{aw}, D_{aw}, Q and S appears in several inequality terms. However in the case of full order anti-windup a convex characterization can be obtained as stated in the next result

Proposition 5.2. *There exists an anti-windup controller (A_{aw}, B_{aw}, C_{aw}, D_{aw}) in the form (5.30) such that the conditions of Proposition 5.1 are satisfied if there exist two positive definite symmetric matrices X, $Y \in \mathbb{R}^{(n_c+n_p)\times(n_c+n_p)}$ and a positive scalar γ such that the following conditions hold:*

$$\begin{bmatrix} \hat{A}X + X\hat{A} & XB_2 & C_2^T \\ * & -I & D_{22cl}^T \\ * & * & -\gamma I \end{bmatrix} < 0 \tag{5.41}$$

$$\begin{bmatrix} Y_1 A_p^T + A_p Y_1 & B_{pw} & Y_1 C_z^T \\ * & -I & D_{zw}^T \\ * & * & -\gamma^2 I \end{bmatrix} < 0 \tag{5.42}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} > 0 \tag{5.43}$$

$$\tag{5.44}$$

where Y_1 is the upper left corner square block, with dimension n_p, of Y.

Proof. First, the matrix Q defined in Proposition 5.1 can be partitioned as follow:

$$Q = \begin{bmatrix} Y & N^T \\ N & W \end{bmatrix}, \quad Q^{-1} = \begin{bmatrix} X & M^T \\ M & W \end{bmatrix}, \quad M^T N = I - XY \tag{5.45}$$

then, four matrices are define as:

$$\Psi_1 = \begin{bmatrix} Y\hat{A}^T + \hat{A}Y & \hat{A}N^T & B_q S - Y^T C_1^T & B_2 & YC_2^T \\ N\hat{A}^T & 0 & -NC_1^T & 0 & NC_2^T \\ (B_q S - Y^T C_1^T)^T & B_2^T & -2S - D_1 S - SD_1^T & -D_{22cl} & SD_2^T \\ C_2 Y & C_2 N^T & D_2 S & D_{22cl} & -\gamma^2 I \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ B_v^T & 0 & -C_{v1}^T & 0 & C_{v2}^T \end{bmatrix}$$

$$G = \begin{bmatrix} N & W & 0 & 0 & 0 \\ 0 & 0 & S & 0 & 0 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} A_{aw} & B_{aw} \\ C_{aw} & D_{aw} \end{bmatrix}$$

$$\tag{5.46}$$

It can be checked that the inequality (5.35) can be written as

$$\Psi_1 + F^T \Omega G + G^T \Omega^T F < 0 \tag{5.47}$$

thus, by using the Elimination Lemma (see appendix A), the relation (5.47) is equivalent to:

$$\mathcal{N}_F^T \Psi_1 \mathcal{N}_F < 0, \tag{5.48}$$

$$\mathcal{N}_G^T \Psi_1 \mathcal{N}_G < 0, \tag{5.49}$$

where $\mathcal{N}_F$ and $\mathcal{N}_G$ denote any basis of the null spaces of F and G, respectively. The basis of $Ker(G)$ can be defined as

$$\mathcal{N}_G = \begin{bmatrix} X & 0 & 0 \\ M & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}, \tag{5.50}$$

it follows that (5.48) is equivalent to the relation (5.41). By the same way, the basis of $Ker(F)$ can be defined as

$$\mathcal{N}_F = \begin{bmatrix} [I_{n_p}\ 0] & 0 & B_{pu} & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & D_{zu} & 0 & I \end{bmatrix}. \tag{5.51}$$

On the other hand, by noting also that the following properties are satisfied:

$$\begin{aligned} [I_{n_p}\ 0]B_v - B_{pv}C_{v1} &= 0 \\ -D_{zu}C_{v1} + C_{v2} &= 0 \\ B_{pu}SB_q^T \begin{bmatrix} I_{n_p} \\ 0 \end{bmatrix} + [I_{n_p}\ 0]B_q SB_{pu}^T &= 0 \\ D_{zu}(-2S - 2D_1 S)D_{zu}^T + 2D_2 S D_{zu}^T &= 0 \end{aligned} \tag{5.52}$$

and by considering $Y_1 \in \mathbb{R}^{n_p \times n_p}$ the upper left corner of Y, it follows that (5.42) is equivalent to (5.51). Finally, relation (5.43) allow one to verify the existence of a positive definite matrix Q, and therefore of a positive definite matrices Y and X that satisfy (5.45). □

It is further to notice that Proposition 5.2 does not provide a constructive method to synthesize the anti-windup filter, such conditions can be found in the particular case of static DLAW [4] where only D_{aw} needs to be computed. However, numerically tractable synthesis algorithm for dynamic DLAW can be defined by fixing some of the variables, as in the following example

- Minimize γ under the LMI constraints (5.41), (5.42), (5.43) with respect to the variables γ, X and Y;
- Compute Q as the solution of (5.45);
- Fix Q in the inequality (5.35) and solve the convex feasibility problem with respect to the variables A_{aw}, $\bar{B}_{aw}$, C_{aw}, $\bar{D}_{aw}$. Then, compute $B_{aw} = \bar{B}_{aw}S^{-1}$ and $D_{aw} = \bar{D}_{aw}S^{-1}$.

5.3.2.2 *DLAW algorithms*

In the previous section the DLAW is analysed. In particular, sufficient condition for the global stability of the anti-windup augmented closed-loop system and a convex characterization for the existence of full-order dynamic anti-windup that ensure global properties are reported. Finally, an example of synthesis algorithm is proposed.

Many others DLAW synthesis algorithm can be defined to ensure global or regional properties and for static or dynamic anti-windup. In [34] an extensive list and descriptions of these algorithm can be found. In table 5.1 a brief description of the algorithms tested in the chapters that follow is listed below:

1. *Static full-authority global* DLAW: Simple architecture, very commonly used but not necessarily feasible for any exponentially stable plant. Global input-output gain is optimized.
2. *Static full-authority regional* DLAW: Extends the applicability of Algorithm 1 to a larger class of systems by requiring only regional properties. Regional input-output gain is optimized.
3. *Dynamic plant-order full-authority global* DLAW: Dynamic anti- windup, with state dimension equal to that of the plant, overcomes the applicability limitations of Algorithm 1. Feasible for any loop containing an exponentially stable plant. Global input-output gain is optimized.
4. *Dynamic plant-order full-authority regional* DLAW: Extends the applicability of Algorithm 3 by only requiring regional properties. Applicable to any loop. Regional input-output gain is optimized.

Table 5.1 comparatively illustrates the applicability, architectures, and guarantees characterizing each algorithm (asterisks mean that some restrictions apply). The applicability is stated in terms of properties of the linear plant involved in the saturated control system, namely, exponentially stable (all the eigenvalues in the open left half plane), marginally stable (same as in the previous case, with possible single eigenvalues on the imaginary axis), marginally unstable (all the eigenvalues in the closed left half plane), and exponentially unstable

Alg. N.	Applicability				Architecture		Guarantee
	Exp Stab	Marg Stab	Marg Unst	Exp Unst	Dyn/ Static	Ext/ FullAu	Global/ Regional
1	√				S	FA	G
2	√	√	√	√	S	FA	G
3	√				D	FA	G
4	√	√	√	√	D	FA	G

Table 5.1: Applicability, architectures and guarantees of the DLAW algorithms.

(plants with at least one eigenvalue in the right half plane). The architecture of the anti-windup solutions is characterized by the presence or not of dynamics in the anti-windup filter, and their interconnection properties: external or full authority. Finally, the guarantees on the compensated closed loop are distinguished as global or regional.

5.3.3 *Model recovery anti-windup*

Model recovery anti-windup (MRAW) was first proposed by Teel and Kapoor (1997) in [30], followed by [35, 34]. This approach has been called $\mathcal{L}_2$ anti-windup for a long time. The term MRAW is used for these algorithms because the structure they use recovers the unconstrained plant model as seen from the viewpoint of the unconstrained controller in order to prevent the system from misbehaving when saturation takes place.

5.3.3.1 *MRAW analysis and synthesis*

The target of these algorithms are linear saturated plants in the form

$$\begin{aligned} \dot{x}_p &= A_p x_p + B_{pu} sat(u) + B_{pw} w \\ y &= C_p x_p + D_{pu} sat(u) + D_{pw} w \\ z &= C_z x_p + D_{zu} sat(u) + D_{zw} w \end{aligned} \tag{5.53}$$

where $x_p \in \mathbb{R}^{n_p}$, $u \in \mathbb{R}^{n_u}$, $w \in \mathbb{R}^{n_w}$, $y_p \in \mathbb{R}^{n_p}$ are the state, the input, the exogenous input and the measured output vectors of the plant, respectively. $z \in \mathbb{R}^{n_z}$ is the regulated output used for performance purposes. Considering the plant (5.53), is assumed that an n_c-order dynamic output stabilizing controller

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c(y + v1) + B_{cw} w \\ y_c &= C_c x_c + D_{cu}(y + v1) + D_{cw} w + v_2 \end{aligned} \tag{5.54}$$

has been designed, where $u_c \in \mathbb{R}^{n_p}$ is the controller input $y_c \in \mathbb{R}^{n_u}$ the output and v_1, v_2 are the two output signals of an external anti-windup.

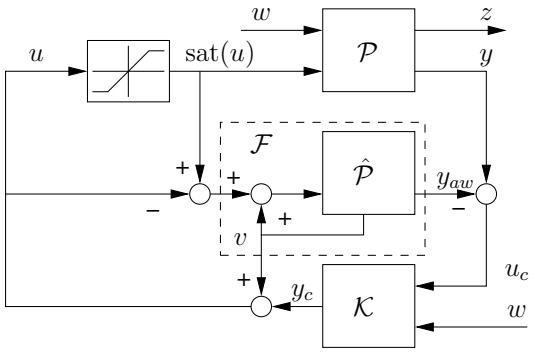


Figure 5.9: A typical block diagram representation of MRAW as an external anti-windup augmentation.

The MRAW augmentation, represented in figure 5.9, is based on selecting the anti-windup compensator as a dynamical system incorporating a model $\hat{P}$ of the plant (5.53). In particular, the compensator is selected as follow:

$$\begin{aligned} \dot{x}_{aw} &= A_p x_{aw} + B_{pu}(sat(u) - u) \\ y_{aw} &= C_p x_{aw} + D_{pu}(sat(u) - u) \\ z_{aw} &= C_z x_{aw} + D_{zu}(sat(u) - u) \\ v_1 &= -y_{aw} \\ v_2 &= k(x_{aw}) \end{aligned} \tag{5.55}$$

where the signal v_2 is purposely left unspecified because it corresponds to a degree of freedom to be exploited in the anti-windup design. The MRAW output v_1 has the effect to modify the controller input as $u_c = y + v_1 = y - y_{aw}$, which corresponds to the unconstrained system response y_l. Therefore, the controller acts as it is connected to the unconstrained plant keeping it from misbehaving and making the MRAW design completely independent of the controller dynamics.

By noticing that the anti-windup filter (5.55) keeps track, via x_{aw}, of the mismatch from the saturated plant and the unconstrained plant $x_{aw} = x_p - x_{p,l}$, the MRAW aims to design the signal v_2 to drive x_{aw} to zero in order to force the plant state x_p to recover the unconstrained response. In brief, the anti-windup goal can be interpreted as a bounded stabilization problem; i.e. v_2 has to be selected in order to drive to zero, or to keep small x_{aw} in spite of the signal y_c. In this context the unconstrained controller output y_c can be regarded as a sort of disturbance, that enters in the saturation function along with v_2, shifting the saturation levels and making the nonlinearity time-varying. This control problem has been extensively considered in the literature, and several solutions have been made available within the

MRAW architecture in order to improve the anti-windup augmented system performance. Here, just a few algorithms are presented in order to show how, also, the MRAW problem can be formulated by means of LMI constrained optimization problems, at least in its simplest version. The algorithms are based on linear compensators in the form (5.55) with v_2 computed as a linear feedback law

$$v_2 = Kx_{aw} + L[sat(u) - u] = Kx_{aw} + L[sat(v_2 + y_c) - y_c] \quad (5.56)$$

and exponentially stable linear plants. In the literature non trivial extensions to unstable plants [31], possibly involving nonlinear laws for the signal v_2 [8], have been proposed along with MRAW solution for special classes of nonlinear plants [19].

Constructive design algorithms are usually laid down considering the anti-windup compensator (5.55) expressed in the equivalent form

$$\begin{aligned} \dot{x}_{aw} &= A_p x_{aw} + B_{pu} v_2 + B_{pu}(sat(u) - u) \\ y_{aw} &= C_p x_{aw} + D_{pu} v_2 + D_{pu}(sat(u) - u) \\ z_{aw} &= C_z x_{aw} + D_{zu} v_2 + D_{zu}(sat(u) - u) \\ v_1 &= -y_{aw} \\ v_2 &= (I - L)^{-1} K x_{aw} + (I - L)^{-1} L(sat(u) - u) \end{aligned} \quad (5.57)$$

where the interconnection law $u = v_2 + y_c$ has been exploited to explicit the signal v_2. When $L \neq 0$ an implicit loop need to be solved in order to implement the scheme, hence the design algorithm has to ensure also well-posedness of this algebraic loop.

The first simple algorithm example ensures global exponential stability of the constrained system, even if no other performance indexes are optimized and with the only assumption that the plant is exponentially stable. The design of matrices K, L can be casted into an LMI constrained optimization problem and relies on the global sector characterization of the saturation function (see Lemma 5.1). It can be outlined as follow:

- solve the following LMI feasibility problem in the variable $Q = Q^T > 0$, $U > 0$ and diagonal, γ, X_1, X_2:

$$\begin{bmatrix} QA_p^T + A_p Q & B_{pu} U + X_1^T \\ UB_{pu}^T + X_1 & X_2 + X_2^T - 2U \end{bmatrix} < 0 \quad (5.58)$$

- select v_2 as in (5.56), with $K = X_1 Q^{-1}$, $L = X_2 U^{-1}$, and construct the anti-windup filter (5.57).

