POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Aerospaziale

Master's Degree Thesis

Study of Prognostic Algorithms for Electrohydraulic Actuators Applied to Aerospace Systems

Thesis advisors
Matteo D.L. **Dalla Vedova**
Paolo **Maggiore**
Pier Carlo **Berri**

Candidate
Leonardo **Forletta**

December 2020

A chi mi è sempre stato vicino,
anche da lontano.

ABSTRACT

The purpose of this dissertation is to conduct prognostic analysis on a numerical model of an electro-hydraulic actuator – used in aerospace environments – to permit early proactive maintenance on the physical system, beneficial for its lifespan extension.

The detailed model of a flapper-nozzle servo valve is used to help identify and analyse possible alterations on the dynamical response by applying several types of commands and faults (single or multiple). The main types of commands used in this study are: two step commands, a basic ramp command, two chirp commands with different decrescent amplitude, and a mixed time-history command for prognostic purposes. The faults I examined are backlash and increasing dynamic friction on the actuation jack.

A comparation between the responses of the detailed model and of a simpler monitoring model is presented to evaluate appreciable differences. Then, the parameters of the monitoring model are modified through the appropriate optimisation algorithms (designed for fault identification purposes) to minimize errors between the dynamic responses of the two models. A detailed illustration of the algorithm simulation is than presented, evaluating the limits and the strengths of the models and the algorithms used.

ACKNOWLEDGEMENTS

I would like to thank my thesis advisor, prof. Dalla Vedova, for the guidance, the humanity and the precious time he decided to donate me.

To my friends, which are countless and scattered all over the country, but who never failed to make me feel cared for.

To my roommates, whose laughs filled this small apartment even when there wasn't anything lo laugh about.

Thanks to all my family, who never lost hope in me even when I did.

And finally, thanks to my partner, who gave me energy, hopes and attentions without any hesitation and when I needed them the most, even if half a nation away.

INDEX

1 INTRODUCTION

Due to a rapid increase in aerospace systems complexity, cost and safety demands, new methods for augmenting their performances and reliability are researched constantly. Diagnostics is an intricate, yet well explored engineering field which allows to comprehend the causes behind a critical failure, and thus avoiding them. For a long time, it was the only tool available in the industry, and it heavily relied on data collection, observation and comparison.

Today new methods are investigated, allowing the prediction of the faults or incipient failures through advanced algorithms, and relieving the companies from old procedures.
This new engineering branch is called prognostics, and is a relatively new field of interest.

PHM (Prognostics and Health Management) has its basis on the design of detailed models which allow the accurate assessment of the health of the studied system. After that, the RUL (Remaining Useful Life) of the system is computed, enabling predictive maintenance, repair, or substitution of a component before the occurrence of a failure.

PHM gathers different steps for its implementation. The main are:

- Data acquisition
- Data pre-processing
- Fault and anomalies detection
- Diagnostics
- Prognostics
- Decision making
- Graphical User Interface (GUI)

The model-based prognostic approach exploits mathematical models and equations to modelize events, phenomena and behaviour, comparing the responses of a monitoring model to a reference model. This approach cuts a lot of costs derived from the laboratory testing, required beforehand. It is more accurate but very challenging due to the complex nature of the studied systems.

The current thesis work exploits a model-based approach to conduct fault analysis and identification on a numerical model of an electrohydraulic actuator present on the main control surfaces of a typical commercial aircraft. As they are some of the most pivotal and critical systems of the aircraft, particular attention to their health assessment is required.

CHAPTER 2 presents a brief historical and technical report of actuation systems in general, explaining their purposes, different designs, functioning and utilization on the most common aircrafts.

In CHAPTER 3 the functioning of each of the components of the EHA system is reported, with particular attention to non-linear phenomena, like magnetic hysteresis and friction.

CHAPTER 4 shows the computational model of the EHA system with some of the mathematical relations used in its design. It is presented in the Matlab/Simulink environment.

CHAPTER 5 revolves around the analysis of the possible faults that can affect the physical – hence the computational – models. I analyse simulations performed in nominal conditions, with single faults and mixed faults. Many different inputs were also used,

stressing and testing the model behaviour to comprehend its limits.

CHAPTER 6 illustrates the prognostic design philosophy, the concepts of mathematical optimization, genetic algorithms and their simulations as the first steps towards the prediction of failures.

CHAPTER 7 collects final impressions, thoughts, advices and flaws that accompanied the entire work.

In the Appendices, a collection of the Matlab scripts I used throughout the thesis.

2 ACTUATION SYSTEMS

2.1 Definition and hystorical background

A servomechanism is a device whose aim is to produce an amplified motion starting from a smaller input or command. It is defined by the physical dimensions controlled: position, speed or force. It generally uses a negative feedback signal to automatically correct the output until it nearly matches the command. The amplification of the motion is commonly achieved by servo valves. They are generally of three types: hydraulic, pneumatic or mechanic. Another classification is based on the type of control that regulates the system: electrical or mechanical.

Servo valve systems have roots in ancient times. We recall the first hydraulic servomechanism in ancient Greece by Ctesibius, a famous inventor and mathematician. He largely studied pneumatics and hydraulics, extensively using pumps and controlling water levels through float valves, both of which we have no previous record.

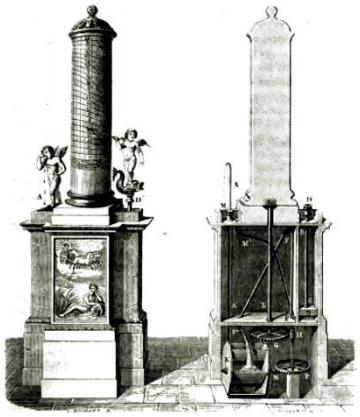


Figure 2.1: Ctesibius water clock

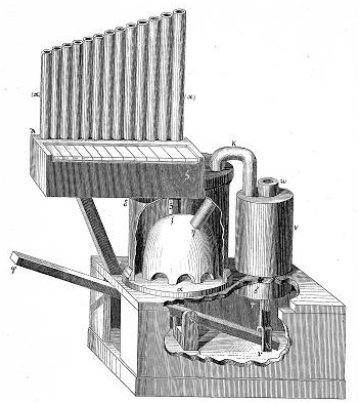


Figure 2.2: Ctesibius water organ

A growing demand for power and precision lead to several inventions, which culminated in the Industrial Revolution. During the 17th and 18th centuries, we recognise the use of the first centrifugal governors, equipped with mechanical feedback control, used to regulate speed levels initially in millstones and windmills, then in steam engines.

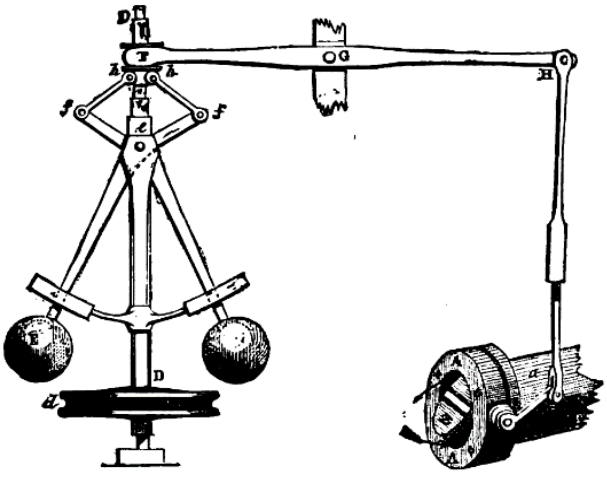


Figure 2.3: Centrifugal governor

In the next centuries, with the widespread application of electrical current, servo valves evolved again, with its main advantages seen during the two World Wars, hence the development of automatic fire control for guns, consisting in solenoid driven spool with a simple spring return.

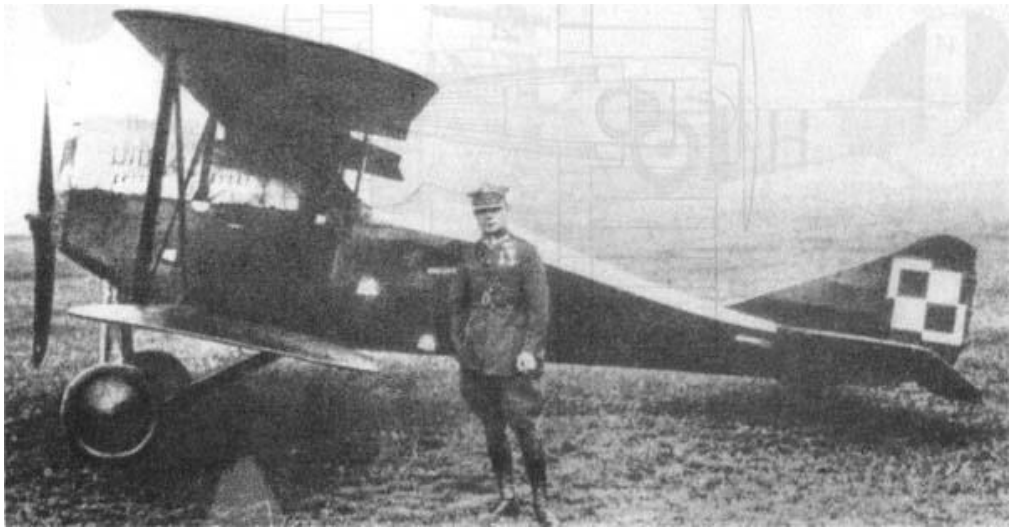


Figure 2.4: Ansaldo A-1, the first Italian plane to enlist in WWI

Until now, aircrafts were small enough to be conducted effortlessly without servomotors, relying on the pilot's strength only. As aircraft's size, complexity and performances grew, more appropriate actuation was needed. Hydraulic-assisted control systems with artificial feel were designed to overcome the already huge aerodynamic forces.

Technological progress greatly improved thanks to the fly-by-wire technology, overcoming the already obsolete and heavy mechanical circuits (rods, cables and chains) with electric actuators, lighter redundancies and digital commands, displaying improvements in maintainability, reliability and safety.

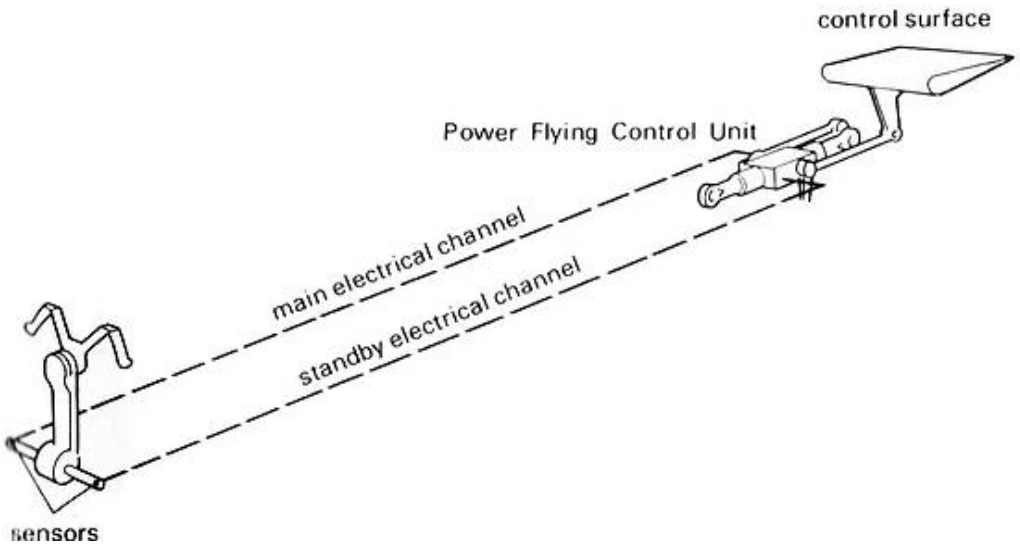


Figure 2.5: Fly-by-wire simple architecture example

2.2 Aircraft Flight Control Systems

A fixed-wing aircraft is a vehicle designed to fly exploiting aerodynamic forces and/or thrust generated by engines to defy gravity. Through the control surfaces, it is possible to manage and modify the overall aerodynamic forces on the aircraft to change its attitude.
Modernly, these are moved through a series of actuators, commanded by a computer (called Flight Control Computer, FCC) that also processes data from sensors and provides stability aid. The FCC digitalizes the input commands from the pilot, filtering and processing it, then deflecting it to the transducers via the avionic bus.
The transmission of the command can be mechanical, hydro-mechanical or electric (digital).

- Mechanical control systems gradually became obsolete with aircraft's increasing size over time. This architecture uses a complex set of mechanical parts like rods, chains and levers to transmit the forces and move the control surfaces. It is highly susceptible to structural stresses, caused mainly by the hinge moment (the actual force required to deflect a surface, considering aerodynamical forces and geometrical quantities). To grant assistance to the pilot, advantageous gearing and servo-tabs are provided, and later analysed. It is still used in small aircraft, like Cessna 172, as the main control system, or in bigger aircrafts as a backup system.

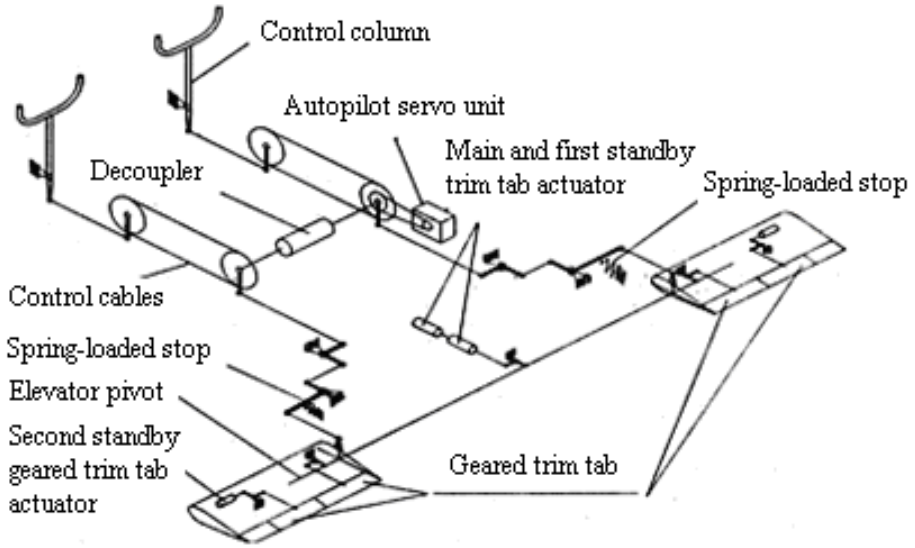


Figure 2.6: A mechanical control system

- Hydro-mechanical control systems are generally lighter and less complex than its counterpart. They are composed of two main circuits: the hydraulic one contains pumps, filters, valves and tanks whose objective is to convert hydraulic pressure into control surface movement, and a mechanical one that connects the pilot to the hydraulics. This type of system can be found in large aircraft from the past like the Lockheed SR-71 "Blackbird".
- Electrical control systems get rid of the mechanical controls from the cockpit to employ only digital signals. This choice benefits the overall weight, the structural stresses and inherent elasticity of mechanical transmission mechanisms. The resulting reduction in the airframe weight is crucial in creating smaller and more performant aircrafts, both in military and civil aviation environment. The mechanical feedback is substituted with

digital closed-loop feedback control, with an instantaneous comparison between sensors and reference signals. Lastly, the pilot does not need to counteract every single movement from the aircraft, because the Flight Control Computer can stabilise the vehicle without his involvement. As a result, inherently unstable aircrafts can be piloted, improving also in manoeuvrability.

2.3 Control surfaces

Control surfaces can be divided into two types:

- main control surfaces
- secondary control surfaces

Other surfaces that have effects on aerodynamics but are not moving parts of the aircraft (like winglets, wing fences, leading-edge extensions, etc...) will not be discussed in this work.

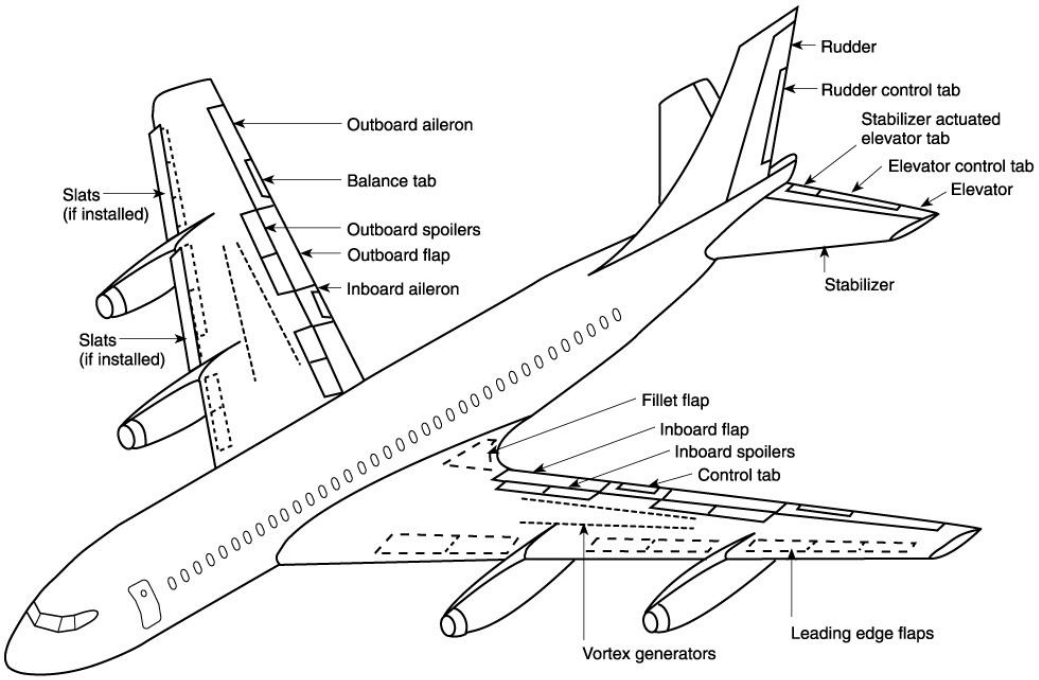


Figure 2.7: A typical airliner control surfaces location

2.3.1 Main control surfaces

The main control surfaces are the moving parts of the vehicle directly and continuously controlled by the pilot to change or maintain its attitude. These are of three types: ailerons, elevators and rudders.

They need to be frequently moved by the pilot to adjust the attitude, to modify the aircraft flight path or to oppose to external forces or dynamic oscillatory modes of the aircraft. Since they are fundamental for the aircraft mission, they need to be precise, safe and reliable throughout their entire life and need servo actuators of comparable performances. The requirements are very strict and are mostly designed to be optimal for the pilot's comfort and to ensure the overall stability and controllability of the aircraft.

The ailerons are usually hinged toward the trailing of the wing, near its tip. They're disposed symmetrically and move in opposite directions, causing the change in direction of lift and drag, hence generating a rotating torque around the longitudinal axis, called roll. The relative control in the cockpit is generated by rotating a control wheel or a joystick, with their centre positions corresponding to the neutral balance angle.

The elevators are located symmetrically at the tail of the aircraft, mounted on the horizontal stabilizers. They move upwards or downwards depending on the front/back

movement of the pilot's control stick, generating a pitching moment around the lateral axis of the aircraft. Henceforth, they are responsible for the change of angle of attack during the flight.

The rudder is hinged to the vertical stabilizer of the aircraft tail. It deflects left or right based on the pilot's movement on the pedals. The movement of the control surface generates a moment around the vertical axis of the plane, called yaw. The yaw can also be generated indirectly by moving the ailerons. In this case it is called adverse yaw.

2.3.2 Secondary control surfaces

The secondary control surfaces are less fundamental for aircraft control, but still significant in improving the performance characteristics and relieving excessive stress from the pilot. The most noteworthy are the following: spoilers, flaps, slats, air brakes and tabs. Given their reduced impact and size, they require less power and precision, unlike the other main control surfaces.

Spoilers are located on the wing, slightly ahead of the trailing edge and their purpose is to reduce lift on the wing, therefore increasing drag, permitting to lose altitude without an increase in speed. They are fundamental in the process of controlled descent and they can be fully or partially deployed.

Flaps are placed in the trailing edge of the wing, between its root in the fuselage and the ailerons. These devices serve as a wing curve modifier and help lessen the stall speed, increasing the maximum lift coefficient and the drag. Their support is significant during take-off and landing, when the speed is low, and the angle of attack is higher than in levelled flight. The construction designs vary a lot between aircraft types. They are generally grouped in five sub-types: plain, split, Fowler, slotted and Krueger, each one based on the curvature or area extension obtained, although there are some rarer and more complex designs.

Slats are the control surfaces designed as extensions of the wing to allow operations at a higher angle of attack. They are positioned in different sections at the leading edge of the wing, spread all over its length. They're usually operated simultaneously with the flaps during take-off and landing phases. Like flaps, slats also have different construction designs based on the performance needed. They can be easily retracted during cruising and high-speed mission phases, as they would experience too much structural stress and compromise the aircraft performances.

The purpose of the air brakes is to slow down the aircraft like the spoilers but increasing the overall drag instead of intervening on the lift. Also, they are not located on the wings but more likely on the fuselage or the tail, largely deflecting the airstream. They are needed especially when a thrust reversal system is missing. When the latter, placed near the engines, is fully deployed, it deflects partially or completely their airstream in the opposite direction, ensuring the decrease of speed. Both these systems are used during the landing, the first during descent and the second on the runway, just after the touch-down.

The remaining control surfaces are the tabs, whose purpose and location vary a lot. Generally, they are used to ease the control of another control surface or to hold them in place, relieving the pilot or the actuators partially from the stress generated when opposing to the aerodynamic forces. The tabs used for holding the other main control surfaces in place are called trim tabs, are set up by the pilot manually and allow "hand-off" flight when opposing constant forces. They're adjusted frequently during the flight.

The balance tabs are used to overcome more easily the forces required to move the control surface, especially for large aircrafts and high speeds. They also have the opposite purpose of anti-balance tabs, used to desensitize flight controls in small aircrafts.

Servo tabs are directly linked to the pilot's commands through mechanical linkages and can be a primary means of control. When deflected, it moves the larger control surface in the opposite direction, changing the characteristics of the airflow.

Spring tabs are servo tabs linked with a spring and a torsion rod to the control surface. They are activated only when high speeds are reached, helping the pilot.

Among all these tabs, the trim tabs are the only ones actuated by servomechanisms.

2.4 Actuators

The actuators are the transducing devices responsible for the movement of the control surfaces. Their design depends on the performance needed and on the parent system that produces power. As said earlier, we can generally find three different types: hydromechanical, electrohydraulic and electromechanical.

2.4.1 Hydromechanical Actuators (HMA)

The command of the pilot is transmitted through a three-centre lever to a servo-valve, which draws its power from the hydraulic system to amplify the command. As evident from the Figure 2.8, the position of the lever determines a movement of a spool, that opens or closes the hydraulic circuit. The pressure generated from the flow operates on the chamber below, moving a jack directly connected to the control surface, determining the output of the actuator. The primary defects of the model are the massive weight and complexity of the plant, which is rather difficult to maintain (expensive and lengthy hour wise) and uses many components. Still, there are some merits, including durability, safety and reliability of the system.

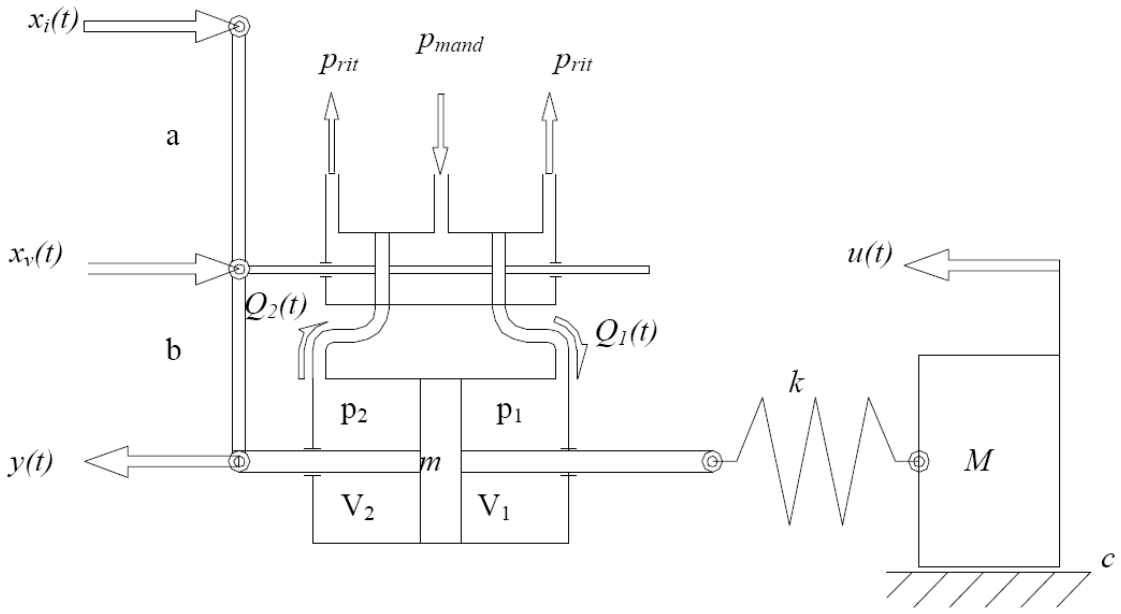


Figure 2.8: Hydro-mechanical system with command lever, servo valve and load

2.4.2 Electrohydraulic Actuators (EHA)

To overcome the deficiencies of hydromechanical actuators, hydroelectrical ones are chosen because of their compactness and efficiency in cost and maintainability. That's the type of actuator discussed in this work and it's one of the most used in the industry.

The command from the pilot is electrical (getting rid of the bulky mechanical linkages), proceeding toward a motor which turns the signal into mechanical movement. The high-pressure circuit then enters the servo-valve, whose purpose is to operate the jack. The servo-valve designs can vary a lot depending on the target of the actuation and will be extensively described in the next chapter. As the hydromechanical actuation, oil filters are required to avoid faults and failures of the system and to ensure that the actuator works as intended. Thanks to the electrical system, high precision sensors and transducers are employed, achieving more

solid performances than its predecessors. For this reason, EHAs are frequently used in both civil and military environments.

