VNIVERSIDAD NACIONAL
AVTONOMA DE
MEXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA

OBESERVACIÓN Y CONTROL POR MODOS DESLIZANTES DE UNA MÁQUINA DE PRESION.

T E S I S

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN INGENIERÍA

INGENIERIA ELÉCTRICA - CAMPO CONTROL

P R E S E N T A

TENOCH GONZALEZ ROJAS

TUTOR:

LEONID FRIDMAN

2011

JURADO ASIGNADO:

Presidente	Dr. Alvarez Icaza Longora Luis Agustn.
Secretario	Dr. Moreno Prez Jaime Alberto.
Vocal	Dr. Fridman Leonid.
1er. Suplente	Dr. Lpez Parra Marcelo.
2do. Suplente	Dr. Dvila Montoya Jorge Angel.

Lugar o lugares donde se realizó la tesis:
UNAM - Campus CU, Mexico D.F. y
Bosch Rexroth AG, Lohr am Main, Alemania.

OBESERVACIÓN Y CONTROL POR MODOS DESLIZANTES DE UNA MÁQUINA DE PRESION.

TENOCH GONZALEZ ROJAS

A mis padres Lenica Rojas Quiones y Octavio Gonzalez Pea

Agradecimientos

A la Universidad Nacional Autónoma de México y la Facultad de Ingeniería por haberme brindado los medios para formarme como un profesional de la Ingeniería.

Agradezco también a mi asesor el Dr. Leonid Fridman por brindarme su confianza al considerarme para desarrollar este trabajo. Así mismo agradezco a todo el colegio de profesores del departamento de control por sus dedicados esfuerzos en mantener un alto estándar de calidad en la Ingeniería nacional desde el campo de la generacin y divulgacin del conocimiento; sus diferentes visiones me han ayudado a tener una visión amplia e integra.

Agradezco a Jan Komsta y mis colegas de Bosch Rexroth AG por la oportunidad de trabajar juntos y las experiencias aprendidas.

Finalmente, agradezco especialmente a mis compañeros estudiantes de posgrado por la oportunidad de compartir ideas y aprender juntos siempre.

Resumen

El problema de eficiencia energética es muy popular en estos tiempos en la industria debido a nuevas regulaciones nacionales e internacionales que intentan proteger el ambiente y hacer un uso más racional de los recursos. En general la mayoría de las maquinas hidráulicas que están en funcionamiento forman parte de un sistema más complejo del que las máquinas hidráulicas son un eslabón básico. Debido a esto se exige que la máquina sea confiable, tenga un alto índice de disponibilidad y sea fácil de operar; en resumen se necesita que lo básico sea sencillo y no una fuente de problemas.

Partiendo de estas exigencias y considerando la naturaleza incierta y no-lineal de la hidráulica, las máquinas fueron diseñadas, en el siglo pasado, sobredimensionando sus capacidades para evitar complicaciones durante la operación; por supuesto, cuando estos diseños se hicieron el consumo de energía no era prioridad. Sin embargo, una vez que se pone atención a la eficiencia energética, es necesario volver a revisar el diseño y revisar como aquellas complicaciones fueron resueltas. Afortunadamente la tecnología ha avanzado y hoy en día el ingeniero tiene a su disposición mejores herramientas para realizar el trabajo.

El presente trabajo se concentra en el problema de mantener una presión deseada en un sistema hidráulico que continuamente experimenta pérdidas de presión debido a filtración o desplazamiento del líquido. Tradicionalmente este problema se soluciona con la ayuda de una fuente constante de flujo mucho mayor a las pérdidas; el flujo de la fuente que entra al sistema y por lo tanto la presión en el sistema puede ser regulada con la ayuda de una válvula. Esta solución cumple con el desempeño deseado y es por ello ampliamente usada; sin embargo la desventaja es que para funcionar la fuente de flujo debe permanecer encendida y al máximo siempre, lo que requiere de mucha energía y genera mucha contaminación. Pensando en eficiencia energética existe ahora una solución que emplea fuentes de flujo variables que solo proporcionan la cantidad adecuada de flujo cuando es necesario, esto reduce la energía necesaria y la contaminación generada. Esta solución se conoce como bomba de velocidad variable o SvP. Para cumplir su función la SvP necesita compensar las pérdidas de flujo, actualmente esta compensación se hace con un complejo esquema de control PID-PI que requiere que el usuario sintonice más de diez parámetros.

La aportación principal del presente trabajo es el diseño e implementación de un esquema de control que utiliza una retro-alimentación proporcional para ajustar la dinámica del sistema y feedfoward para rechazar la perturbación (perdidas de flujo). La perturbación es exactamente identificada en tiempo real utilizando técnicas de modos deslizantes. El producto final es un esquema de control más sencillo, más ligero computacionalmente, más suave y con mejor desempeño que el esquema PID-PI actual. Se destaca que el usuario sólo necesita sintonizar un parámetro y que el control propuesto no necesita ningún cambio en el hardware o firmware actual para implementarse. El esquema ha sido probado exitosamente en condiciones industriales.

Abstract

The problem of energy efficiency is one of the most relevant at the moment in the industry due to new regulations at national and international levels. This new regulations are trying to protect the environment and are aiming to do a more rational use of the resources. In general, the majority of the hydraulic machines, that are operating, are part of a complex system of which the hydraulic machine is a basic link. Because of this, it is a requirement that the machine should be reliable, have a high index of availability and be easy to operate. Summarizing it is a requirement that this basic link is simple and not a source of problems.

Taking these requirements into account and considering the non-linear and uncertain nature of the hydraulics, the original designs, which were made during the past century, were made oversizing the capacities of the machine to avoid complications during operation; of course, when those designs were made consume of energy was not a priority. However once that the focus is on energy efficiency, there is a need to give a second view to the design and to how those complications were avoided. Fortunately, nowadays the technology has advanced and the Engineer has today a bigger set of tools to accomplish this job.

The present work is focused on the problem of keeping a desired pressure in a hydraulic system which is continuously disturbed by pressure losses due to leakage or liquid displacement (load flow). Traditionally this problem is solved with the help of a constant source of flow that is much bigger than the flow losses. A valve regulates the flow of the source that goes inside the system and by regulating the flow the pressure is regulated. This solution gives the desired performance and for this reason is widely used; however, the disadvantage is that, in order for this solution to work, the flow source has to be operative and at maximum flow always. This require a lot of energy and contaminates more. Thinking in energy efficiency there are now solutions that employ variable flow sources that provide just the right amount of flow when it's needed, this approach reduces considerably the energy used and the contamination generated. This solution is known as Speed Variable Pump or SvP. To accomplish its goal, the SvP needs to compensate the flow losses; currently this compensation is done using a complex control scheme PID-PI which requires the tuning of more than ten parameters by the final user.

The main contribution of the present work is the design and implementation of

a control scheme that uses proportional feedback to adjust the system dynamics and a feedforward action to reject all the disturbances (flow losses). These disturbances are exactly identified in real time using sliding mode techniques. The final product is a simpler control scheme, computationally lighter, smother and with a better performance that the current PID-PI. It is remarkable that the final user needs to tune just one parameter and that the proposed control does not need any change in the current hardware or firmware to be implemented. The scheme has been successfully tested in industrial conditions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Hydraulic technology is already consolidated and is widely used across the industry. However, in the last years the main concern in the hydraulics field is the energy efficiency problem which needs to be solved without compromising the good performance that hydraulic technology provide to the industry. It is important to remark that in this case good performance means not just good dynamics, high power density, etc, but also relative simplicity of operation. Hydraulic technology has kept over time simple control schemes that makes tune possible for any engineer or trained technician. Thus, any successful energy efficiency solution needs to remain simple for the user.

It is important not to confuse simplicity for the user with trivial solution or effortless thinking. A user friendly solution means that user can do his tasks without having to learn a new fancy theory or a complex environment. In order for this to happen, Engineers are the ones who need to do the complex thinking, keep themselves updated with new theories and technologies and integrate all these in their design. When a solution is really well thought and for the user is nice and clean, then it is engineering elegance.

1.1. Brief History of Hydraulic Technology

Hydraulics is a topic in applied science and engineering dealing with the mechanical properties of liquids. Fluid mechanics provides the theoretical foundation for hydraulics, which focuses on the engineering uses of fluid properties. In fluid power, hydraulics is used for the generation, control, and transmission of power by the use of pressurized liquids. Hydraulic topics range through most science and engineering disciplines, and cover concepts such as pipe flow, dam design, fluidics and fluid control circuitry, pumps, turbines, hydropower, computational fluid dynamics, flow measurement, river channel behavior and erosion.

Hydraulic machines are machinery and tools that use fluid power to do work. Heavy equipment is a common example. In this type of machine, liquid, called hydraulic fluid, is transmitted throughout the machine to various hydraulic motors and hydraulic cylinders and it becomes pressurized according to the work cycle. The fluid is controlled directly or automatically by control valves and distributed through hoses and tubes.

The popularity of hydraulic machinery is due to the very large amount of power that can be transferred through small tubes and flexible hoses, and the high power density and wide array of actuators that can make use of this power. The main non-linearities in Hydraulics are the compressibility of the fluid, flow dynamics through valves and the friction in hydraulic actuators. These non-linearities are related to the parameters Bulk modulus, viscosity and temperature which are not easy to estimate and, in general, are not available for measure in practice. Due this inconvenience, proportional, and sometimes integral, hand tuned feedback is normally used to achieve control tasks.

A small resume of the history of hydraulics and control is presented here, it is taken from [2], [3], and [1]. Although hydraulic applications can be tracked back for more than a thousand years in Asia, America and Europe; most of the authors establish the beginning of modern hydraulics with the work of Pascal introducing the hydraulic press in 1663. But, it was until 1795 when the hydraulic press was used in an industrial application by Josef Bramah in London. When the industrial revolution started in England steam machines were widely used. Pumps driven by steam engines were pressurizing water that flowed through pipes and actuate pistons. Then, in the second half of 19th century, W. G. Armstrong (1810-1900) developed several hydrostatic machines and control devices, primarily for use in shipbuilding. Some of these devices still resemble the ones used today.

In the beginning of the 20th century, the development of fluid power continued, the hydraulic drives, being a key tool in the industrial revolution, spread into most of the industrial process. Such drives consisted on a flow control device driving the hydraulic actuator in an open loop manner.

After World War Two the developments continued now using closed loop schemes, which allow a better performance and more accurate behavior. Nowadays hydraulic applications with more precise control schemes are used in all kinds of industries from Aeronautic to Packaging.

1.1.1. Applications

The main advantage of fluid power is that it provides a good ratio between torque or force delivered and the weight and size of the actuator. In many applications this fact allows a direct drive solution taking out costly and wear sensitive gear boxes.

Hydraulic servo-systems are used:

- Where relatively large forces or torques are required (Industrial presses, mobile lifting, digging, material handling equipment, etc.).
- Where fast, stiff response of resisting loads is needed (machine tool drives, flight simulators, rolling mills, etc.).
- Where manual control of motion involving substantial forces is essential (heavy machinery, aircraft controls, automotive power steering, etc.).
- As the final actuator subsystem in complex automatically controlled situations (eletro-hydraulic flight simulators, industrial robots, fatigue and other programmable testing rigs, theatre stage control, etc.).

Hydraulic control systems provide many advantages over other types of systems (e.g. electrical motors), some of which are:

- Produce larger forces/torques, and have higher load stiffness.
- Hydraulic fluid acts as a superb lubricant and avoids wear.
- Hydraulic actuators have higher speed of response with fast starts, stops and speed reversals.

- Hydraulic systems can be operated under different conditions (continuous, intermittent, reversing, and stalled).
- Both linear and rotary actuators are available that add great flexibility to hydraulic power elements.
- Overloading protection is easy.
- Smaller and lighter compact systems with long component life are available.

Besides the advantage of fluid power, there are some disadvantages:

- High costs of hydraulic components that result from small allowable tolerances.
- Fire and explosion hazards exist if a hydraulic system is used near a source of ignition.
- Dirty and contaminated fluids cannot be fully avoided. This may lead to clogging of valves and actuators, as well as permanent loss in performance and/or failure. Contaminated oil is the main source of failure; clean oil and reliability are synonyms in hydraulic control.
- The dynamic characteristics of hydraulic systems are highly non-linear and relatively difficult to control.

1.1.2. Control

The will to do engineering is a consequence of human behavior. Naturally, when the man tries to adapt the environment to him, engineering problems arise including control engineering problems. Thus control engineering solutions were known by antique civilizations in America, Asia, Africa and Europe; and control devices were continuously developed trough history. However, it was not until the eighteenth century with the industrial revolution that serious attempts were made to translate ingenious ideas into effective industrial control devices, mainly for manufacturing process.

During the nineteenth century, the Watt engine governor was widely adopted. During this century there also was an enormous range of inventions for temperature, pressure, and flow control devices. These developments continued in the twentieth century.

As an example of early hydraulic control devices we can take the successful controllers based on the flapper-nozzle amplifier. In such control, the movement of the flapper arm towards or away from the nozzle causes a change of back pressure in the pneumatic circuit and this change in pressure results in a movement of the diaphragm bellows. This movement can be applied to a pilot valve which, in turn, controls the opening and closing of the main control valve.

The basic flapper-nozzle mechanism was invented by Edgar H' Bristol of the Foxboro Company during the winter of 1913-1914. The basic flapper-nozzle mechanism is highly nonlinear and actually in the early versions of the Foxboro controllers the flapper-nozzle mechanism was used as an on-off relay. The gain of the flapper-nozzle was such that a change in the measured quantity equal to 1 % of full scale of the measurement would cause 100 % change in the back pressure.

Throughout the 1920s all the companies manufacturing pneumatic controllers attempted to increase the range of linear operation of all the components in the system. In practice, because of the problems caused by the high gain of the controllers, many of the instrument manufacturers recommended using bypass control schemes. In such schemes the controlled medium, for example, steam used for heating, is split into two parts, one controlled by the automatic device and the other, the bypass, controlled by a manually set valve. Large changes in loads or in setpoints are accommodated by adjusting by hand the bypass valve.

The first PI control scheme was incorporated in the Foxboro Model 10 Stabilog controller announced in September 1931. Initially the Stabilog did not sell in large numbers because the users needed some education. The Foxboro Company relaunched it in 1934 and produced a brochure which explained in detail how it operated and the benefits to be gained from its use. Figure 1.1 shows an image of this brochure displaying the control concept. After this, the concept was widely used by the industry and all competitors start producing PI controllers. About ten years after the PI, the PID controller was incorporated. The research then was mainly focused on methods to obtain the optimal gains for a PID. This effort lead to the papers by J.G. Ziegler and N.B. Nichols published in 1942 [3] and 1943. In these papers Ziegler and Nichols showed how optimum controller parameters could be chosen based first on open-loop tests on the plant; and second on closed-loop tests on the plant. This work is still the base for PID tuning.

Is remarkable that this efforts to develop the industrial automatic control were strongly linked to hydraulic technology. Because of this, history of PID control is strongly tied to Hydraulic technology.

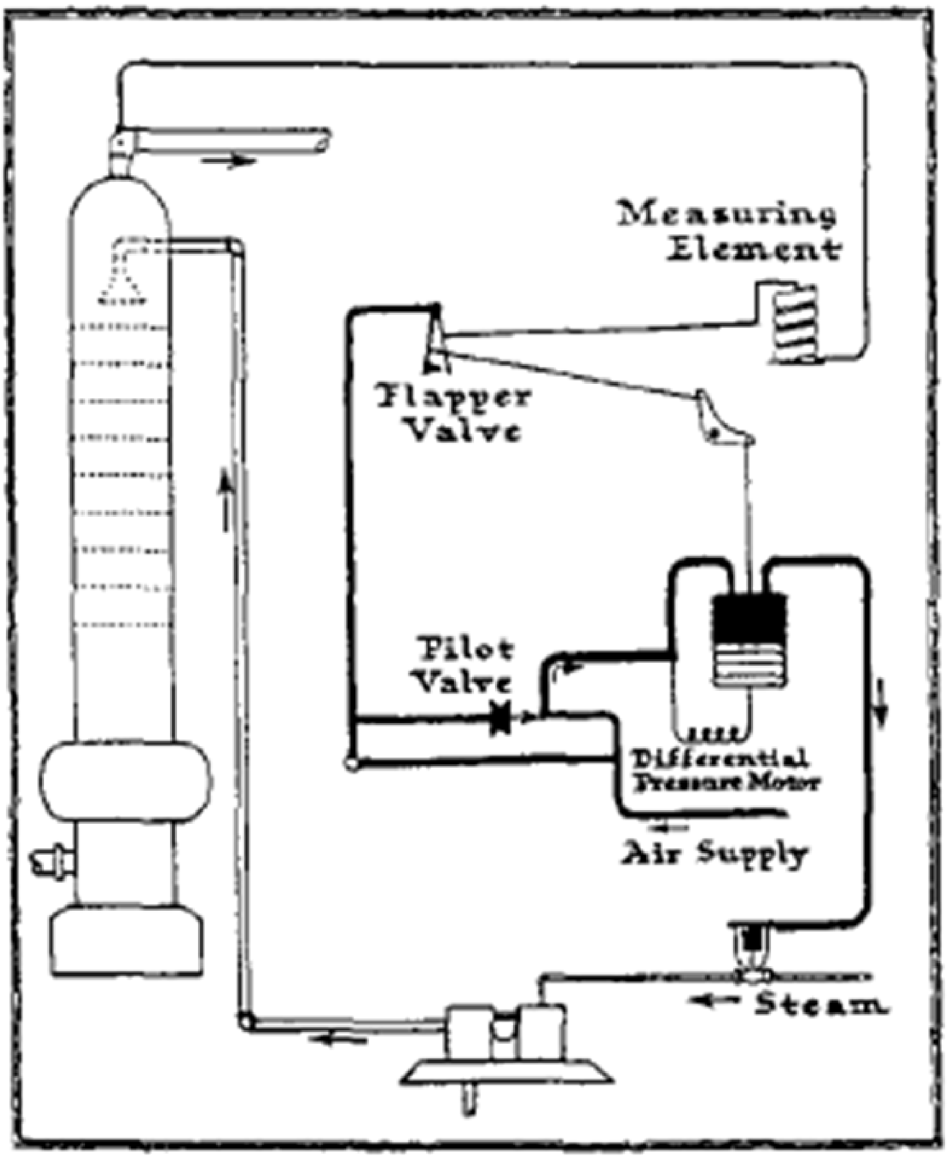

Figure 1.1: Control PI explained in 1934, taken from [1]

1.2. Motivation and state of the art

Pumping systems account for nearly 20 % of the world energy used by electric motors and 25 % to 50 % of the total electrical energy usage in certain industrial facilities. Significant opportunities exist to reduce pumping system energy consumption through smart design, retrofitting, and operating practices. In particular, many pumping applications with variable-duty requirements offer great potential for savings. The savings often go well beyond energy, and may include improved performance, improved reliability, and reduced life cycle costs [4].

Most existing systems requiring flow control make use of bypass lines, throttling valves, or pump speed adjustments. The most efficient of these is pump speed control.

This systems are called Speed Variable Pump SvP and are nowadays offered by several hydraulic companies as the best energy efficiency solution; all the companies present the SvP as a High Tech solution which often needs to be implemented by highly trained professionals.

New energy consumption and contamination (noise) regulations are giving a push to this systems as they help the industry to meet the new requirements. To obtain the best of this momentum the companies have to be able to implement the SvP solution in a simpler and faster way, reducing the need of highly trained professionals.

The SvP always forms part of a bigger system which whole operation relies on the pressure and flow that SvP provides. During operation the pressure is set to a desired value and the SvP provides it; however, the system will also take flow from the SvP, this could shrink the pressure beyond the acceptable limits. It is not known by the SvP when the system is going to take flow; thus the flow taken, called load flow, acts as a random disturbance to the SvP. Because of the unpredictable nature of the disturbance, the control problem is solved using a cascade control PID-PI where the inner loop controls the speed of the pump and the outer loop controls the pressure of the system. While the scheme has potential to overcome the disturbance the tuning of the control gains can be difficult for the technicians who operate the system; using this scheme the technicians have to tune eleven gains.

Sliding Modes is a control technique famous for its robustness against parameter variations, unmodeled dynamics and disturbances. Observers based on sliding modes have also attractive characteristics like insensibility to unknown disturbances and, through the equivalent control, obtaining additional information of the system and disturbances.

With this characteristics it is possible to construct an sliding mode observer that identifies the load flow and other disturbances. Ultimately, this extra information of the sliding mode observer can be used to construct an overall simpler control scheme where the complexity for the final user is drastically reduced.

1.3. Objective of the Thesis

The objective of this work is to present a novel control scheme that has an excellent performance (fast, well damped, no oscillations), is simple in its construction, and significantly reduces the complexity to the user in comparison to the standard controller.

In order to achieve the objective, a novel control is presented for the Speed Variable Pump SvP; this control consists of:

- Linear P-P(I) cascade control where the inner loop controls the speed of the pump (with optionally integral gain) and the outer loop controls the pressure of the system.
- Sliding Mode Observer which identifies the disturbances and rejects them using a feedforward scheme.

The sliding mode observer is away of the final user hands, it rejects all rejectable (matched) disturbances and eliminates the need to use integral or derivative gains in the outer loop.

The P-P cascade scheme is well known by the technicians and normally the inner loop comes previously tuned by the manufacturer (because normally it is the control loop of the actuator), therefore the user/technician tunes, usually, just the outer loop. Since the sliding mode observer is out of the hands of the final user, the complexity during commissioning is drastically reduced to tune just one proportional loop, this is exactly what the user knows how to do.

Additionally, a feedforward scheme is presented for the inner loop that reduces the integral gain in the inner loop. However the inner loop proportional and integral values are given by the manufacturer and the user may prefer to leave the inner loop unchanged. Experimentally, it is shown that, due to special characteristics of the hardware that is used in the inner loop, the scheme that the user selects for the inner loop is not relevant for the overall performance.

1.4. Structure of the Thesis

In chapter 2 the antecedents of the work are given. In chapter 3 a description of the Pressure machine SvP is given including mathematical model, description of the hardware involved and a brief general description of the current control solution. In chapter 4 the control design is presented, first it is shown how the base linear control system is constructed to shape the dynamics, and then, it is shown the sliding mode observer that makes it is possible to reject any disturbance in the context of the machine. In chapter 5 the results of the experimental tests are presented along with comparisons with a a linear control system that uses a Luenberger observer instead of the sliding mode observer, and model based PI-P scheme. Finally in Chapter 6 the conclusions are given. There are two appendices, appendix A presents the code of the main algorithms developed during this work, and appendix B presents additional experiments that were done to find the best solution for the control problem.

Chapter 2

Preliminaries

Is well known that to create a good control for a system a good knowledge of the system is needed. This good knowledge of the system is generally expressed as a mathematical model created following the laws of physics.

The mathematical models can be classified by their grade of complexity and detail:

Research models these are highly nonlinear models that try to describe with high accuracy the phenomena involved in the system. These models help to have a better understanding of the system and can influence not just the control design but also how the whole Engineering problem is solved; for example in the automotive industry a complex model of the air flow against the car helps engineers to design the body of the car. Thus these models are used to have a better understanding of the nature of the phenomena present in the system.

Simulation models these are nonlinear models focused on particular effects or phenomena that are key to the Engineering problem that is being studied. These models are in the middle between the simplest linear model and the research model. An example is the friction forces/torques present on a system; there are several non-linear models that describe in different level of detail the phenomena of friction in different circumstances. The Engineer has to make a decision on which model of friction to use based on how friction affects the system, so in simulation the model can be very simple if there is not a big effect in the system or very complex if the effects of friction are crucial for the desired performance.

Linear model the linear model is used to realize control and control related design. It is obtained, in general, first by ignoring the effects of phenomena in the system that are negligible in the context of the control objective (reduced model), and second by linearizing the non-linear dynamics, if any, around the operation point. Thus the linear model has a limited scope and the designer needs to take this fact always into account.

Normally during the control design the simplest model, the linear model, will be used. However, it is important that the designer keeps in mind the scope of the linear model and the limits of the control created using such model.

2.1. Modeling of hydraulic machines

In the following a brief introduction of hydraulics systems and its modeling is given.

A hydraulic system consists of:

- Hydraulic power supply
- Control elements (valve, sensors, etc.)
- Actuating elements (cylinder, hydraulic motors)
- Other elements (pipelines, hoses)

In the figure 2.1 the standard valve controlled system is shown. The description of such system is:

- The pump converts the available (mechanical) power from the prime mover (electric or diesel motor) to hydraulic power at the actuator.
- Valves are used to control the direction of pump flow, the level of power produced, and the amount of fluid and pressure to the actuator.