The LMI in (5.60) ensures global quadratic stability with respect to a Lyapunov candidate $V = x_p^T Q^{-1} x_p$ of the augmented closed-loop system. A special solutions of the algorithm above consists in setting $v_2 = 0$, i.e. $K = 0$, $L = 0$, hence no degrees of freedom are exploited in

the anti-windup design, and by (5.57), it is straightforward to verify that the resulting compensator dynamics will be an exact copy of the plant. Thus, the so-called internal model (IMC) control-based anti-windup strategy [37] is obtained. It is obvious that this technique relies on the plant stability properties, furthermore no performance measure are optimized.

More evolved techniques have been proposed to include performance indexes into the LMI constrained optimization problem; among those the following example [35] pursues the goals of global exponential stability of the closed-loop augmented system and minimization of the following cost function

$$J = \int_0^\infty x_{aw}^T Q_p x_{aw} + v_2^T R_p v_2 \, dt \tag{5.59}$$

with Q_p, R_p two positive definite matrices, chosen as a design parameters, in a typical LQ control design fashion. The algorithm can be outlined as follow

- select two positive definite matrices Q_p, R_p and solve the following eigenvalue problem (EVP):

$$\begin{aligned} \min_{Q,U,x_1,X_2} \quad & \gamma \\ \text{s.t.} \quad & \begin{bmatrix} QA_p^T + A_pQ & B_{pu}U + X_1^T \\ UB_{pu}^T + X_1 & X_2 + X_2^T - 2U \end{bmatrix} < 0 \\ & \begin{bmatrix} QA_p^T + A_pQ + 2B_{pu}X_1 & Q & X_1^T \\ Q & -Q_p^{-1} & 0 \\ X_1 & 0 & -R_p^{-1} \end{bmatrix} < 0 \\ & \begin{bmatrix} \gamma I & I \\ I & Q \end{bmatrix} > 0, \quad Q > 0, \quad U > 0 \text{ diagonal} \end{aligned} \tag{5.60}$$

- select v_2 as in (5.56), with $K = X_1 Q^{-1}$, $L = X_2 U^{-1}$, and construct the anti-windup filter (5.57).

The first LMI in (5.60) ensures global quadratic stability with respect to a Lyapunov candidate $V = x_p^T Q^{-1} x_p$ of the augmented closed-loop system, while the other two inequalities express the minimization of the LQ index (5.59).

Another interesting approach was presented in [36]; it relies on the hypothesis that in many cases the controller output in the unconstrained control loop, that by means of the signal y_{aw} will coincide with y_c in the augmented system, converges to small values inside the saturation linear region after a transient phase, since the unconstrained controller is commonly designed to achieve fast convergence performance of the unconstrained loop. Keeping in mind

the previous considerations on how the unconstrained controller output y_c acts on the anti-windup dynamics (in the linear case), y_c can be thought as a pulse disturbance that drives x_{aw} away from the origin. Hence it seems reasonable to minimize the integral of the performance output mismatch $z_{aw} = z - z_{uc}$ forcing it to be smaller than the initial condition size of the anti-windup system state x_{aw}. Furthermore since y_c in (5.57) is multiplied by B_{pu}, only the initial conditions belonging to the image of B_{pu}. Formally, K, L are selected in order to minimize γ in the following inequality

$$\int_0^T z_{aw}^T z_{aw} dt \leqslant \gamma |x_{aw}(T)|^2 \tag{5.61}$$

where T is the smallest time such that the control action returns into the saturation linear region $\forall t \geqslant T$, and $x_{aw}(T) \in \mathrm{Im}(B_{pu})$. The measure above represents a sort of $\mathcal{H}_2$ performance index, related to the unconstrained response recovery, moreover global stability of the augmented system can be ensured applying the following procedure

- solve the following EVP in the variable $Q = Q^T > 0$, $U > 0$ and diagonal, γ, X_1, X_2:

$$\begin{aligned} \min_{Q,U,X_1,X_2} \quad & \gamma \\ \text{s.t.} \quad & \begin{bmatrix} QA_p^T + A_pQ + 2B_{pu}X_1 & X_1^T - B_{pu}X_2 - B_{pu}U & X_1^T \\ -UB_{pu}^T - X_2^T B_{pu}^T + X_1 & -2U - 2X_2 & X_2^T \\ X_1 & X_2 & -I \end{bmatrix} < 0 \\ & \begin{bmatrix} QA_p^T + A_pQ + 2B_{pu}X_1 & QC_z^T + X_1^T D_{zu}^T \\ C_zQ + D_{zu}X_1 & -I \end{bmatrix} < 0 \\ & \begin{bmatrix} \gamma I & B_{pu}^T \\ B_{pu} & Q \end{bmatrix} > 0; \end{aligned} \tag{5.62}$$

- select v_2 as in (5.56), with $K = (I + X_2U^{-1})^{-1}X_1Q^{-1}$, $L = (I + X_2U^{-1})^{-1}X_1X_2U^{-1}$, and construct the anti-windup filter (5.57).

Also in this case, the first LMI condition, provides global quadratic stability with respect to the Lyapunov candidate $V = x_p^T Q^{-1} x_p$, while the other two conditions are related to the performance index optimization. Furthermore the LMIs ensures that matrix $(I + X_2U^{-1})$ is non singular, hence the algorithm can be completed with the anti-windup gain computation.

5.3.3.2 *MRAW algorithms*

In the previous section the MRAW augmentation is analysed. In particular, algorithms that only ensure global stability of the augmented

closed-loop or that ensure global stability and optimize LQ or $\mathcal{H}_2$ performance are presented.

In a similar way, other MRAW algorithms can be stated to ensure regional stability properties in order to extend the applicability to plants that are not exponentially stable, or to improve the recovery performance by taking advantage of the trade-off between the guaranteed region of stability and the URR gain (see section 5.3.1.2). In [34] an extensive list descriptions of these MRAW algorithms can be found. In table 5.2 a brief description of the algorithms tested in the chapters that follow is listed below:

5. *Stability-based MRAW for exponentially stable plants*: Special cases include IMC anti-windup and Lyapunov-based anti-windup, neither of which creates an algebraic loop. Global exponential stability is guaranteed but no performance measure is optimized.

6. *Global* LQ*-based MRAW*: A linear quadratic performance index related to URR is optimized subject to guaranteeing global exponential stability.

7. *Regional* LQ*-based MRAW*: Extends the applicability of Algorithm 6 to any loop by requiring only regional exponential stability. Does not create an algebraic loop. A linear quadratic performance index related to small signal unconstrained response recovery is optimized.

8. *Global* $\mathcal{H}_2$*-based MRAW*: An $\mathcal{H}_2$ performance index related to the unconstrained response recovery is optimized subject to guaranteeing global exponential stability.

9. *Regional* $\mathcal{H}_2$*-based MRAW*: Extends the applicability of Algorithm 8 to any loop by requiring only regional exponential stability. An $\mathcal{H}_2$ performance index related to small signal unconstrained response recovery is optimized.

Table 5.2 comparatively illustrates the applicability, architectures, and guarantees characterizing each algorithm (asterisks mean that some restrictions apply). The applicability is stated in terms of properties of the linear plant involved in the saturated control system, namely, exponentially stable (all the eigenvalues in the open left half plane), marginally stable (same as in the previous case, with possible single eigenvalues on the imaginary axis), marginally unstable (all the eigenvalues in the closed left half plane), and exponentially unstable (plants with at least one eigenvalue in the right half plane). The architecture of the anti-windup solutions is characterized by the presence or not of dynamics in the anti-windup filter, and their interconnection properties: external or full authority. Finally, the guarantees on the compensated closed loop are distinguished as global or regional.

Alg. N.	Applicability				Architecture		Guarantee
	Exp Stab	Marg Stab	Marg Unst	Exp Unst	Dyn/ Static	Ext/ FullAu	Global/ Regional
5	√				D	E	G
6	√				D	E	G
7	√	√	√	√	D	E	R
8	√				D	E	G
9	√	√	√	√	D	E	R

Table 5.2: Applicability, architectures and guarantees of the MRAW algorithms.

6

SIMULATOR DEVELOPMENT AND SIMULATION RESULTS

In this chapter the solutions needed to solve the industrial issue are implemented and tested through Simulink models. In particular, as suggested by the system analysis of chapter 3, an inertia identification algorithm is implemented in order to ensure a correct design of the control system and of the PTP motion references. In addition the anti-windup techniques described in chapter 5 are implemented and tested, in order to analyse and demonstrate their effectiveness.

6.1 INERTIA IDENTIFICATION

In chapter 3 it was pointed out that an accurate estimation of the inertia is necessary in order to ensure a correct design of the control system and of the PTP motion references. In particular, it was noticed that an underestimation of the inertia, due to estimation inaccuracy or to unexpected overload, lead to a generation of a reference that requires more torque then the one the motor is capable to deliver. Thus, the saturation phenomenon occurs causing undesirable position oscillations. Here will be presented and tested an efficient and accurate inertia identification algorithm based on the discrete Fourier transform (DFT).

6.1.1 *Inertia identification algorithm*

The DFT of a discrete signal is defined as follow

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-j\frac{2\pi k}{T_s N}n} \tag{6.1}$$

$$= \sum_{n=0}^{N-1} x_n \cdot \left[\cos(2\pi kn/(T_s N)) - j \cdot \sin(2\pi kn/(T_s N))\right], \tag{6.2}$$

where T_s, N are the sample time and the number of samples of the discrete signal $x = [x_0\ x_1\ ...\ x_{N-1}]^T$ and $k = 0, ..., N-1$. Now, defining the vectors

$$C_k := \begin{bmatrix} \cos(\frac{2\pi T_s}{N}0) \\ \cos(\frac{2\pi k}{T_s N}1) \\ \vdots \\ \cos(\frac{2\pi k}{T_s N}(N-1)) \end{bmatrix} \quad S_k := \begin{bmatrix} \sin(\frac{2\pi T_s}{N}0) \\ \sin(\frac{2\pi k}{T_s N}1) \\ \vdots \\ \sin(\frac{2\pi k}{T_s N}(N-1)) \end{bmatrix}, \tag{6.3}$$

it can be notice that equation (6.2) can be rewritten in a vector form that can be easily implemented with Matlab:

$$X_k = C_k^T x - jS_k^T x. \tag{6.4}$$

Considering the normalization factor $1/N$ the amplitude of the fundamental component of x ($k = 1$) at frequency $1/(T_s N)$ is

$$C_{X_1} = \frac{C_1^T x - jS_1^T x}{N}. \tag{6.5}$$

The equation (6.5) is a key result for the identification procedure that follow.

By recalling the equation of electromagnetic torque and mechanical load of a PMSM

$$\tau = \frac{3}{2} p \Lambda_{mg} i_q = K_t i_q = J\ddot{\theta}_m + B\dot{\theta}_m, \tag{6.6}$$

it can be noticed that the parameter J is modelled as linear coefficient between the control signal i_q and the mechanical acceleration $\ddot{\theta}_m$. Thus, by assuming low values of the viscous friction torque $B\omega_m$ during the identification operation, the inertia J can be easily estimated by the following procedure:

- command the motor with a sinusoidal function $i(t) = \sin(2\pi f t)$ at a certain frequency $f = T_s/N$;
- estimate the acceleration $\ddot{\theta}(t)$ by the encoder position measure;
- calculate the DFT coefficient $C_{\ddot{\theta},f}$ of the acceleration signal $\ddot{\theta}(t)$;
- compute the inertia estimation by the DFT coefficient as follow

$$J = \frac{K_t}{2|C_{\ddot{\theta},f}|}. \tag{6.7}$$

6.1.2 *Inertia identification implementation*

The inertia identification procedure is implemented and tested through the Simulink model presented in figure 6.1.

The Simulink subsystem *motor*, presented in figure 6.3, simulates the modified system (see figure 6.5) that consists of the dSPACE controller, the servo drive and finally the motor. The dSPACE introduces a quantization effect due by the DAC. The servo drive, instead, introduces a current control loop (see figure 2.8) that can be approximated by a first order system with cutoff frequency equal to the closed-loop bandwidth obtained with the PI controller. Finally the motor is modelled by the mechanical load equation 6.6 and the quantization effect of the encoder.

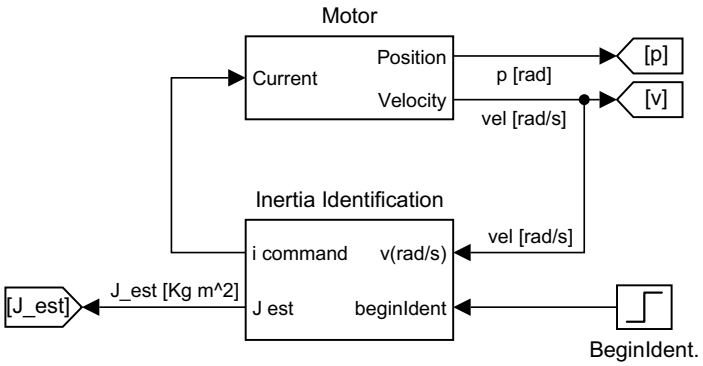


Figure 6.1: Simulink model of the inertia identification procedure.

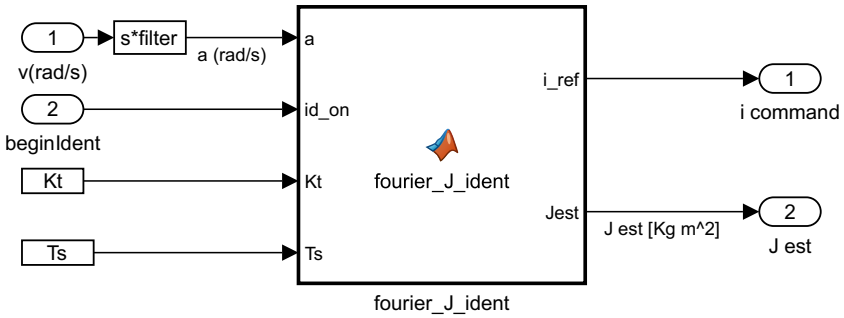


Figure 6.2: Simulink subsystem that consider the effect of the dSPACE controller, the servo drive and finally the motor.