2.4.3 Electromechanical Actuators (EMA)

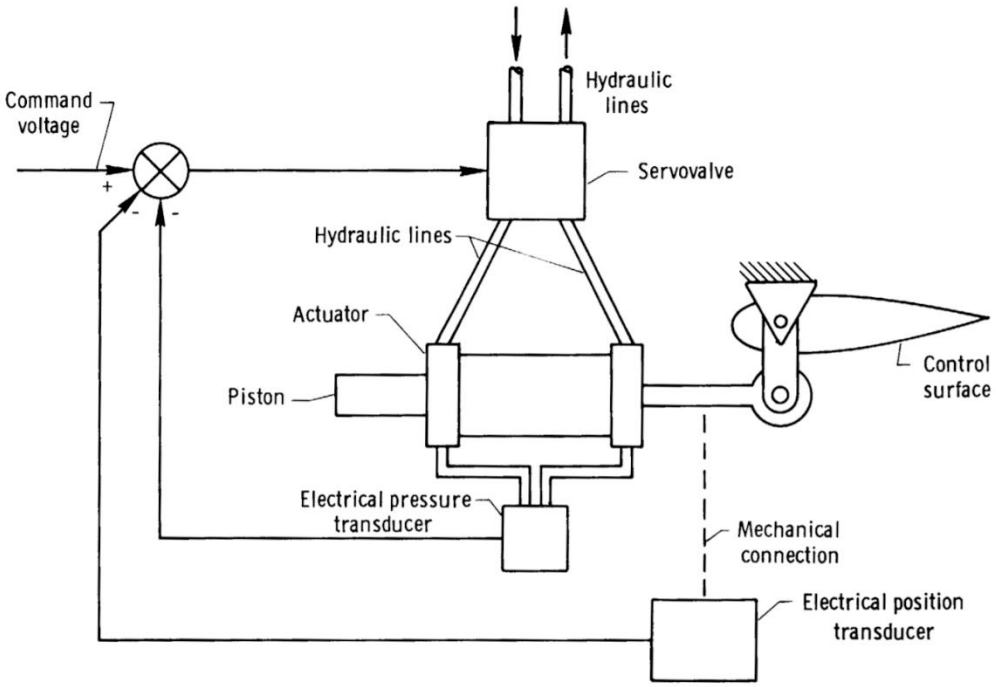


Figure 2.9: Electrohydraulic system example with electrical command, hydraulic circuit, servo valve and control surface

With the advent of more-electric and full-electric configurations, actuation technologies made a step forward, providing even better efficiency and performances. Electromechanical actuators have the benefits of getting rid of the hydraulic system, transforming electrical power directly into mechanical. The pilot's command is still electrical, but now the motor transforms the signal into mechanical movement through a gearbox and a rotary-to-linear mechanism (e.g. ball screw), producing little friction. However, the technology still requires further studies to ensure its reliability and safety, especially considering the vulnerability of the ball screw connection, prone to getting stuck often. There are still few applications in secondary control surfaces thanks to its minor weight and valuable precision.

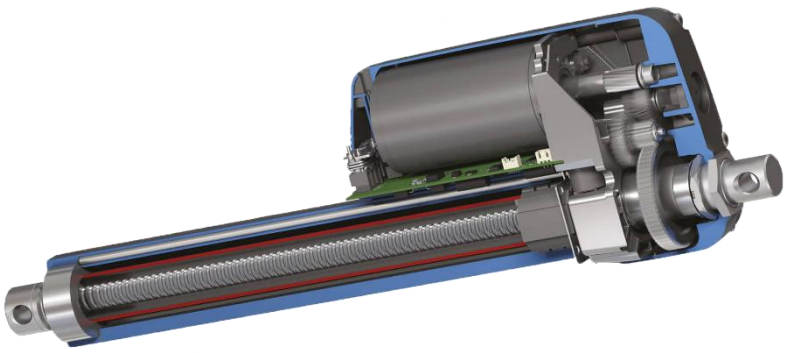


Figure 2.10: Electromechanical ball screw actuator

2.5 Servo valves

Considering electrohydraulic systems, servo valves are the components whose purpose is to deviate the hydraulic flow to the actuation jack after interpreting the small command signal in output from the transducer. They serve as a power amplification component, hence being the core of the servomechanism.

They can be composed of one or more stages, according to the amount of amplification needed. Therefore, a three-stage position-controlled servo valve can generate a huge displacement of a large control surface starting from a small pilot input. In this work, we will discuss only two-stage servo valves, as they are the most common.

Considering that servo valves require rigid precision performances, a closed-loop control logic is adopted. It compares the input reference signal to the one read from the sensors, determining an error that can be reduced by implementing a PID (proportional-integral-derivative) controller, appropriately tuned.

Three main servo valve architectures will be analysed: direct-drive, jet-pipe and flapper nozzle.

2.5.1 Direct Drive Valve (DDV)

This servo-valve uses a direct current linear motor to move the hydraulic spool responsible for the actuation. The electric command signal produces a Pulse-Width Modulated (PWM) current which drives the motor. The latter is composed of a solenoid coil paired with permanent magnets inside of which a shaft (directly connected to the hydraulic spool) moves. A centering spring ensures the neutral position of the shaft, whose position is measured by a Linear Variable Displacement Transducer (LVDT). The spool, upon its displacement, opens or closes the hydraulic circuit. It is suited for position, velocity, force or pressure control. Its benefits are the lack of a pilot oil flow, low magnetic hysteresis, low current consumption, absence of impurities obstruction issues and better dynamic response than its counterparts. Unfortunately, it is extremely vulnerable to high-frequency magnetic interferences, leading to inappropriate behaviour, leaving preference to other types of servo valves.

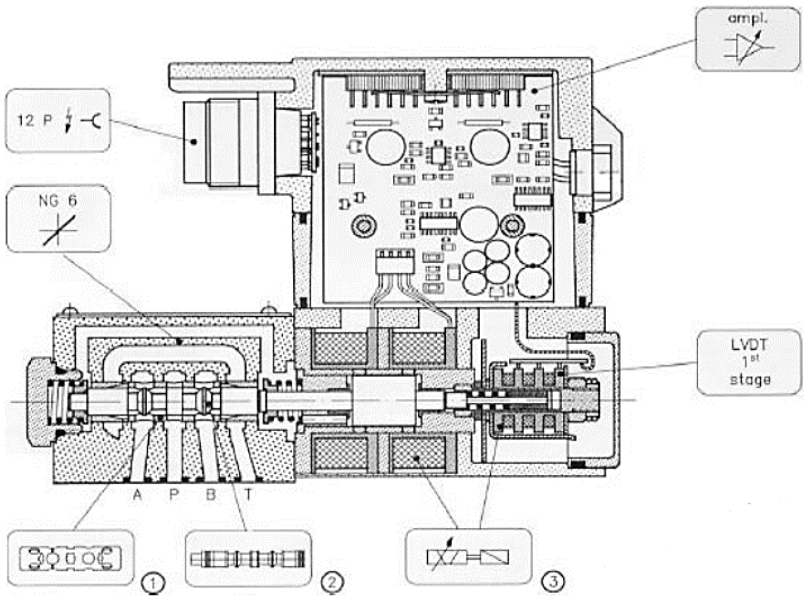


Figure 2.11: Direct Drive Valve section

2.5.2 Jet Pipe valve

The Jet Pipe architecture shows many differences from the Direct Drive Valve. In the first place, the linear motor is replaced with a torque motor. The command signal from the pilot passing through the coils of the motor generates a magnetic torque on the armature (neutrally attracted by permanent magnets), producing a rotation around its fixed point. As it is jointed to a nozzle called jet pipe, it moves too, causing the stream to be redirected to a different hydraulic path. The differential pressure created into the hydraulic chamber displaces its internal spool from the neutral position and opens the hydraulic power circuit. Reproducing this last step, several stages can be replicated, amplifying the output forces on the control surfaces at the cost of higher weight and size of the servo valve. The precision of the valve is guaranteed by an electromechanical transducer, the Linear Variable Differential Transformer (LVDT).

This design is highly vulnerable to obstructions and microscopic particles, therefore needing filters with high contamination removal efficiencies and dirt-holding capacities.

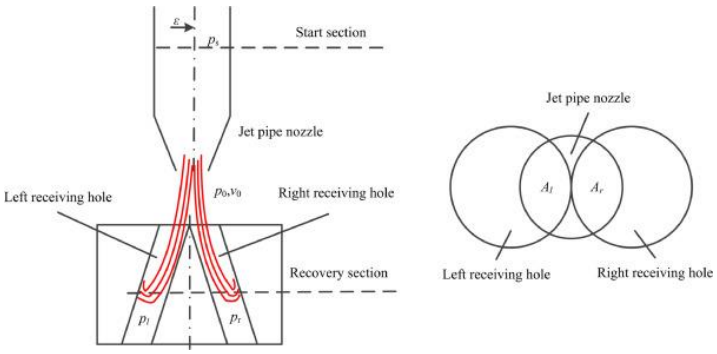


Figure 2.12: Jet-pipe valve lateral section and overhead view

2.5.3 Flapper-nozzle valve

The main difference between this architecture and the Jet Pipe is the presence of a flat rigid surface, called flapper, jointed to the armature of the torque motor. The valve's purpose is similar to the Jet Pipe's, that is to create a differential pressure to the spool extremities. This time the differential pressure will be achieved through partially or fully obstructing the hydraulic flow from two symmetric and horizontal nozzles. Consequently, the spool will displace, opening the second stage hydraulic circuit. The neutral centred position of the flapper is ensured by the presence of a feedback spring, an elastic component attached to the spool with a ball joint. Considering that it's the type of servo valve modelled in this work, it will be in-depth described in the next chapter.

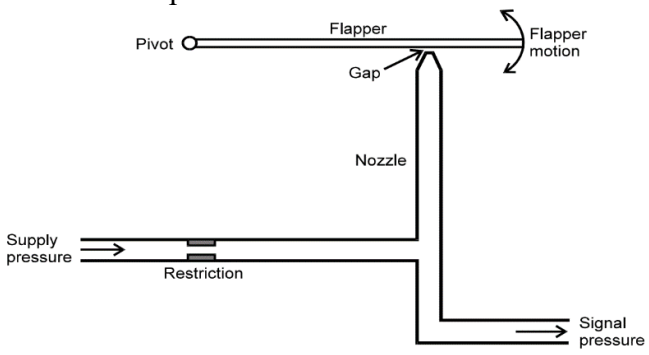


Figure 2.13: Flapper-nozzle system sketch

3 PHYSICAL SYSTEM

The actuation system studied in this work is composed of three subsystems:

- controller subsystem
- electrohydraulic two-stage flapper nozzle servo valve
- hydraulic piston

3.1 Controller subsystem

This system includes the control electronics, which pre-processes the command currents entering in the servo valve. Being part of a closed-loop feedback system means that the output signal from the sensors will be confronted with a reference signal (entered by the pilot), generating an error signal, as illustrated in Figure 3.1. The purpose of the controller is to maintain the error signal as close to zero as possible, enhancing the overall precision of the actuation system. The controller will have a PID architecture.

PID stands for Proportional-Integrative-Derivative, which are the three control terms. Each of them has different purposes.

- The proportional term is used to amplify the current error value through a proportional gain K_p. No error corresponds to no correction.
- The integrative term eliminates the residual error between the reference and the error signal based on its time history. When the steady-state error approaches zero, the integrative contribution fades out. It tends to slow the responsiveness and deteriorate the stability of the system if overused.
- The derivative term acts by the rate of error change, pre-emptively correcting the error and lowering the overshoot values. It works on noise effects, requiring a filter to avoid negative impact on the system.

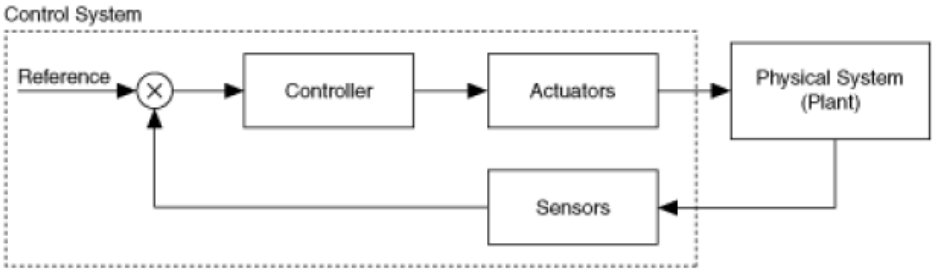


Figure 3.1: Typical closed loop control system

The PID controller can be tuned manually through trial and error, until the system responds as desired, or through the Ziegler-Nichols method, using standard variables (the ultimate gain K_u and the relative ultimate period P_u).

Depending on the needs, the derivative or the integrative terms can be omitted, obtaining a PI or a PD controller. In the current system both terms are absent, considering only the proportional term.

3.2 Servo valve subsystem

The servo valve used is a two-stage flapper nozzle valve [1], which transforms the low power electrical signal into a high-power hydraulic flow.

The first stage (called piloting stage) connects a torque motor to a flapper. The second stage uses the flapper movement to generate a differential pressure around a spool.

The torque motor is composed by a permanent magnet (the frame) which contains a T-shape armature wrapped in coils. The electrical currents, entering the coils from the controller, generate a magnetic torque that causes the armature to twist and the flapper (jointed to the armature) to bend from its neutral position. A feedback spring ensures that the armature and the flapper return to its original location when no current is applied.

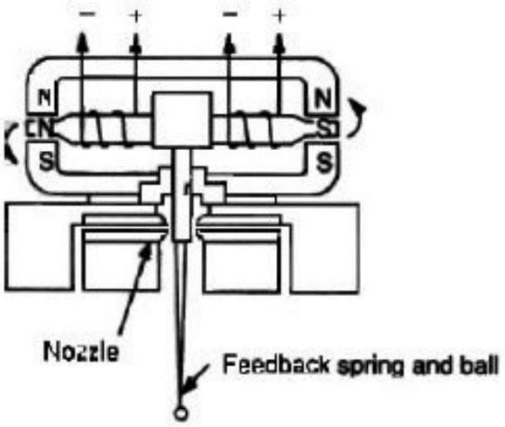


Figure 3.2: Flapper nozzle servo valve piloting stage

When the flapper moves, it obstructs one of the two nozzles of the second stage. Inside the nozzle there is the pilot pressure fed from the hydraulic circuit. As soon as one of the nozzle closes, the pressure upstream of the other increases.

The resulting differential pressure displaces the spool from its neutral position, guaranteed by a null adjustment screw, opening other hydraulic ports, as showed in Figure 3.3.

The components are very critically sized to work effectively, as any contamination in the fluid, feedback spring ball wear and nozzle damage can reduce drastically the accuracy of the valve.

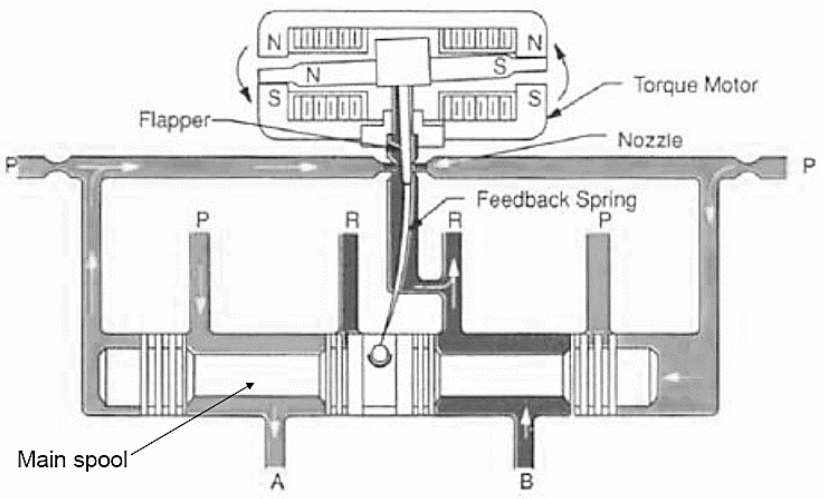


Figure 3.3: First and second stage of the servo valve in function

3.3 Hydraulic piston subsystem

The hydraulic ports, opened or closed by the spool displacement, act on the sides of a bigger actuator, a hydraulic piston or jack, which is directly connected to the control surface through a mechanical transmission. It is a double-acting symmetrical cylinder, equipped with a transducer which precisely tracks its displacement in time, feedbacking it to the reference signal and allowing the controller to reduce incoming position errors. Figure 3.4 shows a representation of the whole system.

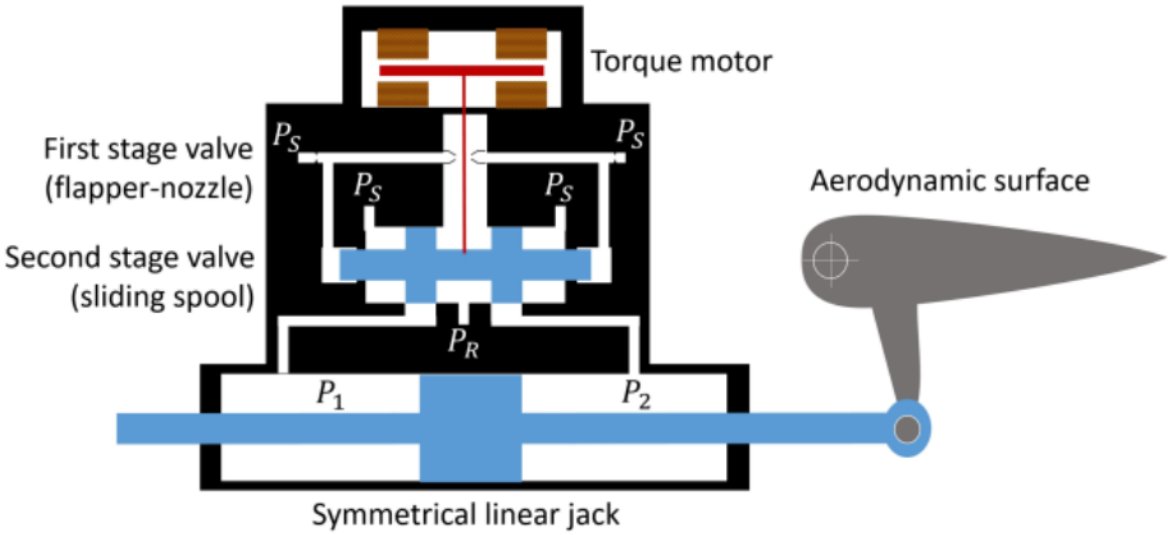


Figure 3.4: Schematic representation of the whole actuation system

3.4 Servo valve characteristics and non-linearities

The servo valve operations are influenced by some design characteristics which can considerably alter its functioning. Most of these are determined during the design phase, others are inevitable degradations which appear after reiterated usage. Before introducing the numerical model, it is important to name and describe these phenomena to be able to model them correctly.

3.4.1 Spool condition

The spool of the servo valve can be found in three different conditions:

- Over-lapped
- Zero-lapped
- Under-lapped

The over-lapped condition requires that the spool covers more space compared to the hydraulic port. Consequently, the spool can hold higher pressures without fluid leaks. The lap percentage is generally not more of 20% of its length. This value can decrease drastically, because an excessive overlap condition creates a slow response from the valve, which is generally not desired.

The zero-lapped condition is the optimal state for the spool. It ensures perfect matching dimensions between the spool and the hydraulic ports, with a near instantaneous response from

the valve. It is the standard condition for most servo valves, but unfortunately it is not reliable, as most zero-lapped valves tend to become under-lapped with usage and wear over time.

Under-lapped valves show smaller dimensions for the spool than for the hydraulic port. It generally implies more responsiveness, but it is not reliable to when high hydraulic loads are applied.

In Figure 3.5 there is a graphical representation of the three conditions. In Figure 3.6 the mathematical representation on the Q-I axes (flow rate – input current).

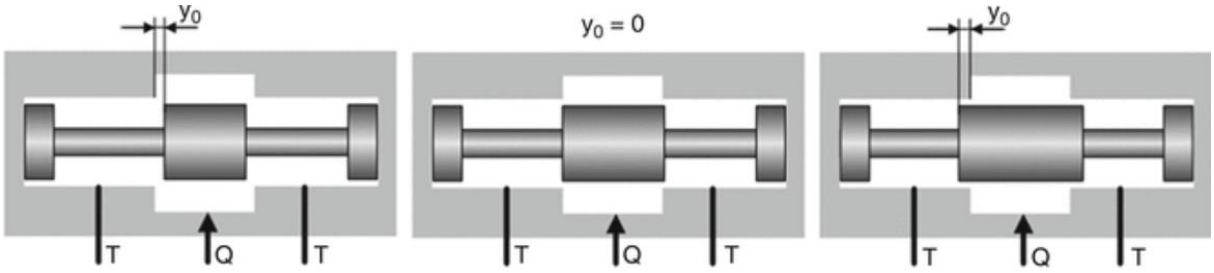


Figure 3.5: Under-lap, Zero-lap and Over-lap conditions

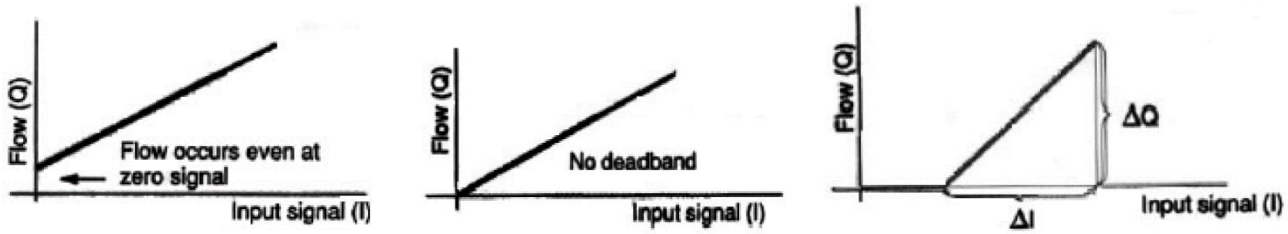


Figure 3.6: Under-lap, Zero-lap and Over-lap plots

3.4.2 Magnetic hysteresis

Hysteresis is a non-linear phenomenon that occurs when applying a magnetic field to a ferromagnetic material. When the magnetic field is removed, the material remains partly magnetized because of the orientation the magnetic domains have obtained. The magnetization curve changes depending on the studied material's characteristics. Its behaviour is non-linear and can change depending on external variables such as temperature. Figure 3.7 shows a typical magnetic hysteresis curve (on the vertical axis *B* represents the magnetization, on the horizontal axis there is the applied field strength *H*). The asymptotes (which represent magnetic saturation), the upward initial magnetization curve and the downward return magnetization

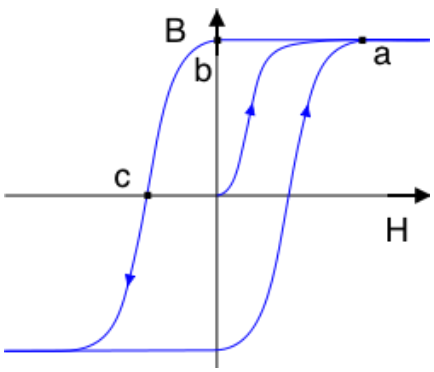


Figure 3.7: Theoretic magnetic hysteresis loop

curve form the main loop curve. This phenomenon is particularly visible in the torque motor, inside the first stage of the servo valve, whose remaining currents must be considered.

3.4.3 Threshold

The spool is subject to friction force inside the servo valve sleeve. Considering that the static friction force acting on the spool is always greater than the sliding friction, the spool may not move when excited by low pressure. For this reason, a small-frequency signal (called dither) with 60-400 Hz frequency and 0-5% amplitude of the command is applied on the spool and superimposed to the main signal, maintaining the spool in constant motion. It is important that the jack stays in its neutral position to not alter the performance of the system. If no dither signal is applied to the system, there is the possibility that the spool does not move until a certain threshold has been overcome.

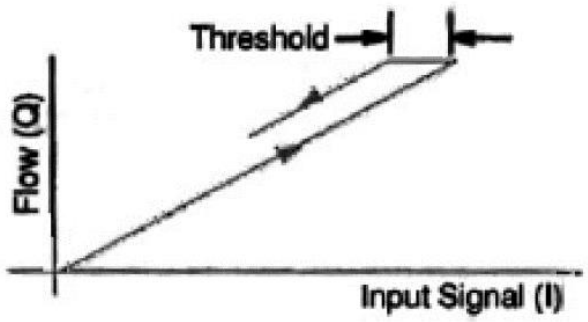


Figure 3.8: Threshold in the Q-I plot caused by static friction

3.4.4 Linearity and symmetry

When studying the characteristic of a servo valve, it is not unusual to see differences between the studied and the ideal behaviour. The actual flow can behave in tricky ways that cannot be studied precisely until field tests are performed.

Also, when inverting the direction of displacement (thus inverting the flow), the spool may act differently from the previous command.

These two conditions are called linearity and symmetry and have to be considered when analysing the model behaviour.

In Figure 3.9 a confrontation between actual and ideal flow is presented.

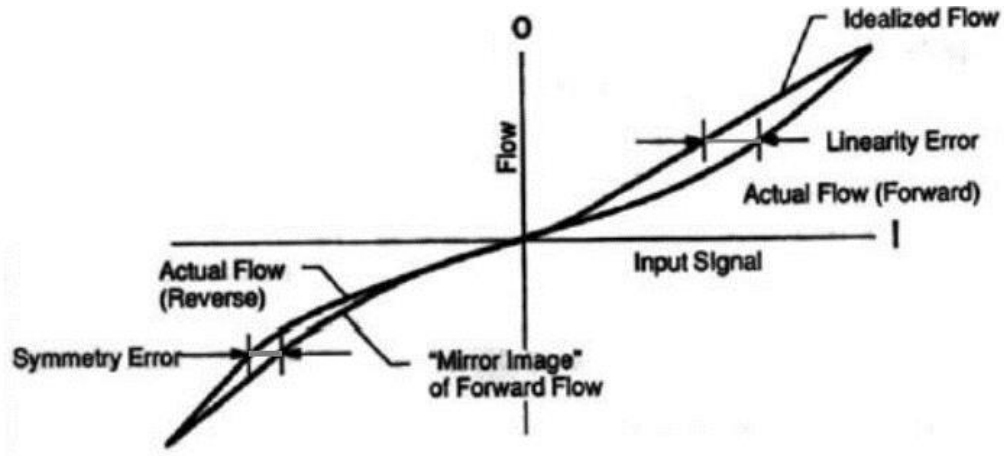


Figure 3.9: Example of deviations from ideal linear and symmetric behaviour

3.4.5 Wear

Due to the amount of small and large movements alike the mechanisms must do during the system's overall life span, it is common to stumble into wear problems. They are caused by abrasion between moving parts, unfiltered particles in the fluid, fatigue over long cycles of operation and sometimes corrosion. It can cause a decrease in accuracy and overall performance, as well as unpredictable behaviour. The most notable ones are:

- nozzle wear
- spool edges rounding
- play between feedback spring ball joint and spool's sleeve (backlash)

When possible, it is helpful to model these conditions to correctly predict the system behaviour even after its initial usage cycles.