- Linear actuator (cylinder) or rotator actuator (motor) converts the hydraulic power to usable mechanical power output at the point required.
- The medium, which is a liquid, provides direct transmission and control, as well as lubrication of components, sealing in valves, and cooling of the systems.
- Connectors link the elements together, direct the power of the pressurized fluid , and fluid-flow return to the tank.
- Finally, fluid storage conditioning equipment ensure sufficient quality, quantity, and cooling of the fluid.

2.1.1. Physical Fundamentals

Density is defined as the amount of mass in a given volume

$$\rho = \frac{m}{V}$$

The density of a hydraulic fluid usually is between 0.85 and 0.91 kg/dm^3. Density in an hydraulic system is a function of temperature and pressure, but generally is considered constant in the temperature range of hydraulic machines. Next equation shows how the density is related to pressure.

$$\rho = \rho_0 + \frac{\rho_0}{E} p(t) \tag{2.1}$$

Where ρ_0 is the density measured at zero, or base, pressure and E is the bulk modulus. The bulk modulus E is the change in pressure divided by the fractional change in volume at a constant temperature.

$$E = -V_0 \left(\frac{\delta p}{\delta V} \right)_\theta$$

The bulk modulus considerable influence the dynamic behavior of hydraulic machines as it is linked to the pressure change. For mineral oils and for common pressures and temperatures ($\theta \epsilon [-40, 120] C$, $p \leq 410 [bar]$), it is possible to assume a mean value of

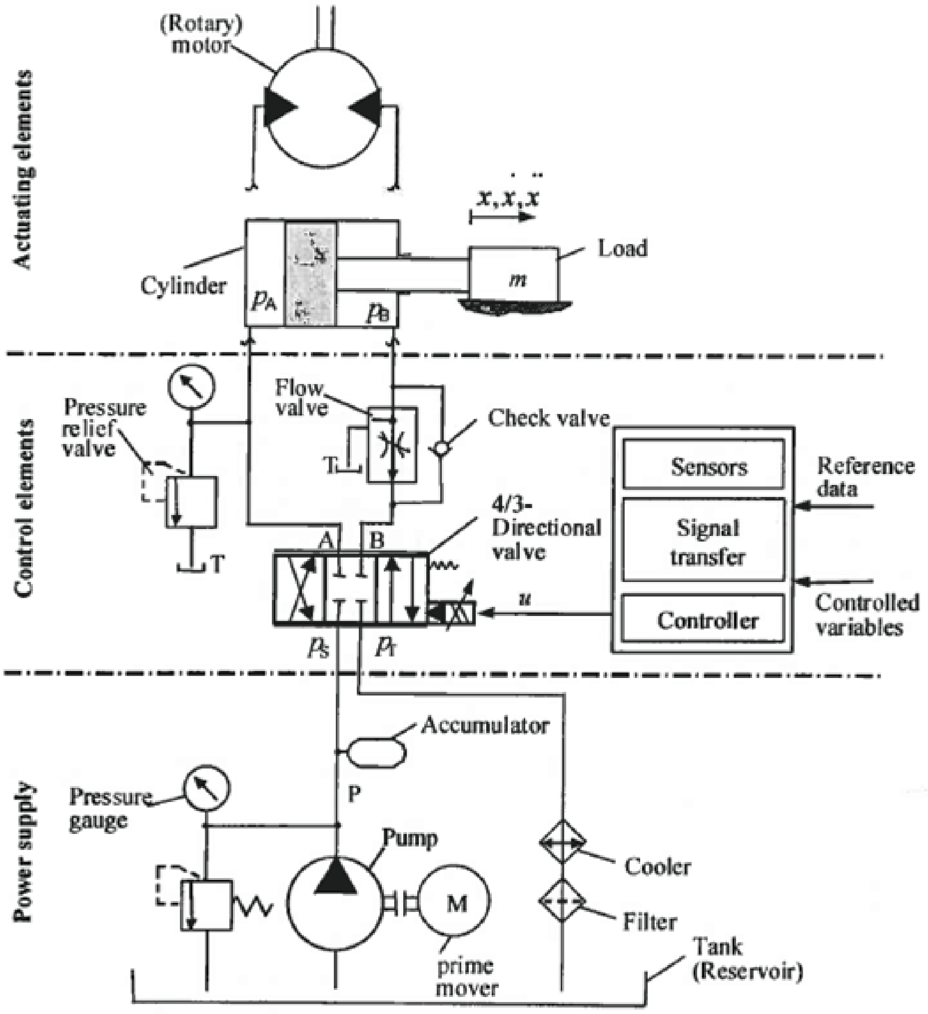

Figure 2.1: General hydraulic system

the bulk modulus. However from a practical point of view this assumption is very rough because the bulk modulus varies considerably with pressure and it is influenced by other factors like air and contaminants present in the oil.

There is a lot of equations that describes the behavior of the bulk modulus, for example

$$E(p) = \frac{1}{2} E_{max} \log\left(90 \frac{p}{p_{max}} + 3\right)$$

Most of these equations have been obtained through experimentation, thus there are also experimental curves under different conditions (pressure, temperature, type of oil) and tables. However, as said before, in practice the bulk modulus is treated as constant and robustness is expected from the controller.

This work deals with the problem of a volume of liquid going in and out of a volume at different rates. This problem can be seen as a continuity problem, see figure 2.1.1, consequently we can start with the integral continuity equation.

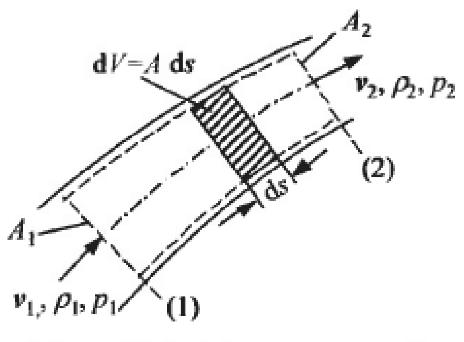

Figure 2.2: Flow in a control tube

$$\int_1^2 \frac{\delta(\rho A)}{\delta t} dt + \rho_2 \nu_2 A_2 - \rho_1 \nu_1 A_1 = 0 \tag{2.2}$$

where A is the area, ν is the velocity and the density $\rho = \rho(t, s)$ is, in general, not constant but in Hydraulics is assumed constant therefore equation (2.2) can be written as

$$\nu_2(t) A_2 = \nu_1(t) A_1 \tag{2.3}$$

The flow through a given area A is defined as

$$Q(t) = \nu(t) A \tag{2.4}$$

therefore the mass flow is given by

$$\dot{m} = \rho \nu(t) A \tag{2.5}$$

The continuity equation is

$$\dot{m}_{in} - \dot{m}_{out} = \frac{d\rho V}{dt} \tag{2.6}$$

A fixed volume V and the density ρ depends of the pressure as showed in equation (2.1). Using equation (2.6), (2.1) we can define the change of pressure in a fixed volume as

$$\dot{p}(t) = \frac{E}{\rho_0 V} \left(\dot{m}_{in} - \dot{m}_{out}\right) \tag{2.7}$$

2.1.2. Hydraulic pump

Using Newton second law of motion is possible to obtain the equation of the pump dynamics

$$J_p \ddot{\phi}_p = T_m - T_f - \eta_{Hp} T_{Hp} \tag{2.8}$$

Where J_p is the pump inertia, T_m is the torque of the external mover (electric motor, diesel motor, etc.), T_f is the torque due to friction, η_{Hp} is the volumetric efficiency of the pump and T_{Hp} is the torque due the high pressure in the hydraulic system.

The ideal pump flow Q_p is given by

$$Q_p = \frac{V_p}{2\pi} \dot{\phi}_p \tag{2.9}$$

Where V_p is the theoretical displacement of the pump. The calculation of the pump displacement depends of the type of pump, however the manufacturer always provides the value. Nevertheless, the real output of the plant is less than the given by the theoretical value due to internal leakage. Hence the effective displacement can be calculated using the volumetric efficiency coefficient ηHp.

The high pressure torque T_{Hp} is present because the pump is integrated into a hydraulic circuit, thus a high pressure load acts on the pump. This torque can be described by the equation

$$T_{Hp} = \frac{V_p}{2\pi} \Delta p \tag{2.10}$$

Where Δp is the difference between the high pressure line and the low pressure line.

2.1.3. Valves

The flow through valves can be described by the next equation

$$Q_v = c_v x_v \sqrt{|\Delta p|} sign\left(\Delta p\right) \tag{2.11}$$

where c_v is the valve coefficient that depends on the type of valve, x_v is the position of the valve spool, Δp is the pressure difference between sides of the valve. Depending on the valve different hydraulic circuits can be described using equation (2.11).

2.1.4. Fixed volume

The pressure dynamics of a fixed volume in a hydraulic circuit can be described as follows using the continuity equation (2.7) divided by the oil density

$$\dot{p} = \frac{E}{V}\left(Q_{in} - Q_{out}\right) \tag{2.12}$$

where Q_{in} is the flow that enters the fixed volume V and Q_{out} is the flow that leaves the volume. It is important to mention that the oil compressibility influences the dynamics of hydraulics control systems drastically. The Hydraulic fluid acts like a spring and, therefore, introduces a second order system whose natural frequency limits the bandwidth of any hydraulic system abruptly. The damping of the system due to leakage or friction is minimal.

There are other elements like cylinders and pipes, however, for this work they won't be described. For more information it is possible to check [2] from where most of these descriptions are taken.

2.2. Sliding modes

This is a brief description of the sliding modes techniques used in the work, for more information or proof of the algorithms presented please check [5], [6], [7], [8], and [9].

Sliding mode techniques form part of the variable structure systems (VSS), which may be regarded as a combination of subsystems where each subsystem has a fixed control structure and is valid for specified regions of the system space. In conjunction, all this structures accomplish a control task, that for the individual structures could be achieved with poor performance or be even impossible to achieve for the structures alone.

In Sliding modes, the switching between structures is defined by the sliding variable. As soon as the properly designed sliding variable becomes equal to zero, it defines the sliding manifold (or the sliding surface in the linear case); movement on the sliding manifold is ensured and consequently the desired designed dynamics are achieved.

Take for example the dynamical system of equation (2.13) where the function $f(x,t)$ is bounded and u is an input.

$$\dot{x} = f(x,t) + u \quad , \quad |f(x,t)| < L \tag{2.13}$$

The objective is to stabilize the system at the point $x = 0$, to accomplish this objective the sliding surface can be selected as $s = x$, then we can select the input $u = -Lsign(s)$. By doing this the system, despite its initial condition, will converge in finite time to the sliding surface $s = 0$.

The objective is accomplished, and even more, by using the sliding mode technique we get more than just finite time stabilization. The technique used in the example allows the reconstruction of the function $f(x,t)$ in finite time by using a low pass filter of the input u; this property is known as the equivalent control [7].

The possibility of reconstruct unknown signals while stabilizing the dynamical system gives to sliding mode techniques a great additional value.

2.2.1. Integral sliding modes

Integral sliding modes were suggested as a tool to reach the following goals:

- compensation of matched perturbations starting form the initial moment, i.e. ensuring the sliding mode starting form the initial moment;

- preservation of the dimension of the initial system, i.e. saving the system dynamics previously designed for the ideal case (without perturbation).

These two attractive properties of ISM allows them to be successfully used in different kinds of applications, e.g. robotics, pneumatics, missile guidance, and design problems (see, for example [10], [11], and [12], [13], and [14]; some related applications are [15], and [16]).

Suppose that a control law $u = u_0(x,t)$ achieving the control objective (e.g. steering, stabilization or tracking) is already available for an ideal, nominal system

$$\dot{x} = f(x,t) + B(x)u \quad , \quad x\epsilon R^n \quad , \quad u\epsilon R^m \tag{2.14}$$

Now suppose that instead of the ideal system (2.14), one has a perturbed system

$$\dot{x} = f(x,t) + B(x)(u+\delta) \tag{2.15}$$

Then, a sliding mode control law $u_1(x,t)$ can be easily included such that the closed-loop system

$$\dot{x} = f(x,t) + B(x)(u_0 + u_1 + \delta) \tag{2.16}$$

Is insensitive to δ (notice that the perturbation δ is matched with the controller). The Integral sliding mode is constructed as follows. Define the sliding variable

$$s(x,t) = g(x) - z(t) \quad , \quad s(x,t)\epsilon R^m \tag{2.17}$$

where

$$z(t) = g(x_0) + \int_{t_0}^{t} G(x)\left[f(x,\tau) + B(x)u_0(x,\tau)\right] \quad , \quad G(x) = \frac{\delta g}{\delta x}(x) \tag{2.18}$$

(gradients are regarded as row vectors) and $g(x)$ is any function such that $G(x)B(x)$ is invertible. Notice that, at $t = t_0$, we have $s = 0$, thus the system starts at the sliding surface (there is no reaching time). Let us now compute the time derivative of s:

$$\dot{s} = G(x)B(x)\left(u_1 + \delta\right) \tag{2.19}$$

It can be seen that if δ is bounded by known functions then it is possible to construct a unit control u_1 ensuring $s = 0$. The equivalent control is:

$$u_{eq} = -\delta \tag{2.20}$$

So the trajectories of the system at the sliding manifold are the ones of the nominal system (2.14) and consequently using ISM there is insensitivity to the disturbance δ.

2.2.2. Generalized super twisting algorithm

The super twisting algorithm (STA) was first proved using a geometrical approach [17], for this reason it was difficult to extend. Some efforts were done to introduce a Lyapunov based approach; in 2009 Prof. Moreno [18] presented a Lyapunov function for the super-twisting algorithm that had the structure of a quadratic function. Using this function is possible to calculate an estimate for the convergence time and also is possible to make the first modification of the STA by including linear terms that improved its convergence and robustness properties; this enhanced STA is called Generalized Super Twisting Algorithm (GSTA) [19], and [20] the name generalized is because it includes the standard STA as one of its possible variants.

Another interesting modification of the STA that was possible due to a Lyapunov analysis is the GSTA with variable gains. It is well known that first order SM algorithms with variable gains improve the performance of the system by decreasing the amplitude of the control signal and thus of the chattering. The GSTA with variable gains (VGSTA) has the following structure

$$\begin{aligned} \dot{x_1} &= -k_1\left(t,x\right)\phi_1\left(x_1\right) + x_2 + g_1(t,x) \\ \dot{x_2} &= -k_2\left(t,x\right)\phi_2\left(x_1\right) + g_2(t,x) \end{aligned} \tag{2.21}$$

where the functions g_1 and g_2 are disturbances and the VGSTA is composed of the gain functions $k_1(t,x)$, $k_2(t,x)$ and of the functions

$$\begin{aligned} \phi_1(s) &= |s|^{\frac{1}{2}} sign(s) + k_3 s \\ \phi_2(s) &= \tfrac{1}{2} sign(s) + \tfrac{3}{2} k_3 |s|^{\frac{1}{2}} sign(s) + k_3^2 s \quad , \quad k_3 > 0 \end{aligned} \tag{2.22}$$

It is assumed that the disturbances satisfy

$$|g_1(t,x)| \leq \rho_1(t,x)|\phi_1(x_1)| \quad , \quad |g_2(t,x)| \leq \rho_2(t,x)|\phi_2(x_1)| \tag{2.23}$$

then the sliding surface $s = 0$ will be reached in finite time if the variable gains $k_1(t,x)$, $k_2(t,x)$ are selected as

$$\begin{aligned} k_1(t,x) &= \delta + \tfrac{1}{\beta} \left\{ \tfrac{1}{4\epsilon} \left[2\epsilon\varrho_1 + \varrho_2\right]^2 + +2\epsilon\varrho_2 + \epsilon + \left[2\epsilon + \varrho_1\right]\left(\beta + 4\epsilon^2\right) \right\} \\ k_2(t,x) &= \beta + 4\epsilon^2 + 2\epsilon k_1(t,x) \ , \end{aligned} \tag{2.24}$$

where $\beta > 0$, $\epsilon > 0$, $\delta > 0$ are arbitrary positive constants.

What is very interesting about this approach is that it can serve as a guidance to develop adaptation methods of the GSTA gains k_1 and k_2.

Chapter 3

Description of the pressure machine

In this chapter a description of the pressure machine is given. As it was briefly mentioned in the section “motivation and state of the art”, the pressure machine that is going to be described has the commercial name of Variable Speed Pump (VsP), since this work is centered in an industrial application the later name will be used.

To begin with the description here is an extract from a press release of Bosch Rerxroth AG [21].

“With the variable-speed pump drives, Rexroth now offers pump control in a highly dynamic intelligent electrical drive which only generates the volume flow actually required. Reduced speed during breaks in the cycle or when not running at full power mean a significant drop in the energy required, in noise emissions and in hydraulic power losses. The pump drive increases the speed of the highly dynamic motors as required as soon as the hydraulic system needs more power. All components come from the standard Rexroth product portfolio...”

From the same source we can find the following description of the the operation concept.

“VsP with servo motor: In closed loop operation, a highly dynamic servo motor with servo converter for complex axis control functions provides for the required torques

Figure 3.1: Variable Speed Pump by Bosch Rerxroth AG

with very short acceleration times. The drive-integrated IndraMotion MLD Motion Logic System by Rexroth analyzes command values and actual values notified via a pressure measuring dose in a decentralized form and controls the required speed accordingly. Via open interfaces to all common field bus systems and Ethernet, the Variable-speed pump drives communicate with the superior control system."

Keeping apart the marketing style it can be summarized that:

1. The system is composed of an electric motor coupled with a pump.
2. The system is integrated into a hydraulic circuit and the measure of the pressure is available.
3. Finally, there is a dedicated programmable logic controller (PLC) to perform the control task.

In the following a description of the hardware, operation conditions and requirements will be given; then the linear mathematical model will be constructed and finally the well known, in industry, cascade control solution will be presented for this model.

3.1. Description of the VsP system

The present work has been tested using real industrial VsP systems by Rexroth. However detailed information of the system will not be revealed; in the following a simplified model of the VsP will be described; nevertheless this model is enough to describe the work.

The VsP system will be considered, for analysis, as part of a generalized hydraulic system where the hydraulic actuator(s) (cylinder(s), motor(s)) has been replaced by a proportional valve which is able to simulate the flow taken by the actuator. The generalized hydraulic system is displayed in figure 3.2. It consist of the following elements.

- Electrical motor coupled with a hydraulic pump.
- Storage tank.
- Fixed volume.
- Load valve.

The pump, driven by the motor, creates an oil flow between the fixed volume and the tank. Because the volume is fixed the pressure dynamic in it depends exclusively on the total flow (the sum of input flows minus the sum of output flows). The flows in the volume are the pump flow, the leakage flow and the load flow. In this case the load flow is given by the spool movement of a proportional valve.

The VsP control system is running on a PLC that, in most of the cases, is performing additional tasks besides realizing the control algorithm of the VsP (network functions, scheduling, system monitoring, etc.). Hence any control solution has to take into account that the processing capabilities are limited; even more a simple and lighter algorithm is better for industrial applications than a heavyweight assuming that they both achieve the same goal.

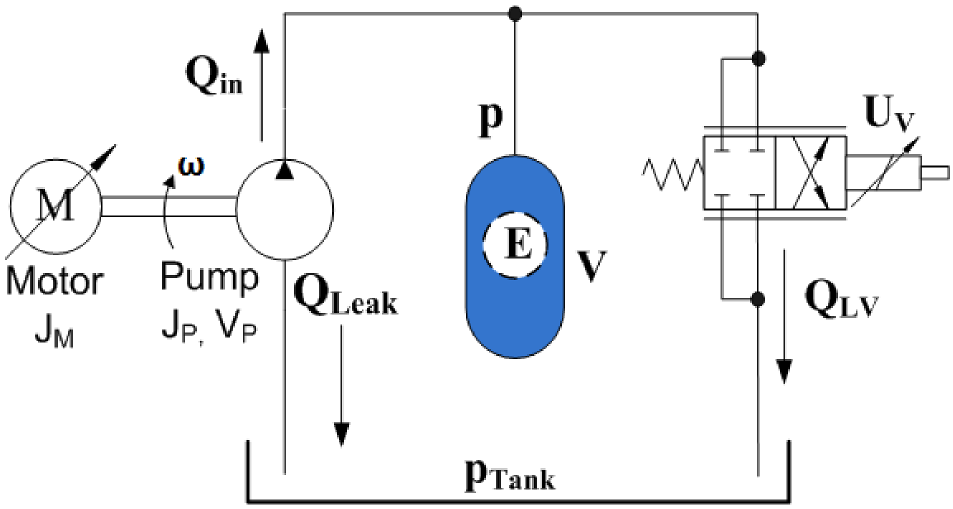

Figure 3.2: General hydraulic circuit for the VsP

As mentioned before, the electrical motor is a servo motor, this means that a control algorithm is already built in firmware. This fact may simplify the control design, since motor control is already done, but it also gives some extra challenges because, due to the protected firmware, there is a level where is not possible to change the control structure and the control algorithm of the motor is not exactly known. Nevertheless, it is well known that for speed control, servo motors use an approach based on the classical proportional integral control.

The main task of the VsP is to keep a desired pressure into the volume despite the leakage flow and the unknown load flow. To accomplish this task the VsP has measures of the motor's speed and the volume's pressure.

3.2. Modeling

As it has been mentioned before, one of the problems to create hydraulics control systems is the non-linear nature of hydraulics and the parameter uncertainties that are always present. In the case of the VsP system, the biggest challenge to create a good model comes from the unknown oil's bulk modulus; as mentioned in the antecedents the oil's bulk modulus depends on the pressure and temperature but the exact way in which this factors are related is still a case of study in the science of fluids; not to mention that the presence of contaminants in the oil, such as air, affect also the oil's bulk modulus. Normally, a constant value is assumed for the oil's bulk modulus and, if necessary, is experimentally tuned [2] . In this work a constant value will be assumed for the oil's bulk modulus and it will be shown later with experimental results that this assumption doesn't compromise the performance of the proposed control.

The generalized hydraulic system used to model the VsP is actually a very common system in the study of hydraulics since it is the base for a lot of industrial systems. Hence the system has been widely investigated including studies of the non-linear effects and pipe dynamics. However, as mentioned in the introduction the different type of models should serve for different purposes.

The present work shows that, with the help of sliding mode techniques, it is possible to construct a simple and easy to use control of a non-linear, parameter uncertain and perturbed plant; without any more information than the given by a simple linear model, and even less. Therefore a linear model of the VsP will be presented.

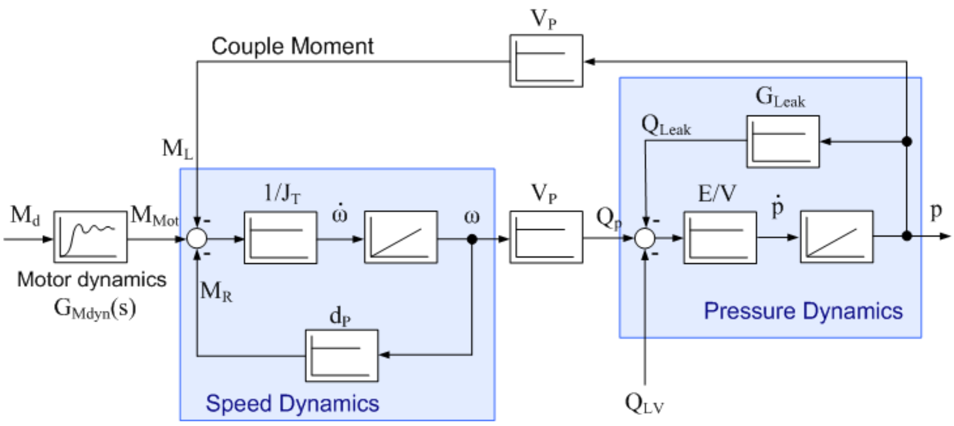

Figure 3.3: VsP Block diagram: the speed dynamics correspond to the pump coupled to the motor, the pressure dynamics correspond to the fixed volume. The main disturbance is the load flow Q_{LV} that is being taken out of the volume.

The block diagram of figure 3.3 shows the system where M_d is the input to the system (desired torque), the block motor dynamics refers to the motor dynamics $G_{Mdyn}(s)$ and firmware control these are considered sufficiently fast to be negligible, M_{mot} is the motor's moment (the output of the motor dynamics block), M_L the load moment due to pressure, M_R is the friction moment, J_T is the total inertia (pump inertia plus motor inertia),d_p is the viscous friction coefficient, ω is the pump speed, V_p is the pump volume displacement, Q_p is the flow from the pump, Q_{Leak} is the leakage flow, Q_{LV} is the flow taken out by the load valve, G_{Leak} is the leakage coefficient, E is the oil's bulk modulus, V is the fixed volume and p is the pressure in the fixed volume.