The Simulink subsystem *inertia identification*, presented in figure 6.3, implements the inertia identification algorithm listed above. In particular it implements the algorithm through a matlab function. This function generates the sinusoidal reference and records the estimated acceleration. Then, through the equation (6.5) and (6.7) it estimates the inertia J.

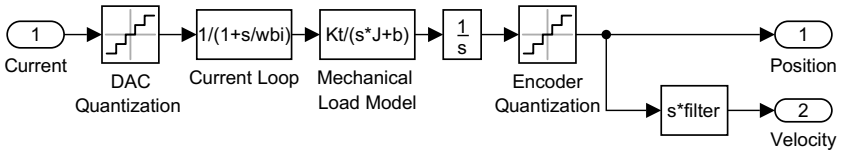


Figure 6.3: Simulink subsystem that implements the inertia identification algorithm.

6.1.3 *Inertia identification results*

The inertia identification procedure is tested driving the motor with a sinusoidal current of amplitude 1 A and frequency $1/(T_s N)$, where

the sampling time is $T_s = 1 \cdot 10^{-4}$ s and the number of samples per period is $N = 2^{12}$.

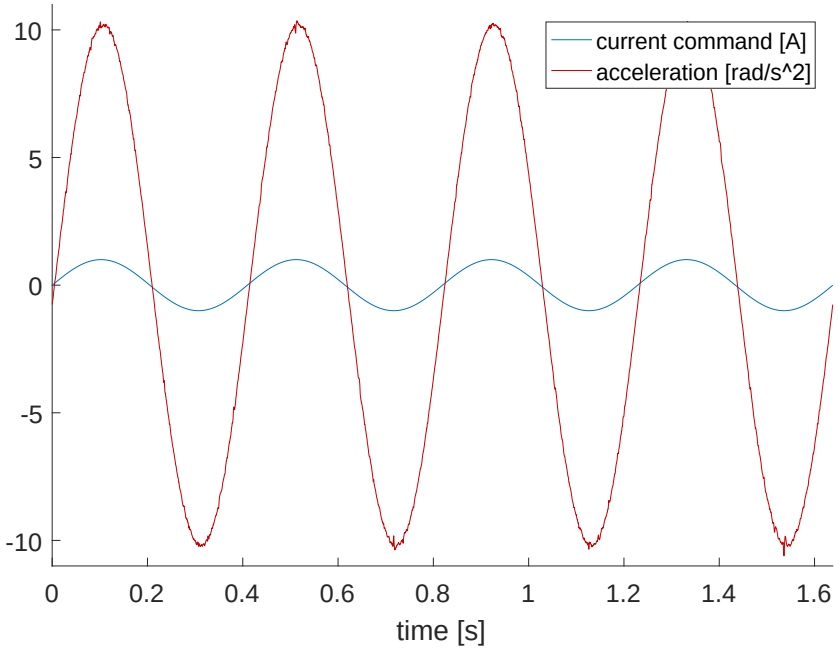


Figure 6.4: Driving current and estimated acceleration.

Nominal Inertia	$J = 9 \cdot 10^{-3}$ Kg m^2
Estimated inertia	$J_{est} = 9.104 \cdot 10^{-3}$ Kg m^2

Table 6.1: Inertia identification results.

The simulation results are depicted in table 6.1 and figure 6.4 where is reported the current command and the resultant estimated acceleration. The identification algorithm, elaborating four period of the acceleration signal after the transitory behaviour, gives and estimated inertia equal to $J_{est} = 9.104 \cdot 10^{-3}$ Kg m^2 when the nominal inertia was setted equal to $J = 9 \cdot 10^{-3}$ Kg m^2. Thus, the inertia is estimated with an error of 1.15%, that is a satisfactory estimation for the analysed application.

6.2 SIMULATOR OF THE SYSTEM

In chapter 2 was presented a modified systems that allows to implement an own position and velocity control loops. In the modified system, recalled in figure 6.5, the LAB Drivebox works as a current drive of the motor. Instead, the position and the velocity control loops are implemented in the dSPACE Controller board by emulating the controller structure already implemented in LAB Drivebox (P and PI controller in the position and velocity loops respectively) and by

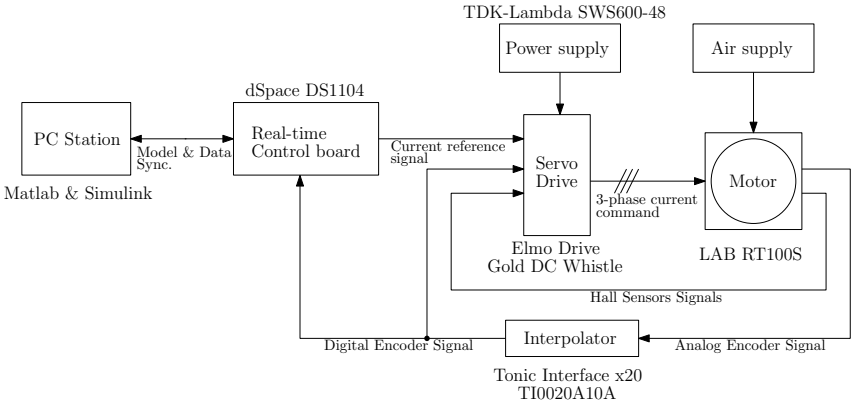


Figure 6.5: Connection representation of the elements of the modified system.

adding an anti-windup augmentation to the velocity controller. In addition, a reference generator for PTP motion references and the above described inertia identification algorithm are implemented as well in the dSPACE Controller board. Here, a simulator of the modified system just described above is described.

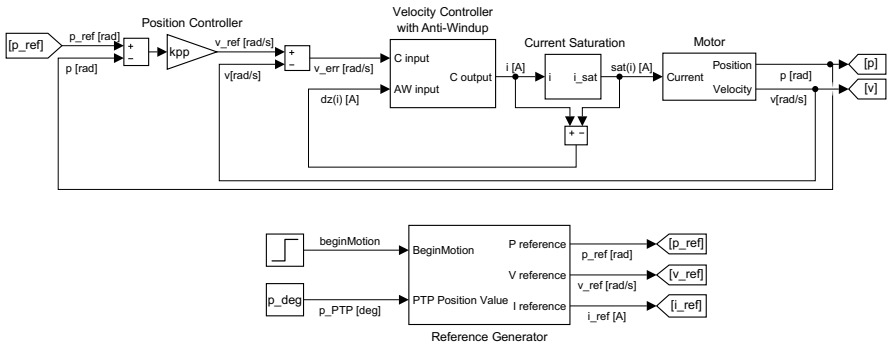


Figure 6.6: Simulink model of the analysed system.

The simulator is implemented by the Simulink model depicted in figure 6.6. It is composed of the following elements:

- the reference generator of constant acceleration motion profiles;
- the P controller, in the position loop;
- the PI controller augmented by an anti-windup compensator, in the velocity loop;
- the current saturation function that emulate the saturation imposed by the LAB Drivebox;
- the motor subsystem: with includes the effect of the dSPACE Controller, the current control loop and the mechanical load model of the motor is already described in section 6.1.2.

The subsystems *reference generator*, *current* and *velocity controller and anti-windup* of the Simulink model in figure 6.6 will be described in the following sections.

6.2.1 *Reference generator*

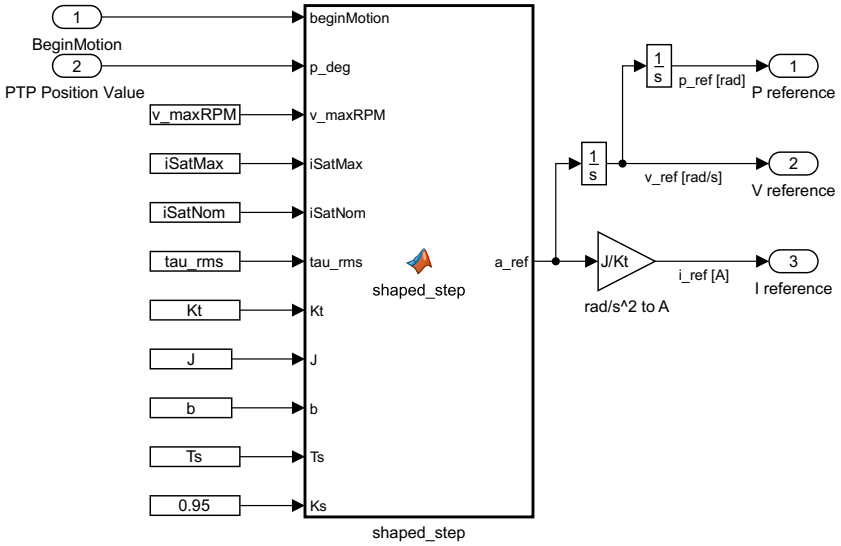


Figure 6.7: Reference generator subsystem of the model in figure 6.6.

The reference generator, presented in figure 6.7, aims to generate a constant acceleration profile for a desired position PTP motion in order to emulate the reference generator in the LAB Drivebox. The reference generator of the servo drive, above other, require to manually indicate the acceleration limit of the profile. Insteas,the Matlab function, here described, automatically computes the acceleration limit by the knowledge of the continuous and maximum current, the inertia, the viscous friction coefficient and the safety factor. If these information are consistent with the real system, the reference generator ensure the feasibility of the motion profile avoiding that saturation occurs.

The continuous and maximum acceleration limits are calculated as follow:

$$\begin{aligned} A_{max,peak} &= K_s \frac{\tau_{peak}}{J} = K_s \frac{I_{peak} K_t}{J} \\ A_{max,nom} &= K_s \frac{\tau_{nom}}{J} = K_s \frac{I_{nom} K_t}{J} \end{aligned} \tag{6.8}$$

The algorithm determines a triagular motion profile with $\lambda = T_A/T = T_D/T = 1/2$, presented in figure 6.8 case a, whenever the velocity and the continuous acceleration limits are not encountered by the generated profile. Otherwise the ratio $\lambda = T_A/T = T_D/T$ is decreased in

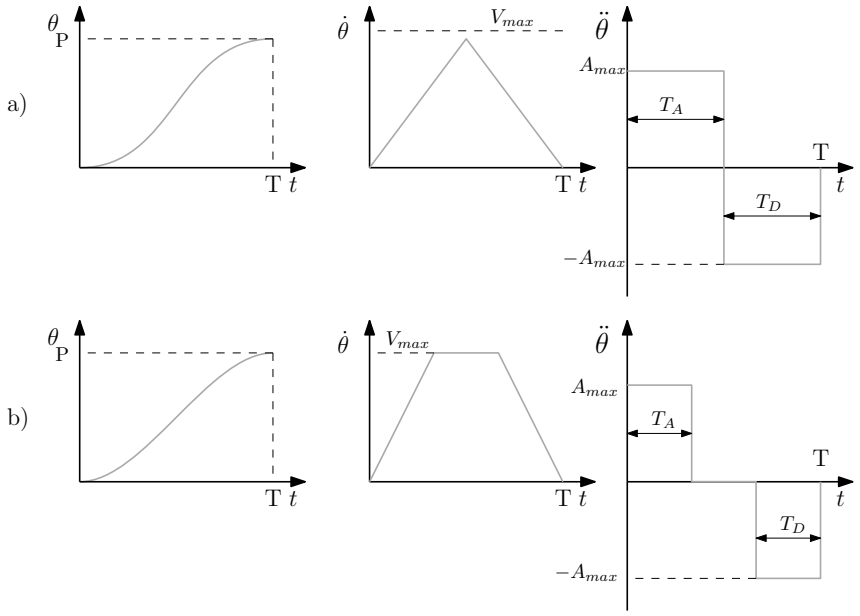


Figure 6.8: Constant acceleration PTP motion profiles: a) triangular profile, b) trapezoidal profile.

order to satisfy those limits resulting in a trapezoidal motion profile depicted in figure 6.8 case b.

The time duration T of the motion profile is calculated as follow:

$$T =; \left(\frac{p_{des}}{A_{max,peak}} \frac{1}{\lambda} \frac{1}{1-\lambda}\right)^{1/2} \tag{6.9}$$

where p_{des} is the desired position displacement.

6.2.2 *Current saturation function*

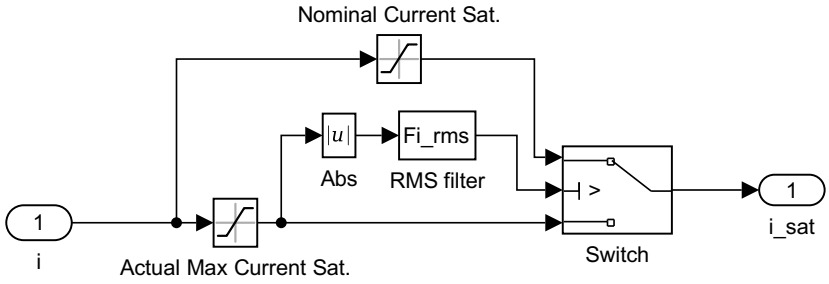


Figure 6.9: Current saturation subsystem of the model in figure 6.6.

The current saturation subsystem, presented in figure 6.9, is developed in order to emulate the current limitation of the LAB Drivebox, and to give a measure of the mismatch between the current reference and its saturated version needed for the anti-windup compensation.

The subsystem implements a saturation that occurs when the maximum current or the continuous current limit are reached. The con-

tinuous limit is checked by passing the current signal through a first order filter as described in section 2.3.2.