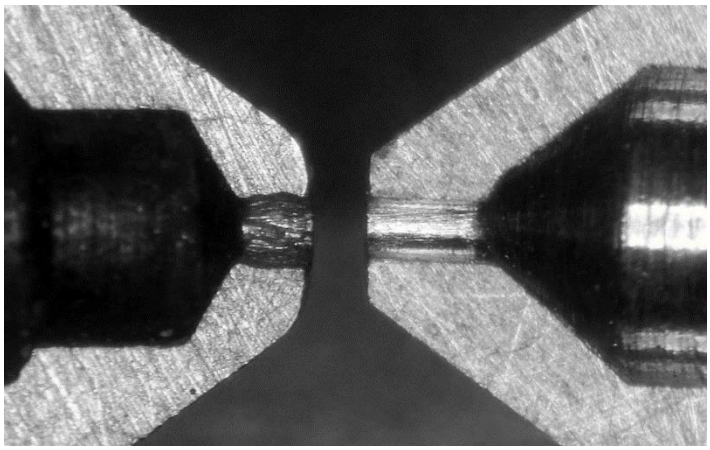


Figure 3.10: On the left, an example of nozzle wear

3.5 Friction models

The friction is one of the most difficult phenomena to study and model correctly. It severely restricts control system performances, acting on most moving parts, especially in complex systems. To improve the precision of the output of the mathematical model, a correct modelization and analysis is needed. As friction is an extremely non-linear phenomenon and is highly dependent on the material used and the ambient variables. It is generally analysed through approximations obtained from empirical evidences.

In time, many friction models were designed and exploited. The most notable are:

- Coulomb friction
- viscous friction
- Stribeck effect
- saturated hyperviscous friction
- Karnopp friction
- Borello friction

3.5.1 Coulomb friction model

The Coulomb friction is probably the most well-known (and the simplest). It differentiates the steadiness condition from the dynamic one. When the velocity is null (DXJ), the static friction varies between two large values (FSJ and $-FSJ$). When the system is moving, the dynamic force is constant (FDJ and $-FDJ$), but lower than its static counterpart. The static condition is difficult to model because of the presence of a jump discontinuity in the origin.

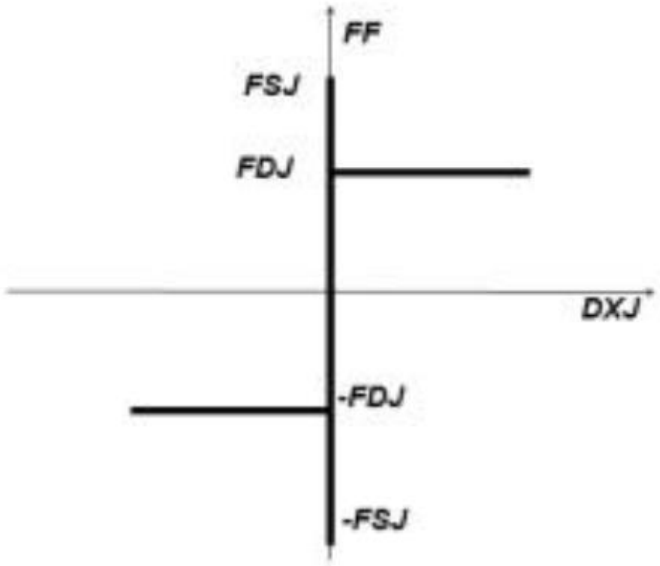


Figure 3.11: Coulomb friction model

3.5.2 Viscous friction model

Similarly to the Coulomb model, the viscous friction model considers the dynamic friction force not constant anymore, but proportional to the sliding speed of a body inside a viscous fluid. It has the same deficits of the Coulomb friction, that is the jump discontinuity in the static condition.

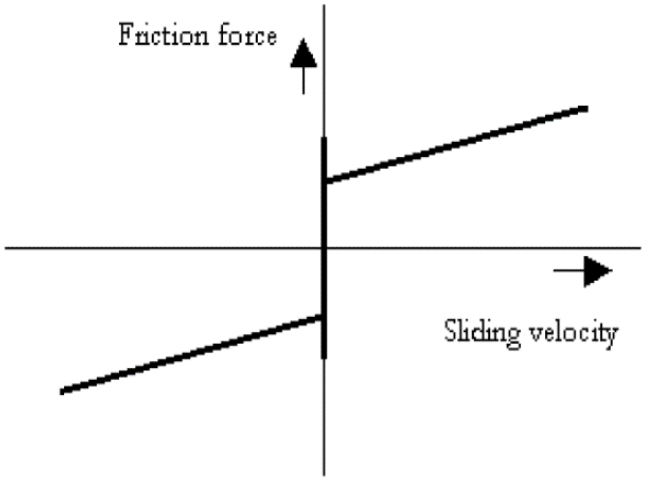


Figure 3.12: Viscous friction model

3.5.3 Stribeck effect

The Stribeck effect (named after its finder) considers the continuous decrease in friction force at low velocities before its increase at higher velocities. Having less discontinuities near the origin results in a more straightforward model for simulations with large loads, but still not useful in static studies.

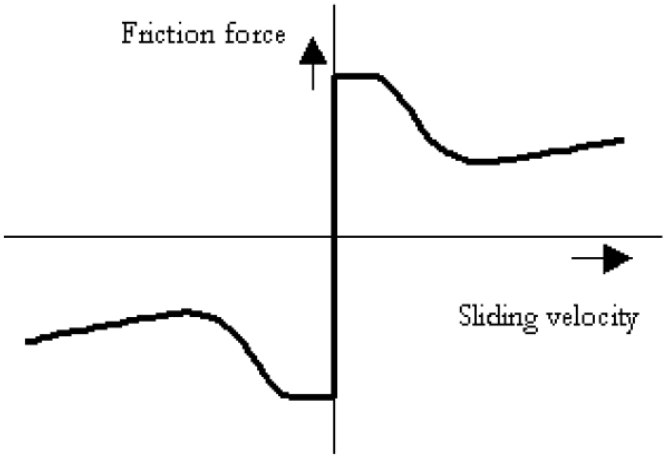


Figure 3.13: Stribeck effect model

3.5.4 Saturated hyperviscous friction model

The saturated hyperviscous friction model has been created to overcome the jump discontinuity near the origin. On its behalf, there is a straight line whose slope is called viscous coefficient. Its half-length projection on the horizontal axis is named ε and it is an important variable for its design. Like in Coulomb friction, the dynamic friction force is considered constant. As shown in Figure 3.13, the axis h represents the load in the direction of the motion (in this case, constant with the sliding velocity).

The model has inevitable limits. The most notable one is the null value of static friction in the origin, with $v = 0$. It means that the friction is not able to contrast external forces (regardless how little they are), inducing movement. This phenomenon is called creep. This is a hint of how the model is unable to discriminate the adherence condition from the dynamic condition, to stop the object and to maintain it stationary despite the presence of external forces.

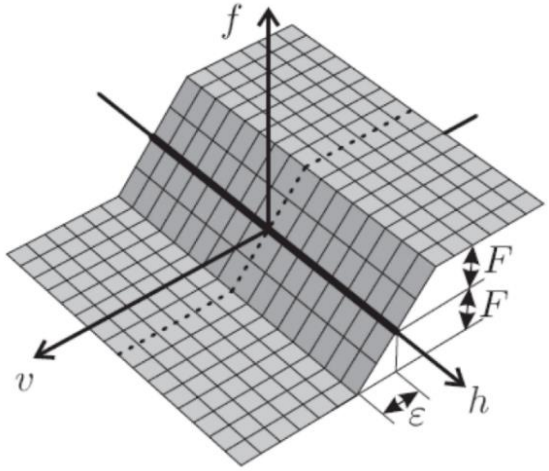


Figure 3.14: Saturated hyperviscous friction model

3.5.5 Karnopp friction model

To overcome the zero-velocity issue in the static condition, Karnopp developed a model which considered a "stick zone" around the origin, with length equal to the variable ε (dead band) and height equal to the static friction maximum force.

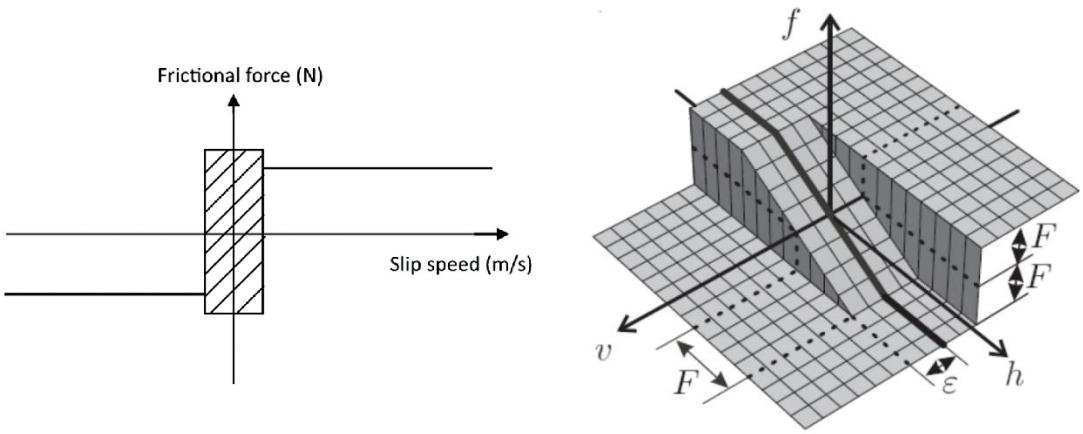


Figure 3.15: ε dead band and Karnopp friction model representations

This "window" allowed the study of the stick-slip phenomena, which happens when there is a sudden and repeated transition between stasis and dynamic condition. This model is much more complex than the previous ones. The dead band must be chosen carefully to avoid dangerous limit cycles and the overall numerical stability of the system.

3.5.6 Borello friction model

As of today, the Borello friction model is one of the most comprehensive ever developed.

It is easily implementable in complex numerical systems and is able to:

- distinguish the adherence from the dynamic condition;
- evaluate the sign-value of the friction force depending on the velocity direction;
- determine the object halt from the dynamic condition and the restart from the static condition;
- correctly maintain the object in motion or stationary.

In addition, the model can acknowledge the presence of limit stops in case of completely inelastic collision. As shown in Figure 3.16, it completely removes the dead band from the Karnopp friction model, considering instead a hyperviscous friction model in static conditions. This is the friction model used in the current work.

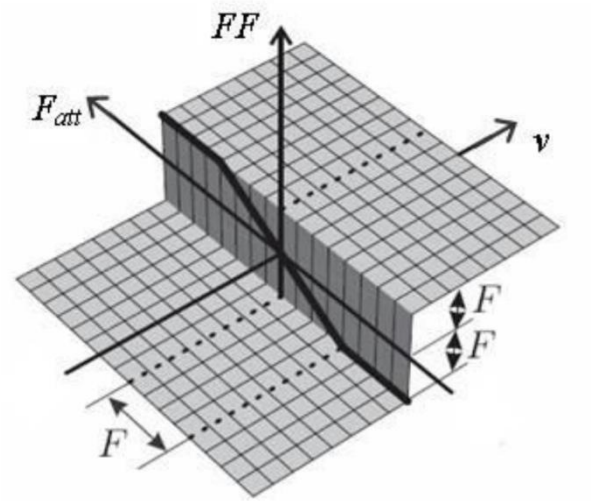


Figure 3.16: Borello friction model

4 COMPUTATIONAL MODEL

Simulating the behaviour of such a complex environment can be difficult using common analytical methods. For this reason, computational and numerical models have been developed through the years to help with non-linear phenomena and complex systems analysis, which can be very helpful when searching for accurate results. The computational model is designed to replicate the behaviour of the physical model through mathematical relations in the most accurate way possible.

The model of this work has been created via Matlab/Simulink (ver. 2016a), a powerful modelling, simulation and analysis environment for dynamic systems. It uses an intuitive block diagram graphical representation of the variables relations. Each block can represent an input, a command, an equation, an operator, a component or a subsystem, starting from an input (in our case, the pilot command) and ending with an output (the position of the aerodynamic surface linkage connected to the actuator jack, called utilizer). Most of the block names and comments of the model here presented are in Italian.

Two fundamental models have been developed: the detailed model (whose purpose is to simulate the behaviour of the physical model with different inputs and faults as precisely as possible), and the monitoring model (which is a simpler and lighter model used to iteratively compare its results with those from the detailed model, and then proceed with the prognostic analysis). The parameters and constant are contained in the script $EHA\_DAT.m$ which needs to be run before the Simulink model. The script can be found in Appendix A.

Therefore, we can subdivide the entire system in seven different subsystems:

- Command block
- PID controller block
- Torque motor model
- Electromechanical servo valve model
- Fluid dynamics servo valve model
- Mechanical jack subsystem
- Monitoring model

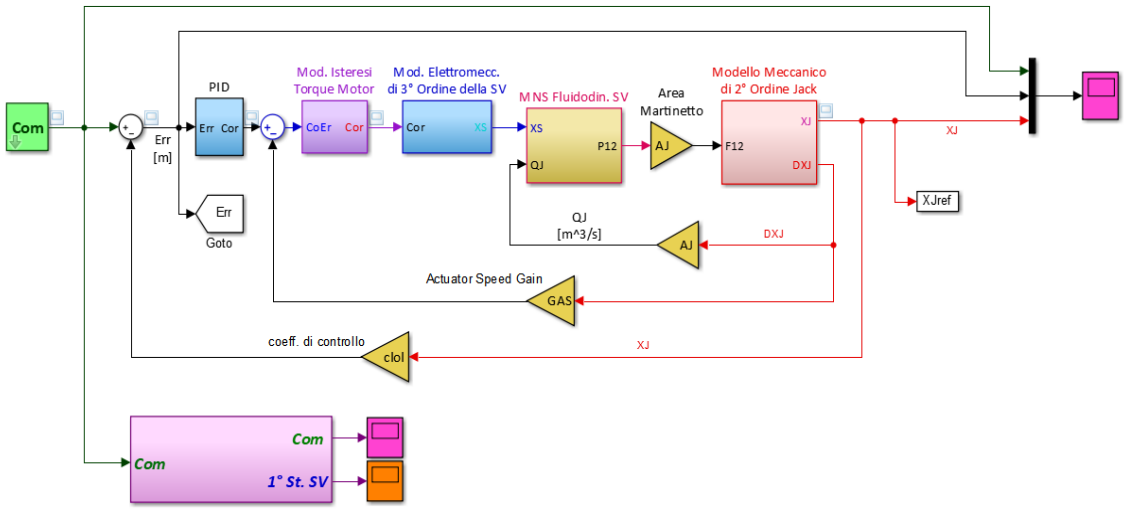


Figure 4.1: The whole actuation system in the Simulink environment. The coloured rectangles represent the subsystems

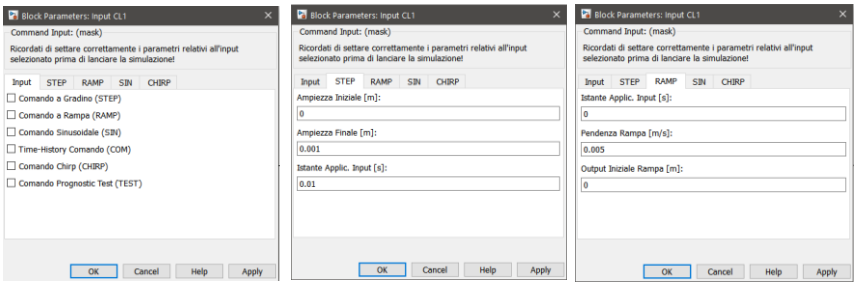
4.1 Command block

The green block on the far left of Figure 4.1 has a mask, a custom interface that allows to decide the pilot's command based on mathematical variables. This is one of the most important blocks, as it decides the input on which the system is going to operate and the output desired. There are different commands to choose from, and each one has its own shape and function:

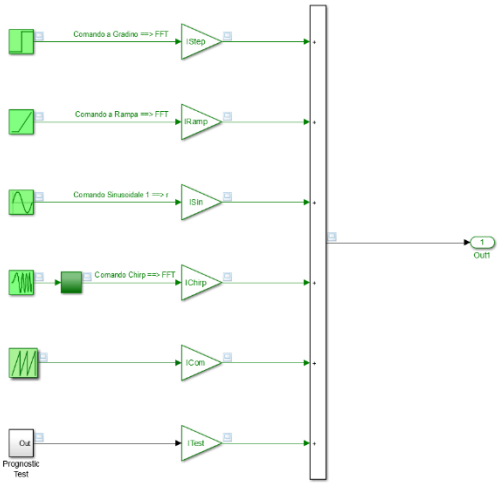
- Step command (STEP)
- Ramp command (RAMP)
- Sinusoidal command (SIN)
- Time-History command (called COM, Time-Com or Mixed)
- Chirp command (CHIRP)
- Prognostic test command (TEST)

Each one of those is completely adjustable through the mask or directly from the model blocks, as shown in Figure 4.2.

(a)



(b)



(c)

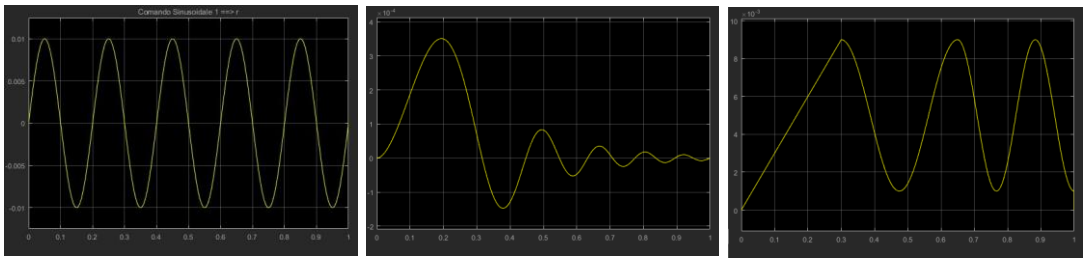


Figure 4.2: (a) details of the command mask; (b) command subsystem signals; (c) from left to right: sine wave command, decrescent amplitude chirp command, mixed command

4.2 PID controller block

As discussed in chapter 3.1, the controller's purpose is to reduce the error between the output signal XJ (the position variable of the jack) and the reference signal (the output from the green block Com). Its structure is shown in Figure 4.3.

GAI, GAP and GAD represent the three PID gains, whose value determines how much impact the integrative, the proportional or the derivative terms will have on the error signal. In this work GAI and GAD are null. The pink block is connected to a switch. Clicking on it will give a constant unit output to the next subsystem, bypassing the PID block.

In output of the subsystem there is the current value in mA, which is compared to the jack velocity times the feedback velocity gain (GAS, located outside of the subsystem), resulting in a current error, CoEr. This signal enters the next subsystem, the torque motor.

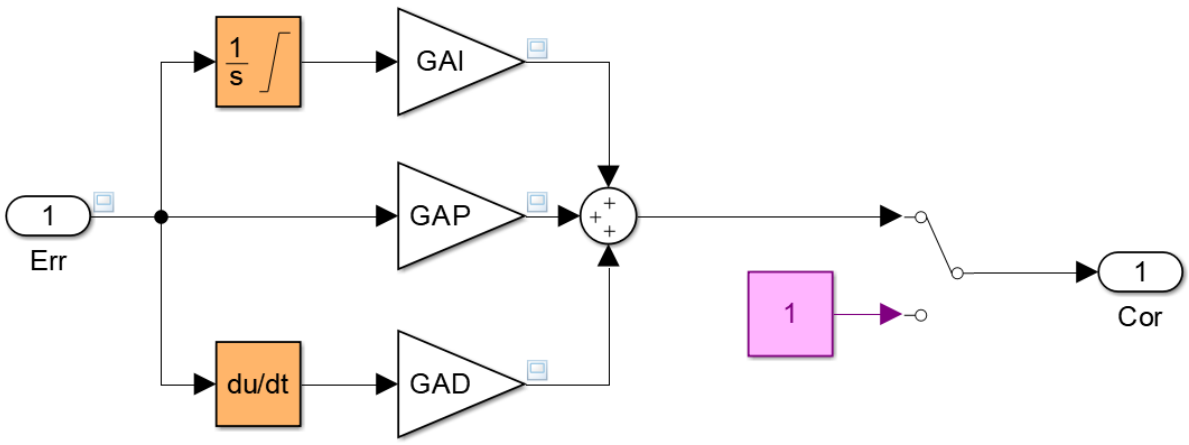


Figure 4.3: PID subsystem block structure

4.3 Torque motor model

The $CoEr$ signal is the error signal of the current. The torque motor subsystem sums the constant offset current value Ofs (which occurs when there are residual currents in the torque motor) and the hysteresis current "His" (whose phenomenon has been described in Paragraph 3.4.2).

As we can see from the signals shown in Figure 4.4, the output value "Cor" will have a value ranging between $+CorM$ and $-CorM$, respectively the upper and lower limits.

The value is calculated as the minimum between:

- $CoEr + Ofs + Hys$
- the maximum value between Cor and $CoEr - Hys + Ofs$
- the maximum input value in the servo-valve

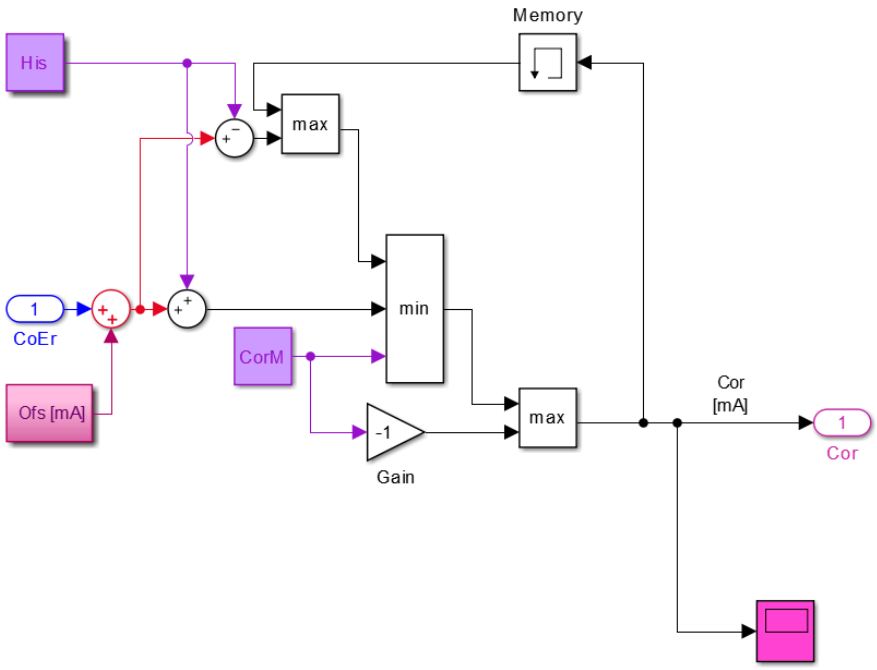


Figure 4.4: Torque motor subsystem block

4.4 Electromechanical servo valve model

The electromechanical servo valve model analysed in this paragraph is a 3rd order system, subdivided into two subsystems. The first is a 2nd order dynamic system which analyses the first stage of the servo valve (flapper and feedback spring). The second is a 1st order dynamic system which takes into account the spool behaviour (the second stage of the servo valve) and its friction model.

We will study those subsystems separately.

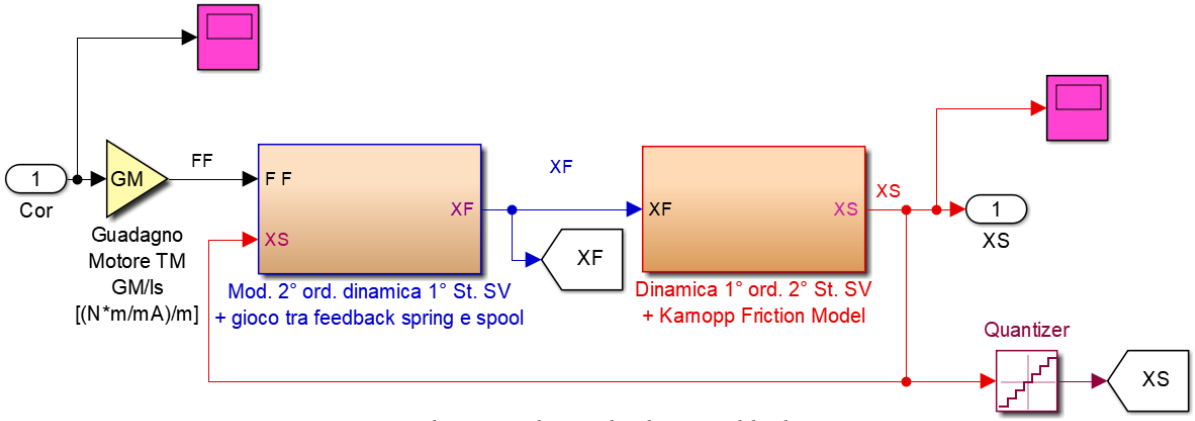


Figure 4.5: Electromechanical subsystem block

4.4.1 First stage of the servo valve

The input of this subsystem is the current Cor, which is the output of the torque motor. Its value times the torque motor gain gives the force acting on the flapper (FF). Starting from it, it is possible to find the acceleration of the flapper and, integrating two times, its velocity and position.