The system is described as a second order linear system. It can be argued that there are several non-linear effects in a hydraulic system being probably the most important the Bulk modulus that changes with pressure. However as mentioned in the introduction, there are different types of models for different purposes.

In this case, it is the intention of this work to use the simplest model in order to highlight the sliding mode robustness against unmodeled dynamics, parameters variation and disturbances. Therefore, it is also valid to use the linear analysis of the plant with all the well known techniques.

The transfer function of the system is given by (3.1)

$$P = \frac{V_p}{d_p G_{Leak} + V_p^2} \cdot \frac{\left(d_p G_{Leak} + V_p^2\right) \frac{E}{J_T V}}{s^2 + \left(\frac{d_p}{J_T} + G_{Leak}\frac{E}{V}\right) s + \left(d_p G_{Leak} + V_p^2\right) \frac{E}{J_T V}} \tag{3.1}$$

From (3.1) the following parameters can be obtained.

Gain

$$K_G = \frac{V_p}{d_p G_{Leak} + V_p^2}$$

Natural Frequency

$$\omega_n^2 = \left(d_p G_{Leak} + V_p^2\right) \frac{E}{J_T V}$$

Damping

$$D = \frac{\frac{d_p}{J_T} + \frac{G_{Leak} E}{V}}{2\sqrt{\left(d_p G_{Leak} + V_p^2\right) \frac{E}{J_T V}}}$$

It is important to notice that usually the designer tries to achieve the minimum viscous friction (represented by the parameter d_p) and also the minimum leakage (represented by the parameter G_{Leak}), so as these parameters tend to zero the damping will tend to zero and oscillations will be present in the system. It is also valid to infer that due to its small value the overall effect of these parameters in the natural frequency of the plant is minimum or even negligible. Thus a good approximation of the natural frequency can be obtained with four parameters. Two of these parameters, motor and pump inertia J_T, and pump displacement V_p, are measured and they can be consulted in catalogues, upper and lower bounds are known for the other two parameters. Therefore it is possible to do a good estimation of the plant natural frequency.

The disturbance caused by the load flow Q_{LV} can be modeled using equation (2.11), however, in practice this disturbance can be caused by different physical elements from which a model may not be available. Nevertheless there are two important properties of Q_{LV}:

Lipschitz continuous

Bounded

$$|Q_{LV}| \leq L$$

Where the bound L is usually known.

3.3. Current control solution

The standard controller solves the following problem:

Control Problem: keep the system pressure at the desired pressure despite parametric variations, uncertainties and disturbances. The desired pressure is in general a square signal.

From a classical point of view there are two possibilities here.

- Cascade control

- State feedback control

Due to reasons related to hardware capabilities and availability of signals (pressure and motor's speed are measured), the cascade scheme was selected to solve the control problem, the scheme that results of this selection is displayed in figure 3.4.

A brief explanation of this scheme is

Inner Loop The inner loop features a PI scheme that controls the motor's speed. It is assumed that the PI scheme decouples the moment due to pressure M_L and also rejects the friction.

Outer Loop The outer loop features a PID scheme that controls the system pressure. The motor's speed is seen as a commanded flow to the system.

It is important to mention that the scheme of the inner loop is already defined by the motor firmware, moreover, the PI gains are preselected by the provider. Normally the user tunes just the outer loop (PID) based on the system volume; additionally to the PID gains, the user tunes other parameters to ensure good performance and a reliable operation. In the general case, the user tunes 11 parameters for a given volume.

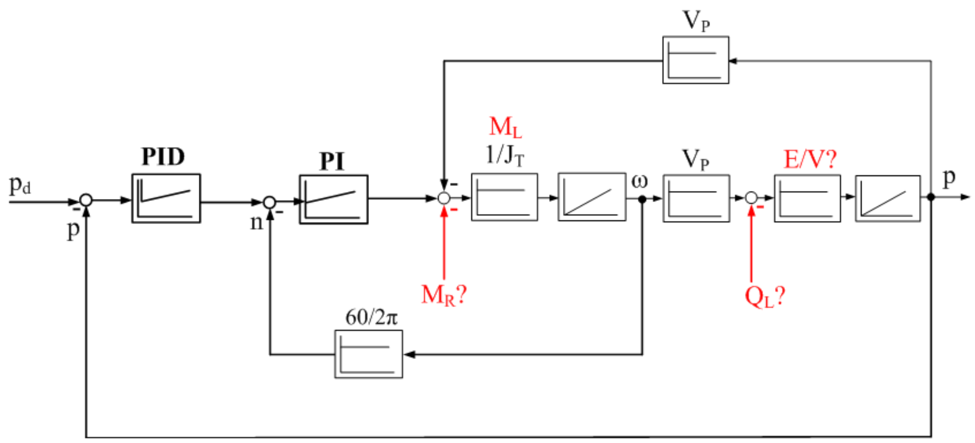

Figure 3.4: The standard solution for this kind of System is the cascade control, in this case the inner loop features a PI that controls the motors velocity, the outer loop features a PID that controls the system pressure.

Once the scheme is tuned the control task is achieved.

Chapter 4

Control design

4.1. Control Problem

The main objective of this work is to present a better alternative to the standard control used today. In this sense, the design has to take into account three equally important goals:

Algorithm formulation: an algorithm that solves the control problem has to be designed.

Implementation: the algorithm has to be implemented and tested using the existing software, firmware and hardware without any change.

Customer/User friendly: users without deep control knowledge should be able to use the system.

The three goals are equally important because a control law or algorithm that solves the control problem is useless if it does not run in the real machine (where the computational resources are limited, underlaying firmware exists and actuators have limited capacities), also final users have to be able to operate the machine for their daily work without taking any course in control theory. The first goal is mandatory, an algorithm that performs the control task is needed. The second goal puts the feet of the solution to the ground, the solution needs to be installed in the existing machine not in an idealized version of the machine. The final goal raise an interesting challenge because the control algorithm has to deal with parameter uncertainties, unknown dynamics and disturbances without high complexity for the user.

The solution proposed here will achieve the goals if it pass the following tests:

- The algorithm works successfully in a real experiment (not simulation).
- The algorithm is tested successfully using the existing machine without any change or modification.
- The solution is less complex for the user than the standard solution.

It will be shown that the key to success is the use of sliding mode techniques.

4.1.1. Basic Formulation

The control problem is the same as for the standard controller

Control Problem keep the system pressure at the desired pressure despite parametric variations, uncertainties and disturbances. The desired pressure is in general a square signal.

The proposed solution is a cascade controller where the inner loop, that as mentioned before, is already defined in the motor firmware, and the outer loop is a linear proportional feedback plus a feedforward of the disturbance. The basic idea is presented in figure 4.1 where the signal $\hat{Q}_L$ is feed forwarded to cancel the disturbance Q_L. The signal $\hat{Q}_L$ is an online estimation of the disturbance Q_L.

The proposed controller can be described as a **Cascade P-PI plus feedforward of the load flow.** The inner and outer feedback loops are used to tune the plant dynamics, once the dynamics are selected the feedforward cancels all the flow losses assuring that the system always keeps the desired pressure. The controller features the following advantages

- Easy and understandable rules for the tuning procedure can be given to the final user (in the simpler case the user needs to tune just one proportional gain).
- Stable preconfiguration can be done before delivery.

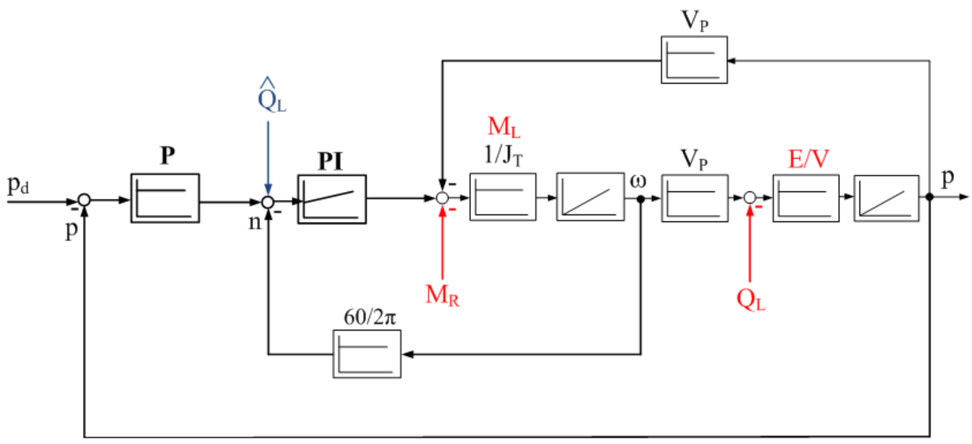

Figure 4.1: The cascade control with the feed forward of the estimated load flow $\hat{Q}_L$.

- All the parameters needed to run the control can be obtained from catalogues.
- There is no need to know the load flow signal in advance.

4.2. Basic linear scheme

We are going to introduce a linear controller which in absence of disturbance or parameters uncertainties will give the desired performance.

Consider the system (3.1), we are going to apply a linear cascade scheme consisting of an inner loop and an outer loop to this system :

- The inner loop will control the speed ω of the motor using a proportional-integral feedback of the speed error:

$$u_n = K_n \left(1 + K_i \tfrac{1}{s}\right) (\omega_d - \omega) \tag{4.1}$$

- The outer loop will control the pressure P of the hydraulic system using a proportional feedback of the pressure error:

$$u_p = K_p \left(P_d - P\right) \tag{4.2}$$

At the same time we are going to cancel the couple moment M_L by doing a feedforward using the desired pressure instead of actual pressure, this feedforward will be detailed in the next section. Figure 4.2 displays the cascade scheme plus the feed forward to cancel M_L.

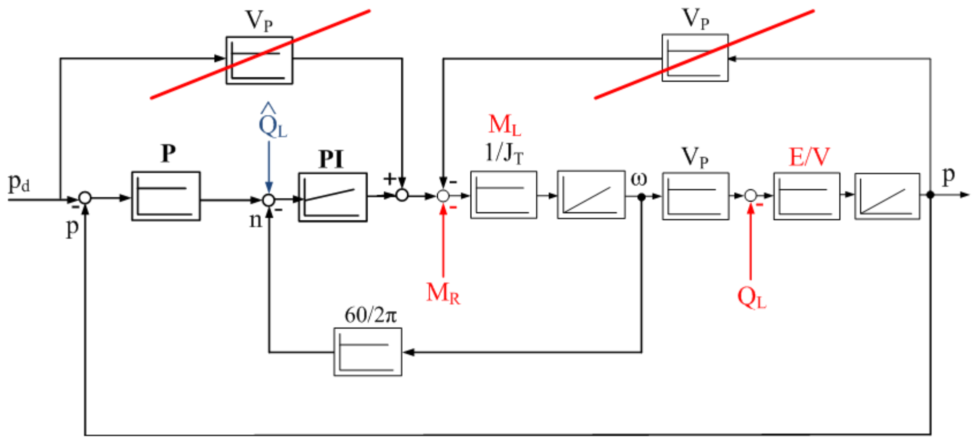

Figure 4.2: The control scheme with the feedforward of the couple moment, the couple moment cancels the moment due to pressure.

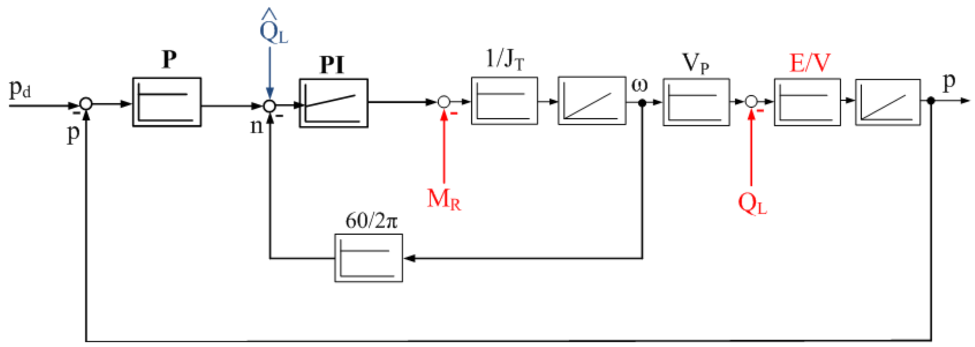

Figure 4.3: The resulting system.

Figure 4.3 displays the resulting decoupled system. For this decoupled system we can get the following closed loop transfer function (4.3). It is important to remark that $K_i > 0$, thus the zero $(s + K_i)$ is always in the open left plane and there are not non-minimum phase zero effects.

$$P(s) = \frac{\frac{K_p K_n V_p E}{V J_T}(s+K_i)}{s^3+\frac{K_n}{J_T}s^2+\left(\frac{K_p K_n V_p E}{V J_T}+\frac{K_n K_i}{J_T}\right)s+\frac{K_p K_n V_p E K_i}{V J_T}} P_d(s) \tag{4.3}$$

A description of the parameters and their units can be found in table 4.1.

Table 4.1: Parameters of the system

Name	Description	Units
J_T	Total inertia	$[kg \cdot m^2]$
E	Bulk Modulus	$[Pa]$
V	System Volume	$[m^3]$
V_p	Pump Displacement	$[m^3/rad]$
K_n	Proportional gain of the inner loop	$[Nm/rad \cdot s]$
K_i	Integral gain of the inner loop	$[1/s]$
K_p	Proportional gain of the outer loop	$[rad/s \cdot Pa]$

Notice that if the integral gain K_i is equal to zero the system order is reduced and we obtain the following transfer function (4.4).

$$P(s) = \frac{\frac{K_p K_n V_p E}{V J_T}}{s^2+\frac{K_n}{J_T}s+\frac{K_p K_n V_p E}{V J_T}} P_d(s) \tag{4.4}$$

System (4.4) is order two, and using the general form of a second-order system, we are able of write down the system parameters as follows:

Natural frequency

$$\omega_n = \sqrt{\frac{K_p K_n V_p E}{V J_T}} \tag{4.5}$$

Damping

$$D = \frac{K_n}{2\omega_n J_T} \tag{4.6}$$

4.2.1. Gains selection

We propose three methods to select the gains K_p and K_n, the user can choose which method to use based on its own knowledge and needs.

First method Select the desired closed loop frequency ω_n and the desired damping D

Then the gains are given by:

$$K_n = 2D\omega_n J_T$$

$$K_p = \frac{\omega_n^2 J_T V}{K_n V_p E}$$

This rule is recommended for users with a good control theory knowledge and also a good knowledge of the plant. The users has to take into account that there is certain values of ω_n that are not reachable.

Second method Fix the proportional gain K_n and select a damping D

$$K_p = \frac{K_n V}{4EJ_T V_p E D^2}$$

This is the best rule for the average user, ω_n is already defined with a fixed K_n thus the users can adjust the damping starting from the value 3 and moving it in steps of 0.100 towards the value of 1.

Third method Fix the proportional gain K_n and then tune manually K_p, avoid values of D close to zero. For the users who claim that they have a good understanding of cascade controllers and they can adjust the proportional gain manually. Nevertheless there is maximum value for the tuning of K_p when the damping D is minor or equal to 0.20

This tuning methods give flexibility to the application, because people with different background are able to tune and use the machine.

With respect to K_n, it can be argued that fixing its value limits the possibilities of the machine, however in practice users are reliant to change the preconfigured gains of the electrical motor because they feel safer when they use the manufacturer tuning.

4.2.2. Load Moment Feedforward

The Load moment or couple moment is defined as

$$M_L = V_p P$$

Where P is the pressure of the system. So it is straightforward to propose the feed forward as the product of the pump volume and the measured pressure, the resulting torque can be fed to the desired torque input of the motor firmware. However, this solution will fail because of the pressure noisy signal (the motor firmware doesn't handle too well the noise in this particular input).

A solution to this problem is to use the desired pressure, which is nice and clean.

- To use the desired pressure P_d it is assumed that the system pressure and the desired pressure will be the same because of the reliable control system.

- The desired pressure is assumed to be a square signal, in order to obtain a smooth signal for the estimated load torque $\hat{M}_L$ the filter (4.7) is used.

- Since this filter represents the behaviour of the real system P_d and P are almost the same.

The equation of the smooth, noise free, feedforward is

$$\hat{M}_L = V_p \frac{\frac{K_p K_n V_p E}{V J_T}}{s^2 + \frac{K_n}{J_T} s + \frac{K_p K_n V_p E}{V J_T}} P_d(s) \tag{4.7}$$

where P_d is a square signal.

4.3. Sliding mode observer

In this section two sliding mode observers are proposed. First the advantage of using an observer is going to be presented. A good survey on state and disturbance observers can be found on [22]. A survey on sliding mode observers can be found on [23]. Moreover, the identification of disturbances and its compensation in control schemes using sliding mode observers has been exhaustedly studied by Ferreira et al [24], [25], [26], [27], [28], [29].

Consider the system as displayed in figure 4.3, where there is a interchange of flows that create a pressure change in the system. Such behavior can be described using equation (2.7), for this system we can write

$$\dot{P} = \frac{E}{V}\left(V_p * \omega - Q_L\right) \tag{4.8}$$

remember that the the flow given by the pump is calculated as

$$Q_p = V_p * \omega \tag{4.9}$$

In this case Q_L represents any flow that is not the pump flow (leakage, flow taken by the system). Notice that ω is measured, V_p is available in the catalogue data and there is an estimate of E and V. The only missing piece here is the disturbance Q_L, which is not negligible.

The behavior of the linear scheme proposed in the last section with respect Q_L is commented in the following points:

- If Q_L is equal to zero and there is no error in the parameters estimation, then the controller presented in the last section will have an ideal performance.
- If Q_L is equal to zero and there is some error in the parameters estimation, then the controller presented in the last section *plus an integral action* in the outer loop will provide a good performance.
- If Q_L is not equal to zero but it is constant and there is some error in the parameters estimation, then the controller presented in the last section *plus an integral action* in the outer loop will provide a good performance.
- If Q_L is not equal to zero, it is not constant, and there is some error in the parameters estimation, then the controller presented in the last section *plus an integral action* in the outer loop will **not** provide a good performance, unless the rate of change is relatively small.

Notice that if a derivative part is added to the outer loop then the standard control scheme is obtained (PID-PI).

Naturally the first case is the simplest, but unfortunately Q_L is not zero and there are parameter uncertainties. However if we know the value of Q_L we can subtract it from the equation giving a total value of zero. In order to do this lets remember equation (4.9) and assume that a estimated value $\hat{Q}_L$ is available, then we can obtain the pump speed that is needed to generate such flow as

$$\omega_c = \frac{\hat{Q}_L}{V_p}$$

in this case the speed ω_c is the one needed to create the flow $\hat{Q}_L$, additionally to the speed ω_c the system still needs the speed ω, commanded by the linear scheme, that builds up the pressure; hence the total speed of the pump is the sum of this two speeds $\omega_{total} = \omega + \omega_c$. The total effect on the fixed volume can be described as

$$\dot{P} = \frac{E}{V}\left(V_p * (\omega + \omega_c) - Q_L + \hat{Q}_L\right)$$

to clarify the intention of each speed, we can rewrite the past equation as

$$\dot{P} = \frac{E}{V}\left(V_p * \omega - Q_L + \hat{Q}_L\right)$$

If $Q_L - \hat{Q}_L = 0$ then any flow that is not the pump flow is equal to zero, this results in the following equation where the pressure dynamics is driven just by the motor speed ω.

$$\dot{P} = \frac{E}{V}\left(V_p * \omega\right)$$

Clearly for this solution to work there is a need of knowing at all time the value of $\hat{Q}_L$. This could be achieved if flow sensors are added to the system, however, this action will increase significatively the total cost of the solution, and it will make maintenance of the system harder. Nevertheless it is possible to obtain the signal of $\hat{Q}_L$ without using flow sensors, thus the signal $\hat{Q}_L$ will be obtained from the available data using a special algorithm. Therefore the control solution presented is *sensorless control.*

The algorithm that makes this possible is a sliding mode observer which identifies the disturbance Q_L as $\hat{Q}_L$, it will be presented in the next section. This algorithm will not only give the value of the disturbance but it will also cancel the problem of the parameters uncertainties.

4.3.1. First order sliding mode observer

The dynamics of the pressure are given by equation (4.8), to make an approximation of this behavior we can construct the following dynamic system:

$$\dot{\hat{P}} = \tfrac{E}{V}\left(V_p * \omega\right)$$

This system is constructed with what is known. Here we assume that the values of E and V are known, experimental results will show that the results are still valid when parameter uncertainties are present. Because of the disturbance there is an error e between the estimate pressure $\hat{P}$ and the real P.

$$e = \hat{P} - P$$

Since the value of P is available from measure and the value of $\hat{P}$ is available from calculation we can calculate the error and use it to correct the estimate pressure $\hat{P}$ and its derivative $\dot{\hat{P}}$.

An exact estimate is achieved whit the use of sliding mode techniques, such as, the first order sliding mode showed in equation (4.10).

$$\dot{\hat{P}} = \tfrac{E}{V}\left(V_p * \omega - L \cdot sign(e)\right) \tag{4.10}$$

In this case the function $sign(\cdot)$ is the signum function, and L is a positive gain. The dynamic error analysis can be done as showed in the following equation.

$$\begin{aligned} \dot{e} &= \dot{\hat{P}} - \dot{P} \\ \dot{e} &= \tfrac{E}{V}\left(V_p * \omega - L \cdot sign(e)\right) - \tfrac{E}{V}\left(V_p * \omega - Q_L\right) \\ \dot{e} &= \tfrac{E}{V}\left(Q_L - L \cdot sign(e)\right) \end{aligned}$$

Let us assume that the system is stable and the error e reaches zero in finite time and stays there; then if the error stays at zero and does not move the derivative of the error is equal to zero $\dot{e} = 0$. With this result it is possible to obtain the value of Q_L as showed in equation (4.11)

$$\begin{aligned} 0 &= \tfrac{E}{V}\left(Q_L - L \cdot sign(e)\right) \\ 0 &= Q_L - L \cdot sign(e) \\ L \cdot sign(e) &= Q_L \end{aligned} \tag{4.11}$$

$L \cdot sign(e)$ is a discontinuous signal, to make it smooth enough so it can be fed as a desired speed we use a first order filter as displayed in equation (4.12).

$$\tau\dot{\eta} = \eta + u \tag{4.12}$$

Where the parameter τ is the time constant of the filter and u is the input.

The condition for this algorithm to remain stable is that the disturbance Q_L is smaller than the gain L. This condition is achieved easily because Q_L will always be smaller than the maximum flow that the pump is capable to provide. This maximum flow is calculated using the maximum speed of the pump ω_{max} and the pump volume displacement V_p, both are available in the pump catalogue. Therefore we can assign the value of L using just the catalogue data as displayed in the following equation:

$$L = V_p\omega_{max}$$

Notice that because of the direct calculation there is no need to tune the gain of the observer.

Finally, as it was mentioned before, the speed ω and the pressure P are measured, therefore the initial condition of the estimated pressure $\hat{P}(t_0)$ can be directly assigned as $\hat{P}(t_0) = P(t_0)$, thus the reaching phase (when the sliding surface e is going from its initial conditions to zero.) is eliminated. This action is implemented within a set of instructions called *reset*. The reset is executed at the beginning of the algorithm, and also when extraordinary events happen, such as alarms, system failures, etc.

Summarizing,

- This scheme is a feedforward that estimates online the disturbance Q_L and rejects it, leaving the linear control disturbance free.
- The value of Q_L can be obtained using the first order sliding mode observer (4.10).
- The signal of the first order sliding mode observer is discontinuous, therefore a first order filter (4.12) is used.
- The output of the filter is the estimated disturbance, and it is fed as a desired speed to the inner loop, after it is converted from flow to speed and scaled to the right units.
- For better results two first order filters are used in a serial scheme.

A formal proof that assures the stability of this algorithm, when uncertain parameters are present, can be found in [7].

4.3.2. Second order sliding mode observer

To complete the feedforward scheme there is a need to include a first order filter in the previous algorithm. As can be seen in equation (4.11) the identification is done through a discontinuous signal. This fact is not itself a problem (high frequency discontinuous signals are used everyday to actuate electrical motors) however the lack of capacity to deliver a high frequency signal of the commonly used control hardware is a problem. To overcome this problem the discontinuous signal is smoothed through the use of low pass filters, described in equation (4.12).

Another method to overcome the problem of the discontinuous signal is the use of high order sliding modes. These techniques deliver a continuous signal and they keep most of the first order sliding mode properties. Actually, in most practical cases nothing is lost in terms of properties between first and high order sliding modes.