6.2.3 *Anti-windup augmentation*

The anti-windup techniques described in chapter 5 can be implemented in two different form: the full-authority or external anti-windup augmentation. The DLAW and OBSAW are commonly implemented in the full-authority framework. Instead, the MRAW is implemented in the external framework.

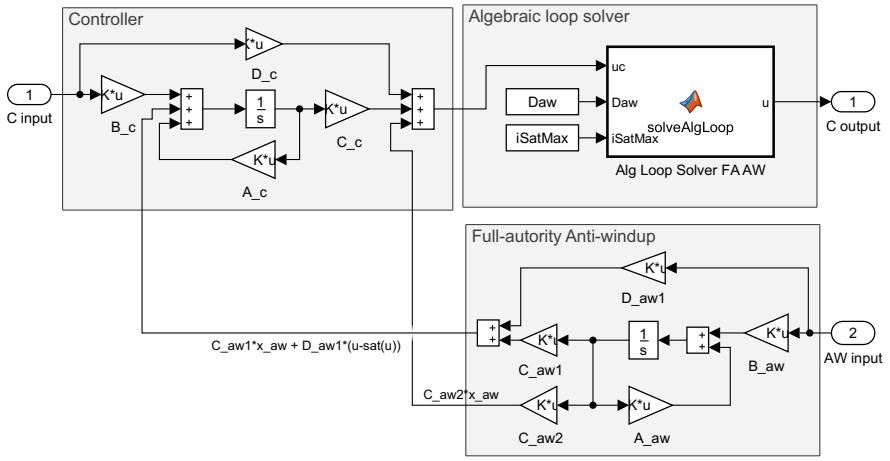


Figure 6.10: Simulink subsystem of the velocity controller with the full-authority anti-windup augmentation.

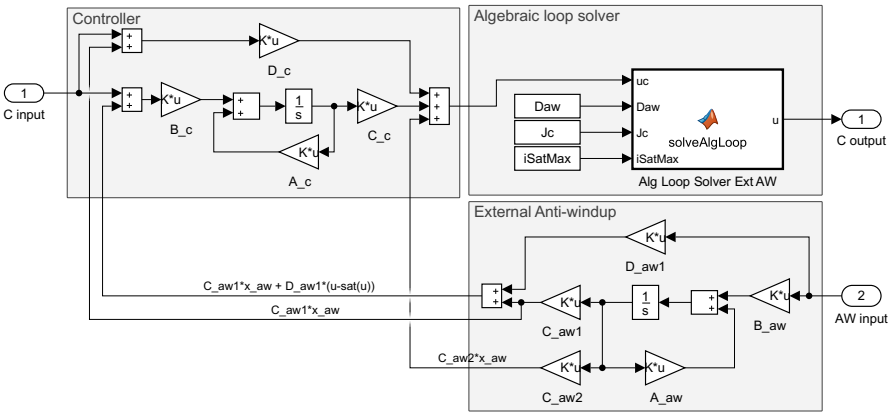


Figure 6.11: Simulink subsystem of the velocity controller with the external anti-windup augmentation.

The full-authority and external anti-windup augmentation of the velocity controller are presented in figure 6.10 and 6.11, respectively. It can be noticed that the connections, between the AWC and the controller, have been modified and a Matlab function have been added respect the common block diagram of an anti-windup augmentation.

These modification are made in order to avoid the creation of algebraic loops (see section 5.1.5.1) that don't allow to properly simulate the model and to make its code-generation.

In the case of the OBSAW the implementation can be simplified as presented in figure 6.12. Indeed, the OBSAW output affect the controller passing through its state, hence no algebraic loops are generated.

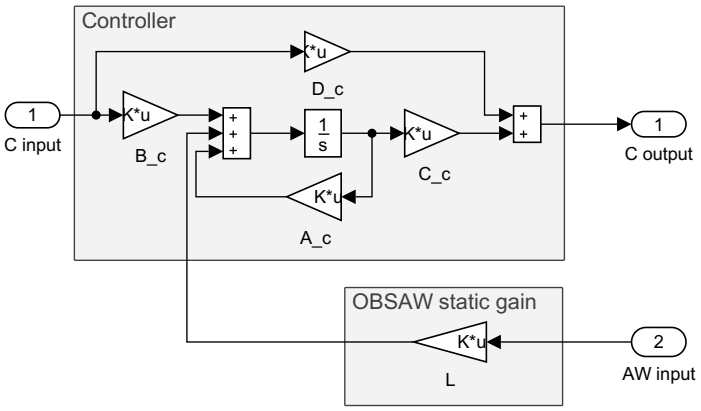


Figure 6.12: Simulink subsystem of the velocity controller with the OBSAW augmentation.

6.3 SIMULATION RESULTS

From the analysis of chapter 3 has been shown that an underestimation of the inertia, due to estimation inaccuracy or unexpected overload, can lead to saturation occurrence during a homing PTP motion operation. In this section, will be tested the anti-windup solutions described in chapter 5 that aim to increase the unconstrained response recovery respect the performance offered by the ad hoc anti-windup solution already implemented in the LAB Drivebox.

As in the analysis of chapter 3 the control system and the reference generator are tuned for a value of the inertia decreased by 20% of its original value, $J_{design} = 0.8\,J$. The control system is tuned with the same parameters used in the previous analysis and they are recalled in table 6.2.

J_{design}	0.0072 Kg m^2
K_{pi}	5.207 V/A
ω_{ii}	1131.29 Hz
K_{pv}	24.57 A/(rad/s)
ω_{iv}	12.52 Hz
K_{pv}	78.54 1/s

Table 6.2: Control system parameters with $J_{design} = 0.8$ J.

The anti-windup techniques that are implemented and tested are listed below :

1. *Static full-authority global* DLAW

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & -3.105 \\ 0 & -0.910 \end{array}\right]; \tag{6.10}$$

2. *Static full-authority regional* DLAW

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & -3.093 \\ 0 & -0.925 \end{array}\right]; \tag{6.11}$$

3. *Dynamic plant-order full-authority global* DLAW

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -68.817 & -0.047 \\ \hline 3.231 & -3.031 \\ 1.939 & -0.864 \end{array}\right]; \tag{6.12}$$

4. *Dynamic plant-order full-authority regional* DLAW

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -5.112 & -0.268 \\ \hline 40.835 & -3.054 \\ 7.428 & -0.921 \end{array}\right]; \tag{6.13}$$

5. *Stability-based* MRAW

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -1.170 & 1.311 \\ \hline 3.192 & 0 \\ -0.170 & -0.672 \end{array}\right]; \tag{6.14}$$

6. *Global* LQ*-based MRAW*

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -8.221 & 1.247 \\ \hline 3.192 & 0 \\ -1.934 & -0.688 \end{array}\right]; \tag{6.15}$$

7. *Regional* LQ*-based MRAW*

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -8.957 & 4.000 \\ \hline 3.192 & 0 \\ -2.118 & 0 \end{array}\right]; \tag{6.16}$$

8. *Global* $\mathcal{H}_2$*-based MRAW*

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -0.486 & 4.34 \\ \hline 3.192 & 0 \\ 0.0000876 & -0.0851 \end{array}\right]; \tag{6.17}$$

9. *Regional* $\mathcal{H}_2$*-based MRAW*

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} -14.833 & 4.407 \\ \hline 2.553 & 0 \\ -3.611 & 0.102 \end{array}\right]; \tag{6.18}$$

10. *OBSAW*

$$\left[\begin{array}{c|c} A_{aw} & B_{aw} \\ \hline C_{aw} & D_{aw} \end{array}\right] = \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & 2.601 \\ 0 & 0 \end{array}\right]; \tag{6.19}$$

To compare the performance of the anti-windup solutions the system is tested by performing a PTP motion of 180 deg that represent the homing operation. In table 6.3 is presented an overview of the results. In particular, in order to compare the unconstrained response recovery of the different solution the value T_{target} is reported in the table. It represents the time required for the position response to reach and stay at the final value with an error equal to the encoder resolution. In the following figures are presented, instead, the results of the test by the plot of position, velocity and current.

PTP deg	$\mathcal{AW}$	J_{design}	T_{target} [s]
1	//	J_{est}	0.1423
1	//	$J_{est} \cdot 0.8$	0.1434
180	//	J_{est}	0.6835
18	DLAW stat. glob.	$J_{est} \cdot 0.8$	1.636
180	DLAW stat. reg.	$J_{est} \cdot 0.8$	1.648
180	DLAW dyn. glob.	$J_{est} \cdot 0.8$	1.635
180	DLAW dyn. reg.	$J_{est} \cdot 0.8$	1.653
180	MRAW stab. based	$J_{est} \cdot 0.8$	//
180	MRAW LQ glob.	$J_{est} \cdot 0.8$	1.458
180	MRAW LQ reg.	$J_{est} \cdot 0.8$	1.441
180	MRAW $\mathcal{H}_2$ glob.	$J_{est} \cdot 0.8$	//
180	MRAW $\mathcal{H}_2$ reg.	$J_{est} \cdot 0.8$	1.1177
180	OBSAW	$J_{est} \cdot 0.8$	1.382

Table 6.3: Results overview of the tested anti-windup solutions.

6.3.0.1 *System response with DLAW*

It can be seen by the figure 6.13 and by the values of T_{trget} that the responses of the system with the four DLAW are really close. This result might be expected by the similarity of the state space matrices. However, the regional design should has led to an improvement of the performance that, unfortunately, here is not seen. The LMI design procedure, in this case, was not able to induce a smaller regional $\mathcal{L}_2$ gain even if by reducing the guaranteed region of stability.

6.3.0.2 *System response with stability-based DLAW*

The simple Stability-based MRAW, as shown in figure 6.14, results effective to guaranteed global exponential stability. However it is designed not minimizing any performance index, indeed the response is really sluggish making this solution not suitable for high-performance industrial application.

6.3.0.3 *System response with LQ-based MRAW*

Differently by the previous case the LQ-based MRAW is designed minimizing a linear quadratic cost function. Indeed it allows to get a desirable unconstrained response recovery as can be seen in figure

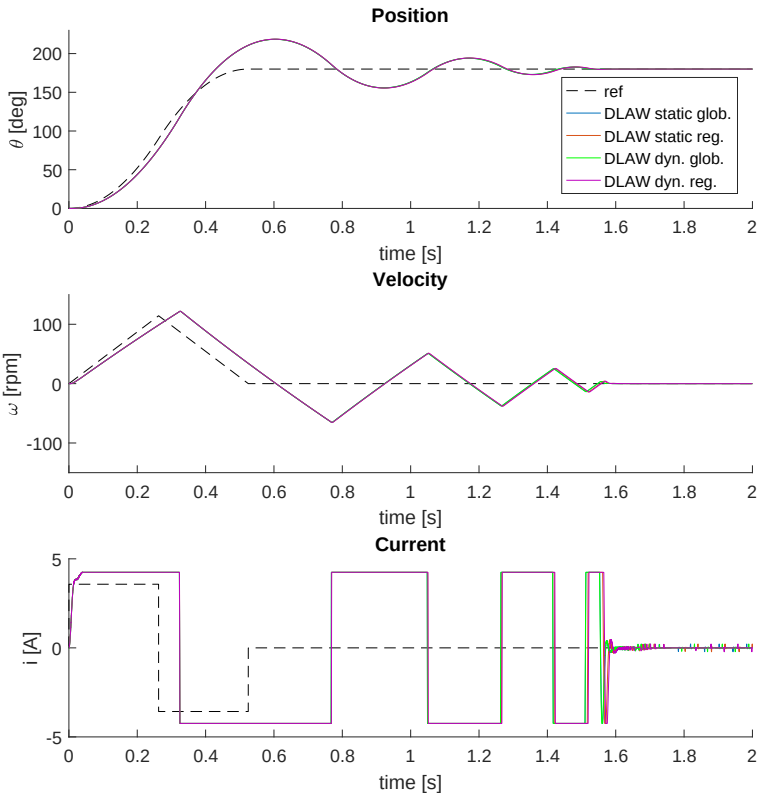


Figure 6.13: 180 deg PTP results with: static global DLAW, static regional DLAW, dynamic global DLAW, dynamic regional DLAW.

6.15. The regional version of the algorithm allows an improvement of the performance that, however, is fewer than expected.

6.3.0.4 *System response with $\mathcal{H}_2$-based MRAW*

Even if the global $\mathcal{H}_2$ MRAW is designed by minimizing a performance index, the response of the system, depicted in figure 6.16, is sluggish as in the case of the stability-based MRAW. However the regional $\mathcal{H}_2$ MRAW results in a high improvement of the unconstrained recovery performance. This algorithm shows a high effectiveness to take advantage of the trade-off between guaranteed region of stability and the URR gain.

6.3.0.5 *System response with OBSAW*

In figure 6.17 is presented the response of the system augmented by the OBSAW. Even if its design does not consider any performance consideration, the value T_{target} is low, demonstrating a unconstrained response recovery similar to the LQ-based MRAW results.

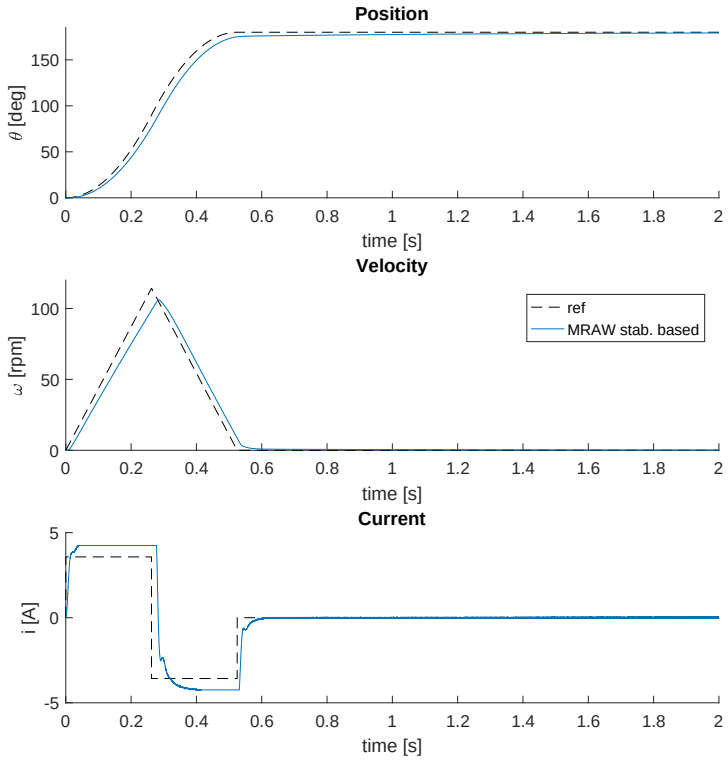


Figure 6.14: 180 deg PTP results with stability-based MRAW augmentation.