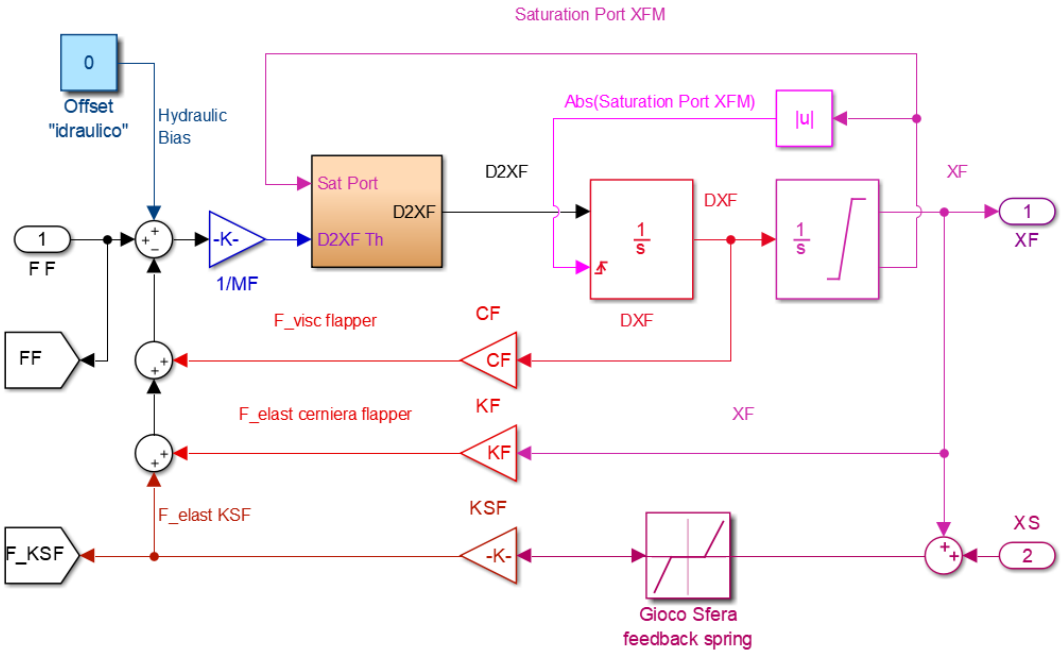


Figure 4.6: Servo valve first stage block

Notice the saturation in the last integrator to simulate the end-stroke. The flapper force is counterbalanced by three forces:

- a viscous force, proportional to the flapper velocity (for low speeds)
- an elastic force proportional to the flapper displacement (related to the torque motor joint)
- another elastic force proportional to the flapper displacement (related to the spool joint as a feedback spring ball)

The backlash block is able to model the mechanical play between the feedback spring ball and its compartment in the spool.

4.4.2 *Second stage of the servo valve*

Considering the limited dimensions of the spool, it is possible to use a first order dynamic system to model its behaviour. As previously mentioned, fluid contamination with small particles can deteriorate the system's performance, so a filter is necessary. To model the progressive damage, we can use new variables – u and K_{intas} – defined as follows:

$$u = 1 - K_{intas}$$

with $0 < K_{intas} < 1$. If $K_{intas} = 0$ the filter is healthy, if $K_{intas} = 1$ the filter is completely jammed.

These variables are important because they influence the gain pressure value GPF, which is used to determine the pressure acting on the spool.

The differential pressure P_{12} is evaluated by multiplying the flapper position with the pressure gain. When multiplied with the section area, the active force on the spool, called F_{att} is found.

The next equation is able to describe the dynamics of the spool and the forces operating on it through a first order ODE:

$$\dot{XS}\left(C_s + ASV^2 \frac{GPF}{GQF}\right) = ASV \cdot GPF \cdot XF - FF$$

In Figure 4.7 the Simulink structure that originates from it:

The friction type used for the spool is the Borello model, described in Paragraph 3.5.6.

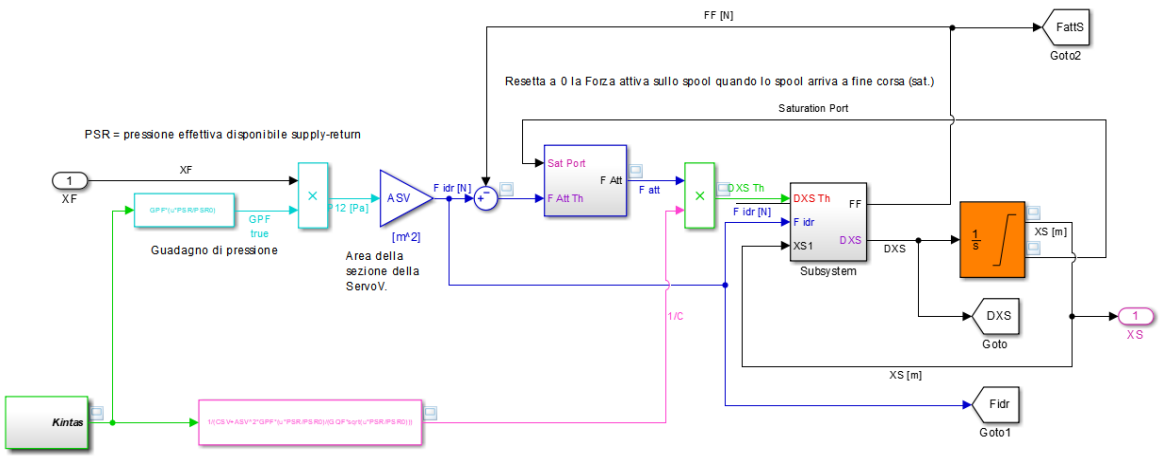


Figure 4.7: servo valve second stage block

Figure 4.8 shows its numeric model:

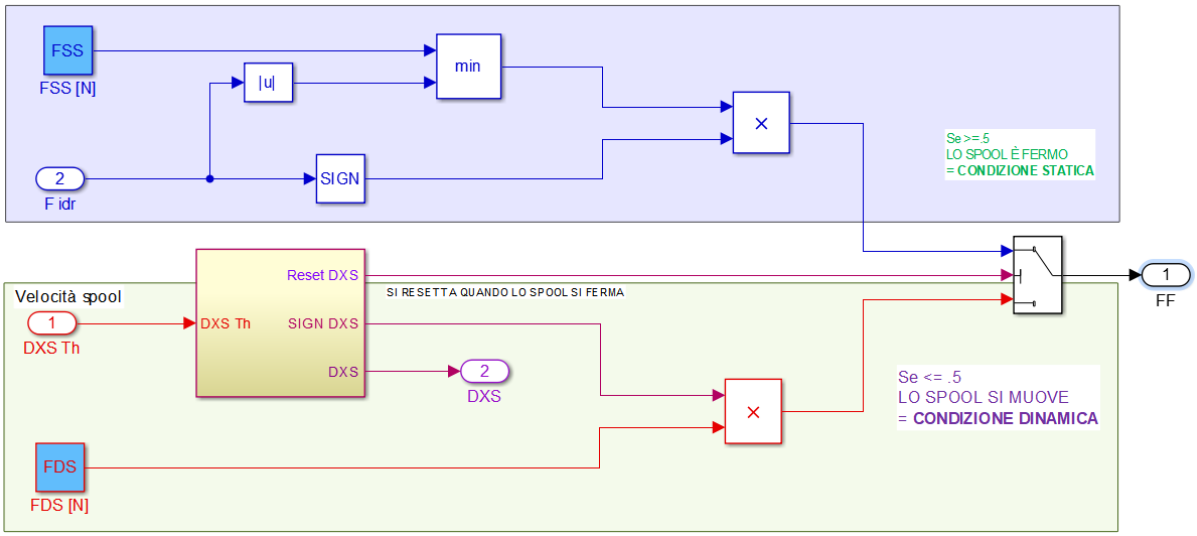


Figure 4.8: Borello friction model

The upper part represents the static condition, the lower part the dynamic condition. The model is able to discriminate between the two of them judging from the spool velocity behaviour, resetting itself when the spool is still (and its velocity null).

The spool displacement value is then digitalized by the last block of the electromechanical subsystem, a quantizer.

4.5 Fluid dynamics servo valve model

The spool displacement opens other hydraulic ports, creating another differential pressure on the jack. Many models with different accuracy levels have been developed. The one studied in this work uses a different – non-linear – approach. The equation used to find the value of the effective differential pressure P_{12} is:

$$P_{12} = \frac{PSR \cdot XS - GPQ \cdot XSS \cdot QJ}{\max(|XS|, XSS) + GPQ \cdot XSS \cdot CL_k}$$

where:

- PSR is the effective differential supply-return pressure
- XSS is the saturation aperture of the spool and defined as $XSS = \frac{PSR}{GP}$
- GPQ is defined as the ratio between pressure gain and flow rate gain $\left(GPQ = \frac{GP}{GQ}\right)$
- QJ is the flow rate on the jack
- CL_k is the leakage coefficient of the fluid

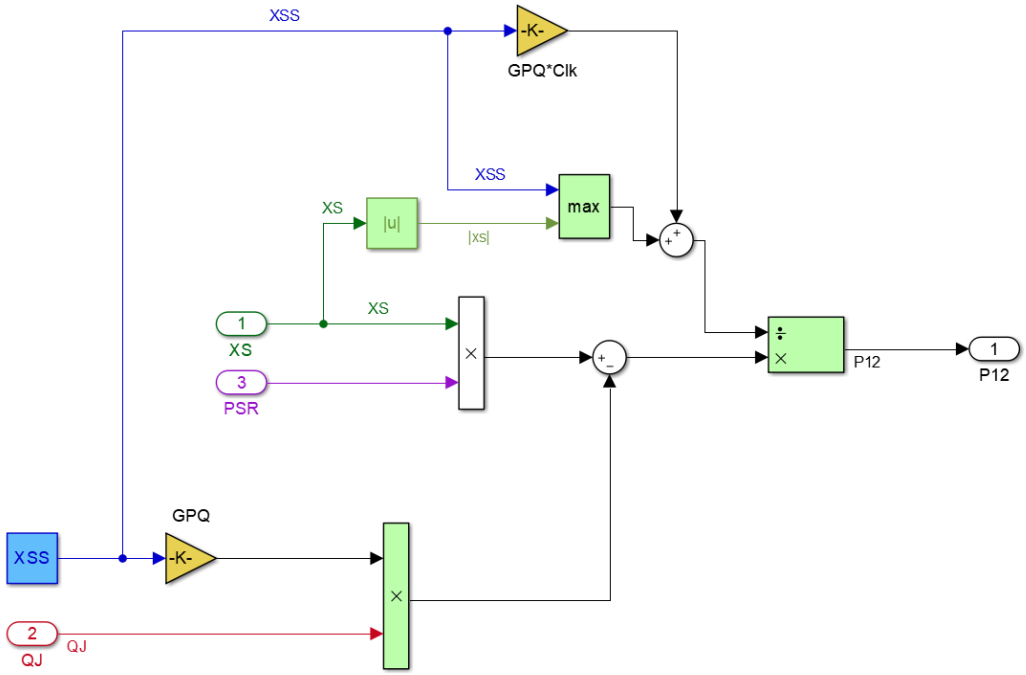


Figure 4.9: Fluid dynamic servo valve subsystem

The previous equation derives from a basic differential pressure balance formula:

$$P_{12} = P_{12P} - P_{12Q} - P_{12LK}$$

where:

- P_{12} is the effective differential pressure agent on the jack;
- P_{12P} is the differential pressure created by the hydraulic ports;
- P_{12Q} is the pressure lost due to the jack draining;
- P_{12LK} is the pressure lost due to leakages.

4.6 Mechanical jack subsystem

By multiplying the previous output P_{12} with the jack section area, we find F_{12}, the hydraulic force acting on the jack. This is the input of the last subsystem of the detailed actuation model, whose purpose is to model the jack's dynamics.

This is the subsystem in which most of the fault analysis and the prognostic analysis of this work take place because of the two faults reproducible by variables presented in the blocks: the backlash of the transmission between the jack and the utilizer, and the friction agent on the jack.

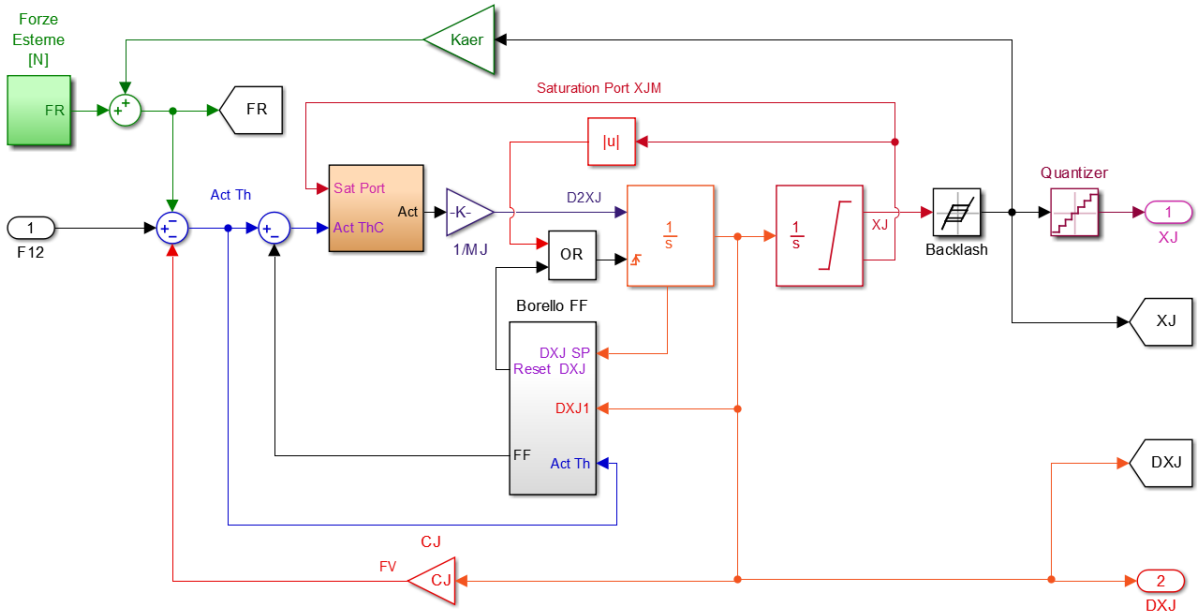


Figure 4.10: Mechanical jack subsystem

To find the actual force on the jack we must subtract the forces opposing the motion from those generated by the differential pressure. These are:

- external forces
- hinge moment
- viscous forces
- friction

The friction model is again a Borello model, like in the spool subsystem.

After discovering the actual force acting on the jack (with an end-stroke saturation), the signal is divided by the mass of the jack, revealing its acceleration. Eventually, the signal passes through two integrators, giving the output signal XJ, that is the jack displacement. The backlash (used to model a mechanical play in the transmission line between the jack and control surface) finally reveals the output signal, which is then linked to the initial controller through a feedback gain, generating an error and closing the loop.

4.7 Monitoring model

The monitoring model has been developed to reduce simulation times compared to the detailed model at the cost of reduced accuracy. The macro-structure is very similar to the detailed one, as shown in Figure 4.11, but the models used to describe its dynamics are much simpler.

Below there is a brief discussion about the differences.

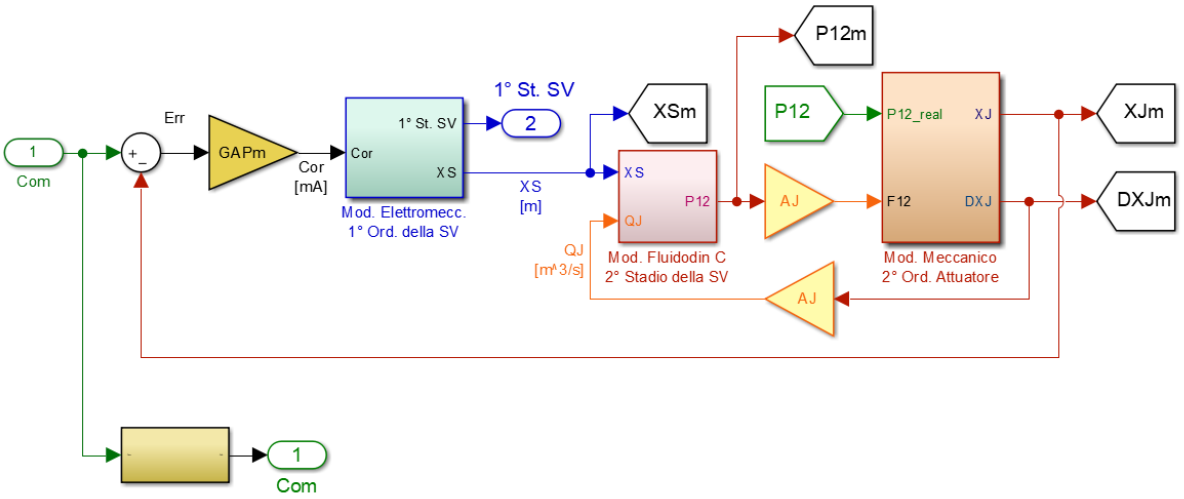


Figure 4.11: Monitoring model macro-structure

The PID controller is not present.

Some non-linearities of the torque motor have been neglected, and the electro-mechanical model has been downgraded to a 1st order system by removing inertial contributions.

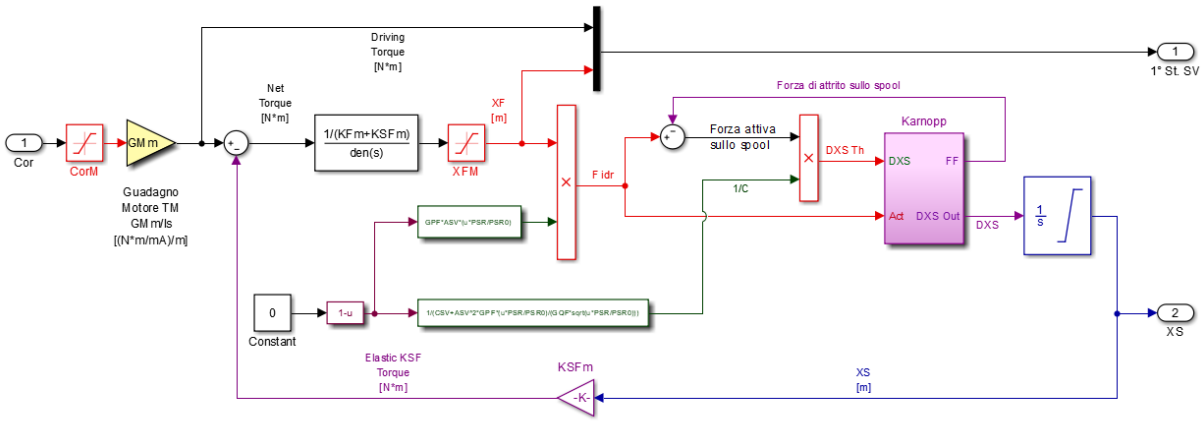


Figure 4.12: Torque motor and electro-mechanical subsystems of the monitoring model

The friction model we used is the Karnopp model, discussed in Paragraph 3.5.5 and here presented:

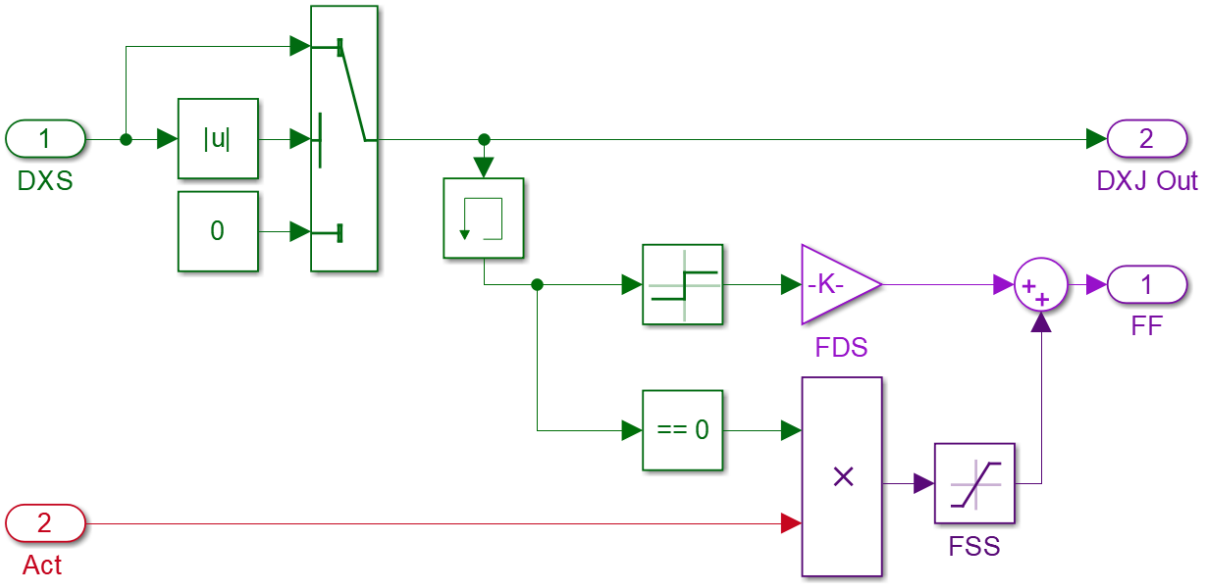


Figure 4.13: Karnopp friction model

If the signal exceeds the threshold (based on the chosen dead band ε), the system is considered in motion. If not, then the spool velocity is null and the static friction is calculated on the active force agent on the spool, with an upper and lower limit (saturation).

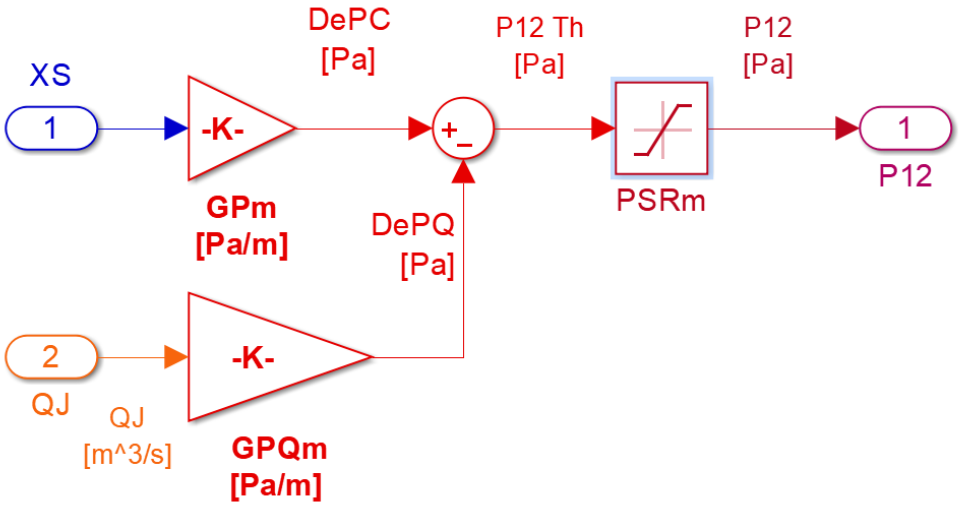


Figure 4.14:Fluid dynamic subsystem of the monitoring model

The fluid dynamic model is the most simplistic one, as it only considers spool position and flow rate with relative gains to determine P_{12}, based on the next equation:

$$P_{12} = G_P \cdot \left[x_s - \frac{Q}{G_Q} \right]$$

The mechanical jack subsystem remains somewhat unaltered, besides the friction model, which is again the Karnopp model.

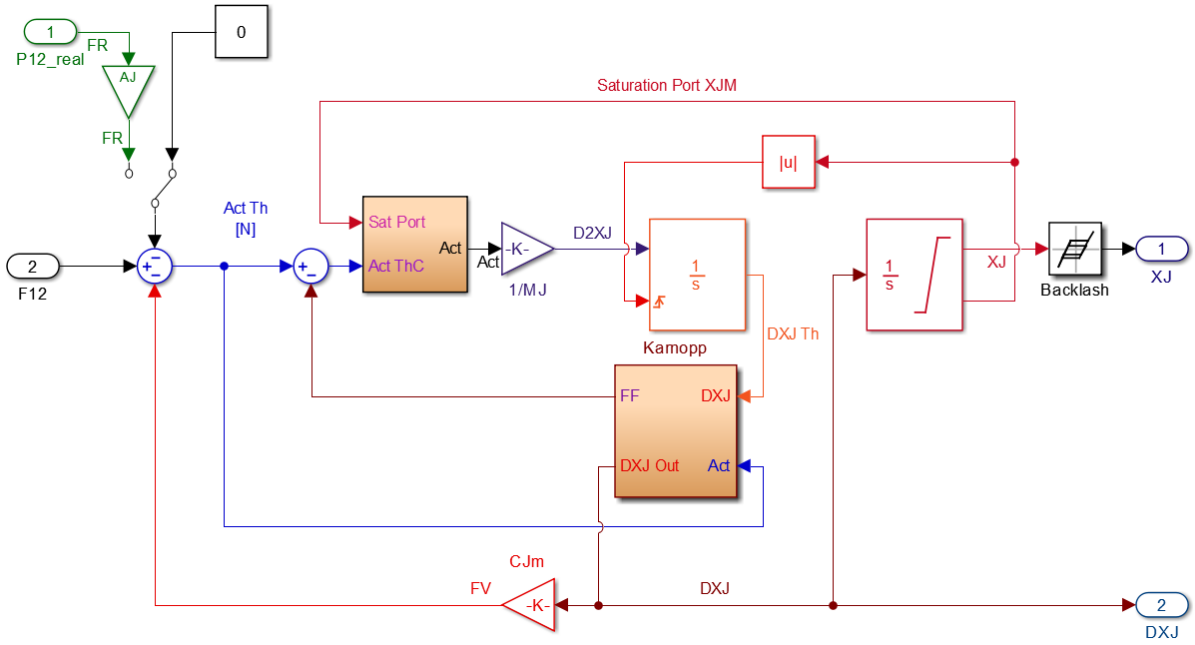


Figure 4.15: mechanical jack subsystem of the monitoring model

5 FAULT ANALYSIS

In everyday life, all types of servo mechanisms experience faults. Differently from critical failures, they do not hinder the mechanisms' ability to complete their work, but they can seriously degrade their performances and reliability over time, making them less robust. Electrohydraulic actuators are definitely crucial, considering that their main job is to move the primary flight controls, which are pivotal for keeping a plane safe and operative. [2]

In the next paragraphs some examples and case studies are presented:

- nominal conditions analysis with lack of faults, to judge the effect of different commands on the response of the model and the minimal differences between the detailed and the monitoring model due to the contrasting design choices;
- individual fault analysis, to evaluate how the detailed model reacts to the presence of simple faults.