For this work we are going to use a second order sliding mode called generalized super-twisting algorithm (GSTA) [19], [18]. Equation (4.13) displays the GSTA.

$$v = -k_1\left(t,x\right)\phi_1\left(e\right) - \int_0^t k_2\left(t,x\right)\phi_2\left(e\right)dt \ , \tag{4.13}$$

where

$$\begin{aligned}\phi_1(e) &= |e|^{\frac{1}{2}}\, sign(e) + k_3 e \\ \phi_2(e) &= \tfrac{1}{2} sign(e) + \tfrac{3}{2} k_3 |e|^{\frac{1}{2}}\, sign(e) + k_3^2 e \quad , \quad k_3 > 0 \ .\end{aligned}$$

Where $k_1(t,x) > 0$, $k_2(t,x) > 0$ are variable gains, and $k_3 \geq 0$ is a fixed gain. This algorithm is stable in finite time ($e = 0$, $\dot{e} = 0$) which means that the error and its derivative will be equal to zero in a time before infinite as it happens in the linear algorithms.

In practical applications this finite time convergence means a faster convergence and the possibility to neglect the transitory response of the algorithm.

The GSTA can be used in equation (4.10) instead of the first order sliding mode as showed in the next equation.

$$\dot{\hat{P}} = \tfrac{E}{V}\left(V_p * \omega - k_1(t,x)\,\phi_1(s) - \int_0^t k_2(t,x)\,\phi_2(s)\,dt\right) \tag{4.14}$$

this can be written using v to make it more simple.

$$\dot{\hat{P}} = \tfrac{E}{V}\left(V_p * \omega + v\right)$$

In this case, after doing the same error analysis, it is possible to find an equivalence like the one of the first order sliding mode (4.11) but in this case for the GSTA.

$$\begin{aligned} 0 &= \tfrac{E}{V}\left(Q_L + v\right) \\ 0 &= Q_L + v \\ -v &= Q_L \end{aligned} \tag{4.15}$$

For this to happen it is necessary that the error e and its derivative $\dot{e}$ are equal to zero, in other words the algorithm should be stable.

The GSTA will be stable in the presence of a disturbance if:

- The disturbance's derivative exists and it is bounded
- The gains are selected according to this bound.

The first point can be assured because the disturbance has the shape of equation (2.11) and therefor the derivative exists and it's bounded. To assume that the derivative of the flow Q_L is not bounded will mean that the flow derivative $\dot{Q}_L$ has an infinite value which is impossible. For the second point a new adaptive method for the gains will presented and experimentally proved. For now we will consider that both conditions are met.

The effect of chattering is significantly reduced between the use of first order sliding modes and the GSTA, however a small level of chattering is still present [30]. Consequently a first order filter, like the one presented in (4.12) its used to make the signal chattering free, the advantage with respect to the first order sliding mode approach is that the filter delay can be smaller and therefore the response will be faster when using the GSTA.

Finally, the algorithm also uses the reset described in the first order sliding mode observer, thus there is no reaching phase.

Summarizing,

- This scheme is a feedforward that estimates online the disturbance Q_L and rejects it, leaving the linear control disturbance free.
- The value of Q_L can be obtained using the Generalized Super Twisting Algorithm as observer (4.13).
- The signal of the GSTA observer still have a small level of chattering, therefore a first order filter (4.12) is used.
- The output of the filter is the estimated disturbance, and it is fed as a desired speed to the inner loop, after it is converted from flow to speed and scaled to the right units.
- Due to the small level of chattering in the GSTA just one first order filter is used.

4.4. Adaptive gains for the GSTA

There are several methods that try to adapt the gain of sliding mode algorithms see for example [31], [32] and [33], a good survey is given in [34]. In this work those methods are classified based on their criterion to change the gain as:

Equivalent control methods In this methods the equivalent control of the sliding mode algorithm is obtained and then this is used as a decision criteria to check if the sliding surface is reached. Based on this criterion it is possible to develop several styles of adaptive schemes. For example, in the first order sliding mode it can be considered that the sliding surface has been reached if the equivalent control c_e is between the values of the sliding mode gain $L > 0$.

$$-L \quad < \quad c_e \quad < \quad L$$

Then if this condition is not met ($|c_e| \geq L$) the gain should be increased, However if the condition is met ($|c_e| < L$) the gain could be smaller.

Sliding surface bound methods In this method a bound S_b is established; then the decision criteria is to check if the distance to the sliding surface is bigger or smaller than the bound. If the distance to the sliding surface is smaller than the bound then it is considered that the sliding surface is reached, if the distance to the sliding surface is bigger than the bound then the it is considered that the sliding surface is not reached. Based on this criterion it is possible to develop several styles of adaptive schemes. For example, in the first order sliding mode the following set of rules can be used

$$\begin{aligned} if \quad s < S_b \quad &\rightarrow \quad L- \\ if \quad s > S_b \quad &\rightarrow \quad L+ \end{aligned}$$

where s is the sliding surface, $L+$ means increase the gain, and $L-$ means decrease the gain.

The advantages of both types of methods is that they allow to estimate the gain needed to stabilize the algorithm. Some of them, additionally try to find a not too overestimated gain in order to improve the performance of the algorithm, according to the idea that a bigger gain gives bigger chattering. There are disadvantages for both types:

Disadvantages of the sliding surface bound methods are:

- There are very few methods to tune the bound S_b, and they do not consider the noise in the signals.
- Due to noise, chattering, and a bad bound S_b tuning the gain could increase infinitely and the whole system could become unstable.

Disadvantages of the equivalent control methods are:

- Most of the methods use a low pass filter to obtain the equivalent control, however tuning the filters is not a simple task.
- Filters used to obtain the equivalent control also induce a delay into the algorithm, thus the sliding mode could be lost before the algorithms has knowledge of it.

Because of the risk of instability present in the sliding surface bound methods in this work a equivalent control method is proposed.

4.4.1. Algorithm formulation

The adaptive gain algorithm proposed here was tested in both simulation and real experiments. The algorithm finds the gains K_1 and K_2 that stabilize the Generalized Super Twisting Algorithm (4.13), moreover the algorithm does not overestimate the gains. Also there is no need to know the bound of the disturbance or the bound of the disturbance derivative, however the conditions for the GSTA are still valid.

The algorithm is formulated in a step by step form:

1. Run the GSTA (4.13) with any valid initial values for the gains K_1, K_2, K_3.
2. Filter the signal $sign(e)$ of the GSTA using the low pass filter (4.12).
3. Label the output of the last filter as η_1.
4. Filter the signal $|\eta_1|$ (absolute value) using a low pass filter (4.12).
5. Label the output of the last filter as η_2.
6. Obtain the quantized value q and the weighted quantized value q_w of η_2.
7. Drop the oldest sample of the set X and add the value q, obtained in the last step, to the set.
8. obtain the media μ of the set X.

9. Apply the following rules

$$\begin{cases} \quad\quad\; if \quad q = 1 & \rightarrow \quad k_n+ \\ else \quad if \quad q > \mu + \sigma_d & \rightarrow \quad k_n+ \\ else \quad if \quad q < \mu + \sigma_d & \rightarrow \quad k_n- \end{cases}$$

where k_n+ is defined as

$$\dot{k_n} = \alpha_2 \cdot q_w$$

and k_n- is defined as

$$\dot{k_n} = -\alpha_1 \cdot (1.01 - q_w)$$

10. Obtain the value of k_n and apply the following saturation:

$$\begin{cases} if \quad k_n > 1 & \rightarrow \quad k_n = 1 \\ if \quad k_n < k_{min} & \rightarrow \quad k_n = k_{min} \end{cases}$$

11. Obtain the value of K_1 and K_2 as follows.

$$\begin{aligned} K_2 &= L_{max} \cdot k_n \\ K_1 &= \delta \cdot K_1 \end{aligned}$$

12. repeat from step 2

The parameters used in the algorithm are briefly explained here:

- The set X has a fixed length k, in the set X the latest values of q are kept to make it possible to calculate a media μ. For example if q_n is the latest quantized value of η_2, then the set should have the $k-1$ values before q_n, i.e.

$$X = \{q_{n-k}, q_{n-k-1}, ..., q_n\}$$

When the algorithm starts X contains just zeros. The algorithm does not output any gain until the population of X is complete for the first time.

- The quantized value q is used to eliminate the residual noise from the less significant numbers.

- The weighted quantized value q_w is used to the same purpose as the quantized value, and also to have a different behavior between the values of η_2 close to zero and the values close to 1 on the gain dynamics. (Code to obtain the quantized values and the weighted quantized values can be found in the appendix A.1)
- The parameter σ_d is a deviation defined by the user.
- The parameters α_1, α_2 are maximum growing and decreasing rates allowed.
- The parameter k_{min} is the minimum value allowed for k_n, the value is defined by the user.
- The parameter L_{max} gives the maximal value for K_2 and K_1.
- The parameter $\delta > 0$ relates K_2 and K_1

Chapter 5

Results

5.1. Experiment description

The test bench is displayed in figure 5.1. This test bench features the industrial VsP system connected to an hydraulic system which allows some modifications in order to recreate the conditions of different industrial plants. The whole system is normally used to tune the standard controller of the VsP before deliver it to the customer, thus, the system recreates accurately the industrial environment.

To get a better insight into the the test bench figure 5.2 displays an schematic of the plant. From left to right and from up to down: the IndraDrive is the Rexroth proprietary PLC, a computer used to configure the PLC and to display the measures, the Cronos Data Acquisition is the device used to work, from the PC, with the sensors and to actuate the valves (it is intended for the user/designer, the PLC does not need it), pressure sensor of the high pressure side of the system (there are other pressure sensors not displayed here), proportional valve simulates the work cycle of an industrial hydraulic system by taking flow from the high pressure to the low pressure, VsP coupling is the motor coupled with the pump, pressure relief valve keeps the safe operation of the system, switch valve (SV) 1, 2, and 3 select the volumes 1, 2, and 3 respectively changing the total volume of the high pressure side of the system.

As mentioned before, the closed loop system when a P-P cascade is used to control

Figure 5.1: The test bench, industrial laboratory, Bosch Rexroth company, Lohr am Main, Germany

can be expressed with equation (4.4), displayed here:

$$P(s) = \frac{\frac{K_p K_n V_p E}{V J_T}}{s^2 + \frac{K_n}{J_T} s + \frac{K_p K_n V_p E}{V J_T}} P_d(s)$$

Table 5.1 shows the parameters of the system with the values for the test bench. It can be observed that just the values of the inertia and the pump displacement are fixed, because they are given by the manufacturer. The value of the Bulk modulus varies with the pressure as mentioned before, however it is considered constant in the model. The value of the system volume can be changed in the range displayed in the table using the switching valves 1, 2, and 3. The maximum pump speed is available in the pump catalogue, the value is given in revolutions per minute [rpm]. The gains of the controller will be specified in each experiment.

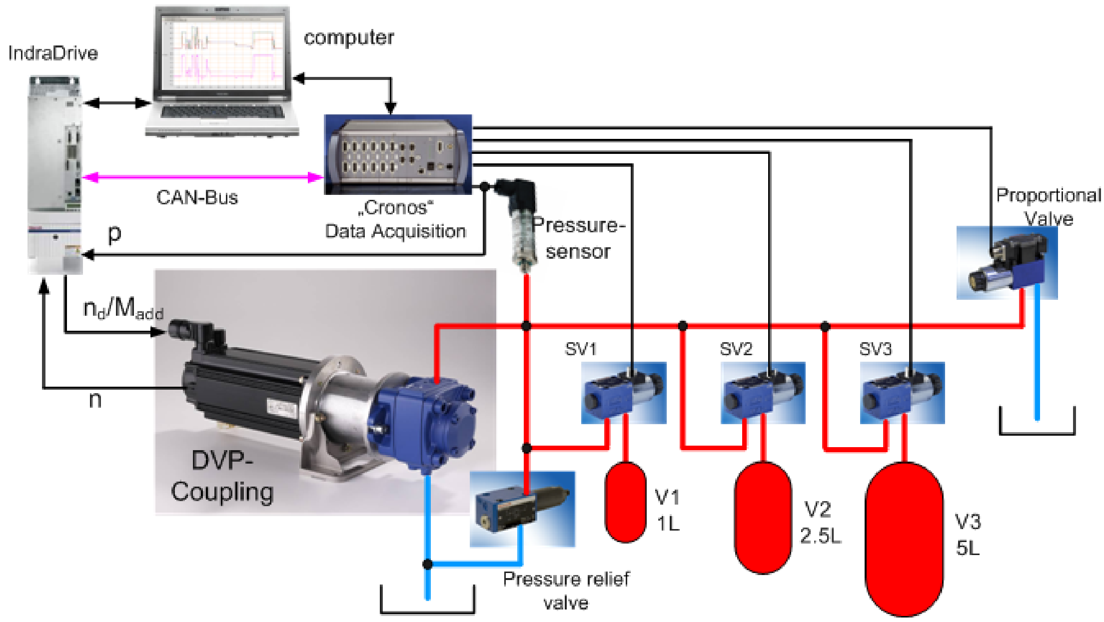

Figure 5.2: Test bench schematic

5.1.1. Open loop step response

The step response is calculated for the system in open loop, equation (3.1) describes the open loop behavior of the system.

In this case the input is a torque and the value of this input is calculated as follows

$$M_{in} = V_p \cdot P_d$$

where P_d is a desired pressure in pascals $[Pa]$. The idea is that the input M_d will be canceled with the couple moment M_L caused by the system pressure P when both the desired pressure and the system pressure are equal. However there is no feedback control, therefore, due parameter uncertainties and disturbances like friction and leakage, the output of system (3.1) will show an error with respect of the desired pressure P_d.

The volume of the system V for this experiment was 1.5 liters, this is the minimum volume of the system.

Figure 5.3 displays the step response of the real system when the input M_{in} is applied using a rectangular signal for the desired pressure P_d that goes from 80 $[bar]$ to 100 $[bar]$. Figure 5.4 shows a zoom in the rising step. Here the frequency of the oscillations can

Table 5.1: Parameters of the test bench

Name	Description	Units	Value
J_T	Total inertia	$[kg \cdot m^2]$	0.0069
E	Bulk Modulus	$[Pa]$	$9000 \cdot 10^5$
V	System Volume	$[m^3]$	$[1.8 - 11.5] \cdot 10^{-3}$
V_p	Pump Displacement	$[m^3/rad]$	$4.0266 \cdot 10^{-6}$
ω_{max}	Maximum pump speed	$[rpm]$	3000
K_n	Proportional gain of the inner loop	$[Nm/rad \cdot s]$	*
K_i	Integral gain of the inner loop	$[1/s]$	*
K_p	Proportional gain of the outer loop	$[rad/s \cdot Pa]$	*

be measured, the result is showed next as the open loop frequency f_{ol} in Hertz $[Hz]$

$$f_{ol} = 4.3478[Hz]$$

It is possible to use the natural frequency measured in this experiment to calculate the value of the parameter Bulk Modulus E. To obtain the Bulk Modulus this open loop response experiment is going to be repeated with the four volumes available in the plant (Using the previously described Switch valves it is possible to change the volume of the system), once the data is obtained with the experiments it is possible to use the model described in (3.1), particulary, the description of the natural frequency can be used to obtain the experimental value of the bulk modulus as displayed in the following equation:

$$E = \frac{\omega_n^2 \cdot V \cdot J_T}{V_p}$$

notice that the values of leakage flow and viscous friction have been neglected ($d_p = 0$, $G_{Leak} = 0$). After doing experiments with different volumes we obtained the data displayed in table 5.2.

Table 5.2: Volume, frequency and Bulk modulus

Volume [l]	frequency [Hz]	Bulk Modulus [bar]
1.8	4.3478	6200
3	3.5928	6600
5.8	2.7497	7474
11.1	2.0134	7669

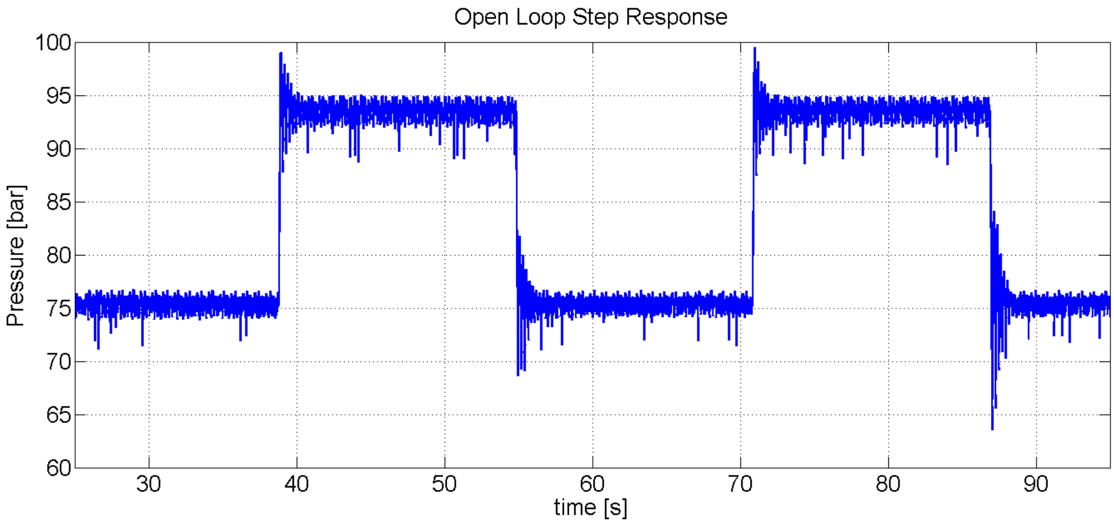

Figure 5.3: Open loop step response showing the output pressure of the system when the input M_in is applied, The desired pressure P_d is a rectangular signal from 80 $[bar]$ to 100 $[bar]$

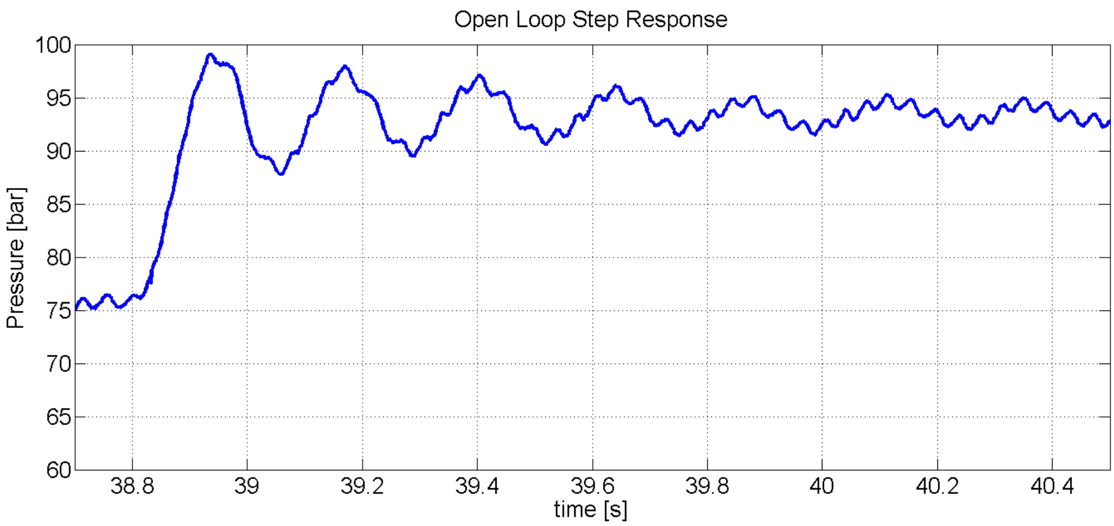

Figure 5.4: Zoom of the open loop step response showing the output pressure of the system

5.1.2. Closed loop model identification

To check how accurate is the model (4.4), it is possible to compare the response of the model and the response of the test bench. Using this comparison it is possible to tune the value of the uncertain parameters bulk modulus E, and system volume V, in the the quotient form $\frac{E}{V}$.

The experiment is conducted as follows:

1. The parameters of the model (4.4), except the value of the control gains K_p, and K_n, are set using the values of Table 5.1.
2. The model (4.4), and the test bench feature a P-P cascade. Control gains K_p, and K_n are equal on both models and they are calculated using the methods that were given in the control design.
3. In order to check the step response the desired pressure P_d is a square signal.
4. The input P_d is given in parallel to the model and to the real plant.
5. The resulting responses are compared to tune the value of the quotient $\frac{E}{V}$.
6. Repeat from step 4 until the responses are similar.

The control of the real plant (and the model) is a P-P cascade, thus it doesn't provide an exponential asymptotic response to the desired value in real conditions, therefore the real plant and the model will converge to different final values. However, since we are just comparing the dynamics of the response (frequency and damping) and not is final value, it is possible to scale the response of the real plant, **in magnitude**, to make the comparison easier.

As part of this work an algorithm that automatic scales the signal of the real pressure to fit the one of the model was developed and tested on Matlab.

Figure 5.5 displays a comparison between the real plant response and the model response. It can be seen that despite the nonlinear nature of the plant it is possible to approximate its dynamics by the linear model. In this experiment the switch valve 1, and 2 were open; this means that the system volume was approximately 5 liters. The desired pressure goes from 80 $[bar]$ to 100 $[bar]$.

The desired pressure range was selected, for this experiment in particular, in order to avoid the saturation of the actuator or the dry friction when the pump stops rotating; to accomplish this objective the selected range was optimal. It is important to remark that this experiment was conducted with all the volume configurations allowed by the switching valves, the data presented here is just one of this experiments, the other experiments presented similar results.

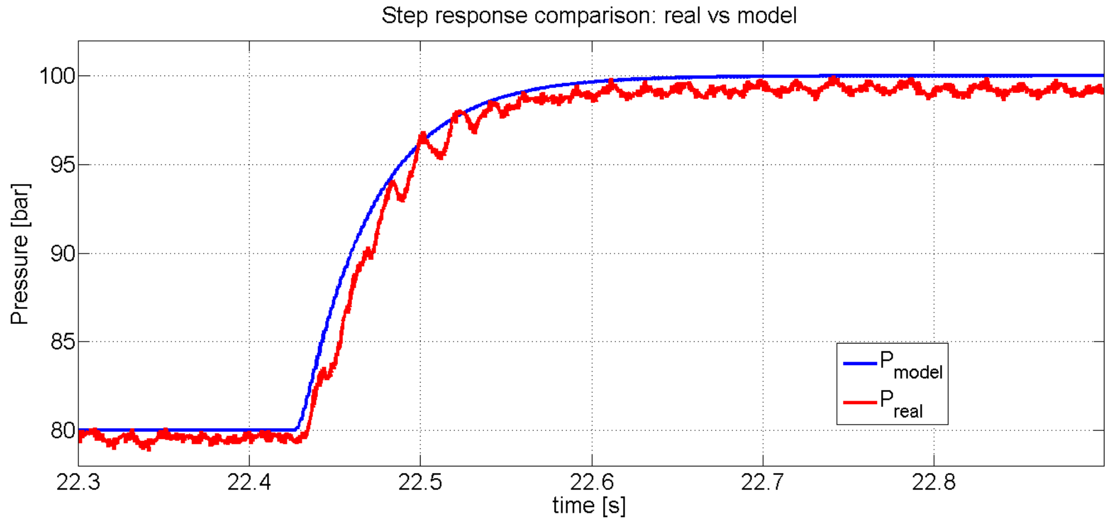

Figure 5.5: Comparison between the real plant step response and the model. The displayed scale correspond to the model, the real plant response has been scaled in magnitude.

By using this method to tune the model, and therefore finding the quotient $\frac{E}{V}$, it's possible to do a better tuning of the linear control using the methods that were mentioned in the chapter control design. Also an accurate tuned model allows the implementation and/or the improvement of control strategies that need a realistic desired pressure.