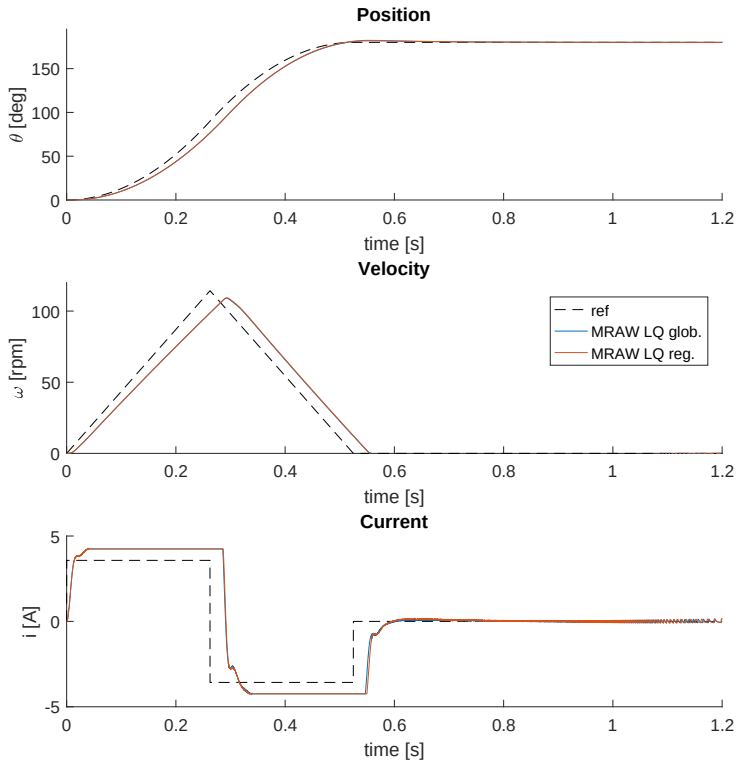


Figure 6.15: 180 deg PTP results with: global LQ-based MRAW, regional LQ-based MRAW augmentation.

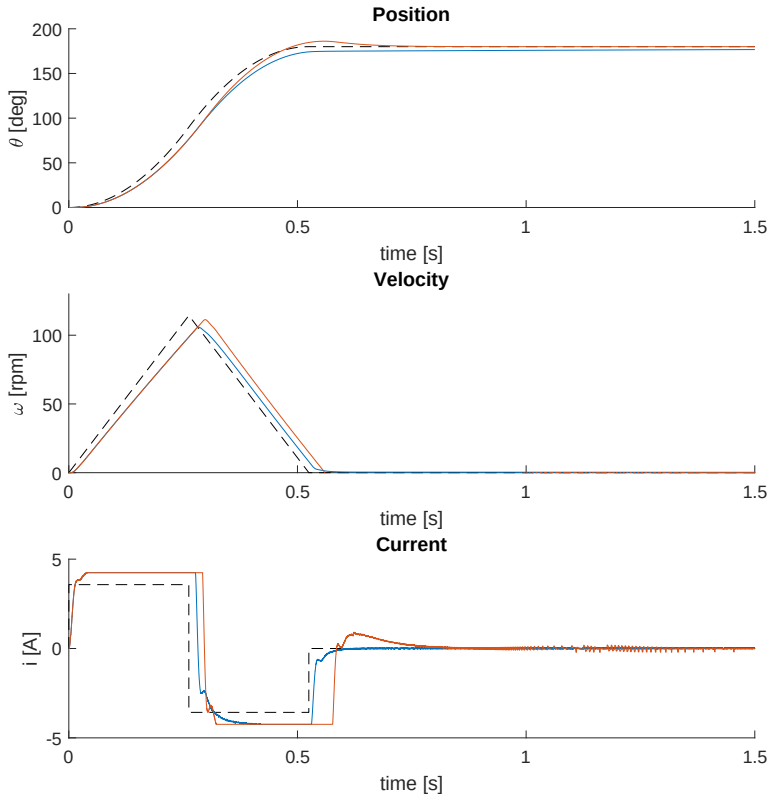


Figure 6.16: 180 deg PTP results with: global $\mathcal{H}_2$-based MRAW, regional $\mathcal{H}_2$-based MRAW augmentation.

6.3.0.6 *System response comparison*

Finally, in figure 6.18 is presented the position responses of the most effective anti-windup, one for each family: the OBSAW, dynamic global DLAW, the regional LQ MRAW and the regional $\mathcal{H}_2$ MRAW.

The figure 6.18 allow to make a comparison of the performance of the several anti-windup here analysed. What results is that the regional LQ and $\mathcal{H}_2$ MRAW allow, over others, to obtained an high response recovery by leading the response to be really close to the unconstrained response during all the motion time.

By looking at the values T_{target} in table 6.3 it is confirmed that the $\mathcal{H}_2$ MRAW is the most effective anti-windup augmentation. On the contrary the response with the LQ MRAW is slower to reach target respect the response with the OBSAW that, however, shows larger oscillations. In this case, a solution can be more preferable then the another depending by the application in which it is implemented.

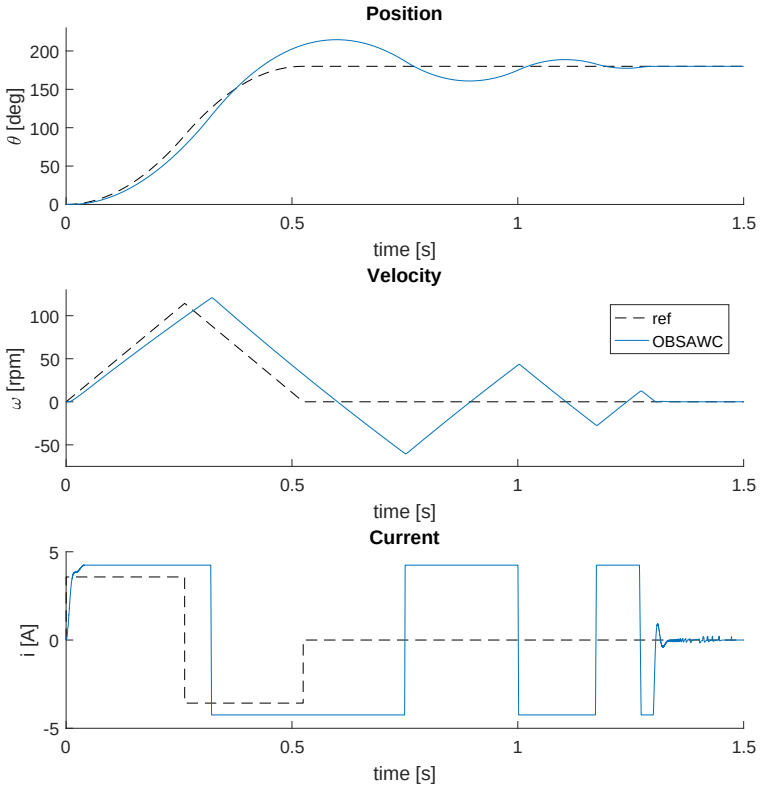


Figure 6.17: 180 deg PTP results with OBSAW augmentation.

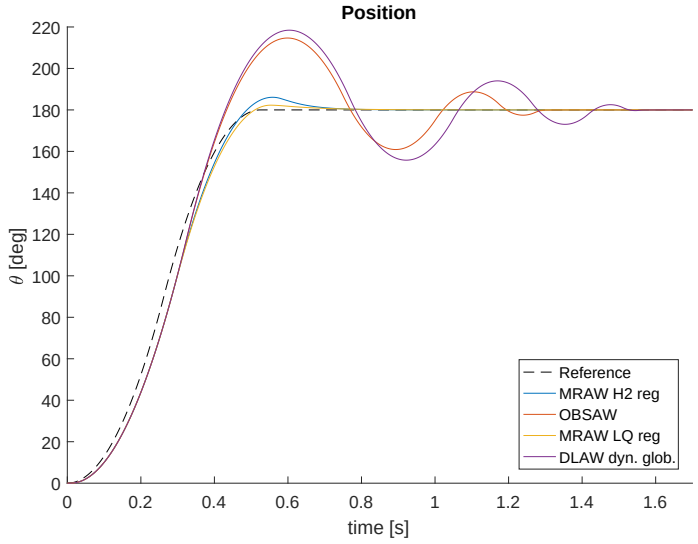


Figure 6.18: 180 deg PTP position response comparison of the most relevant anti-windup compensators.

7

TEST ON THE EXPERIMENTAL SETUP

In this chapter the solutions developed in the previous chapters are validated through tests on the experimental setup.

7.1 EXPERIMENTAL SETUP

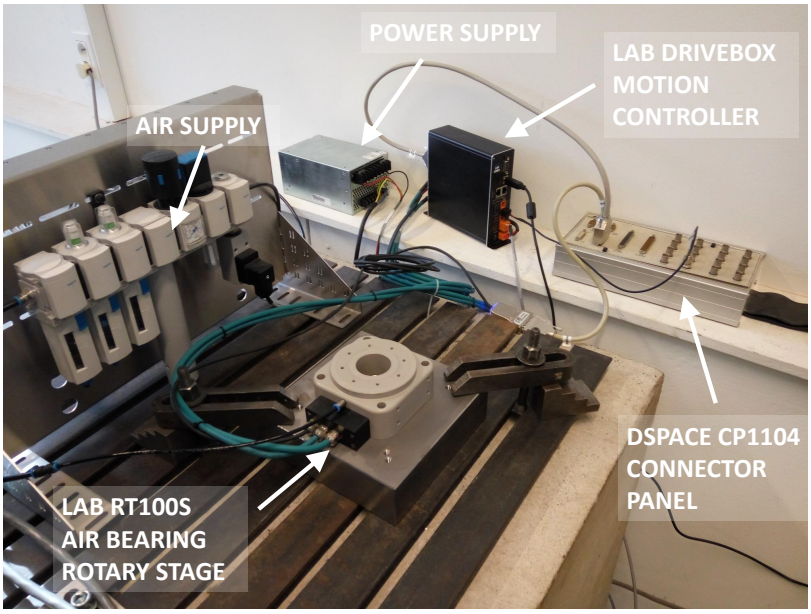


Figure 7.1: Picture of the experimental setup.

The experimental setup used to test the developed solution is presented in figure 7.1. Its connections are described in figure 7.2.

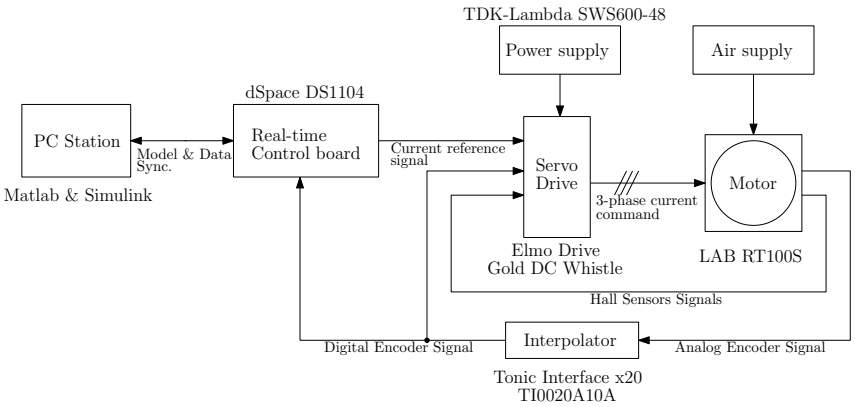


Figure 7.2: Connection representation of the elements of the modified system.

The Simulink model used to implement in the dSPACE controller the algorithms described in the previous chapters is presented in figure 7.3. Differently by the simulator, in this case the subsystem *motor* is substituted by *LAB RT100S* described in figure 7.4. In this subsystem the dSPACE real-time interface blocks are used in order to generate the analog signal that represent the current reference for the servo drive, and to read the position values given by the encoder mounted in the LAB RT100S.

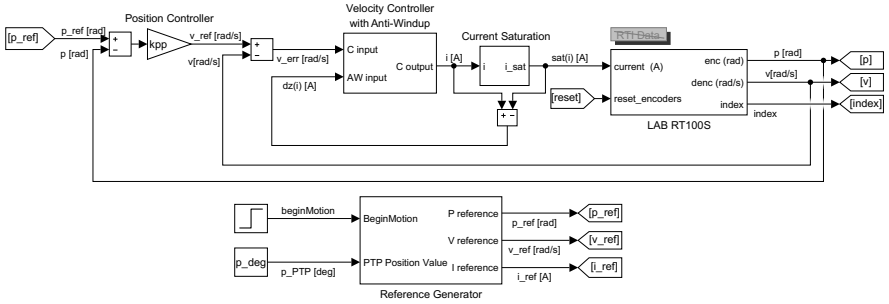


Figure 7.3: Simulink model of the experimental setup.