In both analyses, the coefficients of the monitoring model have not been altered. Because of that, the low-fidelity model can be used as a reference to the ideal nominal conditions.

5.1 Fault modes

Some of the most common fault modes of EHA are:

- clogged filter of the flapper nozzle servo valve;
- worn nozzles of the first stage of the servo valve;
- demagnetization of the torque motor;
- radial play between the spool and its sleeve;
- strain of the elastic feedback spring;
- wear of the feedback spring ball and consequent backlash;
- backlash in the transmission linkage between the jack and the control surface;
- increasing friction on both the spool and the jack.

In the current thesis, only the last two fault modes are studied.

5.1.1 Backlash

The backlash fault can present itself when high loads over an amount of time are applied to a gearing or a transmission. The result is the wear of the structure edges or joints, which culminates in a clearance (or play) between mechanical contacts. Some gears need a small amount of play to allow motion, and are therefore regulated by tolerances based on the intended use. When the tolerances are exceeded, the system can experience a delay in response, a descrease in its precision or dynamic instability, leading to eventual limit cycles, as seen in Paragraph 4.6. Its amount in the analysis is considered ranging between 0.1 and $1\ mm$.

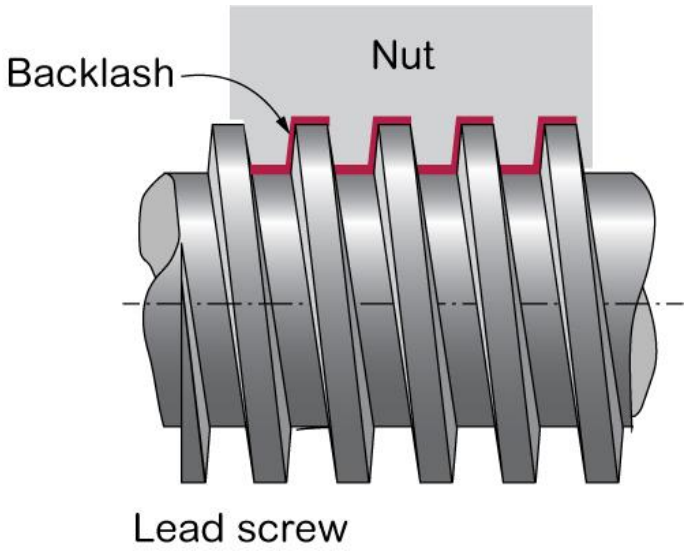


Figure 5.1: backlash in a linear screw mechanism

5.1.2 Friction increment

The increment of friction between the jack and its sleeve can have multiple causes, such as the increment of the lubricant oil viscosity (caused by the temperature change), or the presence of debris and microparticulate.

Friction is defined as a force that opposes to motion. As the friction grows, more pressure is needed to move the jack, increasing the loads on the servo valve, therefore increasing the wear and degradation of the performances of the other components. The value of the dynamic friction (the force acting on the moving object) ranges between 200 and $6400\ N$, the latter being an extreme condition very near to its stall torque, therefore rarely reached.

A detailed description of the friction representations used in the model can be found in Paragraph 3.5.

5.2 Nominal conditions analysis

Nominal condition analysis allows an evaluation, with no considered fault, of the differences in the dynamic responses between the high and the low fidelity models as a validation of the quality of the latter. To ensure that the monitoring model has been modelled correctly, even with major mathematical simplifications, no significant difference should be found. Otherwise, model adjustments should be considered.

For the nominal condition analysis, many inputs from the mask of the model were utilized to test the responses in many circumstances:

- 1 mm step command;
- 1 cm step command;
- ramp command;
- sine wave command;
- decreasing amplitude chirp command;
- mixed command (ramp and double sinusoid).

Each input is represented by two plots, (a) and (b).

The purpose of the first plot is to show how the most important variables detected in the detailed (or high fidelity) model change over $1\ s$, namely the simulation time.

The second compares some variables from the detailed and the monitoring model. In some plots, the time span could be lowered to $0.5\ s$ or less if there is no significant consequent evolution to investigate.

The legend is similar for all plots. Each curve represents the evolution over time of the corresponding variable. The unit of measurement is enclosed by square brackets. Immediately after, if present, the amount of scaling applied to the data curves to make them easier to examine.

The subscript m indicates that the variable is collected from the monitoring model. If absent, its source is the detailed model. The data curve Com refers to the command used as input for the simulation, and it is shared between the two models.

5.2.1 1 mm step command

(a)

(b)

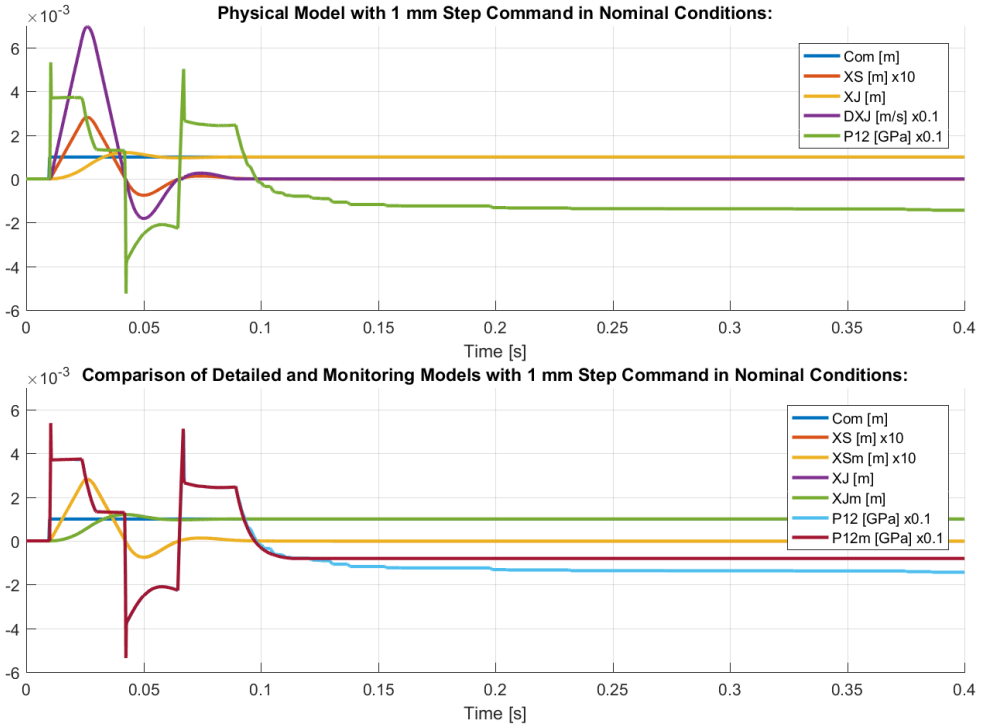


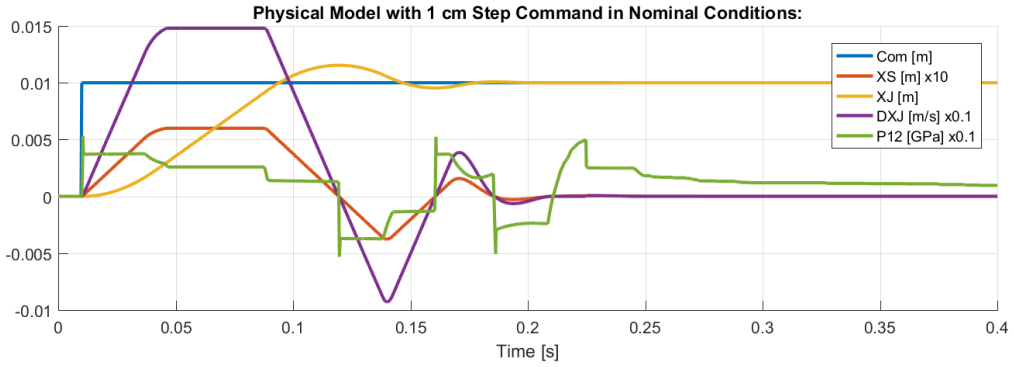
Figure 5.2: (a) and (b)

Figure 5.2*a* illustrates how a step command affects the spool position over time. If we look at the yellow curve, representing the jack displacement XJ, we see that it reaches its steady state around $0{,}06\ s$ with a little overshoot. The spool returns at its neutral position soon after, at 0.09. The positive and negative peaks of the differential pressures P_{12} represent the switching between the static and the dynamic condition, directly correlated to the friction.

In Figure 5.2*b* it is possible to see that the curves from the monitoring model are almost indistinguishable from the ones of the detailed model. Only the differential pressure P_{12} changes noticeably after $0.1\ s$, because of the band-limited white noise dithering signal applied to the offset current of the torque motor, absent in the monitoring model. This little difference will be present in every simulation.

5.2.2 1 cm step command

(a)



(b)

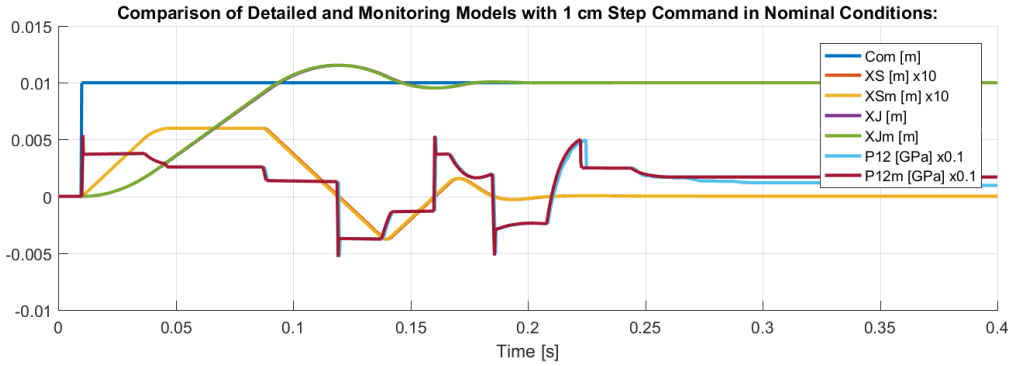


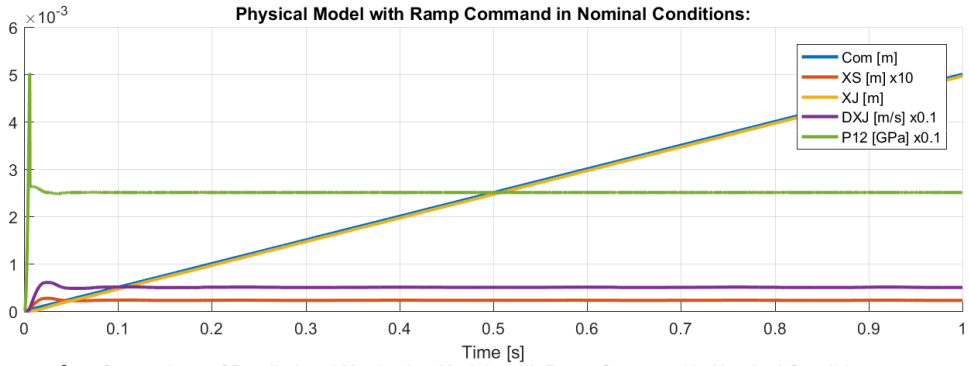
Figure 5.3: (a) and (b)

Differently from what Figure 5.2 shows, the larger command causes the spool to reach the end-stroke near 0.05 s, resulting in a saturation. For that reason, the jack reaches its steady state only later, around 0.18 s.

In Figure 5.3*b* there is no unfamiliar discrepancy from the previous graphs, sign that the monitoring model is simulating correctly the behaviour of the detailed model, although the white noise effect on the differential pressure is still evident, as expected.

5.2.3 Ramp command

(a)



(b)

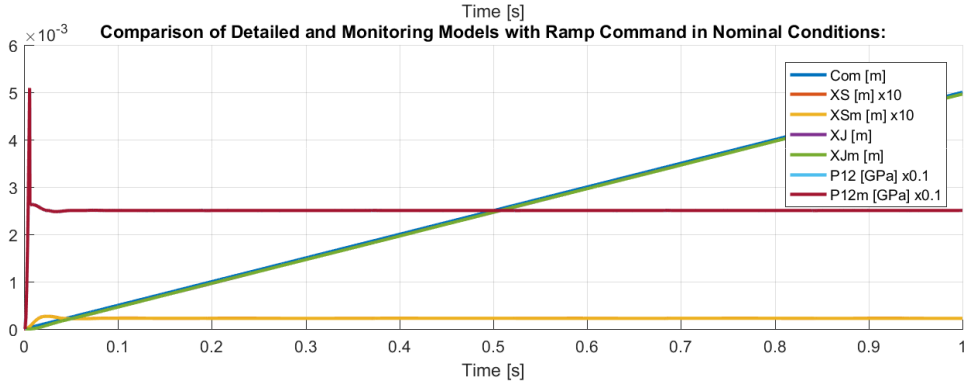


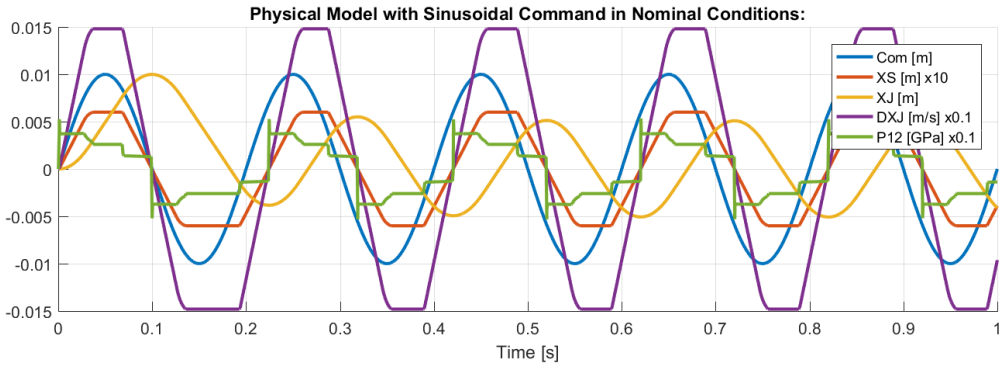
Figure 5.4: (a) and (b)

The detailed model is able to discriminate between the static and the dynamic condition. The brief transient period, seen in Figure 5.4 near the origin of the axes, begins at the first instants of the simulation, switching into dynamic friction as soon as the spool starts moving. A slight overshoot is present, ending at $0.05\ s$. Immediately after, the spool displacement becomes constant in time.

In Figure 5.4*b* we see that the jack response mimics perfectly the command, with their relative curves perfectly overlapping, until it eventually reaches the end-stroke. No difference is recognizable between the two models.

5.2.4 Sinusoidal command

(a)



(b)

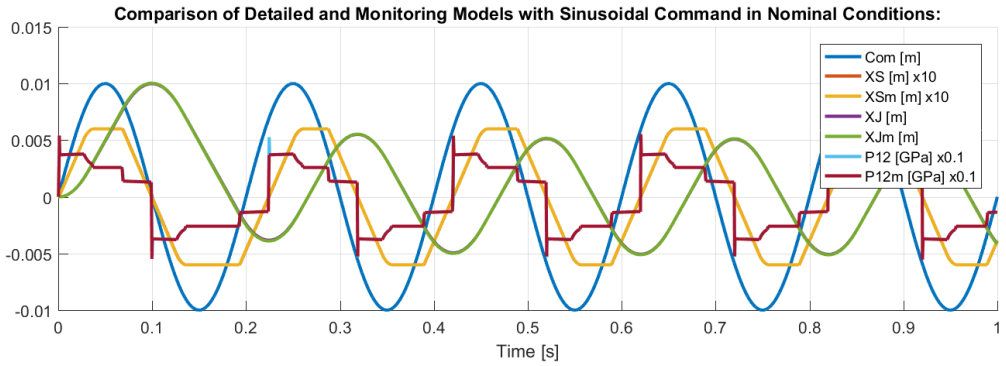


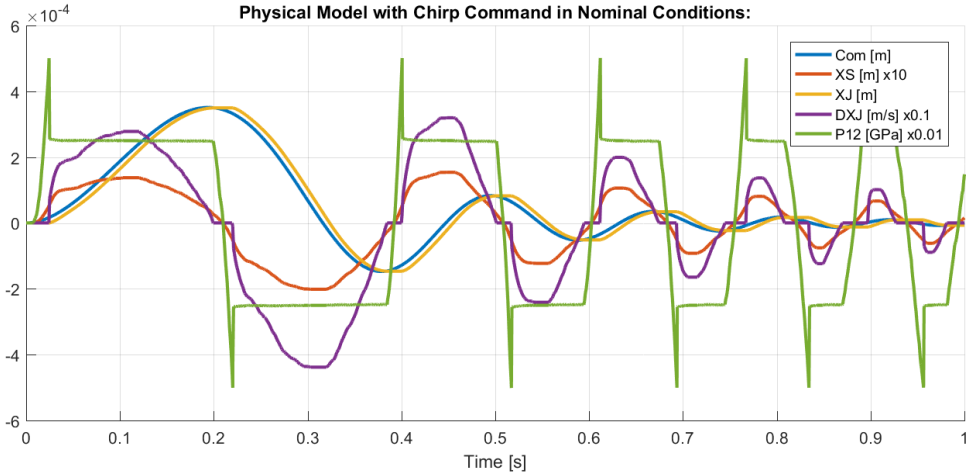
Figure 5.5: (a) and (b)

The sinusoidal command is able to keep the spool in a regular motion, as seen in Figure 5.5. The command has such a large amplitude that the spool reaches its end-stroke every half-period. This causes the model to commutate very often between the static and the dynamic conditions. The jack reaches its steady-state harmonic condition after a brief transition time, around 0.3 s, as shown by the yellow curve in Figure 5.5*b*.

No difference between the two models is evident.

5.2.5 Descreasing amplitude chirp command

(a)



(b)

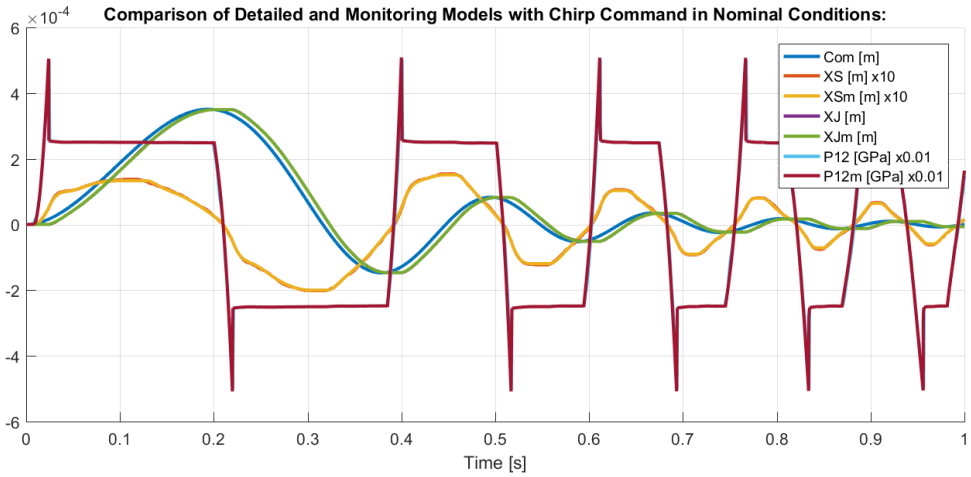


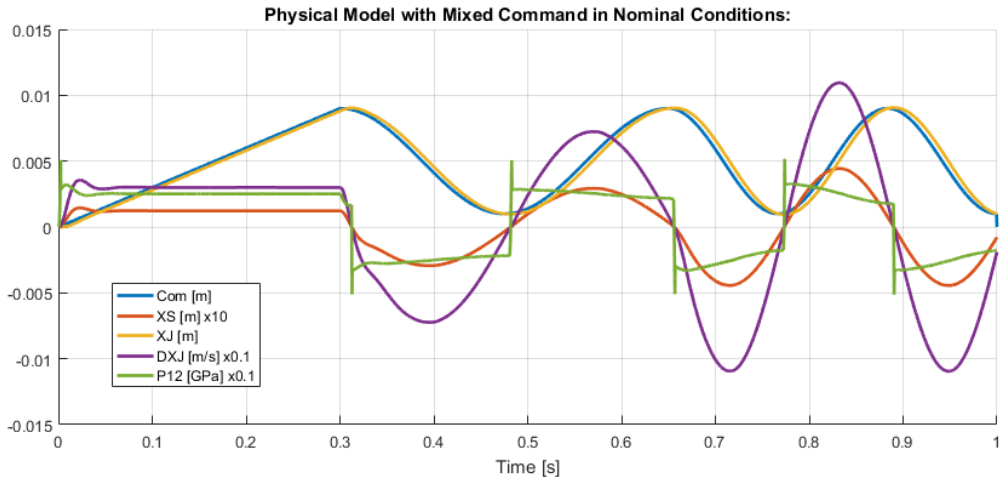
Figure 5.6: (a) and (b)

The chirp command here used is similar to the sine wave, but has crescent frequency and decrescent amplitude. The initial amplitude is much lower than the previous commands, and that has direct consequences on the friction. In fact, it takes considerable time to wait for the static force to rise until its maximum before switching into dynamic force at $0.03\ s$, as shown in Figure 5.6*a*. That causes the jack to remain stuck when it changes direction, delaying its response by a small fraction of a second. Looking at the jack velocity curve DXJ near the pressure peaks, we can see that the delay grows as the command amplitude decreases, proving that the jack has more trouble trying to move when the command has lower amplitude.

In the second graph, Figure 5.6*b*, we see that the curves from the two models match as expected.

5.2.6 Mixed command

(a)



(b)

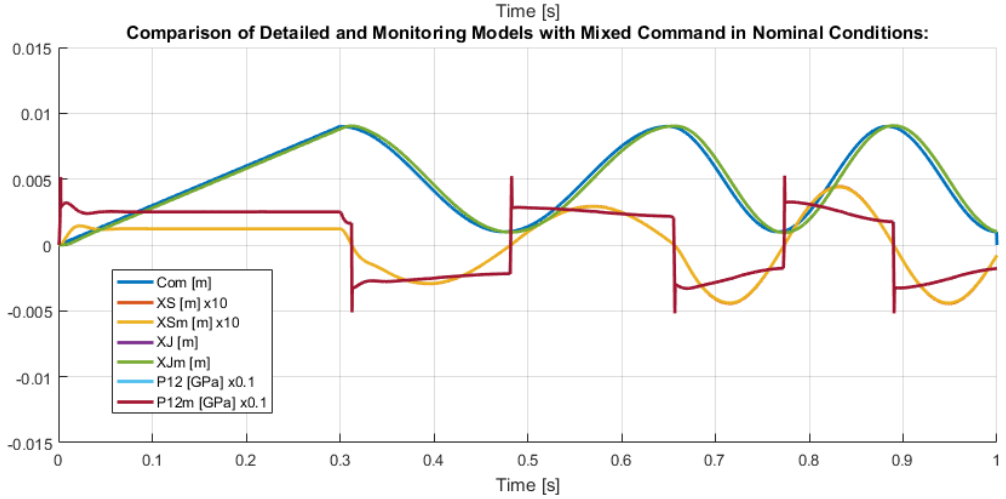


Figure 5.7: (a) and (b)

This type of command is composed of a ramp signal (with an higher slope than in Figure 5.4) and two sine waves with different frequencies, starting at 0.65 *s* and 0.87 *s* respectively. It is used in a testing environment to simulate more behaviours with a single command, especially useful when treating single faults, as we will see in the next chapter.

From Figure 5.7*a* it is possible to see that the ramp section is protracted until the response reaches its steady state, so this part of the command does not interact much with the rest. Its behaviour is similar from Figure 5.4, but with an higher slope. Because of the command slowly ramping up at first, the jack takes a while to move, causing a slight lag from the neutral position, enduring for the entire simulation. The two sine waves commands do not cause the spool to reach its end-strokes, so there are not many noteworthy anomalies, except for the discontinuities where the command changes (0.3, 0.65 and 0.87 *s*) and the slope of the curves changes instantly.

In the second plot, in Figure 5.7*b*, we see that the data curves of the two models match again.

5.3 Single fault analysis on HF model

The previous analysis demonstrated that the monitoring model is able to mimic the behaviour of the detailed model in nominal operating conditions. This time, applying a variety of faults, we can stress the detailed model enough to see how it is affected.

The faults considered in this dissertation are:

- backlash in the utilizer transmission;
- increasing dynamic friction on the jack.

Whereas the static friction is calculated using a constant coefficient ($FSD = 2$), it varies in proportion to the dynamic friction.

Rather, the types of command used in the analysis are:

- ramp command;
- decreasing amplitude chirp command;
- time-history mixed command.

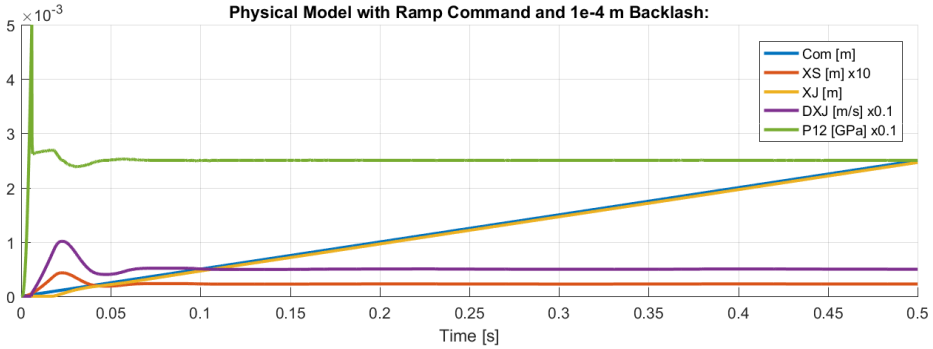
These were already used in the nominal conditions' analysis.

During the analysis of dynamic friction, the value of $FDJ = 1600\,N$ was omitted, as both lower and higher amounts of fault seemed more decisive for the analysis.