5.1.3. Root Locus of the P-P and P-PI scheme

The closed loop system of the plant using a proportional plus integral controller (P-PI) in the inner loop its presented in equation (4.3), and it is displayed here.

$$P(s) = \frac{\frac{K_p K_n V_p E}{V J_T}(s+K_i)}{s^3+\frac{K_n}{J_T}s^2+\left(\frac{K_p K_n V_p E}{V J_T}+\frac{K_n K_i}{J_T}\right)s+\frac{K_p K_n V_p E K_i}{V J_T}} P_d(s)$$

If the outer loop is not closed the system is described by equation (5.1).

$$P(s) = \frac{\frac{K_n V_p E}{V J_T}(s+K_i)}{s^3+\frac{K_n}{J_T}s^2+\left(\frac{K_n K_i}{J_T}\right)s} P_d(s) \quad (5.1)$$

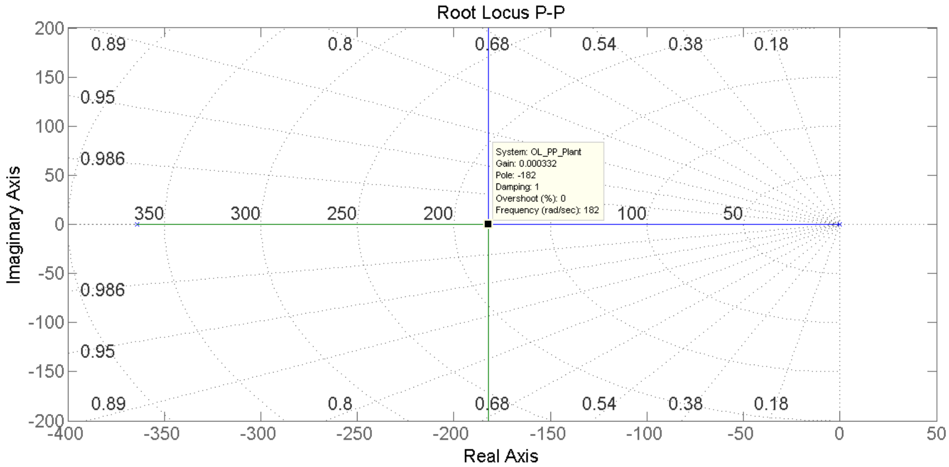

Figure 5.6: Root locus of the system using standard values of table 5.1, the inner loop features a fixed proportional control.

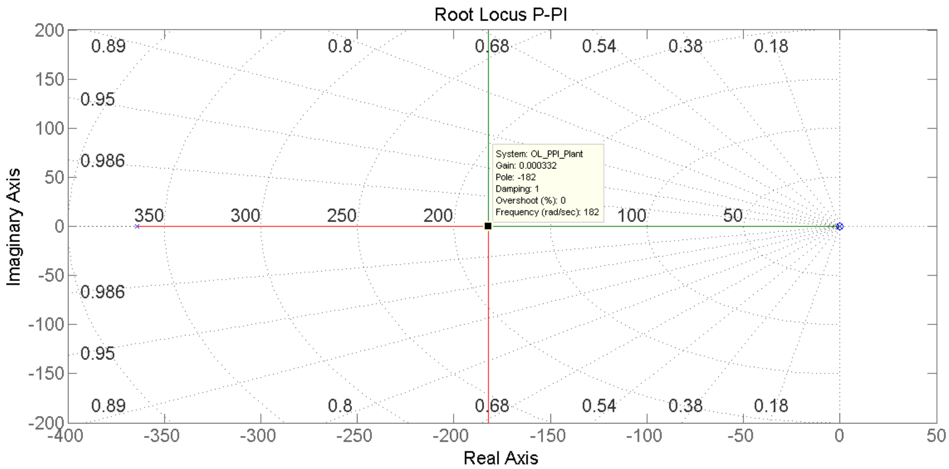

Figure 5.7: Root locus of the system using standard values of table 5.1, the inner loop features a proportional plus integral control, the integral gain K_i is equal to 0.0625.

And using a proportional controller in the inner loop the system is described by equation (5.1).

$$P(s) = \frac{\frac{K_n V_p E}{V J_T}}{s^2 + \frac{K_n}{J_T} s} P_d(s) \tag{5.2}$$

As it is known the open loop poles will move to the closed loop poles when the

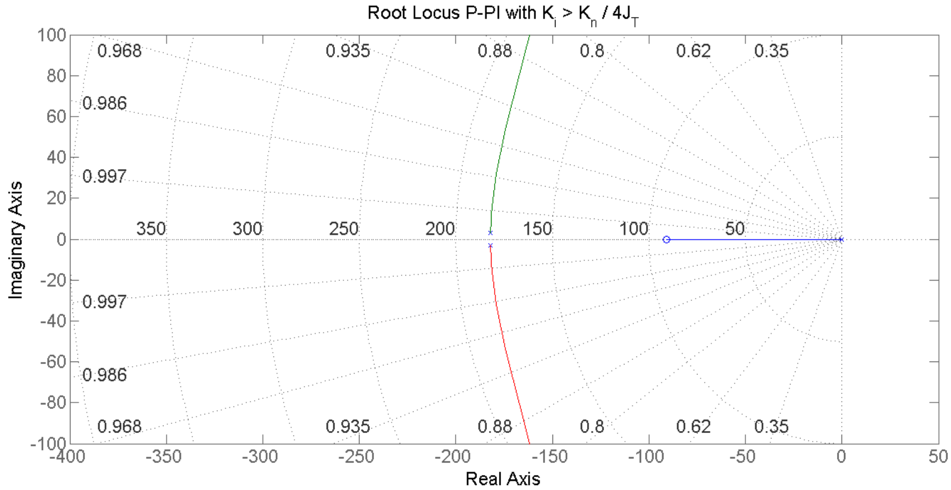

Figure 5.8: Root locus of the system using standard values of table 5.1, the inner loop features a proportional plus integral control, the integral gain K_i is equal to 100.

outer loop is closed as the outer loop gain K_p is increased.

The poles of system (5.2) are:

$$\begin{aligned} p_1 &= 0 \\ p_2 &= -\frac{K_n}{J_T} \end{aligned}$$

The poles of system (5.1) are:

$$\begin{aligned} p_1 &= 0 \\ p_2 &= -\frac{1}{2}\left(\frac{K_n}{J_T} - \sqrt{\frac{K_n}{J_T}\left(\frac{K_n}{J_T} - 4K_i\right)}\right) \\ p_3 &= -\frac{1}{2}\left(\frac{K_n}{J_T} + \sqrt{\frac{K_n}{J_T}\left(\frac{K_n}{J_T} - 4K_i\right)}\right) \end{aligned}$$

The zero of system (5.1) is:

$$z_1 = -K_i$$

As it is known from linear theory [35], [36], and [37], the poles of system (5.2) will move towards each other on the real axis, and then, when they meet, they will leave the real axis and will go to infinity in a direction parallel to the imaginary axis (see figure 5.6).

In system (5.1) the pole located at the origin p_1, will move towards the zero, this movement will happen always on the real axis. The behavior of the other two poles will be similar to system (5.2) if they start on the real axis, see figure 5.7, to start on the real axis the integral gain has to fulfill condition (5.3), this condition is obtained from the equations of the poles p_2 and p_3. However as can be seen in figure 5.8 if this condition is not fulfilled then it is not possible to have an oscillation free behavior.

$$K_i < \frac{K_n}{4 \cdot J_T} \tag{5.3}$$

In the test bench used in the experiment the value of the condition is $\frac{K_n}{4 \cdot J_T} = 90.9753$. This value is obtained using the standard proportional gain K_n given by the manufacturer. The value of the K_i is obtained using the formula:

$$K_i = \frac{t_s}{t_i} \tag{5.4}$$

where t_s is the sampling time of the motor firmware, and t_i is the integral time. In the test bench the parameters are $t_s = 250[\mu s]$, and the integral time given by the manufacturer is $t_i = 4[ms]$. Using this values the integral gain is $K_i = 0.0625$ which mets the condition (5.3). As will be seen later there is no oscillations or overshoot due to the integral action.

5.2. Results

The control design proposed in the last chapter was tested to check if it solves the control problem:

Control Problem keep the system pressure as the desired pressure despite parametric variations, uncertainties and disturbances. The desired pressure is in general a square signal.

The standard test is:

1. Fix the system volume using the switching valves 1, 2, 3.
2. Set the parameters of the system (pump displacement V_p, total inertia J_T, bulk modulus E, and system volume V) available in catalogue data and approximations.

3. Set the additional protection parameters to the system (Maximum speed of the pump ω_{max}, Maximum Torque Tq_{max}) available in the catalogue data.

4. Select the method to calculate the linear gains K_p, and K_n.

5. Set a desired pressure, as a square signal, to the controller.

6. Set a disturbance using the proportional valve.

7. Start the system operation.

In this test all the conditions of the control problem are met. Parametric variations will occur since the bulk modulus will change with different levels of pressure, for the controller the bulk modulus is given as a constant value. The condition of uncertainty is met because all the parameters are approximations due to the fact that even the catalogue data is an approximate of the real value (probably calculated using CAD-CAM tools). The disturbance are present because of the system leakage, and because of the proportional valve, which opens and close following a square signal.

5.2.1. P-P cascade

The first experiment is presented using the P-P cascade without the sliding mode compensator. For this experiment all switch valves are closed, therefore the system is at its minimal volume. The parameters of the controller are in Table 5.3.

Table 5.3: Values of the parameters

Name	Description	Units	Value
J_T	Total inertia	$[kg \cdot m^2]$	0.0069
E	Bulk Modulus	$[Pa]$	$9000 \cdot 10^5$
V	System Volume	$[m^3]$	$1.8 \cdot 10^{-3}$
V_p	Pump Displacement	$[m^3/rad]$	$4.0266 \cdot 10^{-6}$
ω_{max}	Maximum pump speed	$[rpm]$	3000
K_n	Proportional gain of the inner loop	$[Nm/rad \cdot s]$	2.5
K_i	Integral gain of the inner loop	$[1/s]$	0
K_p	Proportional gain of the outer loop	$[rad/s \cdot Pa]$	6
P_d	Desired pressure [lowValue - highValue]	$[bar]$	$[10 - 150]$

The gains K_n, and K_p of the controller were selected using the second method presented in the control design. For this method, the value of K_n was the value used in the standard controller. The damping was selected as 1.2 and with this values K_p was calculated as defined in the method. The desired pressure is a square signal. The desired pressure is also given as an input to the closed loop model (described in the previous section).

Figures 5.9 and 5.10 show the behavior of the system, P_{mod} is the output of the closed loop model, P_{act} is the actual pressure of the system, n_d is the demanded value of the speed, n_{act} is the actual value of the speed.

The response of the system is well damped, there is no presence of overshoot or other oscillations (without taking into account the oscillations due to the pump commonly known as pump pulsations), the only drawback, as it was expected, is that the system does not achieves the desired value and there is an steady state error.

Figure 5.11 shows the performance of the system when the proportional valve is open. It can be seen that due the additional disturbance the error is much more bigger than in the previous figures when the proportional valve is closed. In this case the error is almost of 50 %, which is an unacceptable error.

It important to say that the big delay between the rising of the desired speed (or desired pressure) and the corresponding rising on the actual speed (or actual pressure) is not as long as it seems in the figures. There are two problems with the figures: The first problem is that the signals come from different channels to the PC that is registering all the data, and therefore there are some communication delays. However it is important to remark that this delays does not affect in any sense the performance of the PLC, in the PLC the delay between the desired speed and the actual speed is of 2 $[ms]$. The second problem is that when the system is bellow $7[bar]$ it automatically switches itself to a constant speed mode to protect the hydraulic system from cavitation. Thus when the value of the system is below $7[bar]$, the closed loop is not operative and it has to wait for the system to pass the bound of $7[bar]$, this behavior is the principal responsible of the big delay between the ideal pressure P_{mod} and actual pressure P_{act} that figure 5.11 displays. Summarizing the delay in this figures is because the command signal is not been taken into account by the system at the beginning, because the system pressure is below $7[bar]$, it's until the system pressure

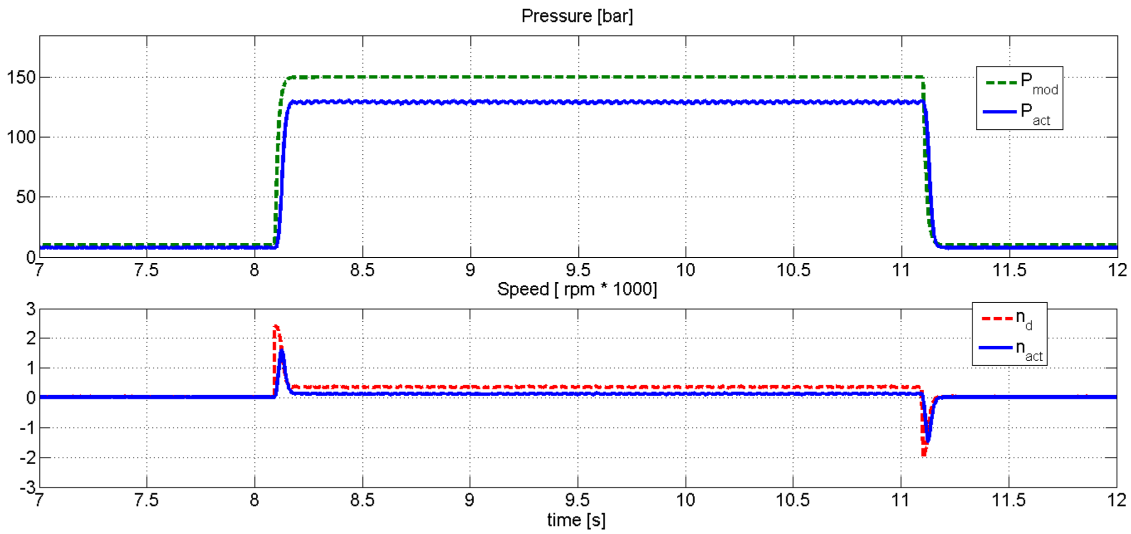

Figure 5.9: Response of the P-P cascade without sliding mode compensator, the load valve is closed.

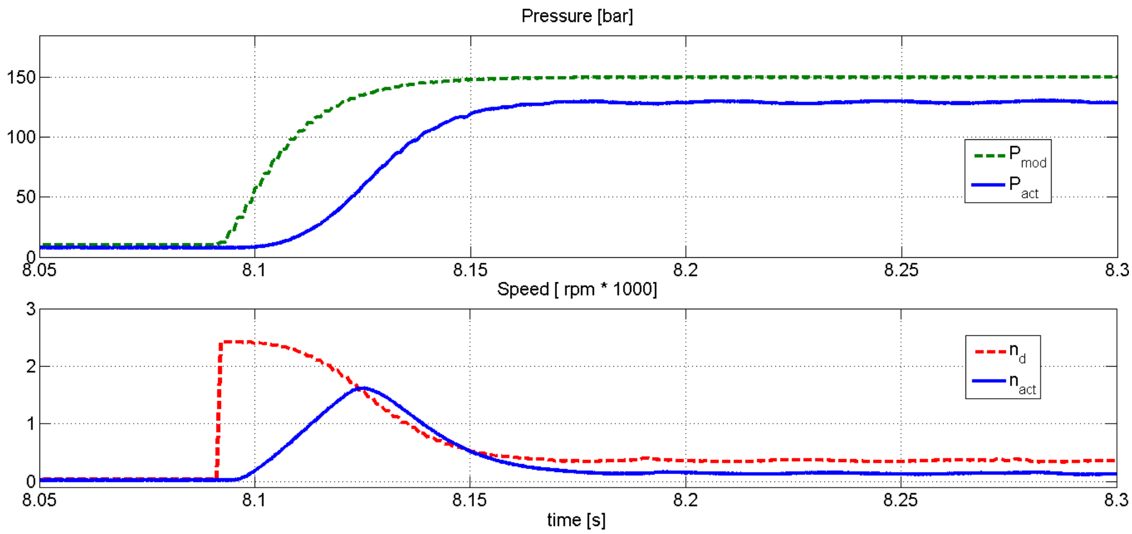

Figure 5.10: Zoom to the response of the P-P cascade without sliding mode compensator, the load valve is closed.

has passed the value of $7[bar]$ that the system starts to respond to the command signal, hence the delay displayed in the figures.

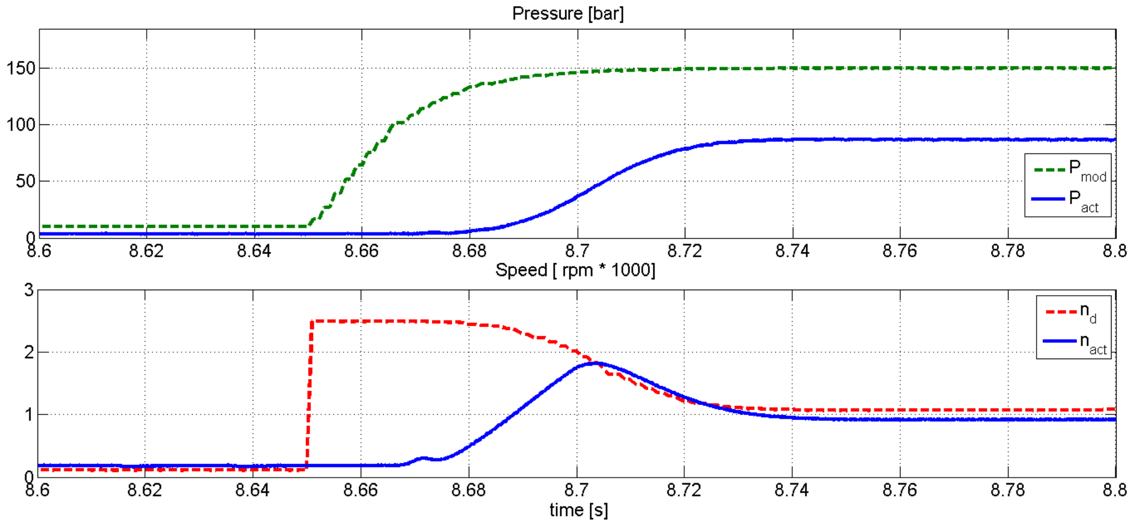

Figure 5.11: Response of the P-P cascade without sliding mode compensator, the load valve is open.

5.2.2. P-P cascade plus first order sliding mode

As it was showed before, the linear P-P scheme produce a nice response: well damped, no overshoot, no oscillations. The only problem is that there is an error in the steady state and this error will grow to unacceptable values when more flow is taken out of the system (something that will happen in any industrial application due to the load flow, simulated in this case by the opening of the proportional valve).

Fortunately it is possible to keep this nice response and eliminate the error by the use of sliding mode techniques without any additional tune.

In this case the first order sliding mode observer, described in the control design, is used to cancel the perturbation. The results can be seen in figures 5.12, 5.13, 5.14, 5.15.

In the figures the top image displays the actual pressure of the system P_{act} along with the ideal pressure of the closed loop model P_{mod}. Bottom image displays the response of the speed of the motor n_{act}, the speed demanded by the linear feedback n_d, and the demanded speed from the first order sliding mode compensator n_c.

As mentioned in the design, the parameters needed to calculate the sliding mode gain are the maximal pump speed, and the pump displacement, these parameters are available in the catalogue data and they are described in the Table 5.1.

The constant τ of the first order filter was selected experimentally as $\tau = 0.032$ $[s]$. Two first order filters were used in series to obtain the compensation signal n_{comp}.

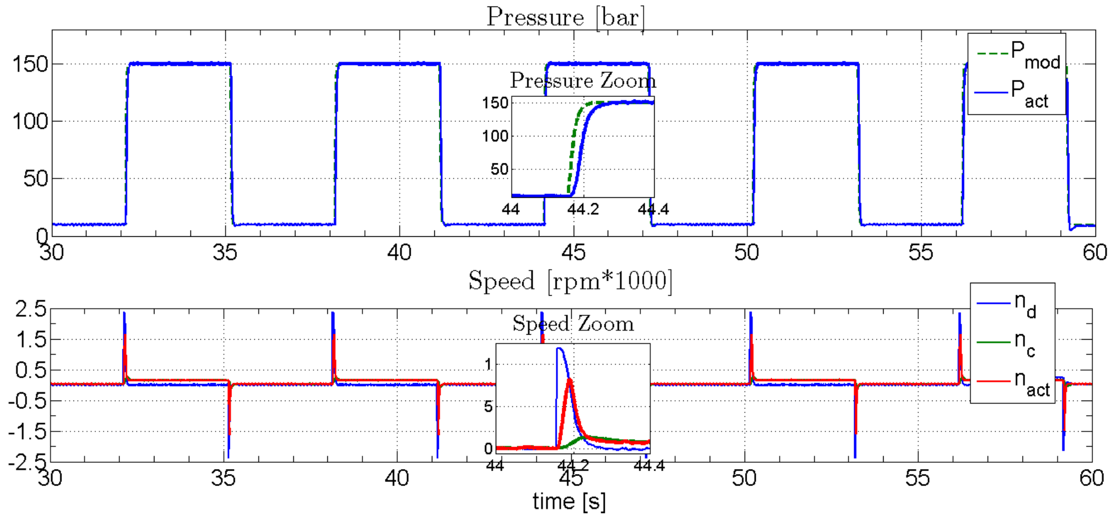

Figure 5.12: Response of the P-P cascade with the first order sliding mode compensator, the load valve is closed.

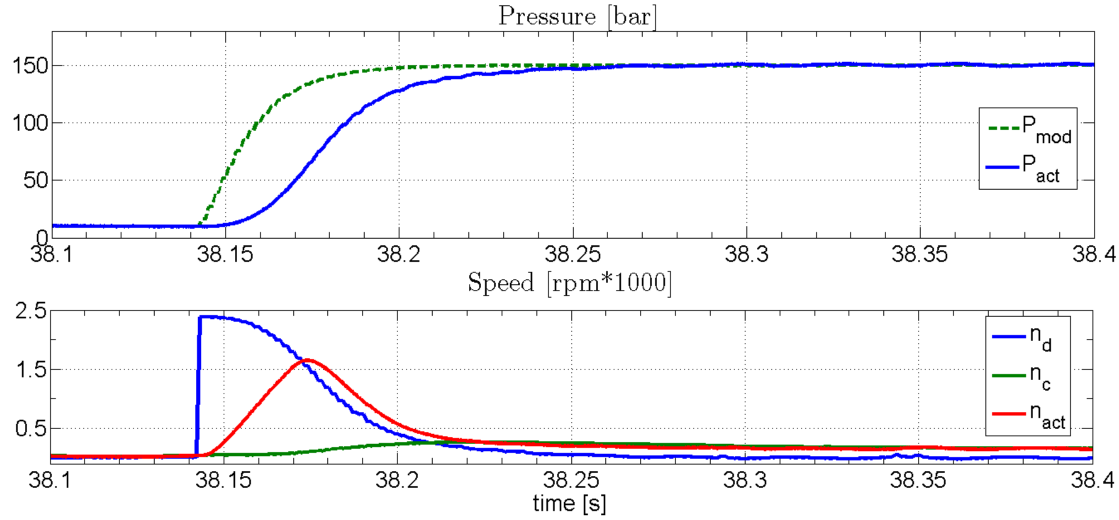

Figure 5.13: Zoom to the step response of the P-P cascade with the first order sliding mode compensator, the load valve is closed.

The results of the sliding mode show that, without tuning any additional gain, it is possible to keep the desired response designed with proportional gains despite disturbances and parameters uncertainties.

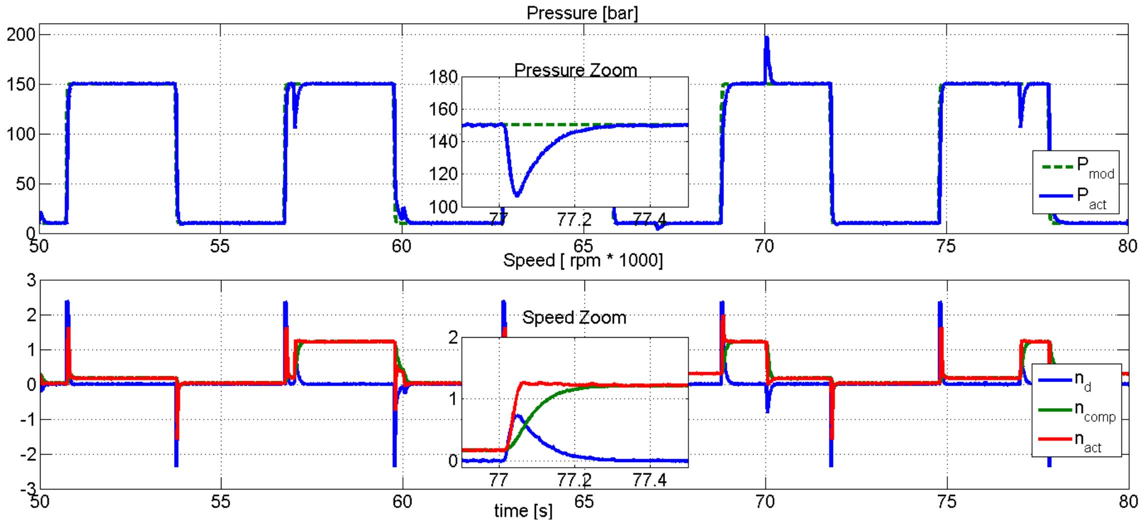

Figure 5.14: Response of the P-P cascade with the first order sliding mode compensator, the load valve opens and closes.