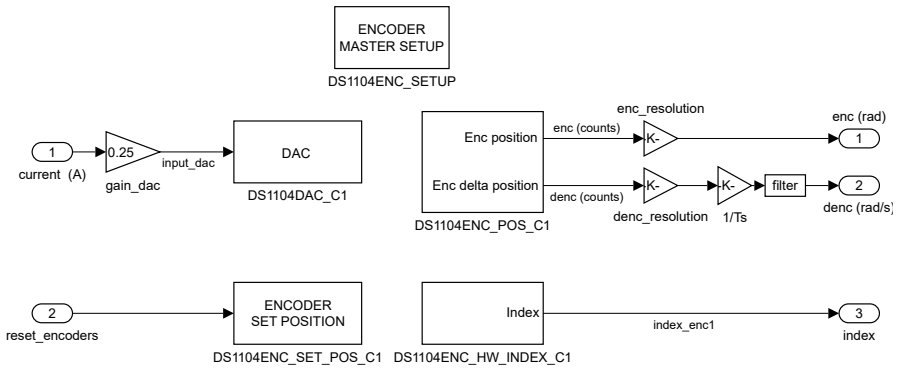


Figure 7.4: Simulink model of the experimental setup.

7.2 INERTIA IDENTIFICATION RESULTS

In order to validate the simulation results, the inertia identification procedure is tested driving the motor with a sinusoidal current of amplitude 1 A and frequency $1/(T_s N)$, where the sampling time is $T_s = 1 \cdot 10^{-3}$ s and the number of samples per period is $N = 2^9$.

Nominal Inertia	$J = 9 \cdot 10^{-3}$ Kg m^2
Estimated inertia	$J_{est} = 9.302 \cdot 10^{-3}$ Kg m^2

Table 7.1: Inertia identification results on the experimental setup.

The experimental results are depicted in table 7.1 and figure 7.5, where it is reported the current command and the resultant estimated

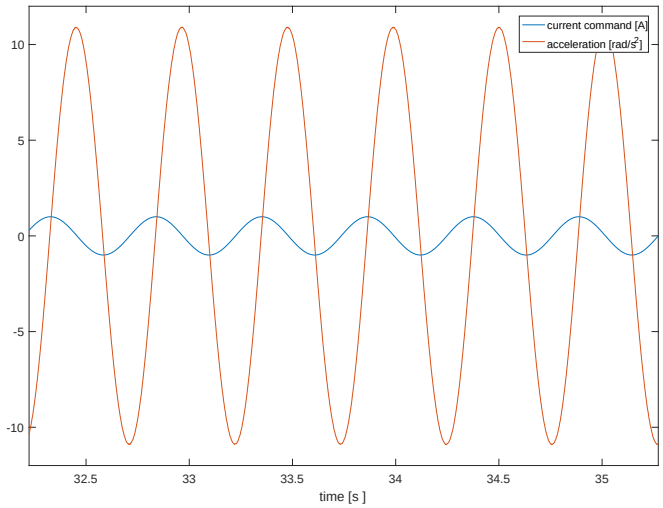


Figure 7.5: Driving current and estimated acceleration of the inertia identification procedure on the experimental setup.

acceleration. The identification algorithm, elaborating four period of the acceleration signal after the transitory behaviour, gives and estimated inertia equal to $J_{est} = 9.302 \cdot 10^{-3}$ Kg m^2 when the nominal inertia was setted to $J = 9 \cdot 10^{-3}$ Kg m^2. Thus, the inertia is estimate with an error of 3.36%, that is a satisfactory estimation for the analysed application.

7.3 EXPERIMENTAL RESULTS

In order to validate the solution developed in this thesis several test are performed in two different scenario:

1. In the first scenario the inertia of the system is accurately estimated with the proposed identification algorithm. Consequently, the control system and the reference generator are well tuned to satisfy the costumer specifications. The value of the inertia used to design is $J_{design} = J$, hence the resultant control parameters are recalled in table 3.2.

2. In the second scenario the value of the inertia used to tune the control system and the reference generator is underestimated. This error can be caused by the inaccuracy of the inertia estimation algorithm used or by an unexpected overload on the motor. That can lead to saturation occurrence during a certain PTP motion operations. The value of the inertia used to design is $J_{design} = 0.8 \cdot J$ and the resultant control parameters are recalled in table 3.3.

In each of this two scenario two PTP motion are performed: one of 1 deg representing a scanning operation, and one of 180 deg representing an homing operation.

7.3.1 *First scenario*

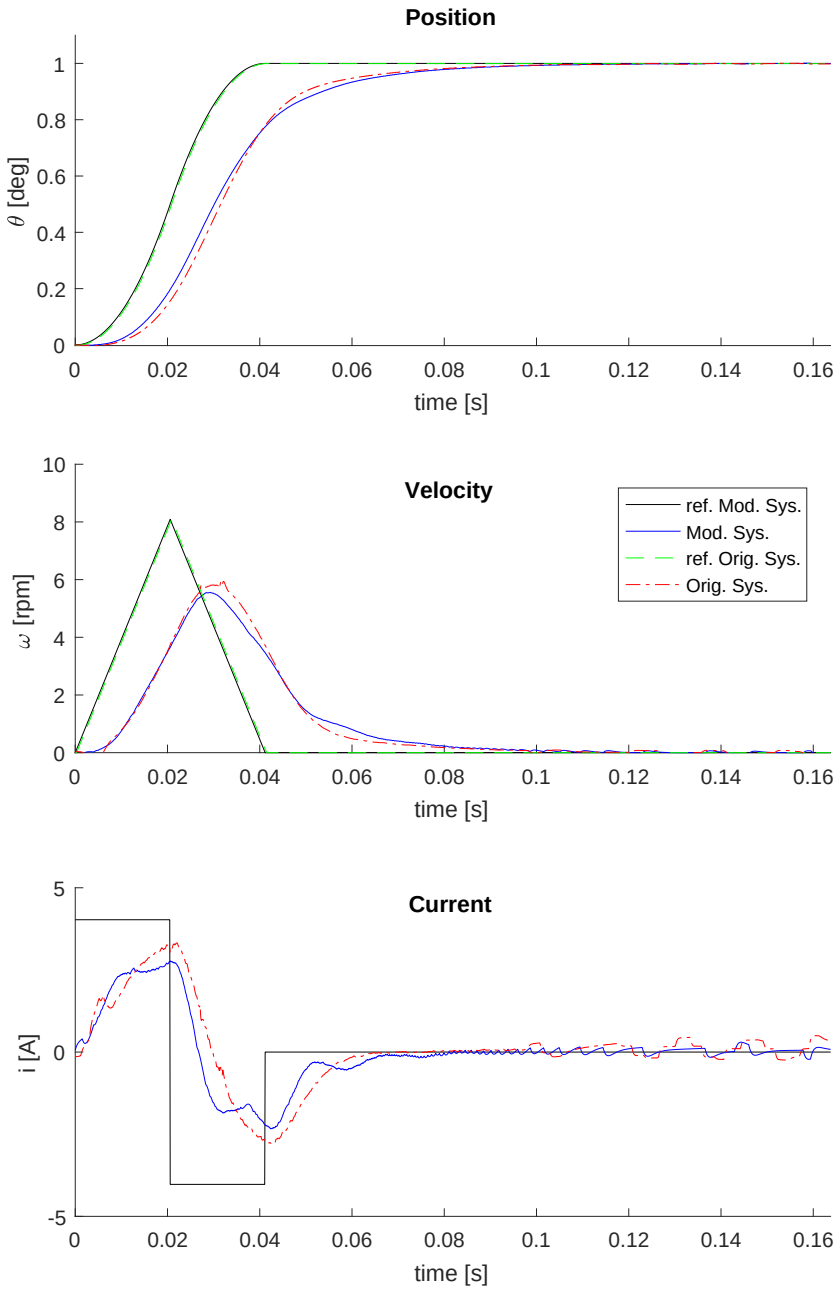


Figure 7.6: 1 deg PTP motion test on the experimental setup: position, velocity and current plot.

In chapter 3 the results of the original system was presented. Here, the system has been modified by implementing the position and ve-

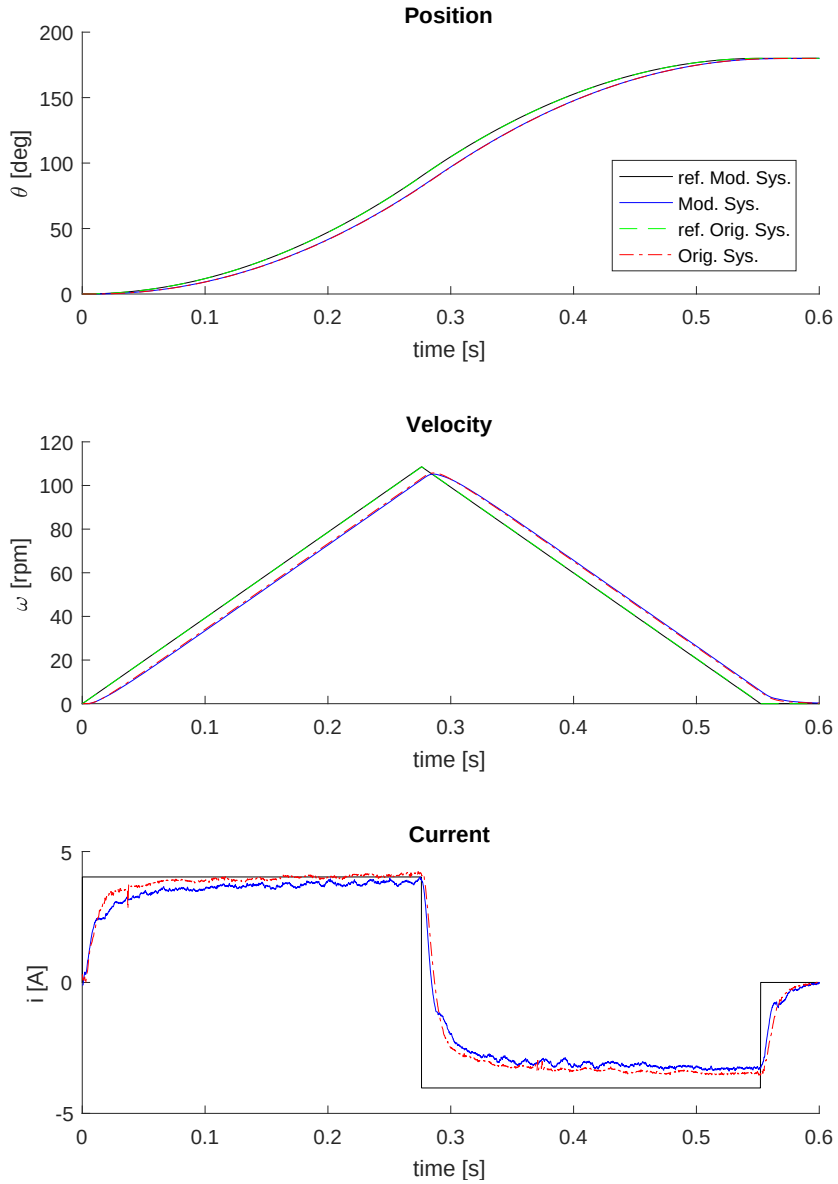


Figure 7.7: 180 deg PTP motion test on the experimental setup: position, velocity and current plot.

locity loop in the dSPACE controller. However, the controller structure has been conserved, hence similar results should be expected.

In figure 7.6, 7.6 the responses to, respectively, 1 deg, 180 deg PTP motion on the original and modified system are reported. What emerges is that the responses of two system are closed to each others validating the comparison of their results. It can be noticed that the responses are satisfactory for the analysed industrial application.

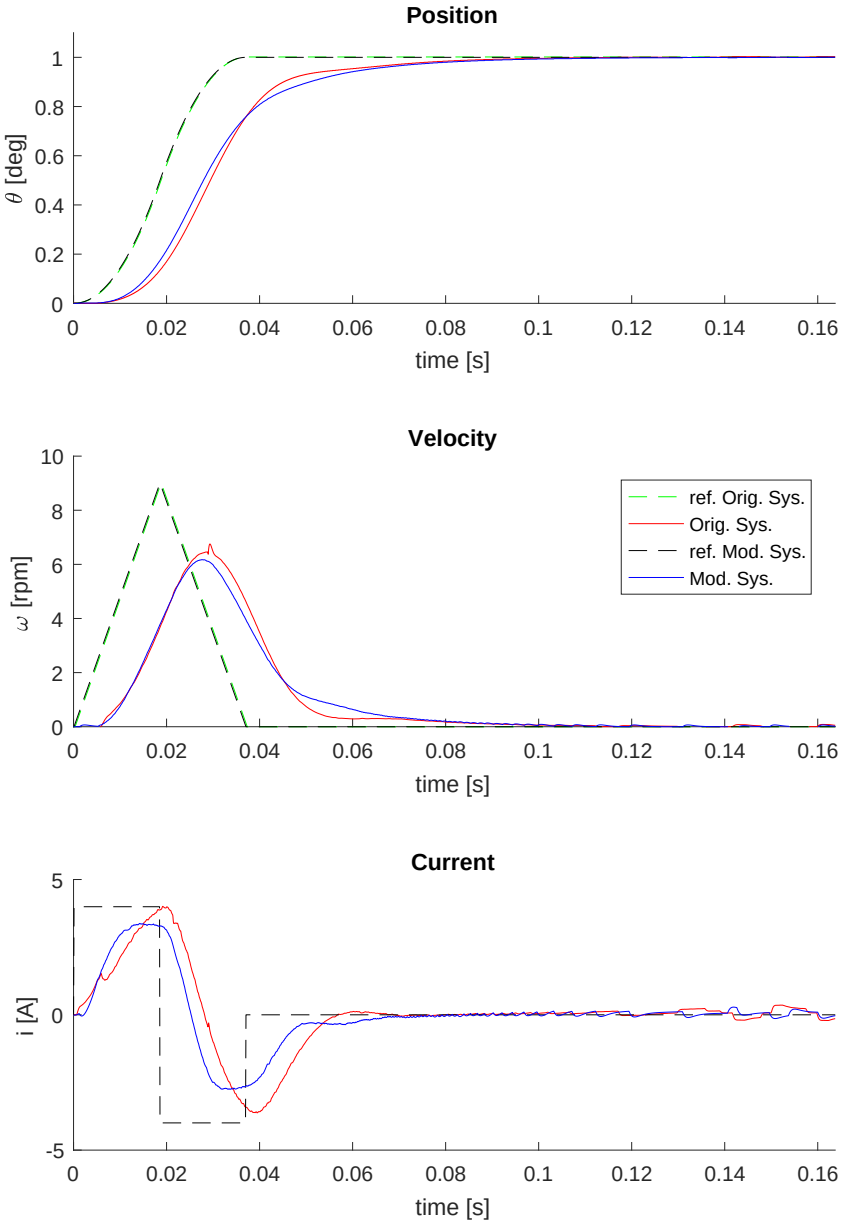


Figure 7.8: 1 deg PTP motion test on the experimental setup with $J_{design} = 0.8$ J: position, velocity and current plot.