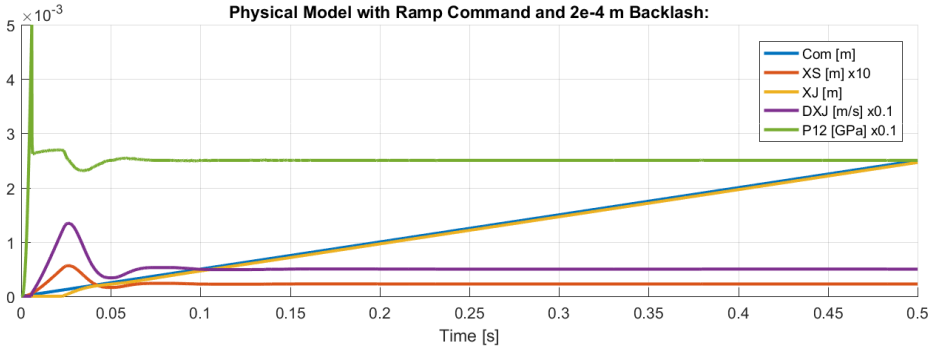
Moreover, some important variables were added, i.e. P_{12_m}, the differential pressure on the jack extremities calculated by the monitoring model (hence the subscript m).

5.3.1 Backlash with ramp command

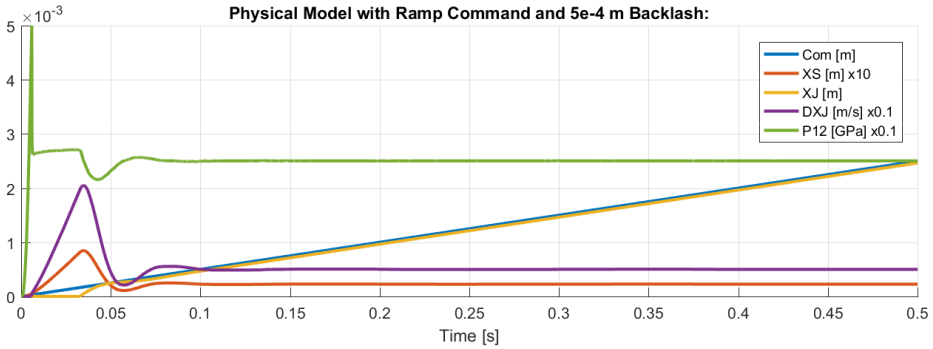
(a)



(b)



(c)



(d)

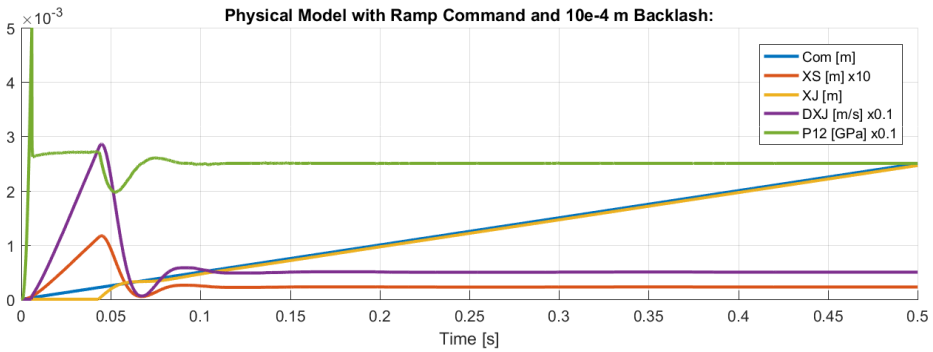


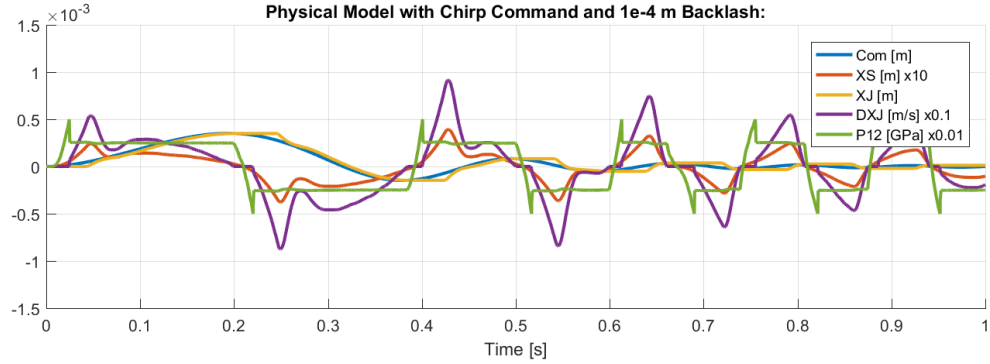
Figure 5.8: (a), (b), (c), (d)

Because of the backlash block (seen in Paragraph 4.6), the signal XJ here presented does not represent the jack displacement, but rather the utilizer displacement, generally a control surface. Instead, DXJ correctly represents the jack's velocity.

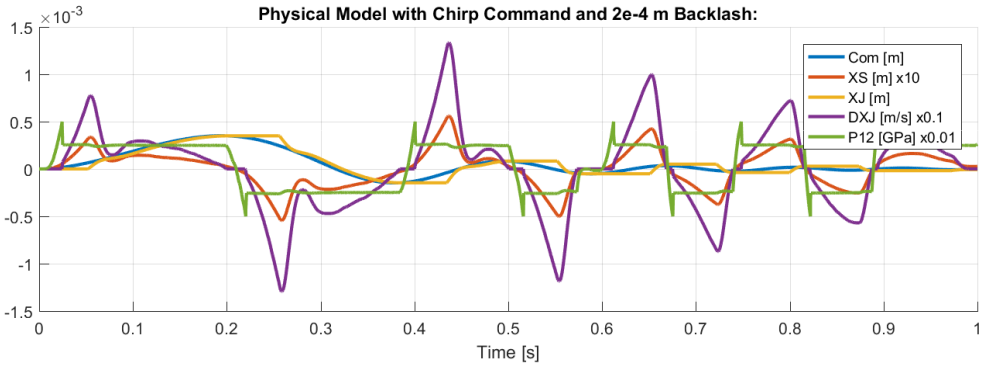
In this set of plots, it is clearly visible that the servo valve has a hard time trying to displace the utilizer (yellow curve, XJ) because of the backlash presence. This causes the spool to move far more than expected to overcome the fault. With a 1 mm backlash (Figure 5.8d), the utilizer starts to move around $0.04\ s$, and reaches its steady displacement condition around $0.07\ s$. Immediately after, the differential pressure has a considerable drop caused by the jack start-up.

5.3.2 Backlash with chirp command

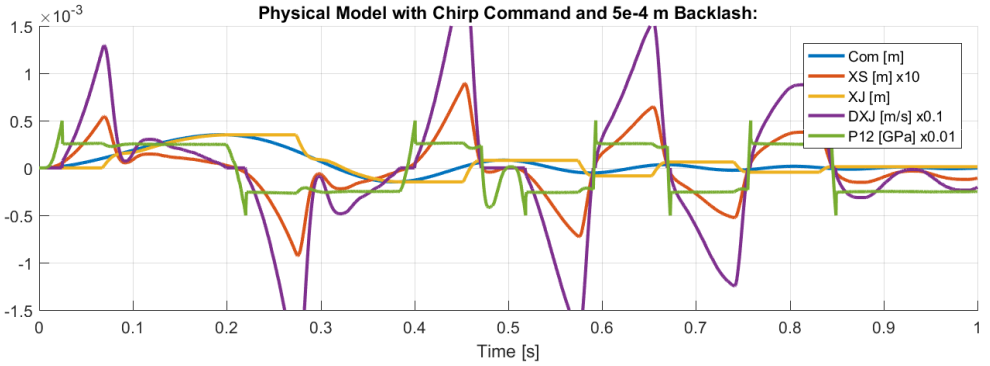
(a)



(b)



(c)



(d)

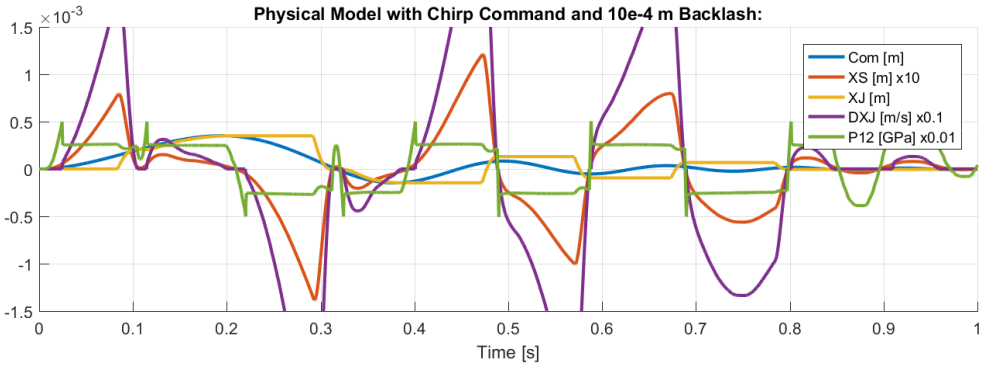
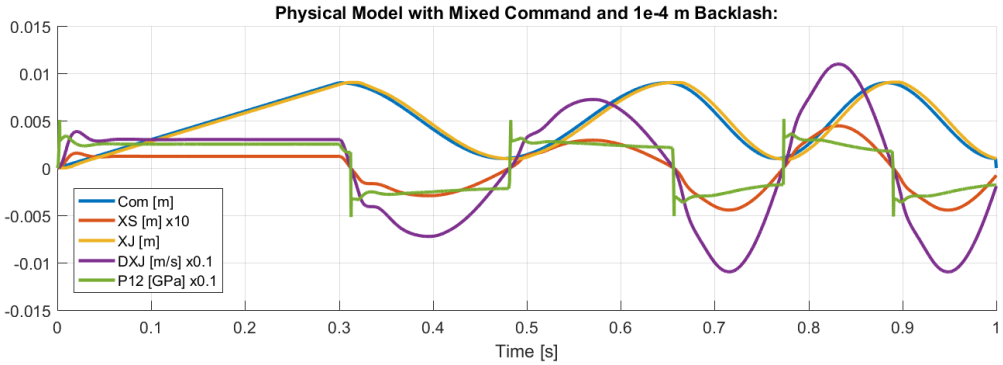


Figure 5.9: (a), (b), (c), (d)

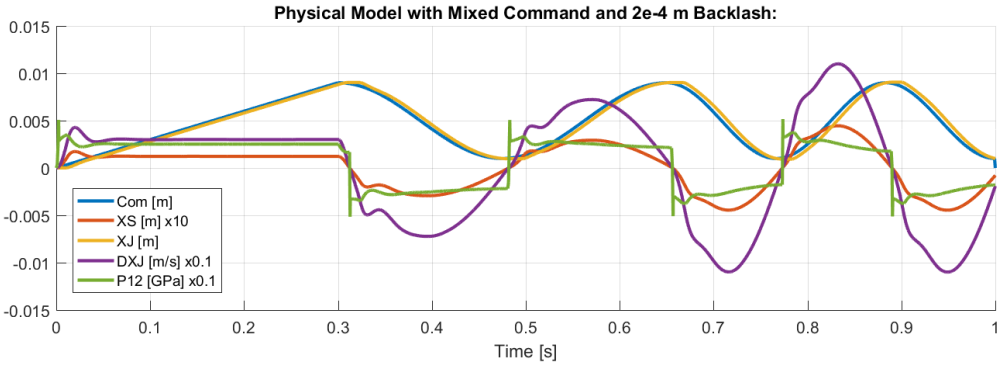
The model response deviates heavily from the nominal conditions seen in Figure 5.6. With the backlash increasing, it becomes more difficult for the utilizer to move, as the movements are more impulsive and less smooth. This causes important delays when changing direction, until the movements become completely out of phase (evident in the yellow curve of Figure 5.9*d*). When the amplitude of the chirp command becomes lower than $0.1\,mm$, the utilizer stops moving completely. The differential pressure (the green curve) shows some bumps instead, caused by the sudden movements of the jack.

5.3.3 Backlash with mixed command

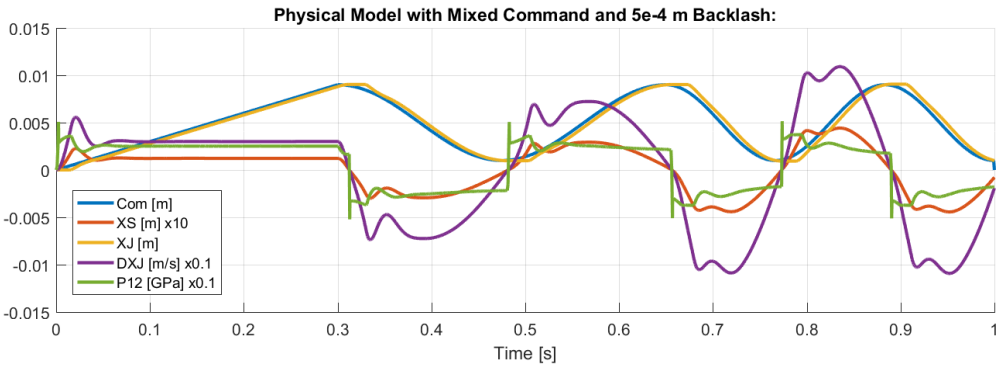
(a)



(b)



(c)



(d)

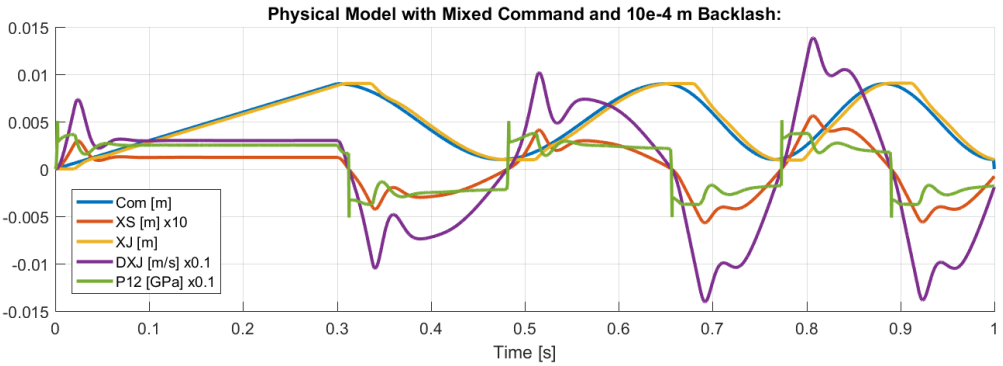
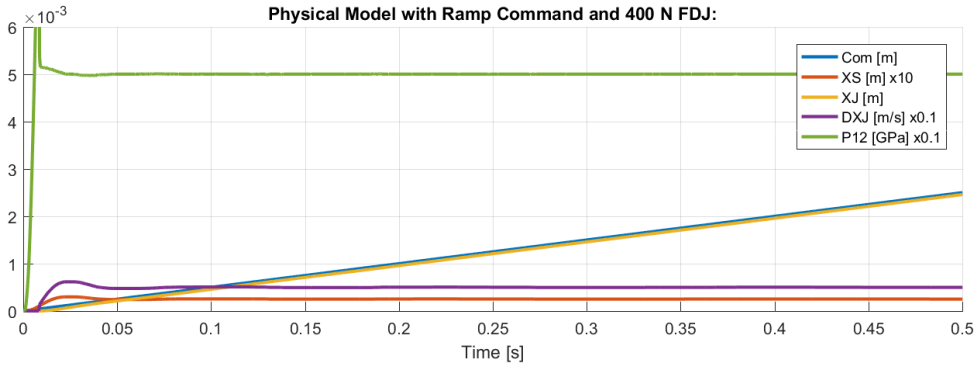


Figure 5.10: (a), (b), (c), (d)

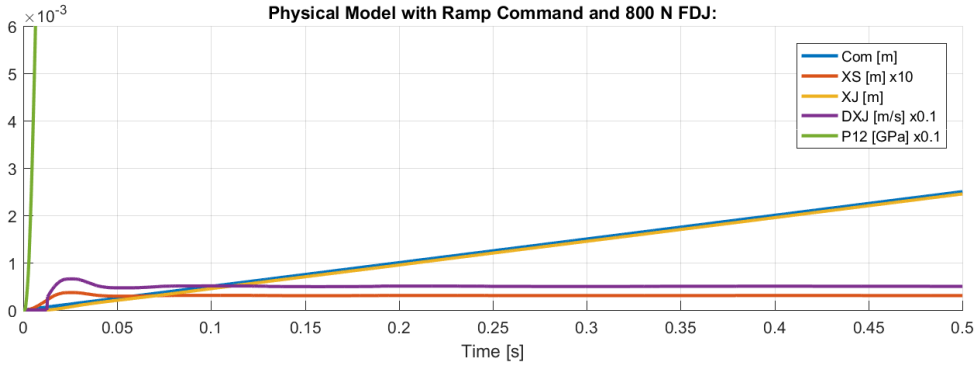
Looking at Figure 5.10, the delay is clearly visible in the direction change of the utilizer. Because the command is larger than in the previous example, the effects of the backlash are not as unfavourable, and there is no inversion of command. However, the spool is subject to many sudden oscillations, which can increase the amount of wear in the long run.

5.3.4 Dynamic friction increase with ramp command

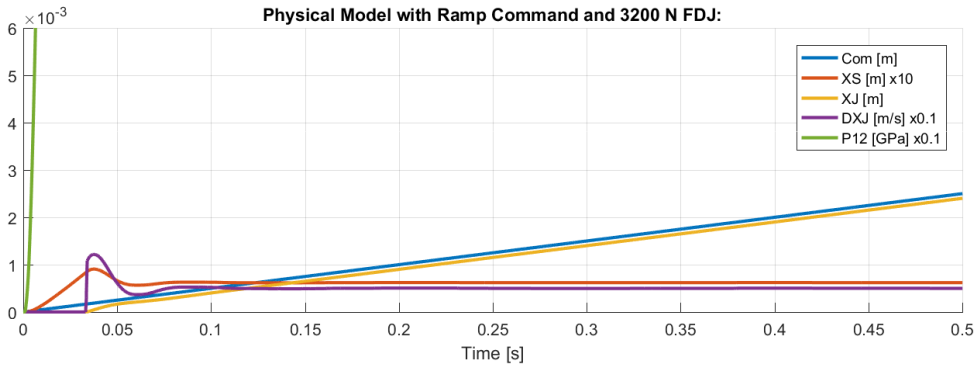
(a)



(b)



(c)



(d)

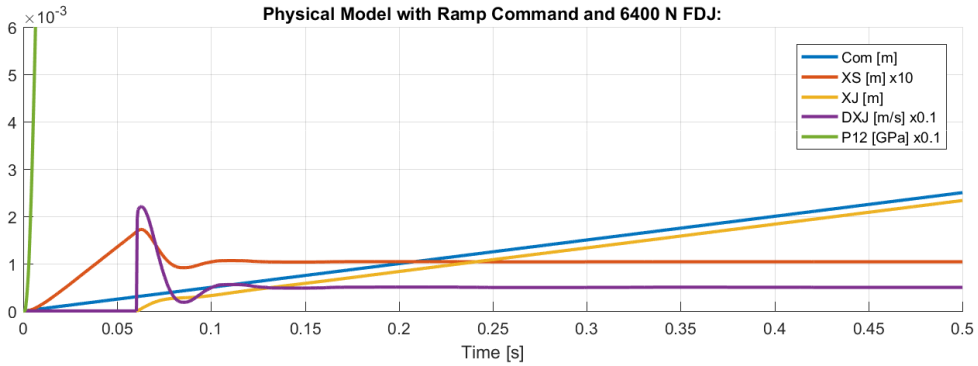
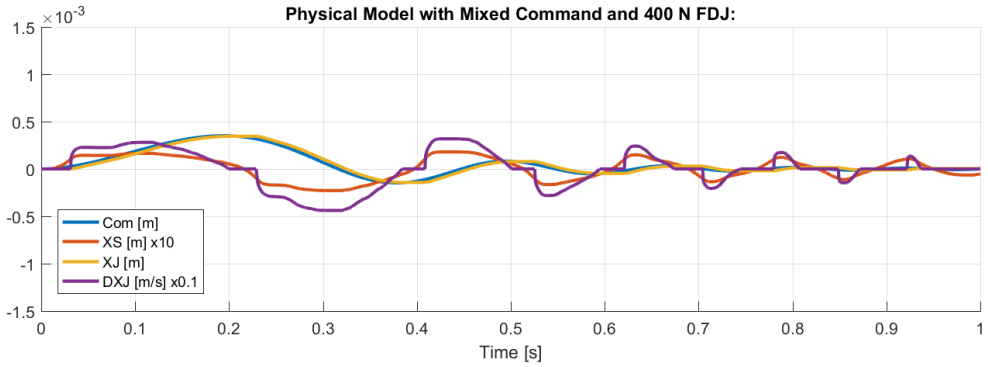


Figure 5.11: (a), (b), (c), (d)

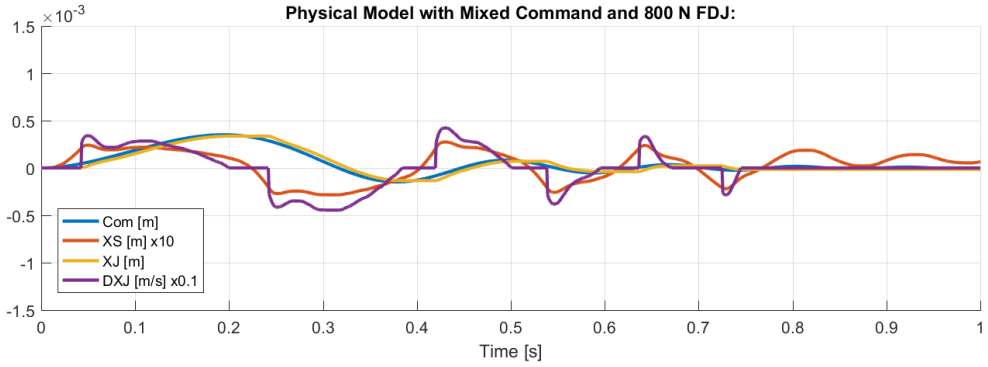
As expected, when increasing the dynamic friction acting on the jack (hence, also the static friction), the differential pressure increases accordingly, rapidly going off the scale. In fact, every time the friction doubles, the pressure doubles too. This delays the switching from the static to the dynamic condition, preventing the jack to move until the static force acting on it arrives at its maximum value (clearly visible in Figure 5.11*d*, around 0.06 *s*). Another effect is the constant lag in the response of the utilizer which increases with the fault amount.

5.3.5 Dynamic friction increase with chirp command

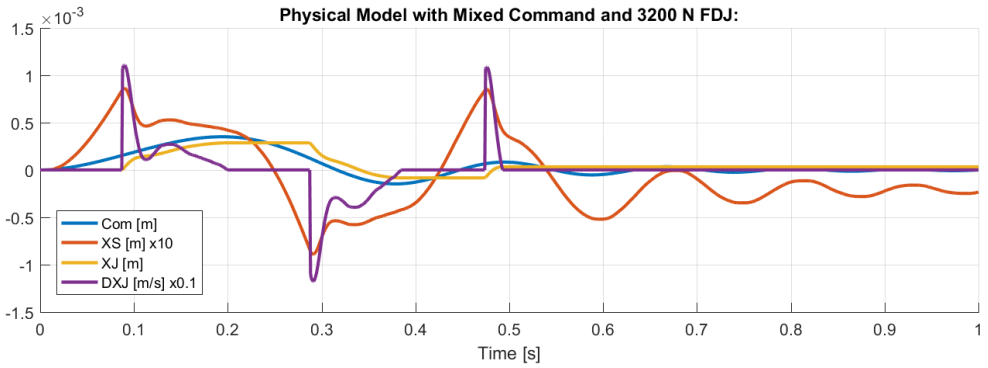
(a)



(b)



(c)



(d)

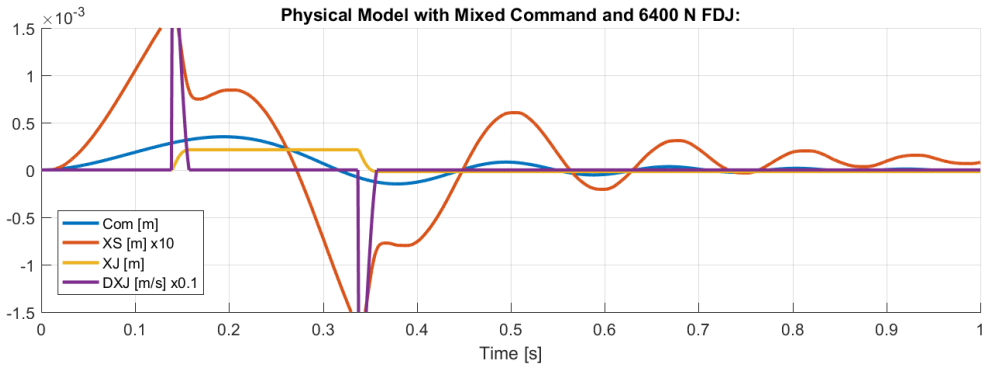
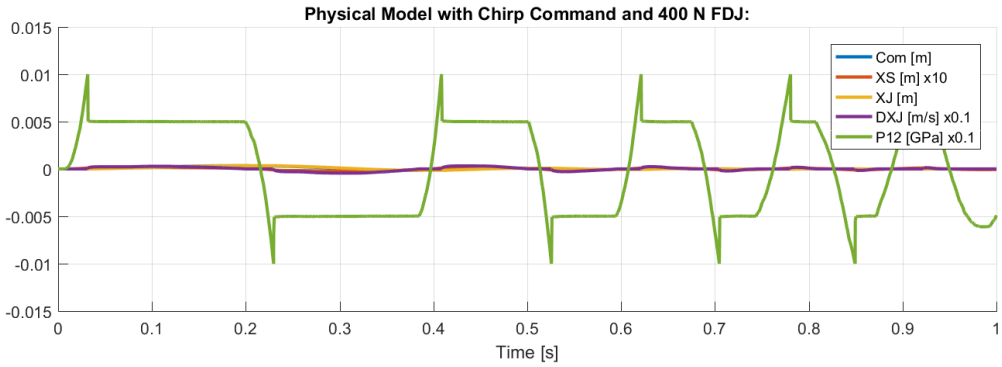


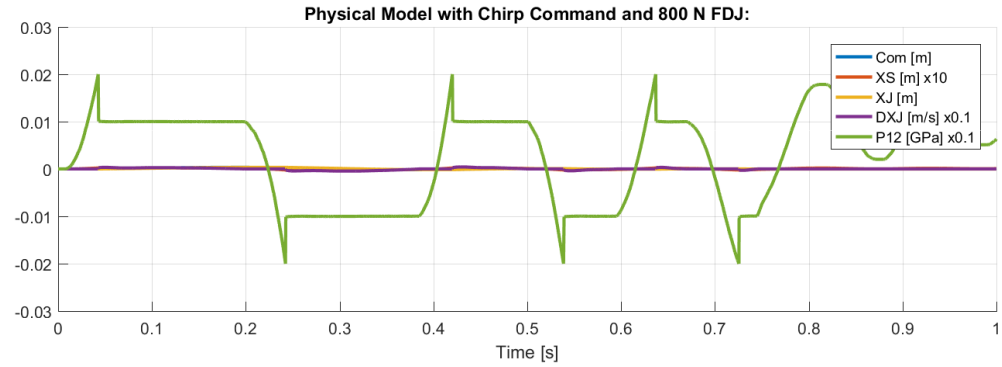
Figure 5.12: (a), (b), (c), (d)

In Figure 5.12*d*, it is evident that the friction has increased so much that is preventing the utilizer to move. The velocity of the jack is null (the purple curve), except for two brief moments (at 0.15 s and 0.34 s). While the jack is still, the spool continues to move with a decreasing harmonic pattern, analogue to the command. However, the differential pressure generated by opening the hydraulic ports is not large enough to displace the jack.