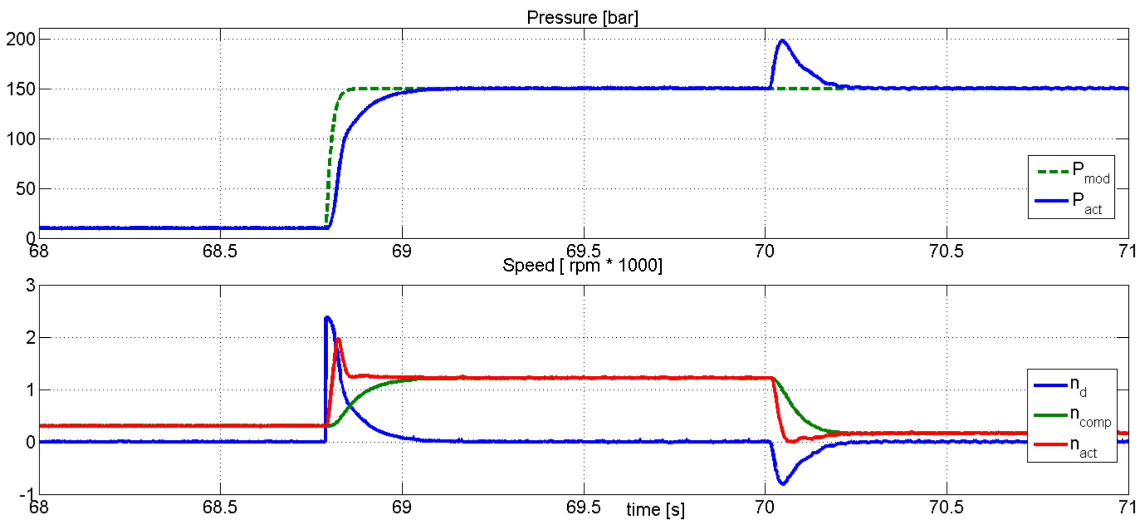

Figure 5.15: Zoom to the step response of the P-P cascade with the first order sliding mode compensator, the load valve is open before and after the step.

5.2.3. P-P cascade plus second order sliding mode

Using the second order sliding mode instead of the first order sliding mode gives the advantage of using a faster first order filter because the signal is already continuous. For this case the adaptive scheme for the observer gains, described in the control design, was also tested.

For this experiment, the system volume was the maximal volume possible, this means that the three switch valves of the test bench were open. The gain of the P-P cascade scheme were selected with the same method as described before, the only difference is that this time the damping was defined as 2 because a smaller value gave some additional oscillations. The parameters of the adaptive algorithm were selected experimentally.

Figure 5.16 displays the result of using the generalized super twisting observer, as described in the control design, it can be seen that the results are similar to using the first order observer. However in this case the compensation signal n_{comp} is obtained using just one first order filter instead of two as it's done with the first order sliding mode.

The filter constant is the same as it was with the first order sliding mode observer $\tau = 0.032$ $[s]$.

Figure 5.19 shows that the sliding surface is not zero at all time, even more it is lost when the disturbance acts or the step is realized; however the sliding surface is always taken back to zero (actually around zero due to chattering, noise, and delays) by the reaction of the adaptive gain. This figure also shows that the behavior of the observer has some oscillations due to the lost of the sliding surface (lightly undamped behavior), the first order sliding mode does not present this oscillations, however, in order to be fair it is necessary to consider that in this experiment using the GSTA the magnitude of the disturbance is not known in advance and there is an additional adaptive gain algorithm finding the right gain in real time.

5.2.4. System failure test

This experiment shows the robustness and reliability of the sliding mode solution. The experiment was conducted as the previous experiments with the test bench, the three switch valves were open thus the volume was maximal. The proportional gains were selected using the second method given in the control design and the desired damping was 2.

Additionally the plant is not working in normal conditions, there is a **failure** in the test bench, the pressure relief valve is failing and it opens randomly during operation, most of the times it opens when a step is performed but also at other times during operation. This pressure relief valve should protect the system from high pressures that could compromise

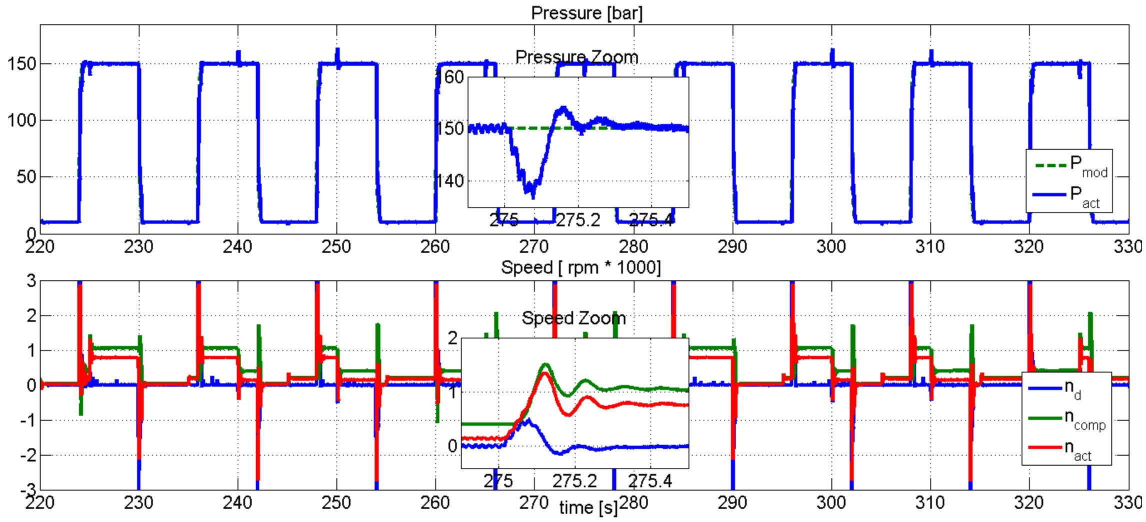

Figure 5.16: Response of the P-P cascade with the second order sliding mode compensator, the load valve opens and closes. The zoom window displays the disturbance (caused by the opening of the load valve) and the rejection by the compensation speed n_{comp}.

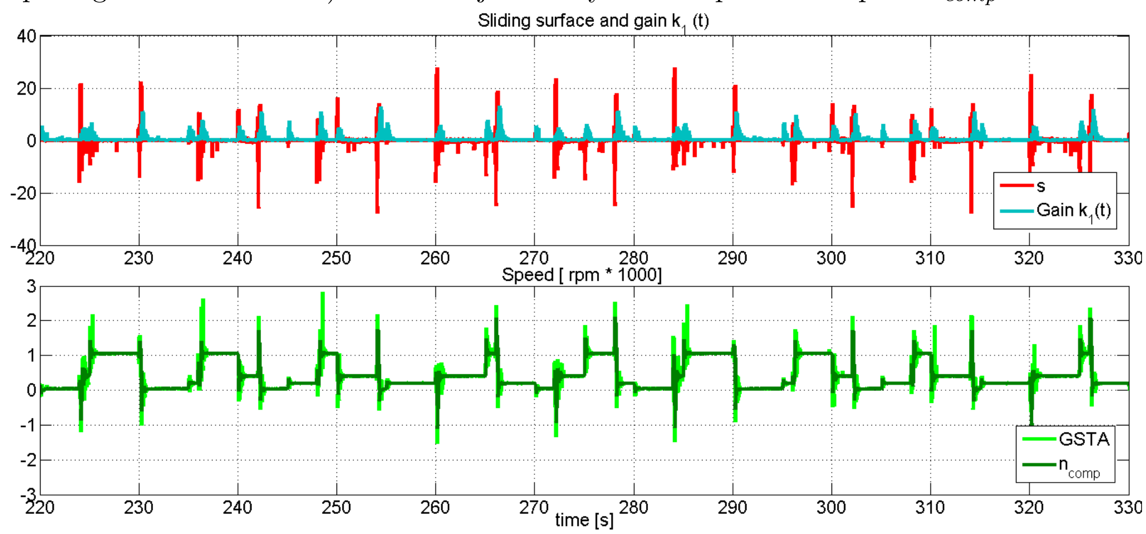

Figure 5.17: At the top are the gain $k_1(t)$ of the generalized super twisting algorithm, and the sliding surface s. At the bottom the GSTA output and its filtered value n_{comp}. The period of time for these graphics is the same as the previous image.

the system integrity (more than $350[bar]$), but in this case due to a failure it is opening randomly and the system is working in a normal range $10 - 150[bar]$.

The pressure relief valve is strategically placed at the output of the pump, thus when it opens its similar to causing a short circuit in an electrical system.

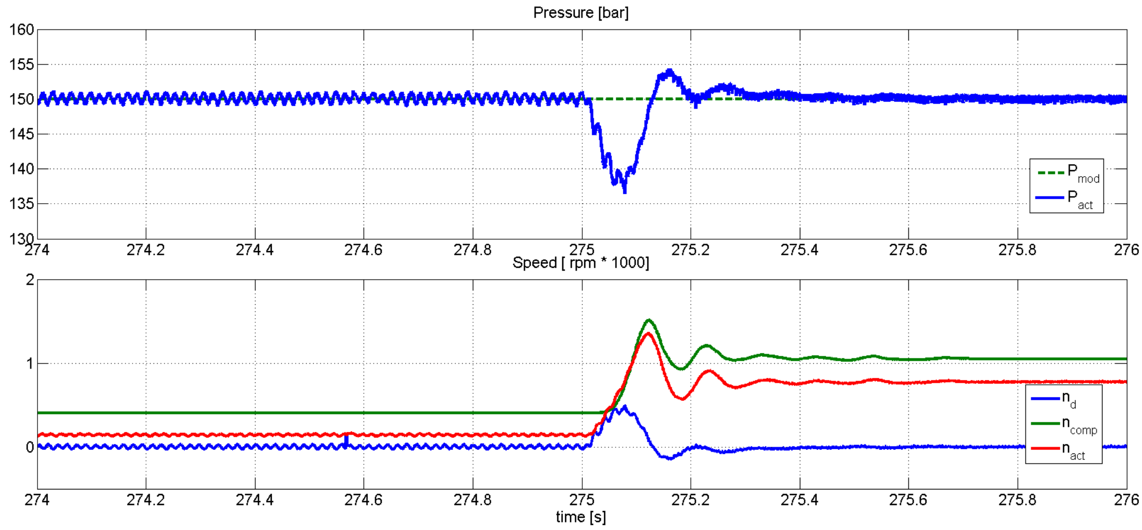

Figure 5.18: This figures is a zoom of the previous figure 5.16. At the top are the actual pressure P_{act} of the system, and the ideal pressure from the disturbance free model P_{mod}. At the bottom the actual speed n_d, the demanded speed from the linear shceme n_d, and the compensation speed from the sliding mode observer n_{comp}.

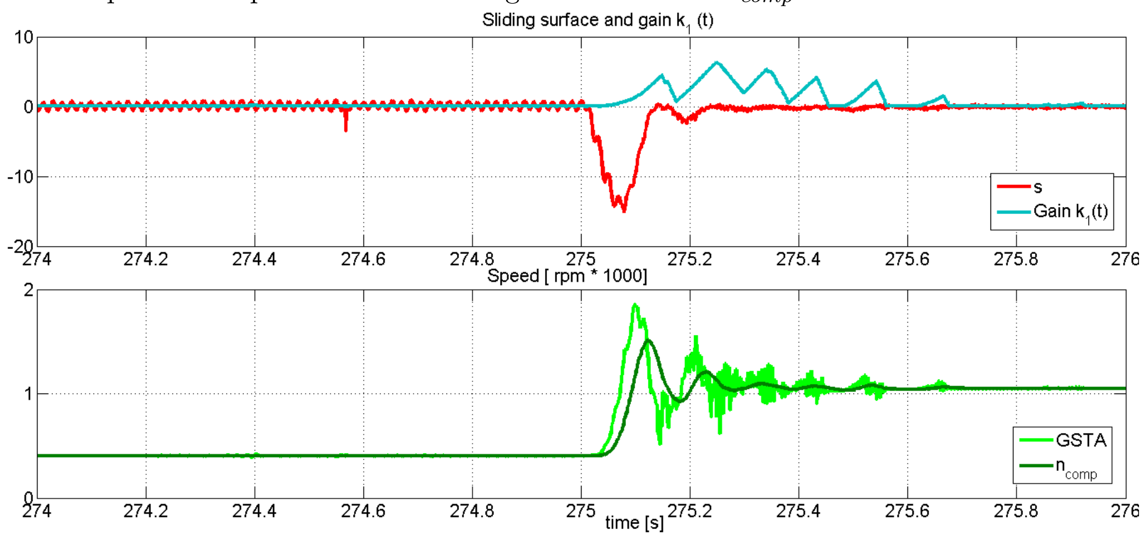

Figure 5.19: This figures is a zoom of the previous image. At the top are the gain $k_1(t)$ of the generalized super twisting algorithm, and the sliding surface s. At the bottom the GSTA output and its filtered value n_{comp}.

Additionally during this test the proportional valve is opening and closing in its own cycle.

At the beginning of this test a first order sliding mode observer is working and

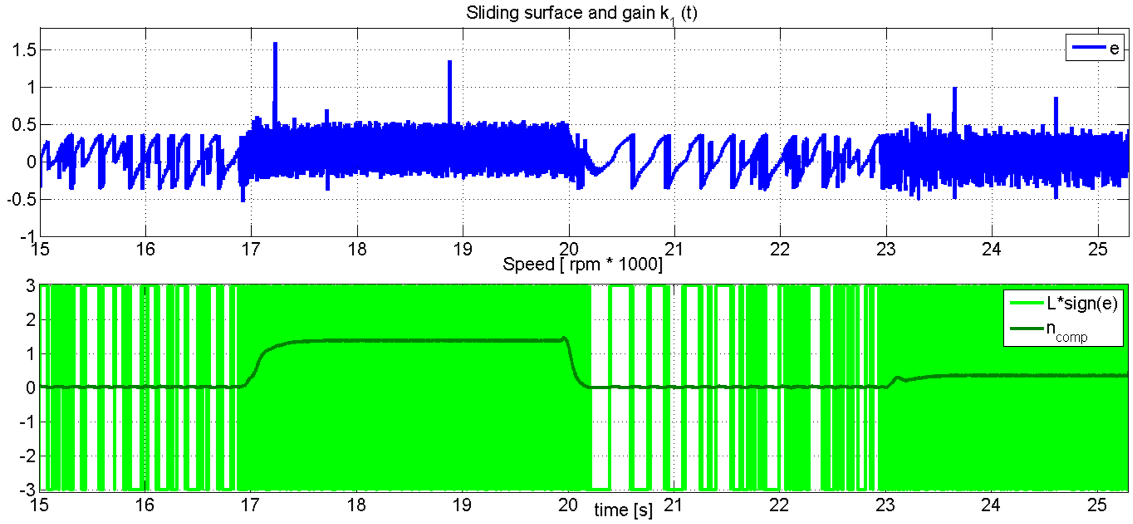

Figure 5.20: At the top the sliding surface e, at the bottom the discontinuous signal from the first order sliding mode observer $L \cdot sign(e)$, and the compensation signal n_{comp}.

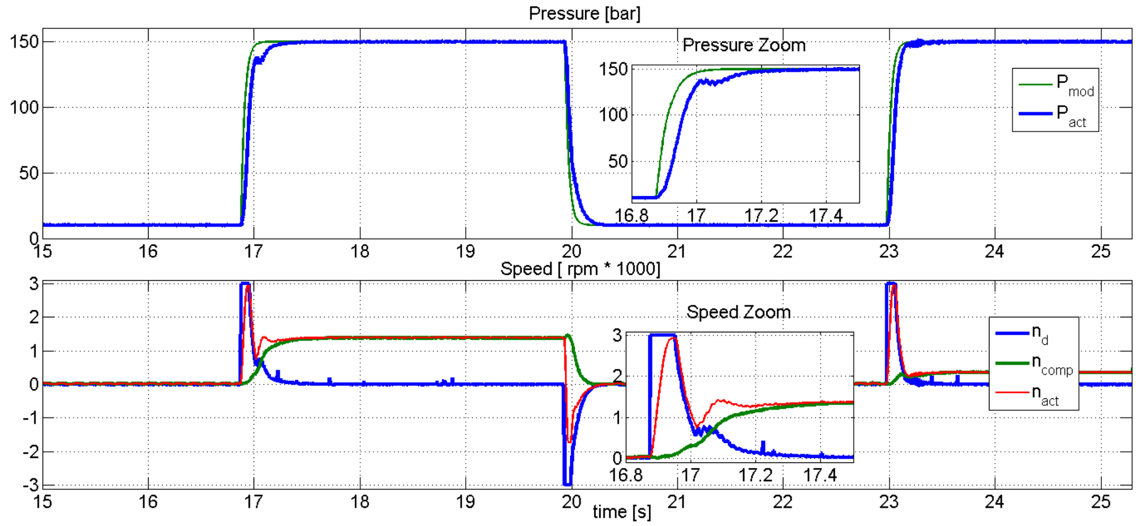

Figure 5.21: At the top the real pressure of the system, and the ideal pressure extracted from a closed loop model of the system. At the bottom the actual speed n_{act}, the demanded speed n_d from the linear feedback, and the compensation speed n_{comp} from the first order sliding mode observer. The zoom presents the moment when the pressure relief valve opens (at approximately $t = 17$).

then after the time $t = 25.29[s]$ the algorithm is changed to the second order sliding mode observer. This change is made when the machine is operating (and there is no big oscillations or instabilities, see figures 5.22, 5.23).

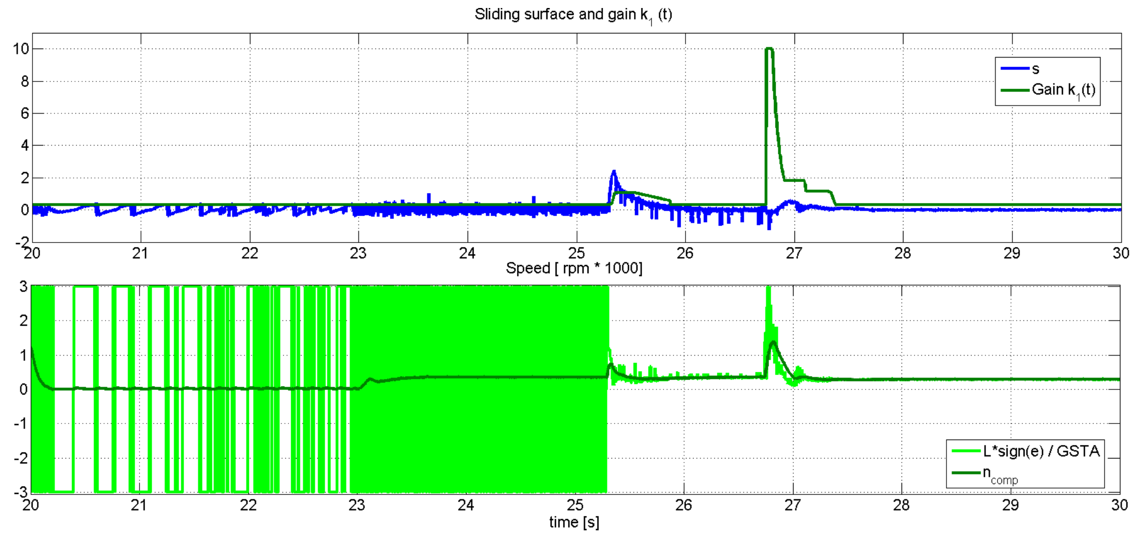

Figure 5.22: Here the switch is being made between first order sliding mode and second order sliding mode while the machine is operating (between second 25 and 26). At the top the sliding surface e, at the bottom the discontinuous signal from the first order sliding mode observer $L \cdot sign(e)$, and the compensation signal n_{comp}.

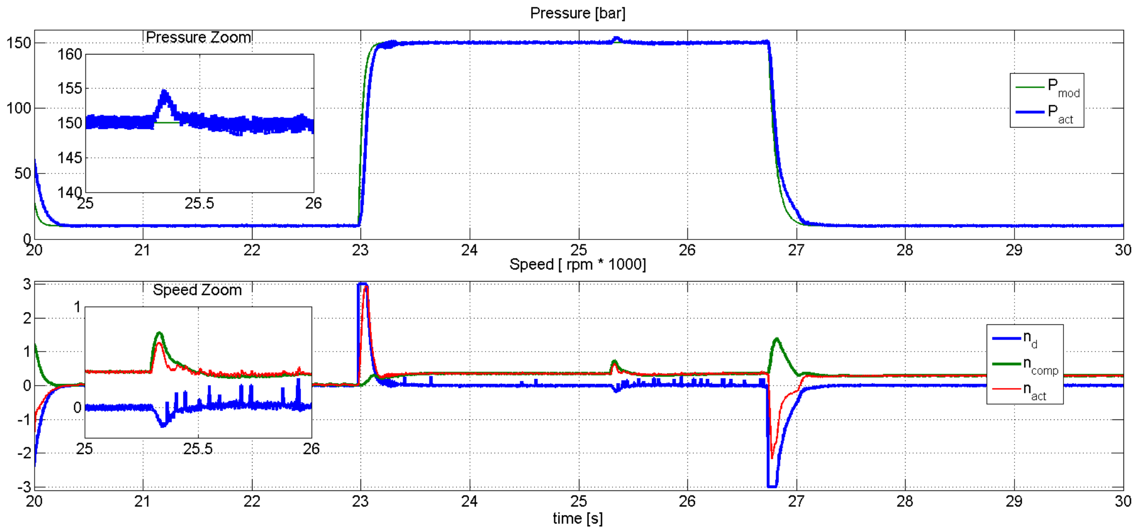

Figure 5.23: Here the switch is being made between first order sliding mode and second order sliding mode while the machine is operating (between second 25 and 26). At the top the real pressure of the system, and the ideal pressure extracted from a closed loop model of the system. At the bottom the actual speed n_{act}, the demanded speed n_d from the linear feedback, and the compensation speed n_{comp} from the first order sliding mode observer. The zoom presents the moment when the switch is being made.

The first and second order sliding mode algorithm are “off” when the actual speed of the system or the demanded speed from the linear feedback gets closer to the maximal speed (in this case this value is $3000[rpm]$), how close can be set in the range from $90-100[\%]$. Because of this preventive rule, the compensation signal from the sliding mode observer is fixed as zero, because adding more speed to the maximal speed has no sense and it could be dangerous. When the actual speed or the demanded speed leaves the set where the sliding mode algorithm is “off” there is a small waiting time before the algorithm goes “on” again. This waiting time is longer in the second order sliding mode observer than in the first order sliding mode observer. (It is important to notice that all this transitions between “on” and “off” happen without any oscillations because of the robustness and flexibility of the sliding mode algorithms.)

Figure 5.20, and 5.21 display the performance of P-P plus first order sliding mode observer. In the zooms of pressure and speed, it can be seen the moment when a rising step is performed, the demanded and actual speed hit the $3000[rpm]$ and therefore n_{comp} remains at zero; also at this moment the pressure relief valve opens (at approximately $t = 17$). It can be seen that, due to this failure, the demanded speed n_d from the linear feedback does not reach its lower value (as it normally happens) and starts to oscillate; nevertheless, the n_{comp} from the sliding mode observer does not oscillate and compensates the failure.

Figure 5.24, and 5.25 shows the performance of the P-P plus second order sliding mode observer. In figure 5.24 the failure of the pressure relief valve can be easily identified, because when it happens the sliding surface escapes far from zero (in some cases it goes to the value 10) and later it takes a triangular shape when it comes back, also the behavior of the adaptive gain is displayed. It always reacts and takes the sliding surface back to zero. The only rising step when the pressure relief valve does not seem to open completely, is the one between $t = 75[s]$ and $t = 80[s]$.

It is also remarkable how the GSTA and n_{comp} have a great performance, the compensation signal behaves always well damped, without oscillations even when the pressure relief valve is failing. The compensation signal is actually smother than the signal from the linear feedback.