7.3.2 *Second scenario*

In figure 7.8 the responses to 1 deg PTP motion on the original and modified system are reported. As in the previous cases, the responses of the two system are closed to each others. It can be noticed that in spite of the wrong value of the inertia used the required torque effort is low enough to avoid saturation.

Differently, saturation occurs during the responses to 180 deg PTP motion. Thus, to ensure stability and unconstrained response recovery is necessary the action of an anti-windup compensator. Based

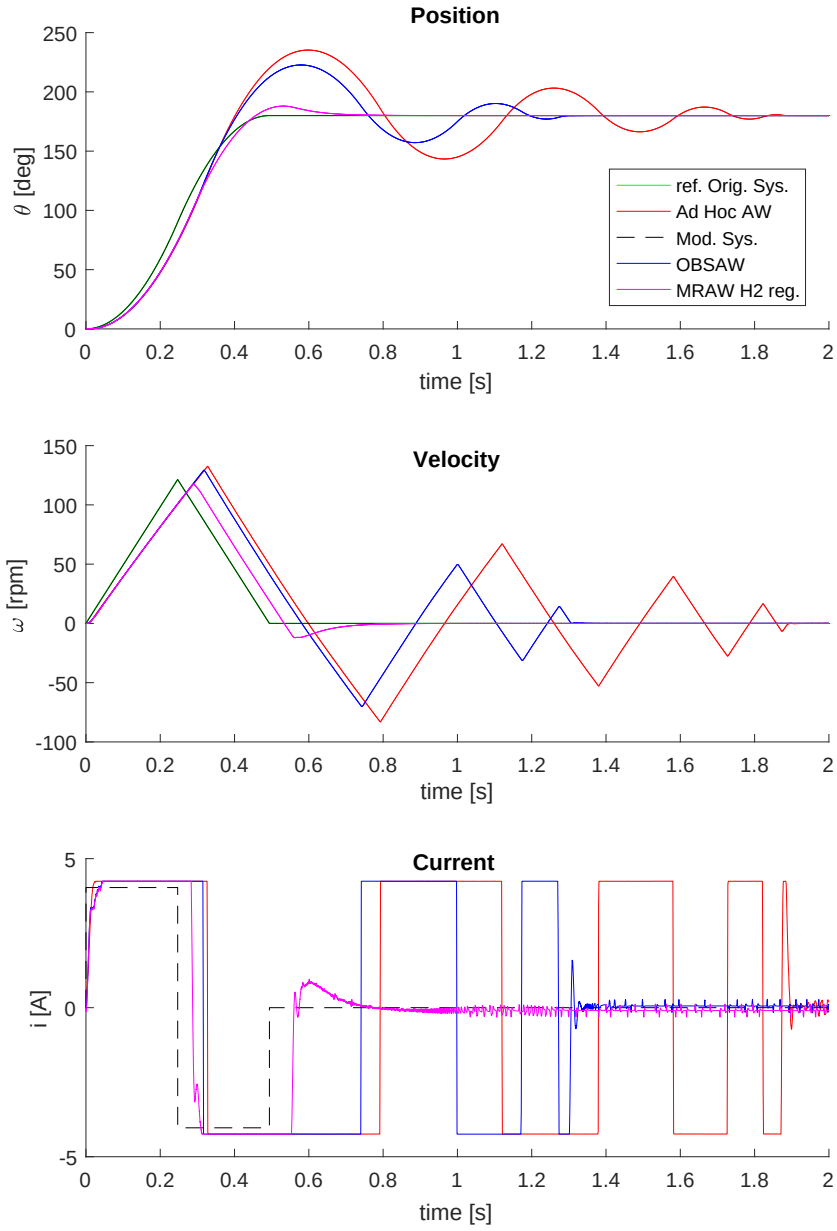


Figure 7.9: 180 deg PTP motion test on the experimental setup with $J_{design} = 0.8\,J$ and several anti-windup compensators: position, velocity and current plot.

on the results of chapter 6 the OBSAW and the regional $\mathcal{H}_2$ MRAW have been choosen to compare their effectiveness with the ad hoc anti-windup implemented in the original system. The responses comparison is presented in figure 7.9 and table 7.2.

The experimental results confirm what seen in simulation. The regional $\mathcal{H}_2$ MRAW and the OBSAW are really effective on recovering the unconstrained response and on leading to a remarkable improvement of the tracking performance, compared to what ensured by the ad hoc anti-windup. It should be noticed that with the modern anti-windup

the time needed to reach the target is almost split in half respect the value T_{target} obtained with the ad hoc anti-windup.

	$\mathcal{AW}$	J_{design}	T_{target} [s]
1 deg	//	J_{est}	0.1230
1 deg	//	$J_{est} \cdot 0.8$	0.131
180 deg	//	J_{est}	0.6522
180 deg	AdhocAW	$J_{est} \cdot 0.8$	1.975
180 deg	OBSAW	$J_{est} \cdot 0.8$	1.380
180 deg	MRAW $\mathcal{H}_2$reg.	$J_{est} \cdot 0.8$	1.166

Table 7.2: Results overview of the tested anti-windup solutions.

8

CONCLUSION

The aim of this thesis has been to study a solution of an industrial issue given by the company, LAB Motion Systems. The issue has been recognized as due to the undesirable effects of the actuator saturation. Two approaches are given in this thesis to solve the issue:

- an accurate inertia identification algorithm: in order to generate feasible references;
- modern anti-windup techniques: in order to ensure stability and high unconstrained response recovery performance.

The first approach consists of an easy challenge widely studied and implemented in industry. Indeed, many control software of servo drives already implement automatic procedure to identify the load inertia installed on the motor. Differently, the second approach consists of an involved challenge that is still not tackle by the industry even if modern anti-windup are widely described in literature. Still nowadays in industrial applications ad hoc anti-windup compensator for PI controllers are commonly used.

In this thesis the principle to understand and design the advanced anti-windup are given. In addition, has been shown that the augmentation of a control system with a modern anti-windup technique can remarkably increase the unconstrained response recovery performance and can give guarantee of stability. However, to be able to design the modern anti-windup techniques a practitioner should be willing to invest the energy required to gain competency with an LMI optimization problem and LMI solvers.

With the advanced anti-windup techniques in many application there will be no more reason to tackle the saturation problem by over sizing the actuators and hence by decreasing the efficiency of the system and by increasing the cost.

In the application analysed in this thesis the implementation of the anti-windup techniques on the dSPACE controller consists of a limitation for a future use on a commercialized product. Instead of use the dSPACE controller should be used controller more easily integrable in an industrial control system.

Finally, by noticing the similarity between the design of the modern anti-windup technique and the design of $\mathcal{H}_\infty$ robust optimal controller, both through LMI optimization, a possible future development of this work is to add the anti-windup feature to the LCToolbox (a Linear Control Toolbox), developed by MECO Research Team at KU

Leuven, that gives to the control engineers a support on the identification and on the robust control design of a system.

APPENDIX

A

LINEAR MATRIX INEQUALITY (LMI)

This appendix provides an overview of the mathematical tools and of a numerical software package commonly exploited to formulate and solve the LMI-constrained optimization problems, that as reported in chapter 5, arise in modern anti-windup solutions.

A.1 INTRODUCTION TO LMI

LMIs are matrix inequalities which are linear or affine in a set of matrix variables. They are essentially convex constraints and therefore many optimization problems with convex objective functions and LMI constraints can easily be solved efficiently using many existing software. This method has been very popular among control engineers in recent years. This is because a wide variety of control problems can be formulated as LMI problems.

A generic LMI can be expressed in the following form

$$F(x) := F_0 + x_1 F_1 + ... + x_m F_m = F_0 + \sum_{i=0}^{m} x_i F_i > 0 \tag{A.1}$$

where $x \in \mathbb{R}^m$ is the *decision variable*, $F_0, F_1, ..., F_n$ are given constant symmetric real matrices, i.e. $F_i = F_i^T$, $i = 0, ..., m$. The inequality symbol in the equation means that $F(x)$ is positive definite, i.e. $u^T F(x) u > 0$ for all nonzero $u \in \mathbb{R}^m$. This matrix inequality is linear in the variables x_i. Equation (A.1) is a sort of *explicit representation* of the LMI, while often problems are formulated letting the matrices to be the variables.

As an example, the Lyapunov inequality

$$A^T P + PA < 0 \tag{A.2}$$

where $A \in \mathbb{R}^{n \times n}$ is given and $X = X^T$ is the decision variable that can be expressed in the form of LMI (A.1) as follows: let $P_1, ..., P_m$ be a basis for the symmetric $n \times n$ matrices ($m = n(n+1)/2$), then take $F_0 = 0$ and $F_i = -A^T P_i - P_i A$.

A.1.1 *LMI feasibility problem, Eigenvalue Problem and Generalized Eigenvalue Problem*

Here some standard convex and quasi-convex problems arising in control theory are presented. The most basic problem we can think about is to determine if a set of linear matrix inequalities $F(x) > 0$ is

feasible, i.e. there exist a x_{feas} such that $F(x_{feas}) > 0$. This problem is usually refereed as *LMI feasibility problem*.

Another common convex problem is the so-called eigenvalue problem (EVP), where the objective is to minimize the maximum eigenvalue of a matrix, which depends affinely on a variable, subject to an LMI constraint. Formally a generic EVP is formulated as

$$\begin{aligned} &\min_{x} \lambda \\ &\quad \text{s.t. } \lambda - A(x) > 0,\ B(x) > 0 \end{aligned} \tag{A.3}$$

where A, B are symmetric matrices depending on the decision variable x. Equivalently an EVP can arise in the form of minimizing a linear function subject to an LMI

$$\begin{aligned} &\min c^T x \\ &\quad \text{s.t. } F(x) > 0 \end{aligned} \tag{A.4}$$

which reduces to a linear programming (LP) problem if $F(x)$ is composed by all diagonal matrices. Another equivalent form for the generic EVP, commonly appearing in control problems is

$$\begin{aligned} &\min \lambda \\ &\quad \text{s.t. } A(\lambda, x) > 0 \end{aligned} \tag{A.5}$$

where A is affine in (x, λ).

A third standard problem arising in control theory applications is the so-called generalzed eigenvalue problem (GEVP) which consists in minimizing the maximum generalized eigenvalue of a pair of matrices that depend affinely on a variable, subject to an LMI constraint. A GEVP can be expressed as

$$\begin{aligned} &\min \lambda \\ &\quad \text{s.t. } \lambda B(x) - A(x) > 0,\ B(x) > 0,\ C(x) > 0 \end{aligned} \tag{A.6}$$

where A, B, C are symmetric matrices depending affinely on x. Equivalently it can be expressed as

$$\begin{aligned} &\min \lambda_{max}(A(x), B(x)) \\ &\quad \text{s.t. } B(x) > 0,\ C(x) > 0 \end{aligned} \tag{A.7}$$

where $\lambda_{max}(A(x), B(x)$ denotes the largest eigenvalue of the matrix $B^{-1/2} A B^{-1/2}$. Hence we can see that this is a quasi-convex problem, since the constraint is convex (LMI) while the objective function is quasi-convex.

As for the EVP problem, the GEVP problem can appear in a third equivalent form

$$\begin{aligned} &\min \lambda \\ &\quad \text{s.t. } A(x, \lambda) > 0 \end{aligned} \tag{A.8}$$

where A is affine in x for fixed λ and viceversa, furthemore it satisfies the monotonic condition with respect to λ.

A.1.2 *Tricks used in LMIs*

Although many problems in control can be formulated as LMI problems, some of these problems result in nonlinear matrix inequalities. There are certain tricks which can be used to transform these nonlinear inequalities into suitable LMI forms. Some of the tricks which are often used in control are described herewith suitable examples.

CHANGE OF VARIABLES By defining new variables, it is sometimes possible to linearise nonlinear matrix inequalities.

Example (Synthesis of state feedback controller). The objective is to determine a matrix $K \in \mathbb{R}^{m\times n}$ such that all the eigenvalues of the matrix $A + BK \in \mathbb{R}^{n\times n}$ lie in the open left-half of the complex plane. Using the Lyapunov theory, it can be shown that this is equivalent to find a matrix F and a positive definite matrix $P \in \mathbb{R}^{n\times n}$ that the following inequality holds:

$$(A + BF)^T P + P(A + BF) < 0 \tag{A.9}$$

or

$$A^T P + PA + F^T B^T P + PBF < 0 \tag{A.10}$$

Note that the terms with products of F and P are nonlinear or bilinear. Let us multiply either side of (A.10) by $Q = P^{-1}$. This gives

$$QA^T + AQ + QF^T B^T + BFQ < 0 \tag{A.11}$$

This is a new matrix inequality in the variables $Q > 0$ and F. But is still nonlinear. A second new variable is defined $L = KQ$. This gives

$$QA^T + AQ + L^T B^T + BL < 0. \tag{A.12}$$

This gives an LMI feasibility problem in the new variables $Q > 0$ and $L \in \mathbb{R}^{m\times n}$.