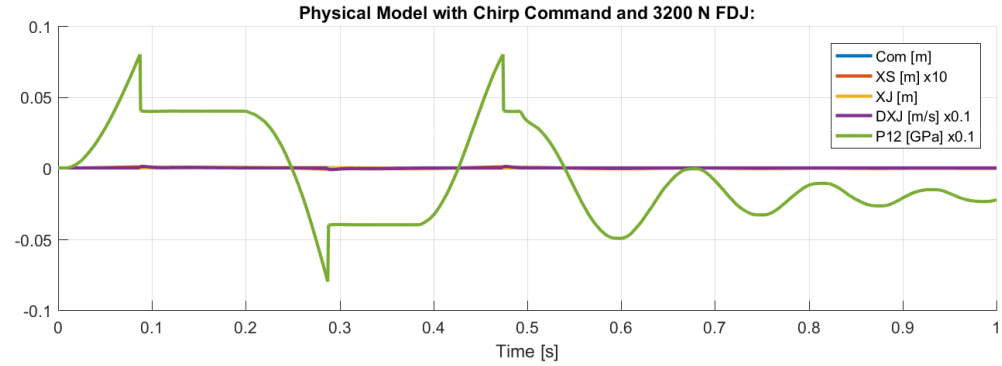
(a)



(b)



(c)



(d)

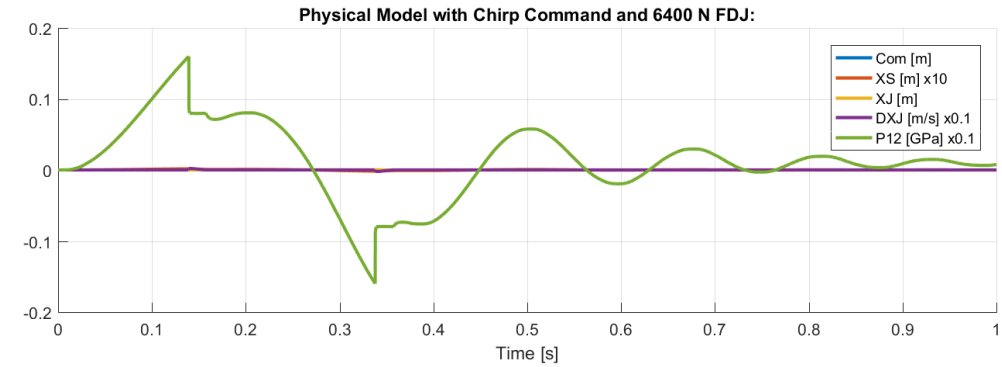
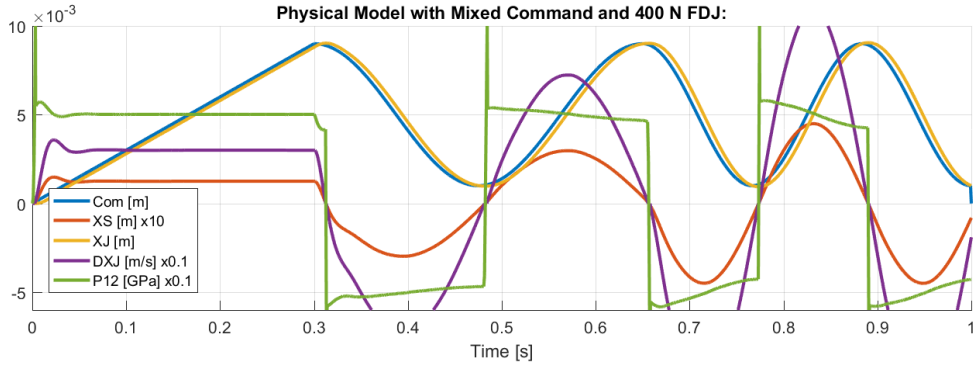


Figure 5.13: (a), (b), (c), (d)

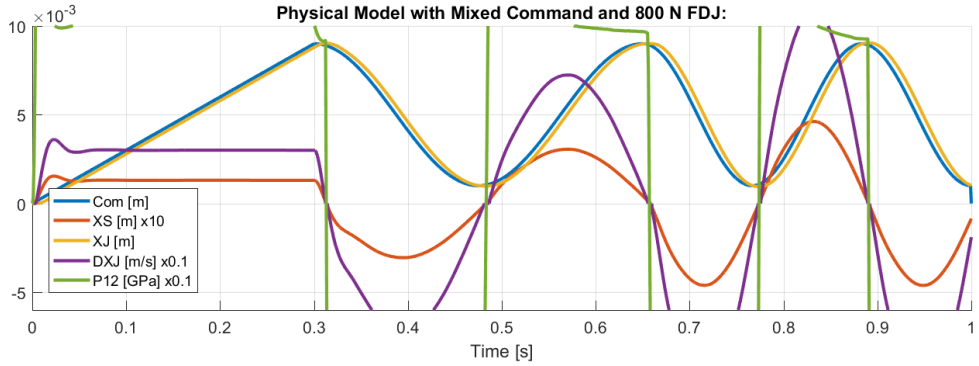
As it is shown in Figures 5.13, the differential pressure P_{12} has a similar behaviour, although the curves develop in different and increasing scales.

5.3.6 Dynamic friction increase with mixed command

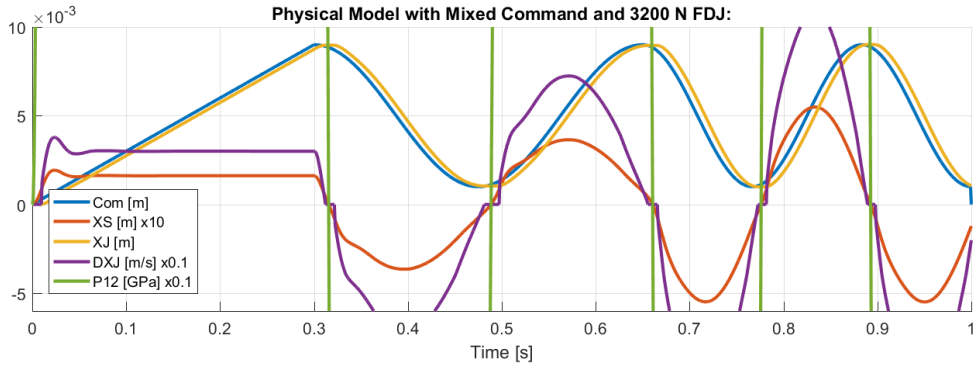
(a)



(b)



(c)



(d)

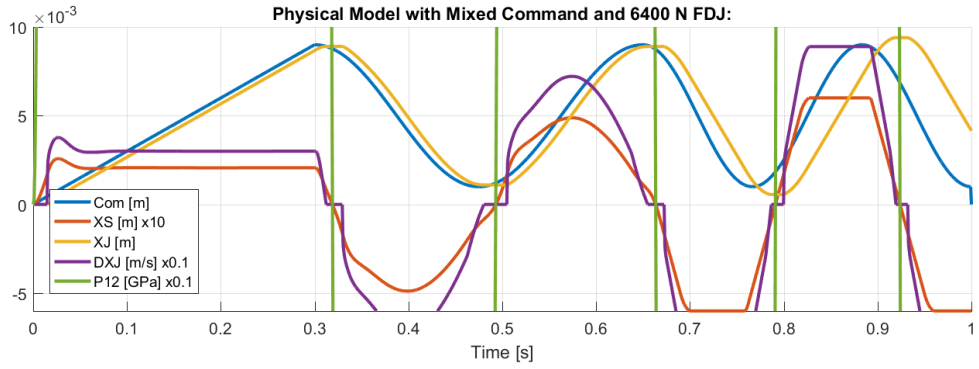


Figure 5.14: (a), (b), (c), (d)

As we have already seen previously, the differential pressure varies proportionally with the amount of dynamic friction. The spool must travel more to generate enough pressure.

Figure 5.14*d* presents the limit case, where the spool eventually reaches its end-strokes in 0.7 s 0.82 s and 0.95 s. That saturation causes a major delay in the utilizer displacement, which shifts from a sinusoidal shape to a straight line while the spool is saturated (particularly evident in the yellow curve between 0.7 and 0.76 s). Furthermore, with the increase of friction, the jack stays still longer when changing direction.

5.4 Response comparison of HF and LF models with single faults

As we have seen in the fault analysis, the monitoring model does not always perfectly match the behaviour of the detailed model because of his simplified design. In the previous paragraphs we have treated the low-fidelity model as a reminder of the ideal nominal conditions, useful to identify and better analyse the effects generated by the faults.

The faults applied to the models are the same of the last chapter:

- backlash in the utilizer transmission;
- increasing dynamic friction on the jack.

Likewise, the commands are:

- ramp command;
- decreasing amplitude chirp command;
- time-history mixed command.

As the first steps of prognostic analysis were conducted, more commands were needed, as the previous were considered inadequate:

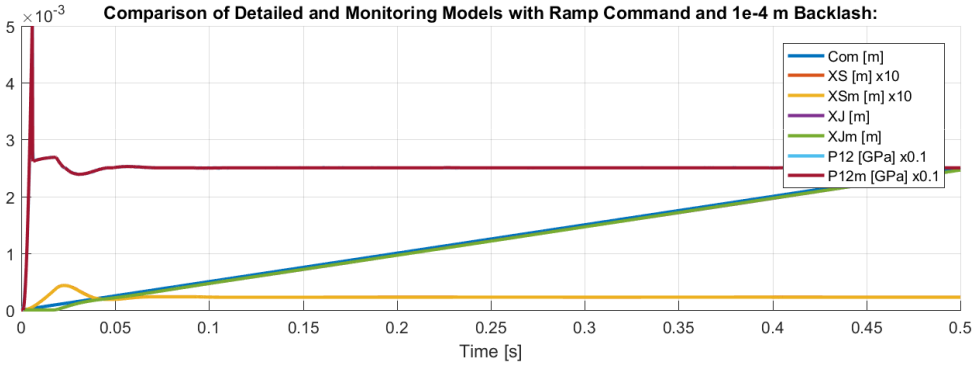
- 1 mm step command;
- 1 cm step command;
- sinusoidal command;
- 1 cm chirp command.

The first three were used within the dynamic friction simulation, but they will not be discussed, considering that their results are not useful for the purposes of this work.
The latter, however, will be reported with both faults. The reasons behind these choices are explained in the next chapter (Paragraph 6.3.1).

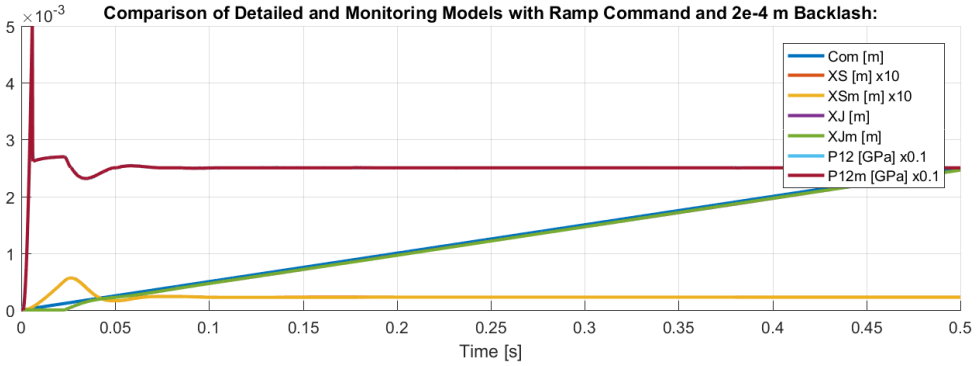
The structure of the plots and the legends is the same as those in the previous chapter.

5.4.1 Backlash with ramp command

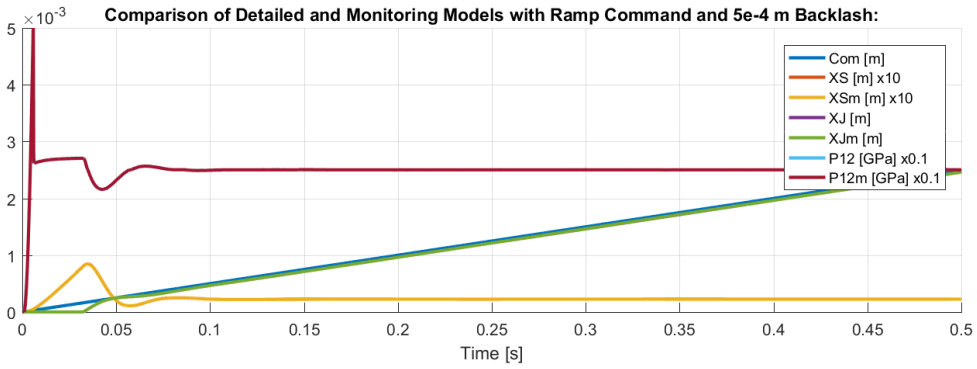
(a)



(b)



(c)



(d)

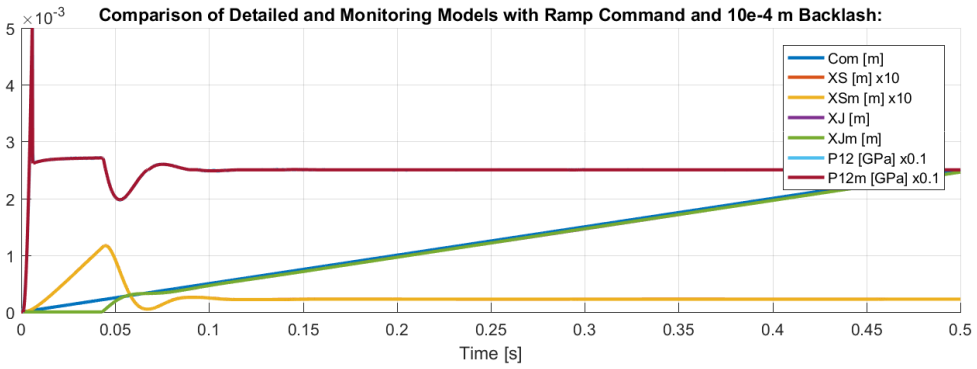
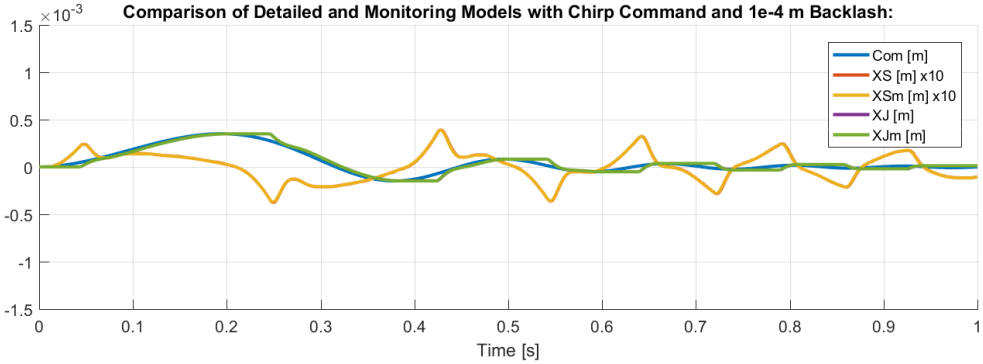


Figure 5.15: (a), (b), (c), (d)

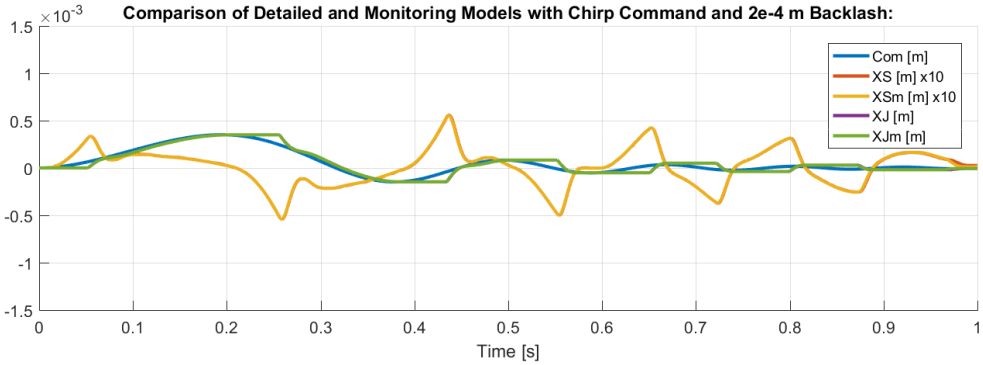
In Figure 5.15 it is particularly evident how the detailed model is able to reproduce the backlash effects correctly. The only difference, imperceptible in these graphs because of its small amplitude, is the presence of the dithering signal applied in the offset signal of the torque motor inside the detailed model, unlike in the monitoring model.

5.4.2 Backlash with chirp command

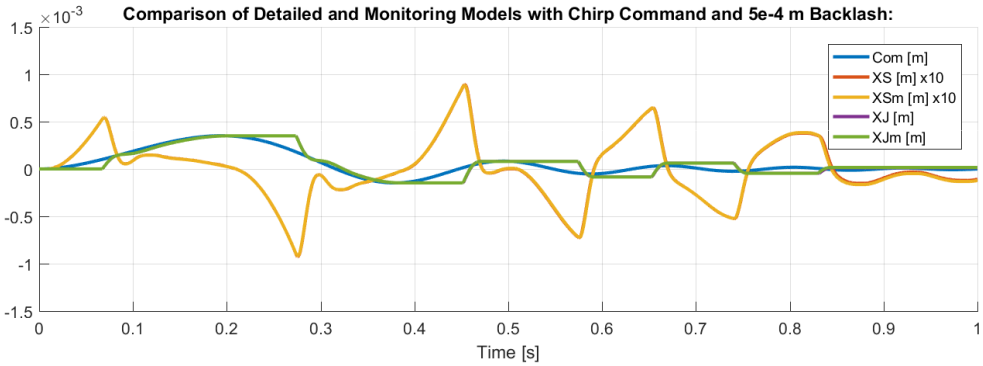
(a)



(b)



(c)



(d)

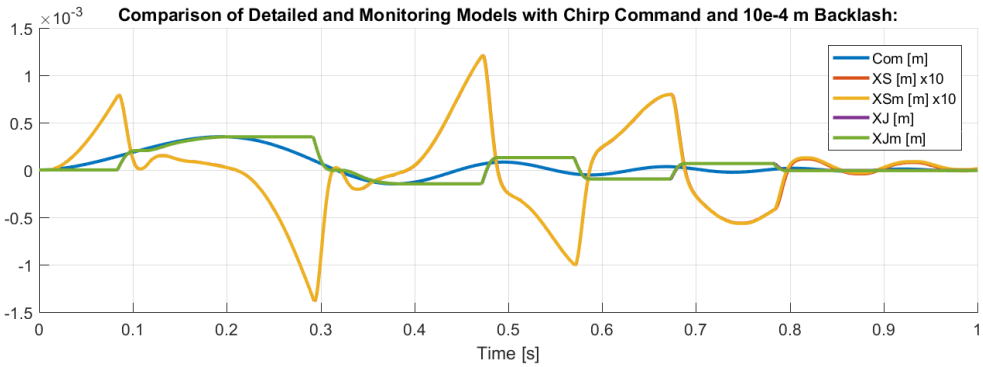
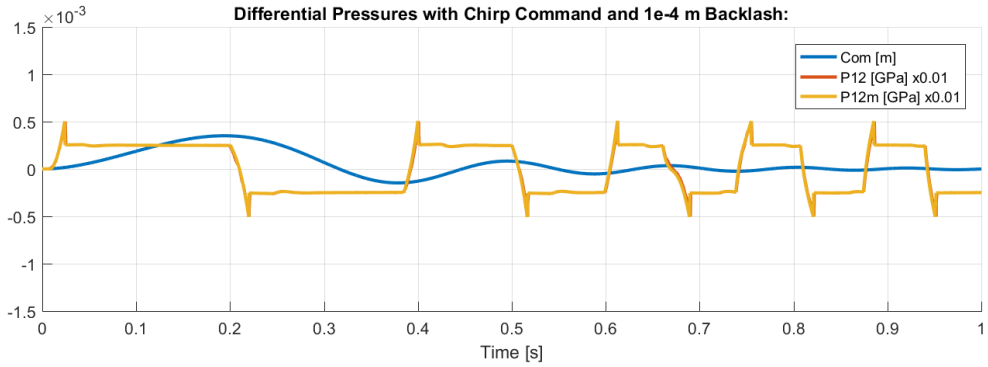


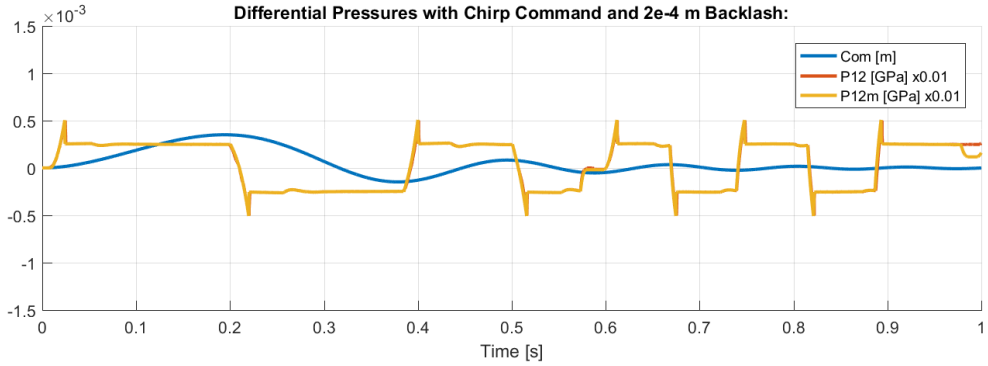
Figure 5.16: (a), (b), (c), (d)

In Figure 5.16 no major difference is perceivable. There are few instants where the red curve – the spool displacement in the high-fidelity model - becomes slightly visible (e.g. in Figure 5.15*b* at 0.96 *s*, in Figure 5.15*c* after 0.84 *s* and in Figure 5.15*d* after 0.77 *s*, which highlights that the differences become more clearly visible when the fault amount increases).

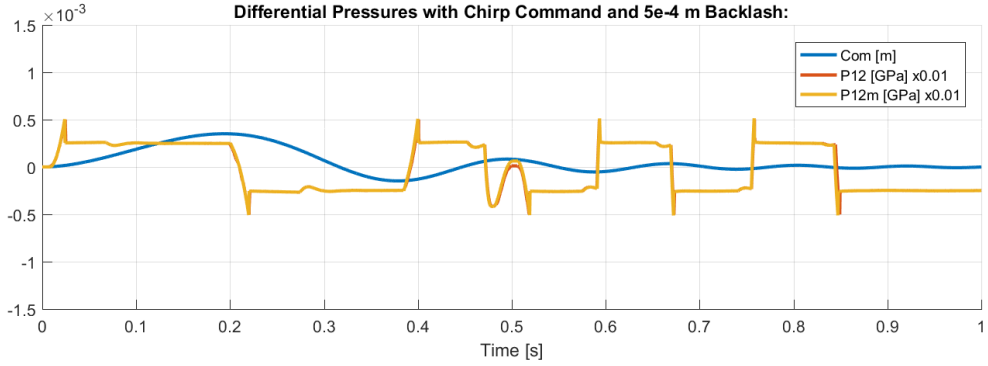
(a)



(b)



(c)



(d)

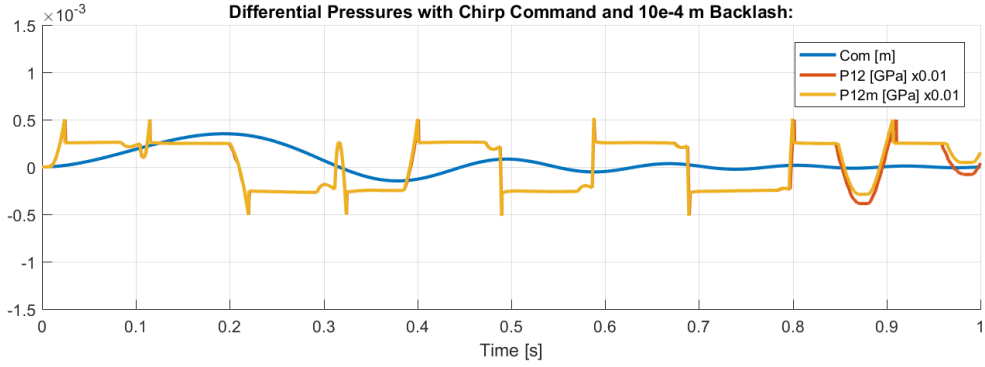


Figure 5.17: (a), (b), (c), (d)

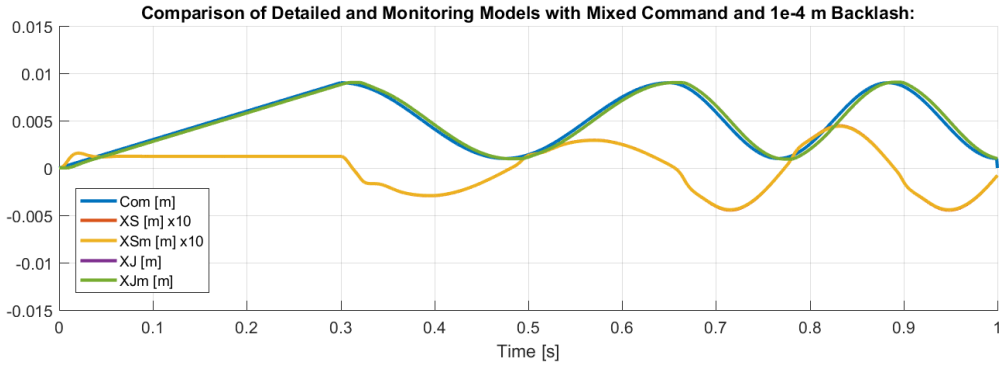
The simulations of the differential pressures P_{12} and P_{12_m} have been separated from the previous variables for the sake of comprehension.