Figure 5.26, and 5.27 presents a step where the pressure relief valve does not fail

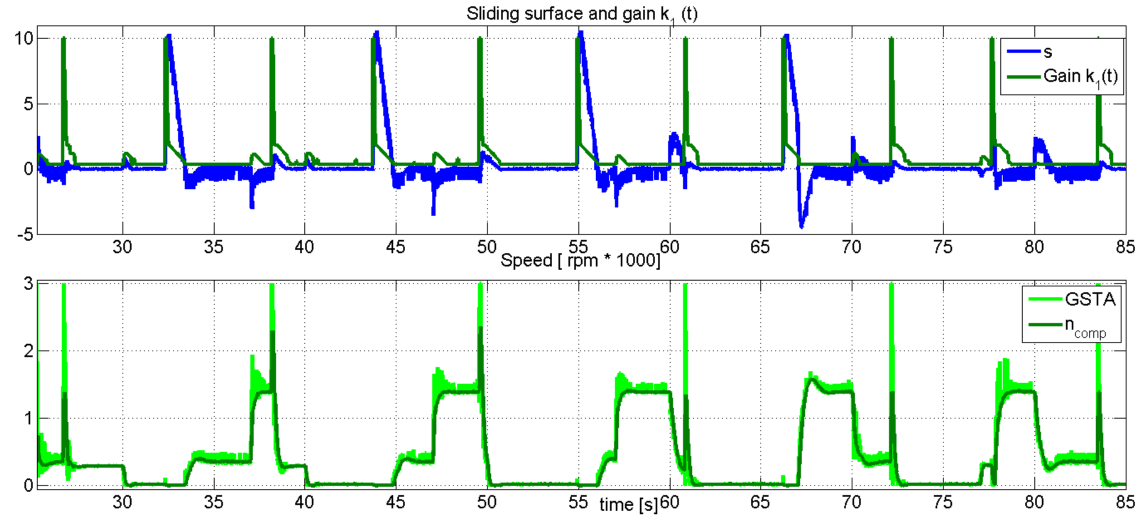

Figure 5.24: At the top the sliding surface s, at the bottom the output of the generalized super twisting algorithm from the second order sliding mode observer, and the compensation signal n_{comp}.

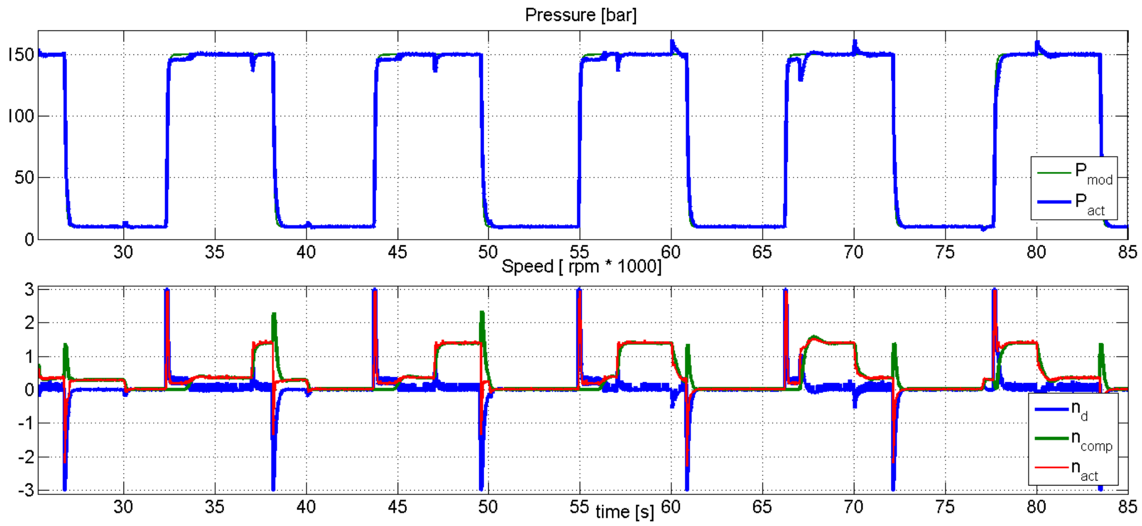

Figure 5.25: At the top the real pressure of the system P_{act}, and the ideal pressure extracted from a closed loop model of the system P_{mod}. At the bottom the actual speed n_{act}, the demanded speed n_d from the linear feedback, and the compensation speed n_{comp} from the second order sliding mode observer.

completely. A chronological description of the figure is presented:

1. Before the step the proportional valve opens in its normal work cycle.

2. The compensation signal from the observer n_{comp} reacts and compensates the distur-

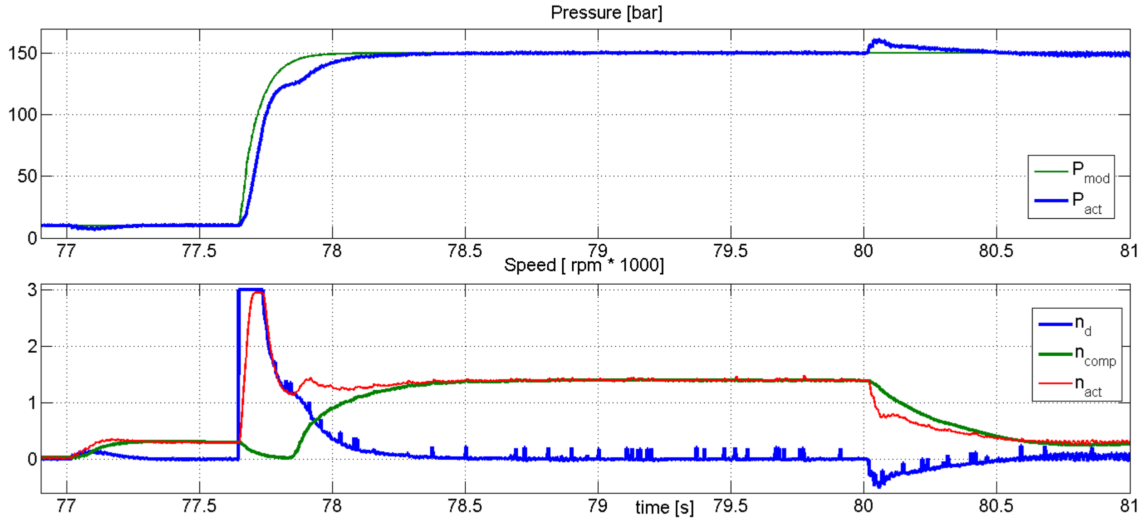

Figure 5.26: The rising step. At the top the real pressure of the system P_{act}, and the ideal pressure extracted from a closed loop model of the system P_{mod}. At the bottom the actual speed n_{act}, the demanded speed n_d from the linear feedback, and the compensation speed n_{comp} from the second order sliding mode observer.

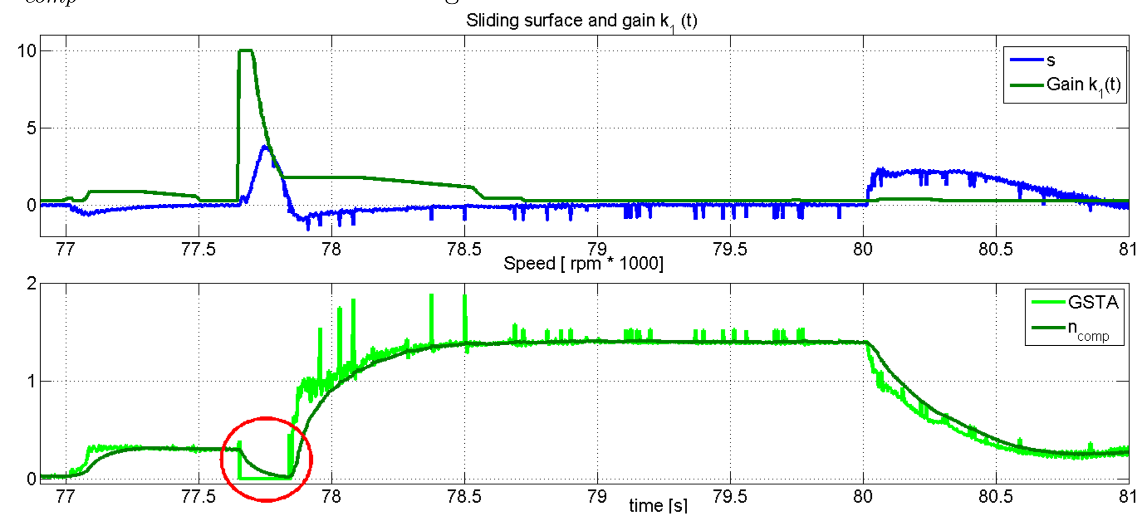

Figure 5.27: The rising step (same as the previous plot). At the top the sliding surface e (displayed as s), and the gain $k_1(t)$. At the bottom the output v of the GSTA, and its filtered value n_{comp}; notice the red circle, it indicates the moment when the GSTA goes off due to safety reasons (the speed is hitting the maximal value of 3000 $[rpm]$), after the speed returns to normal operation values, the algorithm starts working again without any overshoot or aggressive oscillations.

bance.

3. The step rises, the demanded speed n_d, and the actual speed n_{act} goes to 3000[rpm]. Because of this the GSTA signal goes to zero, and n_{comp} follows it.

4. After the demanded speed n_d, and the actual speed n_{act} leave a neighborhood of 3000[rpm], the compensation speed starts to compensate again the disturbance caused by the proportional valve.

5. The proportional valve closes.

It is remarkable that the compensation signal n_{comp} is smoother than the demanded speed n_d from the linear feedback.

5.3. Comparison with integral control PI-P

In this section a comparison between a PI-P controller and the proposed scheme P-P plus sliding mode observer is presented. The first order sliding mode is used.

The PI-P controller is a cascade controller similar to the P-P described in the control design. The difference is that the outer loop presents an integral term in order to eliminate the steady state error, however this integral term is slightly modified from the traditional integral term.

To the describe the PI-P scheme is better to separate the inner loop from the outer loop and to separate the outer loop in its two terms proportional and integral.

Inner loop The inner loop is exactly the same as the inner loop of the P-P cascade described in the control design.

Outer loop The outer loop receives as its input the desired pressure P_d specified by the user, usually this is a constant set value, therefore, it is given as a step.

Proportional Term Obtains the pressure error e_p between the desired pressure P_d and the system pressure P ($e_p = P_d - P$). The output of the proportional term is the result of the multiplication of the pressure error e_p, and the proportional gain K_p

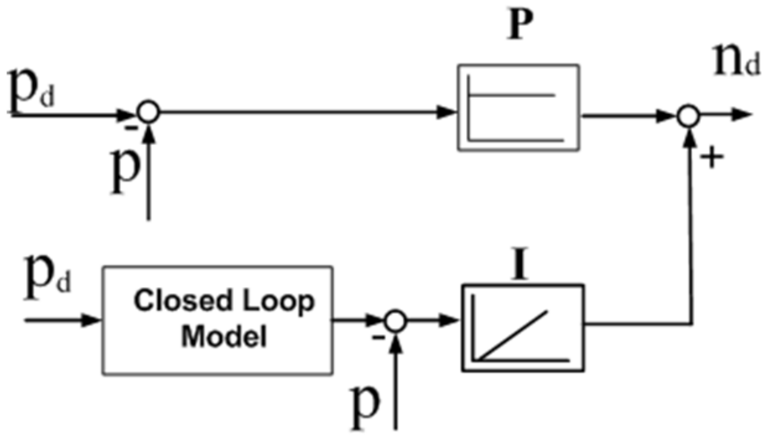

Figure 5.28: Scheme of the modified PI-P.

Integral Term The desired pressure is given as an input to the closed loop model of the plant (4.4), then the output of this model is label as the filtered desired pressure $P_{d_{fil}}$, and it is used to calculate the filtered pressure error $e_{p_{fil}} = P_{d_{fil}} - P$. The output of the integral term is the result of the multiplication of the filtered pressure error $e_{p_{fil}}$, the proportional gain K_p, and the integral gain K_i.

The following equation (5.5) describes the outer loop of the PI-P.

$$u = K_p \left(e_p + K_i \cdot e_{p_{fil}}\right) \tag{5.5}$$

It is impossible for the plant to respond as fast as the step signal P_d, therefore, there will always be an error at the beginning of the step response, thus if the integral term tries to correct this error it is likely to cause an overshoot. In order to get a more adequate desired pressure this scheme of the PI-P takes advantage of a good knowledge of the system to tune a model of the plant (The model is the one described in (4.4)), and use that model to generate a "realistic" desired pressure or filtered desired pressure $P_{d_{fil}}$ that the plant can achieve. The idea is not to integrate the error of the system pressure with the step signal P_d, but with the filtered value $P_{d_{fil}}$ in order to eliminate the overshoot caused when the step signal is used in the integral term. Of course a bad tuning of the integral or proportional gain can still cause overshoot, oscillations and instability of the plant. Figure 5.28 shows a block diagram of the PI-P outer loop.

5.3.1. Test results

For this experiment, the proportional gains of the PI-P and of the P-P plus first order sliding mode observer were the same, they were selected using using the second method described in the chapter control design. The proportional gain of the inner loop was $K_n = 2.5$ and the damping was $D = 2$, The three switch valves of the system are open so the volume of the system is maximal. The integral gain was tuned to avoid overshoot or additional oscillations. The gain of the first order sliding mode observer does not need to be tuned, it is the same as described in the control design.

The results of the test show that despite of the efforts to avoid the overshoot in the PI-P scheme an overshoot is present, on the other side the linear scheme plus sliding mode compensator is faster, does not present any overshoot, and is faster than the PI-P in the same circumstances (see figure 5.29). When it comes to reject the disturbance the linear scheme plus sliding mode observer is four times faster than the PI-P scheme(see figures 5.30, and 5.31).

5.4. Comparison with linear observer

In this section a comparison between a P-P plus linear observer scheme and the proposed scheme P-P plus sliding mode observer is presented. The first order sliding mode is used.

The P-P plus linear observer features the same cascade P-P described in the chapter control design. Thus both algorithms, the P-P plus linear observer, and the P-P plus sliding mode observer are the same except in the observer.

To describe the linear observer, consider the first order sliding mode observer (4.10), the equation is reproduced here:

$$\dot{\hat{P}} = \tfrac{E}{V}\left(V_p * \omega - L \cdot sign(e)\right)$$

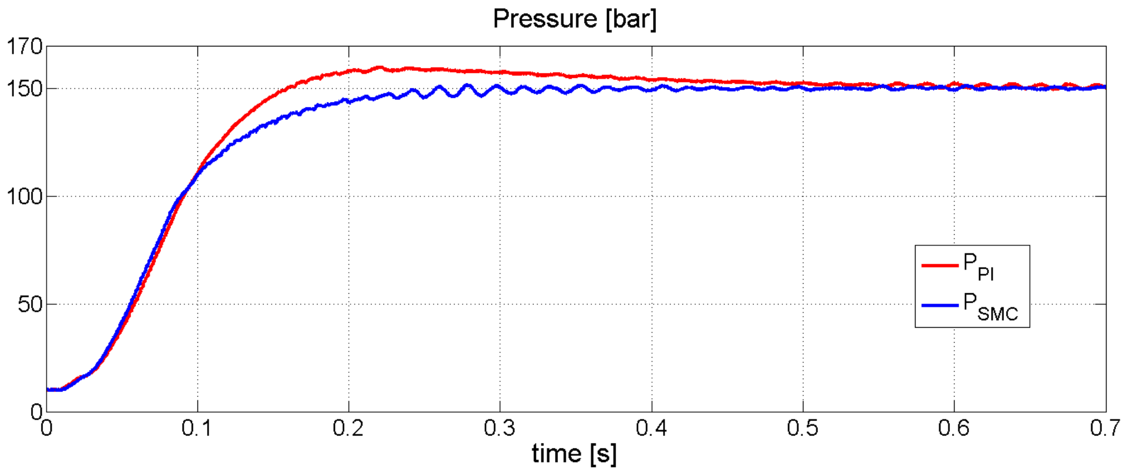

Figure 5.29: Comparison between the step response of the PI-P and the P-P plus sliding mode observer. The PI-P features an overshoot caused by the saturation of the motor's torque.

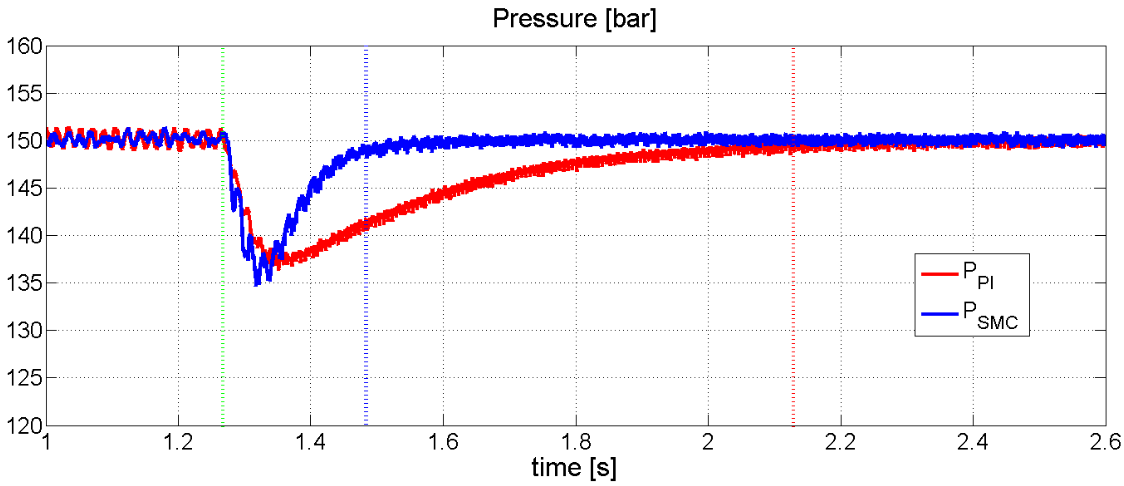

Figure 5.30: Response of the PI-P when the proportional valve opens, the P-P plus sliding mode observer is approximately four times faster.

Where $e = \hat{P} - P$. To make a linear version of this algorithm it is enough to change the function signum with its argument e as displayed next.

$$\dot{\hat{P}} \quad = \quad \frac{E}{V}\left(V_p * \omega - L \cdot e\right) \tag{5.6}$$

Then, assuming that the algorithm is stable, it is possible to follow the steps used

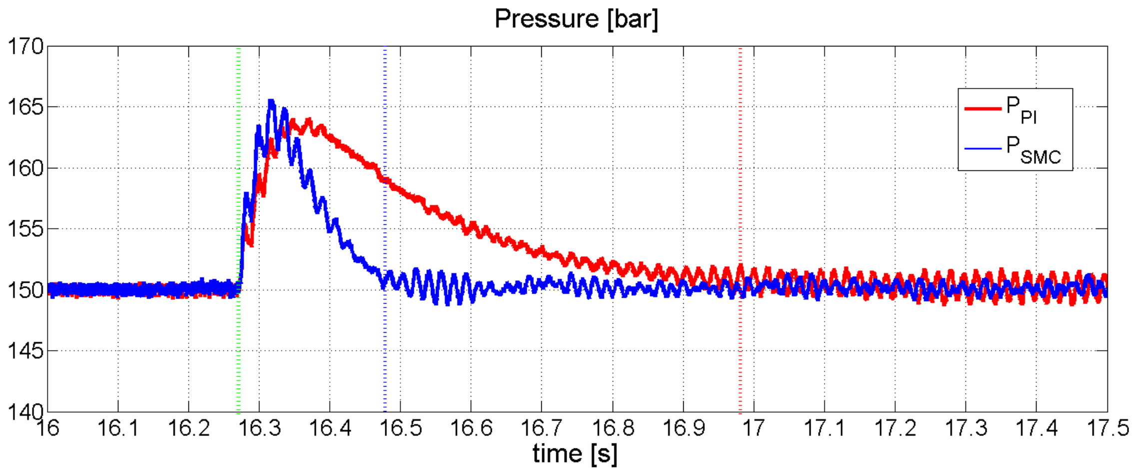

Figure 5.31: Response of the PI-P when the proportional valve closes, the P-P plus sliding mode observer is approximately four times faster.

for the sliding mode observer to get to the conclusion that:

$$L \cdot e \quad = \quad Q_L$$

And then after scaling it is possible to get the compensation speed n_{comp} from the linear observer. In this algorithm there is no need to use low pass filters.

5.4.1. Test results of the linear observer

For this experiment the proportional gains of the P-P plus linear observer were selected using using the second method described in the control design. The proportional gain of the inner loop was $K_n = 2.5$ and the damping was $D = 2$, The three switch valves of the system are open so the volume of the system is maximal.

The desired pressure P_d is a pulse signal in order to check the step response of the system. The lower value is 80 $[bar]$ and the higher value is 100 $[bar]$.

As it can be seen in figures 5.4.1, and 5.32 the P-P plus linear observer does not have a good performance and it is almost unstable. Since the stability and good performance of the P-P plus sliding mode observer have already been tested there is no need to do a comparison.

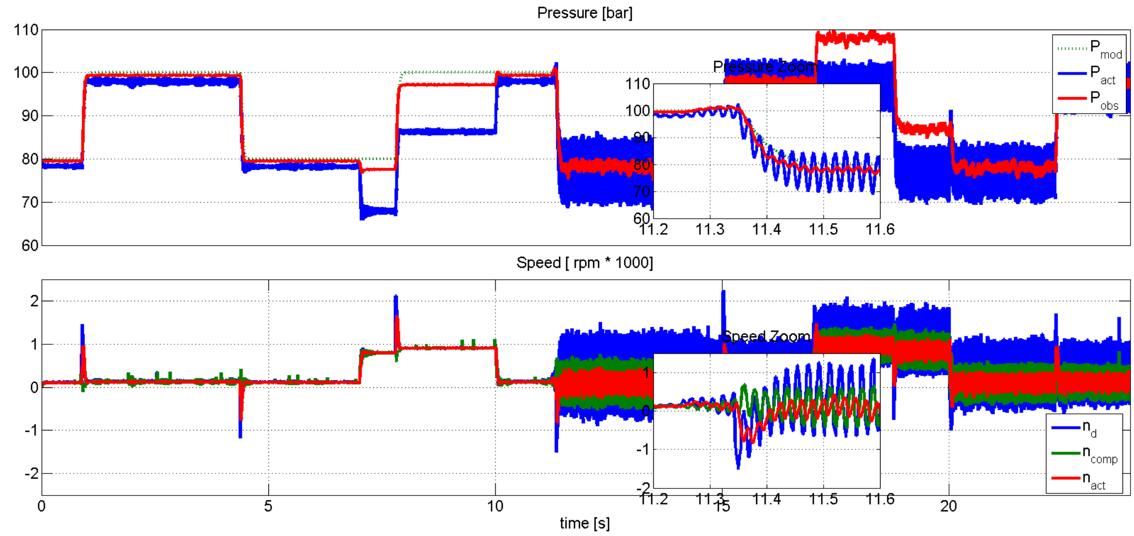

Figure 5.32: Before 11.2 [s] the linear observer is working but the compensation signal n_{comp} (from the linear observer) is not feeded to the system. Adter 11.2 [s] the compensation signal n_{comp} is given to the system, the system become marginally stable (In practical applications this behavior is considered unstable).

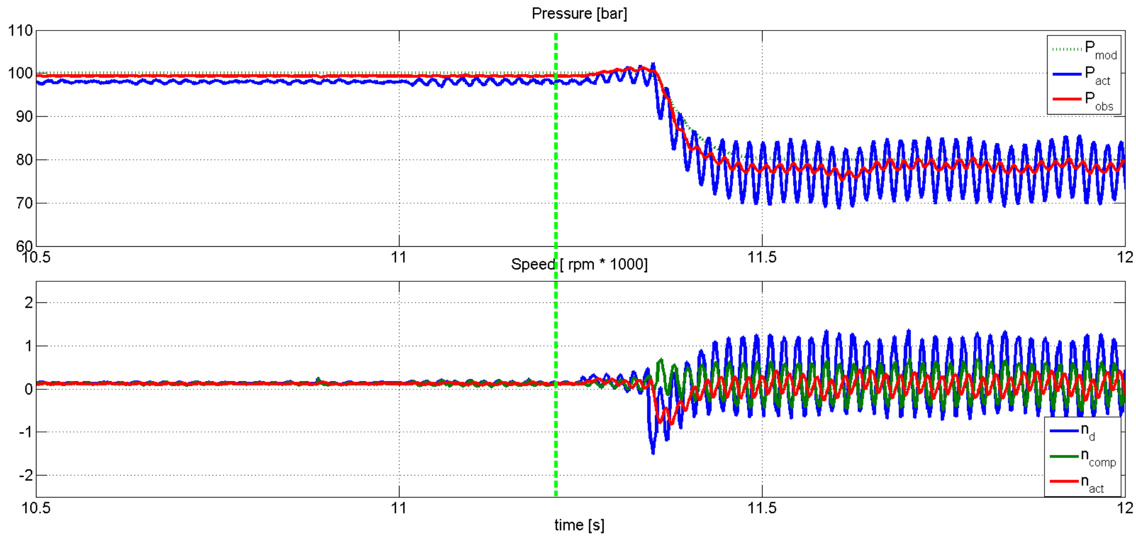

Figure 5.33: Zoom of the moment when the system becomes unstable, the green bar marks the moment when the compensation signal n_{comp} is given to the system.