After solving this LMI, the feedback matrix K and the Lyapunov variable P can be recovered from $F = LQ^{-1}$ and $P = Q^{-1}$. This shows that by making a change of variables, an LMI can be obtained from a nonlinear matrix inequality.

SCHUR COMPLEMENT Shur's formula is used in transforming nonlinear inequalities of convex type into LMI.

$$\begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} < 0 \tag{A.13}$$

where $Q(x) = Q(x)^T$, $R(x) = R(x)^T$ and $S(x)$ depends affinely on x, is equivalent to

$$R(x) < 0,\ Q(x) - S(x)R(x)^{-1}S(x)^T < 0. \tag{A.14}$$

In other words, the set of nonlinear inequalities (A.14) can be transformed into the LMI (A.16).

Example. Consider the following matrix inequality:

$$A^T P + PA + PBR^{-1}B^T P + Q < 0 \tag{A.15}$$

where $P = P^T > 0$ and $R > 0$, is equivalent to

$$\begin{bmatrix} A^T P + PA + PBR + Q & PB \\ B^T P & -R \end{bmatrix} < 0 \tag{A.16}$$

S-PROCEDURE This procedure is adopted when we want to combine several quadratic inequalities into one single inequality. In many problems of control engineering we would like to make sure that a single quadratic function of $x \in \mathbb{R}^m$ is such that

$$F_0(x) \leqslant 0,\ F_0(x) := x^T A_0 x + 2b_0 x + c_0, \tag{A.17}$$

whenever certain other quadratic functions are positive semi-definite, i.e.,

$$F_i(x) \geqslant 0,\ F_i(x) := x^T A_i x + 2b_i x + c_i,\ i \in (1, 2, ..., q). \tag{A.18}$$

Example. To illustrate the S-procedure, consider the case $i = 1$ for simplicity. We need to guarantee that $F_0(x) \leqslant 0$ for all x such that $F_1(x) \geqslant 0$. If there exist a positive (or zero) scalar τ such that

$$F_{aug}(x) := F_0(x) + \tau F_1(x) \leqslant\ \forall x,\ \text{s.t. } F_1(x) \geqslant 0, \tag{A.19}$$

then our goal is achieved. This is because $F_{aug}(x) \leqslant 0$ implies that $F_0(0) \leqslant 0$ if $\tau F_1(x) \geqslant 0$, since $F_0(x) \leqslant F_{aug}(x)$ if $F_1(x) \geqslant 0$. By extending this to q number of inequality constraints gives the following:

$$F_0(x) \geqslant 0 \text{ whenever } F_i(x) \geqslant 0 \text{ holds if} \quad F_0(x) + \sum_{i=1}^{q} \tau_i F_i(x) \leqslant 0,\ \tau_i \geqslant 0. \tag{A.20}$$

FINSLER'S LEMMA This lemma is a property closely related to the previously presented S-procedure. The lemma is reported in the following

Lemma A.1. *Given two real symmetric matrices P, A if the quadratic inequality*

$$x^T P x > 0 \tag{A.21}$$

holds for any $x \neq 0$ such that $x^T A x = 0$, then there exist a scalar λ such that

$$Q - \lambda A > 0 \tag{A.22}$$

ELIMINATION LEMMA Finsler's lemma is exploited to derive another useful lemma, the so-called elimination lemma [15], used to reformulate matrix inequalities eliminating some of the original variables. The lemma is reported in the following

Lemma A.2. *Consider the matrix inequality*

$$T(x) + W(x) + FV^T(x) + V(x)F^T W^T(x) > 0 \tag{A.23}$$

with $T \in \mathbb{R}^{n\times n}$*, and* W*,* V*,* F *of suitable dimension. Assume that* T*,* W*,* V *are independent from* F *and denote with* $W^\perp(x)$*,* $V^\perp(x)$ *the orthogonal complements of* $W(x)$*,* $V(x)$ *respectively. Then (A.23) holds for some* F *if and only if the following inequalities are satisfied for* $x = x_0$

$$\begin{aligned} W^\perp(x)T(x)W^\perp(x) > 0 \\ V^\perp(x)T(x)V^\perp(x) > 0. \end{aligned} \tag{A.24}$$

Furthermore, by applying lemma A.1 we can state that there exist $\lambda \in \mathbb{R}$ *such that*

$$\begin{aligned} T(x) - \lambda W^\perp(x)W^\perp(x) > 0 \\ T(x) - \lambda V^\perp(x)V^\perp(x) > 0. \end{aligned} \tag{A.25}$$

A.2 SOLVING LMI USING YALMIP TOOLBOX

Most of the MATLAB programs which are being used to synthesise the modern anti-windup tested in this work are formulated using YALMIP toolbox. YALMIP (*Y*et *A*nother *LMI P*arser) is a modelling language for advanced modeling and solution of convex and non-convex optimization problems. It is available freely to be used by researchers. It should be noted that YALMIP is only a modelling language and it requires some underlying solvers such as Mosek, Sedumi, SDPT3, CPLEX etc. these solvers in your computer.

In this section, some basic commands of YALMIP toolbox will be introduced [17].

A.2.1 *Defining decision variables*

The decision variables constitute the most important components in any optimization problem. In YALMIP these variables are represented by the command `sdpvar`. A rectangular matrix P with n rows and m columns is represented by the command

```
P=sdpvar(n,m);
```

A square matrix $P \in \mathbb{R}^{n\times n}$ is by default, a symmetric matrix and is defined either by the following command:

```
P=sdpvar(n,n,'symmetric','real');
```

or it is defined using only the dimension argument as

```
P=sdpvar(n,n);
```

We can represent different types of variables such as Toeplitz, Hankel, diagonal matrices using the following commands:

```
X=sdpvar(n,1);
D=diag(X);     % Diagonal matrix
H=hankel(X);   % Hankel matrix
T=toeplitz(X); % Toeplitz matrix
```

A.2.2 *Defining constraints*

The most commonly used constraints in YALMIP are element-wise, semi-definite, and equality constraints. For example, the code below-generates a constraint to define matrices X to be positive definite and Y to be negative definite.

```
X=sdpvar(n,n);
Y=sdpvar(n,n);
constraint = [X > 0, Y < 0];
```

In addition to these standard constraints, YALMIP also supports definition of integrity constraints, second-order cone constraints, and sum of squares constraints.

A.2.3 *Solving optimization problem*

Once all variables and constraints have been defined, the optimization problem using the command `optimize` can be solved. For example, a linear program

$$\begin{aligned} &\min\ c^T x \\ &\quad \text{s.t. } Ax \leqslant b \end{aligned} \tag{A.26}$$

can be solved with the following piece of code

```
x=sdpvar(lenght(c),1);
constraint = [A*x <= b];
objective=c'*x;
optimize(constraint,objective);
solution = value(x);
```

If the aim is to test the feasibility the objective function can be omitted and used `optimize(constraint)`.

BIBLIOGRAPHY

[1] K. J. Astrom and L. Rundqwist. Integrator windup and how to avoid it. pages 1693–1698, June 1989.

[2] Martin Bak. *Control of systems with constraints*. PhD thesis, Technical University of Denmark, Department of Automation, 2000.

[3] J.-M. Biannic and S. Tarbouriech. Optimization and implementation of dynamic anti-windup compensators with multiple saturations in flight control systems. *Control Engineering Practice*, 17(6):703 – 713, 2009. ISSN 0967-0661. doi: https://doi.org/10.1016/j.conengprac.2008.11.002.

[4] J. M. G. da Silva and S. Tarbouriech. Antiwindup design with guaranteed regions of stability: an lmi-based approach. *IEEE Transactions on Automatic Control*, 50(1):106–111, Jan 2005. ISSN 0018-9286. doi: 10.1109/TAC.2004.841128.

[5] C. Edwards and I. Postlethwaite. Anti-windup and bumpless-transfer schemes. *Automatica*, 34(2):199 – 210, 1998. ISSN 0005-1098. doi: https://doi.org/10.1016/S0005-1098(97)00165-9.

[6] *Elmo Application Studio (EAS) User Guide*. Elmo Motion Control, 12 2012.

[7] H. A. Fertik. Direct digital control algorithm with anti-windup feature. *ISA Transaction*, 6:317–328, 1967.

[8] Sergio Galeani, Andrew R Teel, and Luca Zaccarian. Constructive nonlinear anti-windup design for exponentially unstable linear plants. *Systems & Control Letters*, 56(5):357–365, 2007.

[9] Sergio Galeani, Sophie Tarbouriech, Matthew C. Turner, and Luca Zaccarian. A tutorial on modern anti-windup design. *European Journal of Control*, 15(3–4):418–440, 2009. doi: http://dx.doi.org/10.3166/ejc.15.418-440.

[10] Gene Grimm, Jay Hatfield, Ian Postlethwaite, Andrew R Teel, Matthew C Turner, and Luca Zaccarian. Antiwindup for stable linear systems with input saturation: an lmi-based synthesis. *IEEE Transactions on Automatic Control*, 48(9):1509–1525, 2003.

[11] R. Hanus, M. Kinnaert, and J.-L. Henrotte. Conditioning technique, a general anti-windup and bumpless transfer method. *Automatica*, 23(6):729 – 739, 1987. ISSN 0005-1098.

[12] Tingshu Hu, Andrew R. Teel, and Luca Zaccarian. Anti-windup synthesis for linear control systems with input saturation: Achieving regional, nonlinear performance. *Automatica*, 44 (2):512 – 519, 2008. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2007.06.003.

[13] Rockwell Automation Inc. Logix 5000 controllers general instructions reference manual. https://www.rockwellautomation.com/global/literature-library/overview.page, 2018.

[14] N. Kapoor, A. R. Teel, and P. Daoutidis. An anti-windup design for linear systems with input saturation. *Automatica*, 34(5):559 – 574, 1998. ISSN 0005-1098.

[15] H.K. Khalil. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2002.

[16] M. V. Kothare, P. J. Campo, M. Morari, and C. N. Nett. A unified framework for the study of anti-windup designs. *Automatica*, 30 (12):1869 – 1883, 1994. ISSN 0005-1098. doi: https://doi.org/10.1016/0005-1098(94)90048-5.

[17] Johan Lofberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289. IEEE, 2004.

[18] J. Lozier. A steady state approach to the theory of saturable servo systems. *IRE Transactions on Automatic Control*, 1(1):19–39, May 1956. ISSN 0096-199X.

[19] F. Morabito, A. R. Teel, and L. Zaccarian. Nonlinear antiwindup applied to euler-lagrange systems. *IEEE Transactions on Robotics and Automation*, 20(3):526–537, 2004. doi: 10.1109/TRA.2004.824933.

[20] E. F. Mulder, M. V. Kothare, and M. Morari. Multivariable anti-windup controller synthesis using linear matrix inequalities. *Automatica*, 37(9):1407–1416, 9 2001. ISSN 0005-1098. doi: 10.1016/S0005-1098(01)00075-9.

[21] LAB Motion Systems NV. Lab motion systems product catalog 2017, 2017.

[22] LAB Motion Systems NV. http://www.labmotionsystems.com/, 2018.

[23] L. Peretti and M. Zigliotto. *Azionamenti Elettrici Industriali*. UniversitÃ degli Studi di Padova, 2013.

[24] Dario Richiedei. Control of mechanical systems, lecture notess, 2017.

[25] Siemens. Simatic s7 s7-1200 programmable controller. https://support.industry.siemens.com, 2018.

[26] S. Tarbouriech. Anti-windup design: an overview of some recent advances and open problems. *IET Control Theory and Applications*, 3:1–19(18), January 2009.

[27] S. Tarbouriech and M. Turner. Anti-windup design: an overview of some recent advances and open problems. *IET Control Theory Applications*, 3(1):1–19, January 2009. ISSN 1751-8644.

[28] S. Tarbouriech, G. Garcia, J.M.G. da Silva, and I. Queinnec. *Stability and Stabilization of Linear Systems with Saturating Actuators*. SpringerLink : Bücher. Springer London, 2011.

[29] MECO Research Team. Ku leuven. https://www.mech.kuleuven.be/en/pma/research/meco, 2018.

[30] A. R. Teel and N. Kapoor. The l2 anti-winup problem: Its definition and solution. pages 1897–1902, July 1997.

[31] Andrew R. Teel. Anti-windup for exponentially unstable linear systems. *International Journal of Robust and Nonlinear Control*, 9(10):701–716, 1999. doi: 10.1002/(SICI)1099-1239(199908)9:10<701::AID-RNC429>3.0.CO;2-8.

[32] K. S. Walgama and J. Sternby. Inherent observer property in a class of anti-windup compensators. *International Journal of Control*, 52(3):705–724, 1990.

[33] K. S. Walgama, S. Ronnback, and J. Sternby. Generalisation of conditioning technique for anti-windup compensators. *IEE Proceedings D - Control Theory and Applications*, 139(2):109–118, March 1992. ISSN 0143-7054.

[34] L. Zaccarian and A.R. Teel. *Modern Anti-windup Synthesis: Control Augmentation for Actuator Saturation*. Princeton Series in Applied Mathematics. Princeton University Press, 2011.

[35] Luca Zaccarian and Andrew R. Teel. A common framework for anti-windup, bumpless transfer and reliable designs. *Automatica*, 38(10):1735 – 1744, 2002. ISSN 0005-1098. doi: https://doi.org/10.1016/S0005-1098(02)00072-9.

[36] Luca Zaccarian and Andrew R. Teel. Nonlinear scheduled anti-windup design for linear systems. *IEEE Transactions on Automatic Control*, 49:2055–2061, 2004.

[37] ALEX ZHENG, MAYURESH V. KOTHARE, and MANFRED MORARI. Anti-windup design for internal model control. *International Journal of Control*, 60(5):1015–1024, 1994. doi: 10.1080/00207179408921506.