Chapter 6

Conclusions

This work is in the context of energy efficiency in the industrial production of goods. Ideally, everywhere where the change from the old scheme to a variable speed pump is possible, there is an opportunity to save energy. However in order to go global and massive the solution has to offer not just a good performance, but it has to be easy to use. There will be no sense in sending every user or potential user in the world to a special training in control theory, and there will be no sense either in sending special trained technicians to help users all over the world. Therefore the solution has to solve the control problem assuring a good performance, and it needs to solve the control problem without giving high complexity to the final user.

The proposed P-P(I) plus first order sliding mode observer achieves this challenge. It was show experimentally that it solves the control problem with a good performance (fast, well damped, without oscillations), and it also reduces the complexity of tuning 11 parameters (in the standard controller) to tune just 1 parameter. This is in fact an outstanding reduction of complexity since the user has to tune just one proportional gain (with the second method given in the control design) to operate the machine.

It was tested that the P-P plus first order sliding mode observer performs better, and faster than an improved PI-P. The first order sliding mode observer is also a simpler algorithm than the improved PI-P and needs less information of the system, therefore it is less susceptible to fail than the improved PI-P.

Also the reliability, robustness, and flexibility was experimentally demonstrated with a failure test. Here flexibility means that the algorithm can be turned *off* and *on* without causing undesirable effects like overshoots or oscillations.

It was also experimentally showed that trying to match the capabilities of the sliding mode observer with a linear observer is not possible.

Additionally in this scheme it is possible to send the machine to the industrial plant pretuned, assuring stability by tuning it with an overdamped P-P scheme using an estimation of the parameters. This advantage is possible because the first order sliding mode observer does not need to be tuned.

The proposed P-P plus second order sliding mode observer with adaptive gain gave good results and it should be developed further. But unfortunately for now, due to complexity of the algorithm, it is not ready for industrial applications.

Apendix A

Code

The code for the first order sliding mode observer, the generalized super twisting sliding mode observer and the adaptive gain is presented here. There are of course additional algorithms that coordinate the work, input, and output of these algorithms (namely the master program), and auxiliary programs that preprocess the data that these algorithms need. Those programs were created also in the development of this work but they are not displayed here because they are related to the implementation process.

The code is written in structured text (ST) a standard PLC programming language.

```
FUNCTION_BLOCK fbFlow1stSMCompensator
(*#################################################################################################*)
(*#  Name: fbFlow1stSMCompensator                                         Date: 11.03.2010/TGR #*)
(*#################################################################################################*)
(*#  DESCRIPTION:                                                                               #*)
(*#   - Implements First Order Sliding Mode Flow Compensator                                   #*)
(*#                                                                                             #*)
(*#################################################################################################*)
(*#  Change Log:                                                                                #*)
(*#    11.03.2011, TGR                                                                          #*)
(*#    - Program Created                  #*)
(*#                                                                                             #*)
(*#################################################################################################*)


VAR_INPUT

bEnable: BOOL:= FALSE;
bReset: BOOL := FALSE; (*reset the Sliding Mode*)
rActualSpeed: REAL:=0; (* Motor Speed [rpm] *)
rPs: REAL:=0;  (* Actual Pressure *)
rLGain:  REAL:=0; (* Maximal Gain possible equal to Maximal Pump Flow [l/min] *)
rPumpVh: REAL := 20.1E-6;          (* Pump Volume Displacement[m/rev] *)
rMaxSpeed: REAL; (*Maximal Pump speed [rpm]*)
```

```
rTs: REAL:=0.001; (*Sample time [sec]*)
rHydC: REAL := 0; (*Hydraulic Capacity [bar/m]/60*)

END_VAR

VAR_OUTPUT

rNComp: REAL; (*Additional Speed [rpm]*)
rDiscN: REAL := 0; (*Discontinuos Output [rpm]*)
rSlidingS: REAL; (*Sliding Surface [bar]*)

END_VAR

VAR

rZ: REAL; (*Virtual Pressure [bar]*)
rSignum: REAL; (*Signum of the sliding Surface*)
rQtotal: REAL; (*Total Flow [l/min]*)
rZp: REAL; (*Virtual Pressure Dot [ bar/s ]*)
rWarning: REAL; (* Speed Warning 0..1 *)

END_VAR

VAR CONSTANT

rWarnBound: REAL := 1.10; (* Maximal control Action default 1.10 *)

END_VAR

(* @END_DECLARATION := '0' *)

IF bEnable THEN

rWarning := rActualSpeed/ rMaxSpeed;

IF rWarning > rWarnBound OR bReset THEN

bReset := FALSE;
rZ := rPs;
rSlidingS := 0;
rSignum := 0;

ELSE

rZ := rZp*rTs + rz;
rSlidingS := rZ - rPs;

IF rSlidingS > 0 THEN
rSignum := 1;
ELSIF rSlidingS < 0 THEN
rSignum := -1;
ELSE
rSignum := 0;
END_IF

END_IF

rQtotal:= rActualSpeed * rPumpVh * 1E3 - rLGain * rSignum; (* Qt = Pump flow [l/min] - L*Sign(s) *)
rZp:= rHydC * rQtotal;

rDiscN := rLGain * rSignum/( rPumpVh * 1E3 );

ELSE
```

```
rNComp := 0;
rDiscN := 0;
rSlidingS := 0;

END_IF

END_FUNCTION_BLOCK

ACTION aInit:
rZ := rPs;
rHydC:= rHydC/60;
END_ACTION

FUNCTION_BLOCK fbFlow2ndSMCompensator
(*#################################################################################################*)
(*#  Name: fbFlow2ndSMCompensator                                          Date: 29.03.2010/TGR #*)
(*#################################################################################################*)
(*#  DESCRIPTION:                                                                               #*)
(*#   - Implements Second Order Sliding Mode Flow Compensator                                  #*)
(*#                                                                                             #*)
(*#################################################################################################*)
(*#  Change Log:                                                                                #*)
(*#    29.03.2011, TGR                                                                          #*)
(*#    - Program Created                  #*)
(*#                                                                                             #*)
(*#################################################################################################*)


VAR

rZ: REAL; (* Observed Pressure [bar] *)
rAbsSqrOfS: REAL := 0; (*Square root of the absolute value of S*)
rU2: REAL := 0; (*Second term of the GSTA*)
rK3_2: REAL; (*power 2 of gain K3*)
rZ2: REAL:=0; (*Observed Pressure Dot *)
rU1: REAL := 0; (*First Term of the GSTA*)
rz1p: REAL := 0; (* Observed Pressure Dot (Total) *)

bTestWd: BOOL := TRUE;
END_VAR
VAR_INPUT

bEnable: BOOL := FALSE;

(********************** Initial Set Up **********************)

rTs: REAL := 0.001; (*Sample time *)
rVh: REAL := 0.000001; (*Volume Displacement of the Pump [m/U]*)
rC: REAL := 0; (*Hydraulic Capacitance [bar/liter]*)

(********************** Working Values **********************)

rP: REAL; (* Real Pressure [bar] *)
rK3: REAL := 2; (*Gain K3*)
rK2: REAL := 0; (*Gain K2*)
rK1: REAL := 0; (*Gain K1*)
rQpump: REAL := 0; (*Flow pumped [l/min]*)
bReset: BOOL := FALSE; (*Reset ONCE the Sliding Mode*)
rMaxSpeed: REAL := 3000;  (* Maximal speed of the Pump *)
rWarnBound: REAL := 0.99;  (* Activate reset with this bound for the Speed *)
bWdesired: BOOL := FALSE;

END_VAR
```

```
VAR_OUTPUT
rComp: REAL := 0; (*Compensation Signal [rpm]*)
rCompSmooth: REAL:= 0; (*Smooth compensation signal [rpm]*)
rQtotal: REAL := 0; (*Total Flow in the System [l/min]*)
rQLoadFast: REAL := 0; (*Load Flow in the System [l/min]*)
rQLoadSmooth: REAL := 0; (*Load Flow in the System [l/min]*)
rSign: REAL := 0; (*Signum of rSlide*)
rZ1: REAL := 0; (*Pressure Dot  [bar/s]*)
rSlide: REAL := 0; (* Sliding Surface [bar] *)
rPdot: REAL := 0; (* P dot from the Algorithm [bar/s] *)
END_VAR
(* @END_DECLARATION := '0' *)
IF bEnable THEN

(************* GSTA U2 ************************)

rU2 := 0.5 * rSign + rK3_2 * rSlide + 1.5 * rK3 * rAbsSqrOfS;
rZ2 := rK2 * rU2 * rTs + rZ2;

(************ Get Ready *********************)


IF bReset THEN

rZ := rP;
rZ2 := 0;
bReset := FALSE;
rSign := 0;
rSlide := 0;
rAbsSqrOfS := 0;

ELSE
rZ  := rZ1 * rTs + rZ;
rSlide := rP - rZ;
IF rSlide > 0 THEN
rSign := 1;
ELSIF rSlide < 0 THEN
rSign := -1;
ELSE
rSign := 0;
END_IF
rAbsSqrOfS := SQRT( rSlide  * rSign ) * rSign;
END_IF

(*********** GSTA U1 *************************)

rU1 := rAbsSqrOfS + rK3 * rSlide;
rz1p := rK1 * rU1 + rZ2;

(********** Outputs ************************)
IF  bWdesired THEN
IF (rz1p/(rVh*1000)) > rMaxSpeed THEN
rz1p := rVh*1000*rMaxSpeed;
IF (rZ2/(rVh*1000)) > rMaxSpeed THEN
rZ2 := rVh*1000*rMaxSpeed;
END_IF
ELSIF (rz1p/(rVh*1000)) < -rMaxSpeed THEN
rz1p := -rVh*1000*rMaxSpeed;
IF (rZ2/(rVh*1000)) < -rMaxSpeed THEN
rZ2 := -rVh*1000*rMaxSpeed;
END_IF
END_IF
ELSE
```

```
IF (rz1p/(rVh*1000)) < -rMaxSpeed THEN
rz1p := -rVh*1000*rMaxSpeed;
IF (rZ2/(rVh*1000)) < -rMaxSpeed THEN
rZ2 := -rVh*1000*rMaxSpeed;
END_IF
ELSIF (rz1p/(rVh*1000)) > 0 THEN
rz1p := 0;
IF bTestWd THEN
IF (rZ2/(rVh*1000)) > 0 THEN
rZ2 := 0;
END_IF
END_IF
END_IF
END_IF

rComp := -rz1p/(rVh*1000);
rCompSmooth := -rZ2/(rVh*1000);
rQtotal := rz1p + rQpump;
rQLoadFast := rz1p;
rQLoadSmooth := rZ2;
rZ1 := rC * rQtotal / 60 ;
rPdot := rZ1;

ELSE
(* All Outputs are zero*)

rComp := 0;
rCompSmooth := 0;
rQtotal := 0;
rZ1 := 0;
rSign := 0;

END_IF
END_FUNCTION_BLOCK
ACTION aInit:
rK3_2 := rK3 * rK3;
rZ2 := 0 ;
rz1p := 0;
END_ACTION

FUNCTION_BLOCK fbGainGSTA
(*###############################################################################################*)
(*#  Name: fbGainGSTA                                                     Date: 29.03.2010/TGR #*)
(*###############################################################################################*)
(*#  DESCRIPTION:                                                                              #*)
(*#   - Implements the adaptive gain for the GSTA                                              #*)
(*#                                                                                            #*)
(*###############################################################################################*)
(*#  Change Log:                                                                               #*)
(*#    29.03.2011, TGR                                                                         #*)
(*#    - Program Created                    #*)
(*#                                                                                            #*)
(*###############################################################################################*)

VAR_INPUT
bEnable: BOOL := FALSE;
(********************** Initial Set Up **********************)

rTf: REAL:= 0.016; (*Filter Time [sec]*)
rTs: REAL:=0.001; (*Sample time [sec]*)
rTresHold: REAL:= 0.2; (* Treshold for the filtered signal *)
rOffset: REAL := 0.03; (*Define the offset of Quantized value *)
rMotorTqMax: REAL; (*Maximal Torque of the motor*)
rAlpha: REAL := 5; (*Growing Coefficient*)
```

```
rAlpha2: REAL := 100; (*Growing Coefficient*)
rQmax: REAL; (* Maximal Pump Flow [l/min] *)


(********************** Working Values ***********************)

rQtotal: REAL := 0; (*Total Flow [l/min]*)
rSign: REAL; (*Signum of the sliding variable*)
rK2normal: REAL := 0.21;
rPpCommand: REAL := 0;
rPpTresHold: REAL := 350;
rPpAverageMax: REAL := 2000;

END_VAR

VAR_OUTPUT

rK2: REAL := 0; (*K2 GSTA Gain*)
rK1: REAL := 0; (*K1 GSTA Gain*)

END_VAR

VAR

ifb1PT1: fbPT1; (* First Order Filter *)
ifb2PT1: fbPT1; (* First Order Filter *)
rFilterSign: REAL := 0; (*Filtered Sign*)
rQuantSign: REAL := 0;
rMin: REAL := 0.1;
rTau: REAL := 0;
rFilterAbsOut: REAL := 0;
rQuantQ: REAL;
rMinK1: REAL := 0.1;
rQpast: REAL := 0;
rTauK1: REAL := 0;
arDataC: ARRAY [0..45] OF REAL := 0;      (* Array with samples                       *)
rFactorK1: REAL := 0.2;
bSqrt: BOOL := FALSE;
iCount: INT := 0;
iSize: INT := 14;
bInitSAmple: BOOL := FALSE;
rAcumulator: REAL := 0;

bSelectQuantMode: BOOL := FALSE;
bLogGrow: BOOL := TRUE;

rDeviation: REAL := 0.09;
iCDicrease: INT := 0;
iDicreaseBand: INT := 50;
rGrowBound: REAL := 0.4;
rQuantSignLog: REAL := 0;

bNoQuant: BOOL := TRUE;
END_VAR
(* @END_DECLARATION := '0' *)
IF bEnable THEN

rPpCommand := ABS(rPpCommand);

ifb1PT1.rInput := rSign;
ifb1PT1();
ifb2PT1.rInput := ABS( ifb1PT1.rOutput );
ifb2PT1();
```

```
rFilterAbsOut :=  ifb2PT1.rOutput;

(************* K2 GSTA ***************************)


IF bSelectQuantMode THEN
rQuantSign := fUpQuantizer( rFilterAbsOut );
ELSIF bNoQuant THEN
rQuantSign := rFilterAbsOut;
ELSE
rQuantSign := fUpQuantizerLog( rFilterAbsOut );
END_IF

IF bLogGrow THEN
rQuantSignLog := fUpQuantizerLog( rFilterAbsOut );
ELSE
rQuantSignLog := 1;
END_IF

FOR iCount := 0 TO iSize DO
arDataC[iCount] := arDataC[iCount+1];
END_FOR;
arDataC[iCount] := rQuantSign;

rAcumulator := 0;
FOR iCount := 0 TO iSize DO
rAcumulator := rAcumulator + arDataC[iCount];
END_FOR;
rAcumulator := rAcumulator/iSize;

IF rQuantSign = 1 THEN
rK2normal := rAlpha2 * rQuantSignLog*rTs + rK2normal;
IF rK2normal >= 1 THEN
rK2normal := 1;
END_IF

iCDicrease := 0;
ELSIF rQuantSign > ( rAcumulator + rDeviation ) THEN
rK2normal := rAlpha2 * rQuantSignLog*rTs + rK2normal;
IF rK2normal >= 1 THEN
rK2normal := 1;
END_IF
iCDicrease := 0;
ELSIF rQuantSign < (rAcumulator - 0.5*rDeviation) THEN
IF rAcumulator > rGrowBound THEN
iCDicrease := iCDicrease + 1;
IF iCDicrease > iDicreaseBand THEN
rK2normal := -rAlpha * rTs*(1.01-rQuantSignLog) + rK2normal;
ELSE
rK2normal := rK2normal;
END_IF
ELSE
rK2normal := -rAlpha * rTs*(1.01-rQuantSignLog) + rK2normal;
iCDicrease := 0;
END_IF
IF rK2normal < rOffset THEN
rK2normal := rOffset;
END_IF

ELSE
rK2normal := rK2normal;
IF rK2normal >= 1 THEN
rK2normal := 1;
```

```
END_IF
iCDicrease := 0;
END_IF

IF rPpCommand > rPpTresHold THEN
rK2normal := rPpCommand/ rPpAverageMax;
IF rK2normal >= 1 THEN
rK2normal := 1;
END_IF
IF rK2normal < rOffset THEN
rK2normal := rOffset;
END_IF
END_IF
rK2 := 0.5*rMotorTqMax * ( rK2normal );

(************** K1 GSTA ****************************)

IF bSqrt THEN
rK1 := SQRT(rK2)*rFactorK1;
ELSE
rK1 := rK2*rFactorK1;
END_IF
ELSE
rK1 := 0;
rK2 := 0;
END_IF
END_FUNCTION_BLOCK

ACTION aInit:
ifb1PT1.rConst1:= rTs/rTf;
ifb1PT1.rConst2:= 1 - rTs/rTf;
ifb1PT1.aInit();

ifb2PT1.rConst1:= rTs/rTf;
ifb2PT1.rConst2:= 1 - rTs/rTf;
ifb2PT1.aInit();
rK2normal := rOffset;
END_ACTION
```

References

[1] S. Bennett, "Development of the pid controller," *IEEE Control Systems*, vol. 6, pp. 58–65, December 1993.

[2] K. A. Jelali M., *Hydraulic Servo-systems Modelling, Identification and Control.* Springer, 2002.

[3] J. G. Ziegler and N. B. Nichols, "Optimum Settings for Automatic Controllers," *TRANSACTIONS OF THE A.S.M.E.*, pp. 759–765, November 1942.

[4] U. D. of Energy., "Variable speed pumping, a guide to succesful implementation.," *U.S. Department of Energy Industrial Technologies Program*, May 2004.

[5] J. Slotine, W. Li, *et al.*, *Applied nonlinear control.* Prentice hall Englewood Cliffs, New Jersey, 1 ed., 1991.

[6] W. Perruquetti and J. Barbot, *Sliding mode control in engineering.* CRC Press, 2002.

[7] V. Utkin, J. Guldner, and J. Shi, *Sliding mode control in electro-mechanical systems.* Automation and control engineering, CRC Press, 2009.

[8] K. Young, V. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *Control Systems Technology, IEEE Transactions on*, vol. 7, no. 3, pp. 328–342, 1999.

[9] C. Edwards and S. Spurgeon, *Sliding mode control: theory and applications.* Taylor & Francis systems and control book series, Taylor & Francis, 1998.

[10] J. Komsta, J. Adamy, and P. Antoszkiewicz, "Input-output linearization and integral sliding mode disturbance compensation for electro-hydraulic drives," in *Variable Structure Systems (VSS), 2010 11th International Workshop on*, pp. 446 –451, june 2010.

[11] T. H. J. A. J. Komsta, P. Antoszkiewicz, "New nonlinear robust control concept for electro-hydraulic drives," *in Proc. 7th International Conference on Fluid Power (IFK) Aachen*, 2010.

[12] J. A. J. Komsta, P. Antoszkiewicz, "Load pressure and p-cascade control with integral sliding mode disturbance compensation for electrohydraulic cylinder drives," *The Twelfth Scandinavian International Conference on Fluid Power, Tampere, Finland*, May 2011.

[13] H. L. J. Shi and N. Bajeinca, "Robust control of robotic manipulators based on integral sliding mode," *International Journal of Control*, 2008.

[14] V. Utkin and H. Shi, "Integral sliding mode in systems operating under uncertainty conditions," *In Proceedings of the 35th Conference Decision and Control, Kobe, Japan.*, p. 45914596, 1996.

[15] A. Girin, F. Plestan, X. Brun, and A. Glumineau, "High-order sliding-mode controllers of an electropneumatic actuator: Application to an aeronautic benchmark," *Control Systems Technology, IEEE Transactions on*, vol. 17, pp. 633 –645, may 2009.

[16] Y. B. Shtessel, I. A. Shkolnikov, and A. Levant, "Smooth second-order sliding modes: Missile guidance application," *Automatica*, vol. 43, no. 8, pp. 1470 – 1476, 2007.

[17] A. Levant, "Sliding order and sliding accuracy in sliding mode control," *International Journal of Control*, vol. 58, no. 6, pp. 1247–1263, 1993.

[18] J. Moreno, "A linear framework for the robust stability analysis of a Generalized Super-Twisting Algorithm," in *Electrical Engineering, Computing Science and Automatic Control, CCE, 2009 6th International Conference on Decision and Control*, pp. 1–6, IEEE, 2009.

[19] J. Moreno and M. Osorio, "A Lyapunov approach to second-order sliding mode controllers and observers," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 2856–2861, IEEE, 2008.

[20] J. Moreno, "Lyapunov approach for analysis and design of second order sliding mode algorithms," in *Sliding Modes after the First Decade of the 21st Century* (L. Fridman, J. Moreno, and R. Iriarte, eds.), pp. 113–149, Springer Berlin / Heidelberg, 2012.

[21] B. R. AG, "Up to 70 percent energy saving in tool machines with Rexroth Blue Hydraulic Drives," *Press Release*, no. PI 104/09, 2009.

[22] A. Radke and Z. Gao, "A survey of state and disturbance observers for practitioners," in *American Control Conference, 2006*, p. 6 pp., june 2006.

[23] S. Spurgeon, "Sliding mode observers: a survey," *International Journal of Systems Science*, vol. 39, no. 8, pp. 751–764, 2008.

[24] M. Basin, A. Ferreira, and L. Fridman, "Sliding mode identification and control for linear uncertain stochastic systems," in *Decision and Control, 2006 45th IEEE Conference on*, pp. 1339 –1344, dec. 2006.

[25] M. Basin, A. Ferreira, and L. Fridman, "Lqg-robust sliding mode control for linear stochastic systems with uncertainties," in *Variable Structure Systems, 2006. VSS'06. International Workshop on*, pp. 74 –79, june 2006.

[26] A. Ferreira, F. Bejarano, and L. Fridman, "Output hierarchical super twisting observation based robust feedback control: With or without chattering?," in *Decision and Control, 2007 46th IEEE Conference on*, pp. 4335 –4340, dec. 2007.

[27] A. Ferreira, F. Bejarano, and L. Fridman, "Robust output regulation with exact unmatched uncertainties compensation based on hosm observation," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 6692 –6696, dec. 2009.

[28] L. Capisani, A. Ferrara, A. Ferreira, and L. Fridman, “Higher order sliding mode observers for actuator faults diagnosis in robot manipulators,” in *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, pp. 2103 –2108, july 2010.

[29] A. Ferreira, F. Bejarano, and L. Fridman, “Robust control with exact uncertainties compensation: With or without chattering?,” *Control Systems Technology, IEEE Transactions on*, vol. 19, pp. 969 –975, sept. 2011.

[30] I. Boiko, L. Fridman, A. Pisano, and E. Usai, “Analysis of chattering in systems with second-order sliding modes,” *Automatic Control, IEEE Transactions on*, vol. 52, pp. 2085 –2102, nov. 2007.

[31] V. Brgeault, F. Plestan, Y. Shtessel, and A. Poznyak, “Adaptive sliding mode control for an electropneumatic actuator,” in *Variable Structure Systems (VSS), 2010 11th International Workshop on*, pp. 260 –265, june 2010.

[32] A. G. Y. S. F. Plestan, V. Bregeault and E. Moulay, “Advances in high order and adaptive sliding mode control - theory and applications.,” *In 11th IEEE International Workshop on Variable Structure Systems, Mexico City, Mexico*, 2010.

[33] Y. Shtessel, J. Moreno, F. Plestan, L. Fridman, and A. Poznyak, “Super-twisting adaptive sliding mode control: A lyapunov design,” in *Decision and Control (CDC), 2010 49th IEEE Conference on*, pp. 5109 –5113, dec. 2010.

[34] F. Plestan, Y. Shtessel, V. Brgeault, and A. Poznyak, “New methodologies for adaptive sliding mode control,” *International Journal of Control*, vol. 83, no. 9, pp. 1907–1919, 2010.

[35] K. Ogata, *Modern control engineering.* Prentice Hall, 2010.

[36] E. Umez-Eronini, *Dinámica de sistemas y control.* Thomson, 2001.

[37] N. Nise, *Control Systems Engineering.* John Wiley & Sons, 2010.