Some slight differences are present because of the different friction models adopted in the two subsystems (Borello for the high-fidelity, Karnopp for the low-fidelity). They are particularly visible in Figure 5.17b (after 0.98 s), in Figure 5.17c (near 0.5 s and after 0.84 s) and in Figure 5.17d (after 0.84 s).

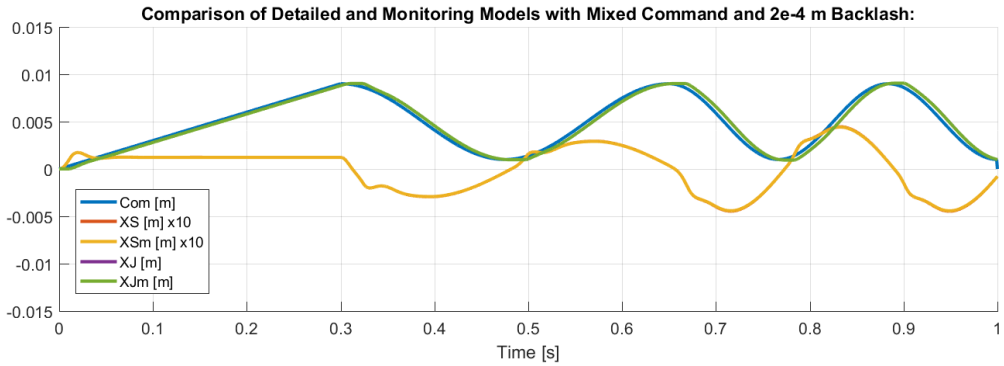
The majority of them occurs when the command becomes really small in amplitude, so an amplification of the divergences is expected.

5.4.3 Backlash with mixed command

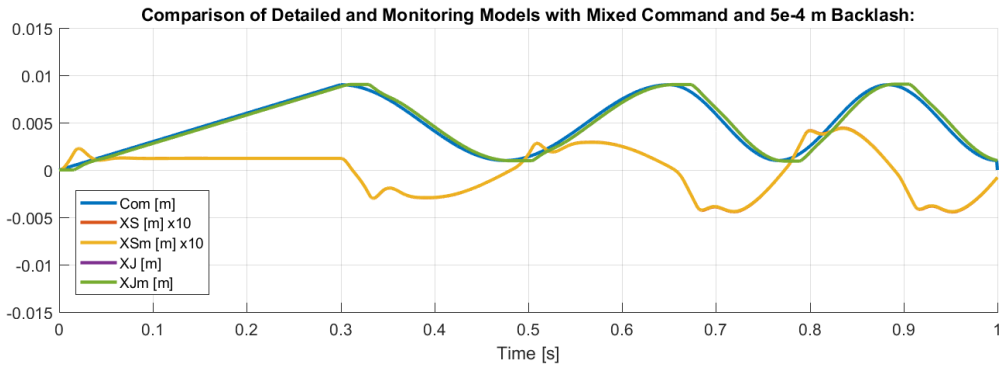
(a)



(b)



(c)



(d)

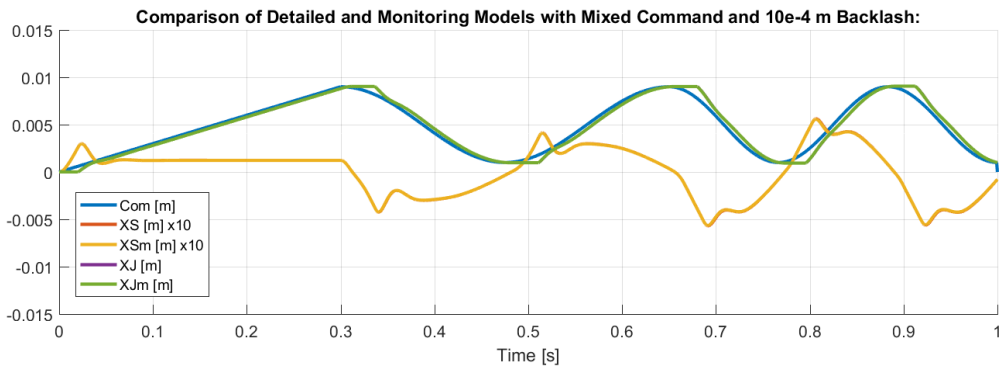
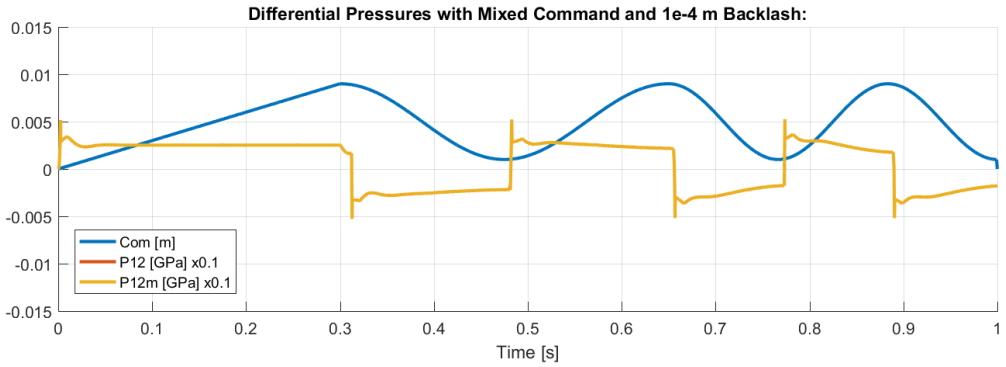


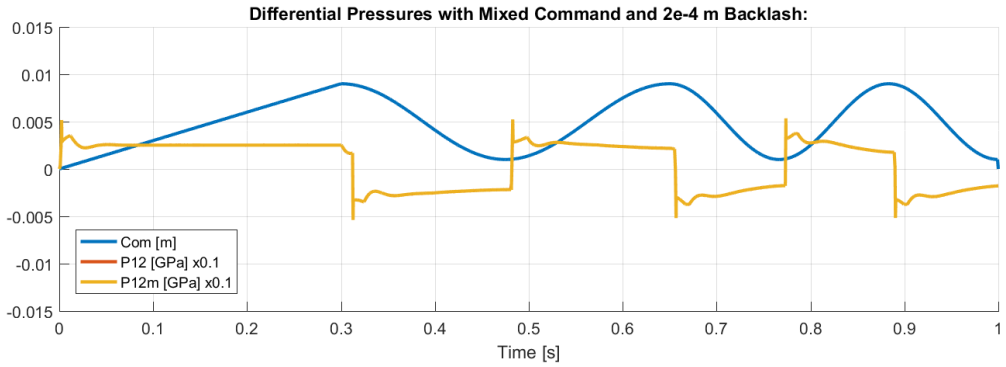
Figure 5.18: (a), (b), (c), (d)

Looking at Figure 5.18, the differences are imperceptible. This means that the detailed model, even with the high simplification and simulation speed achieved, is able to correctly simulate – in the current conditions – the position data, even with different amounts of the backlash faults. In the next graphs (Figure 5.19), it is evident how the same results emerge with the differential pressure curves.

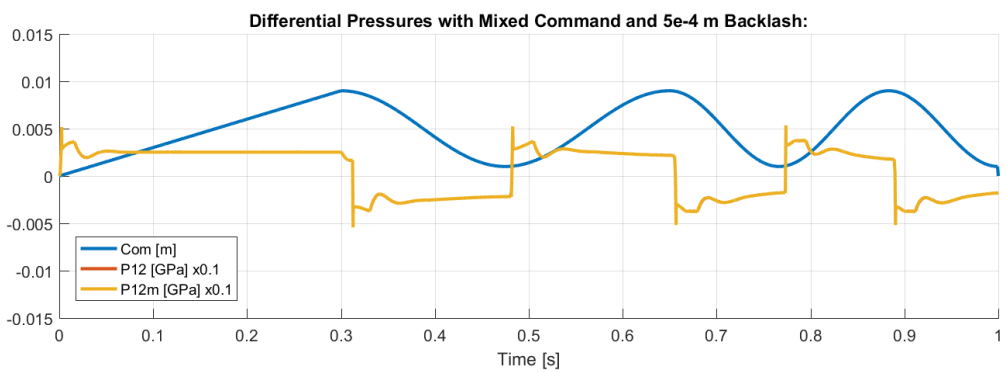
(a)



(b)



(c)



(d)

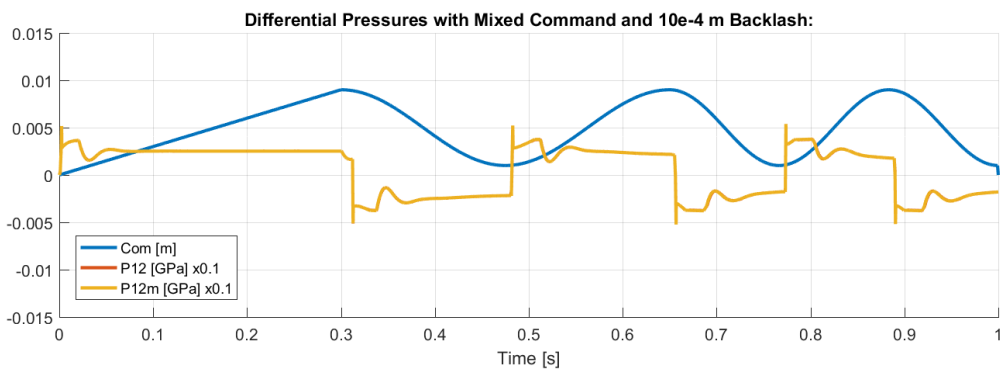
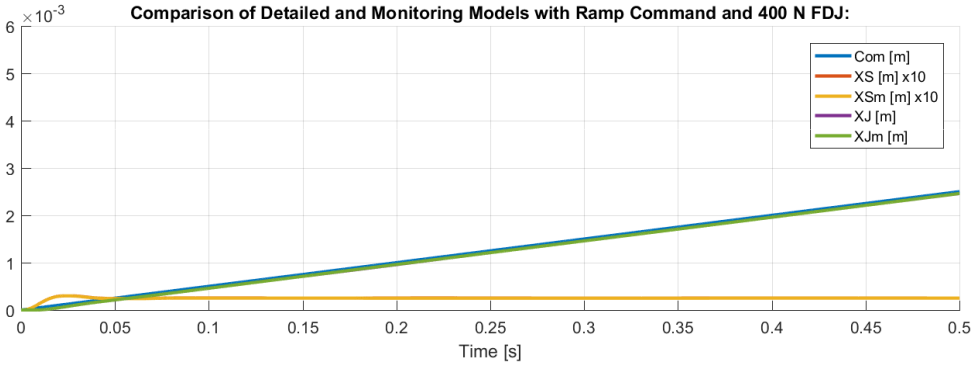


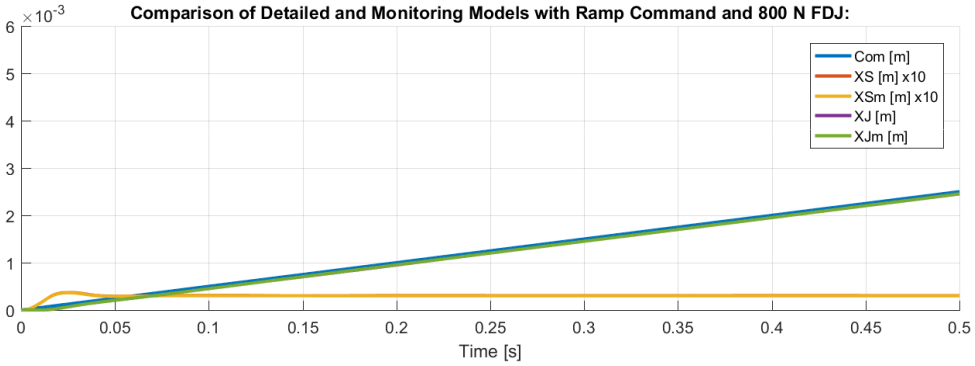
Figure 5.19: (a), (b), (c), (d)

5.4.4 Dynamic friction increase with ramp command

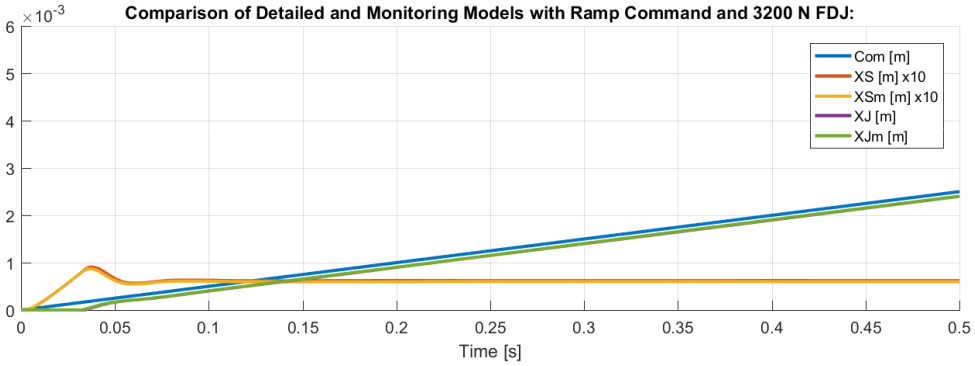
(a)



(b)



(c)



(d)

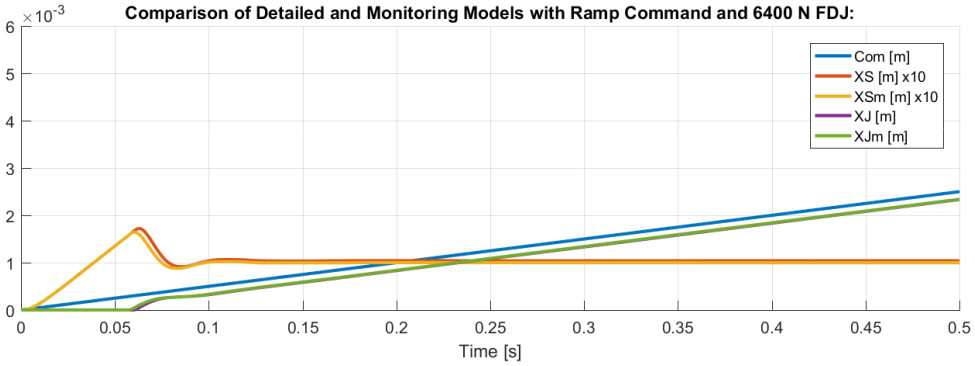
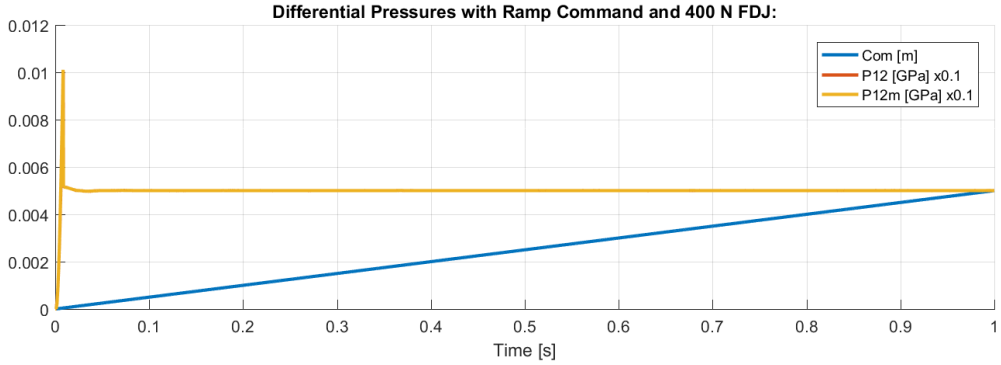


Figure 5.20: (a), (b), (c), (d)

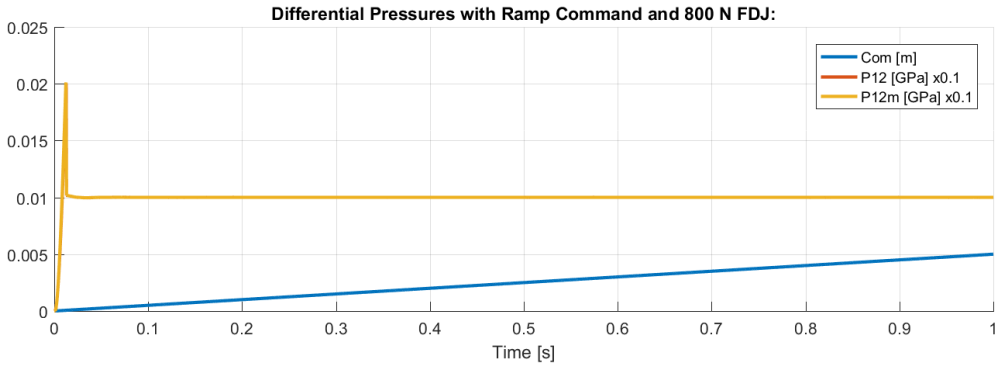
Differently from the backlash fault analysis, the dynamic friction increase generates greater differences between the two models than in previous cases, although still minor. Those are easily notable in Figure 5.20*d*, near the spool position peak (yellow curve, 0.07 s) and the

utilizer start-up (purple curve, 0.06 *s*). These are probably caused by the differences in the friction models (Karnopp and Borello) we have already discussed in Paragraphs 3.5.5 and 3.5.6.

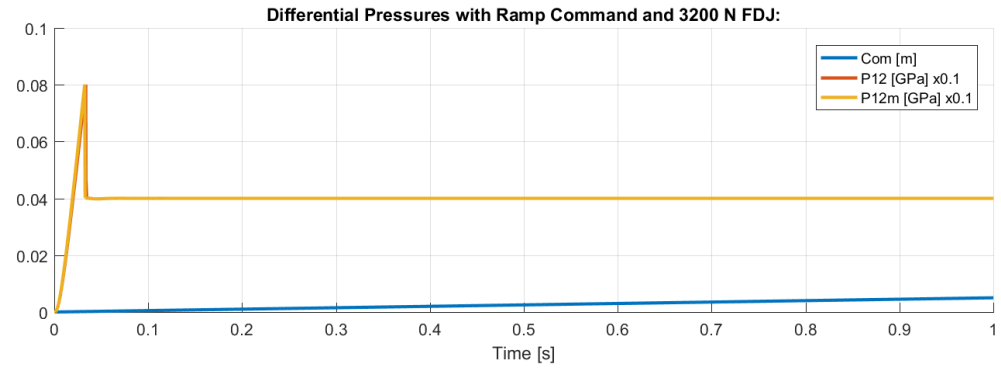
(a)



(b)



(c)



(d)

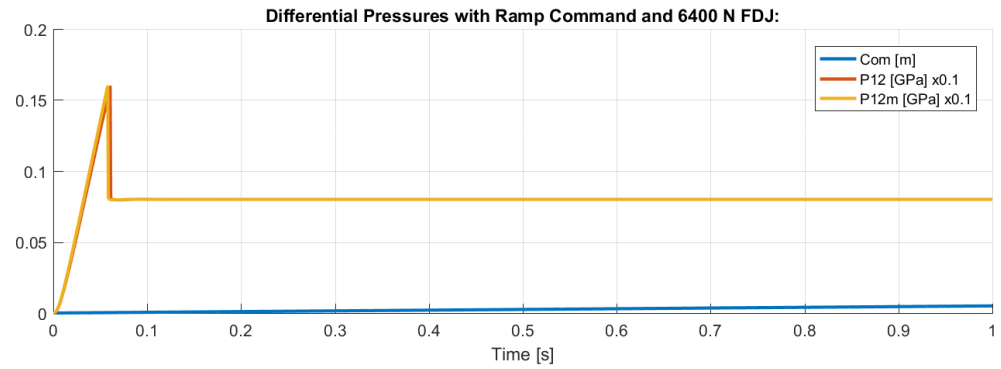
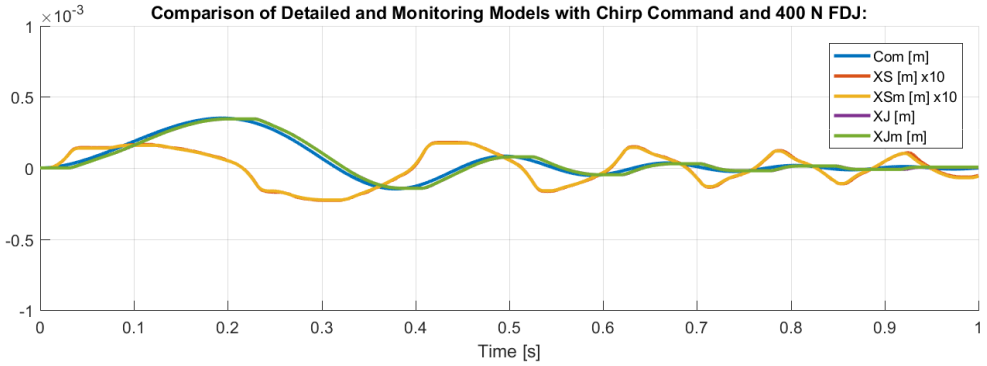


Figure 5.21: (a), (b), (c), (d)

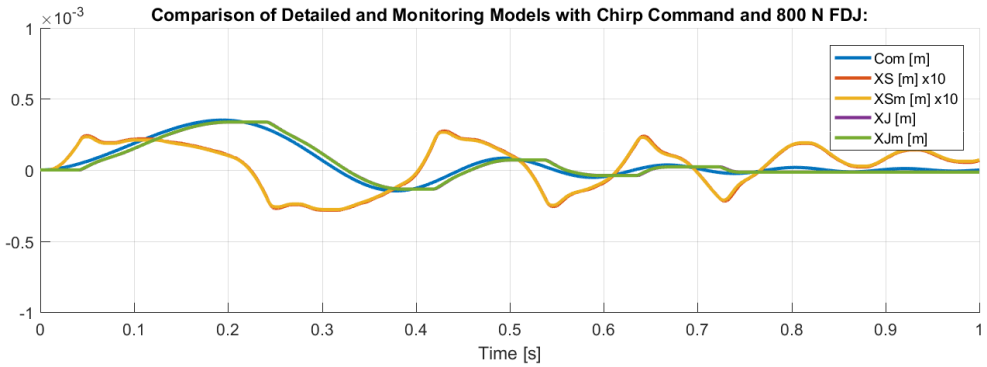
The same thing happens in the differential pressure curves, scaled accordingly with the fault amount to allow an easier reading. The differences become evident in Figure 5.21*d,* when the dynamic friction is already prominent.

5.4.5 Dynamic friction increase with chirp command

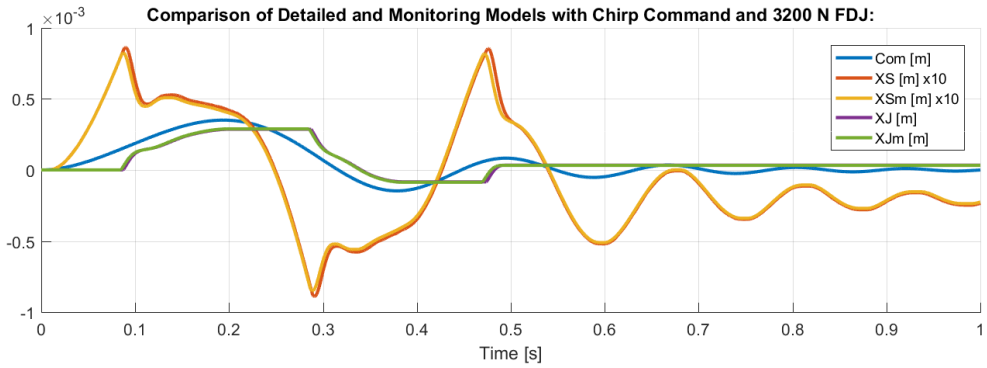
(a)



(b)



(c)



(d)

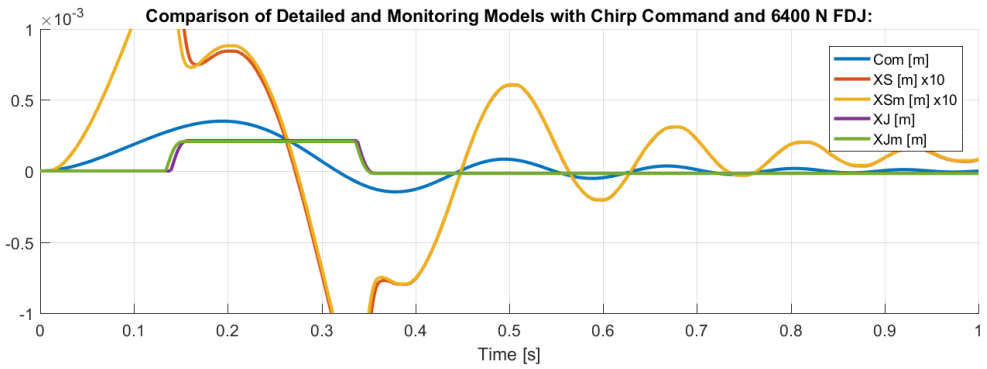
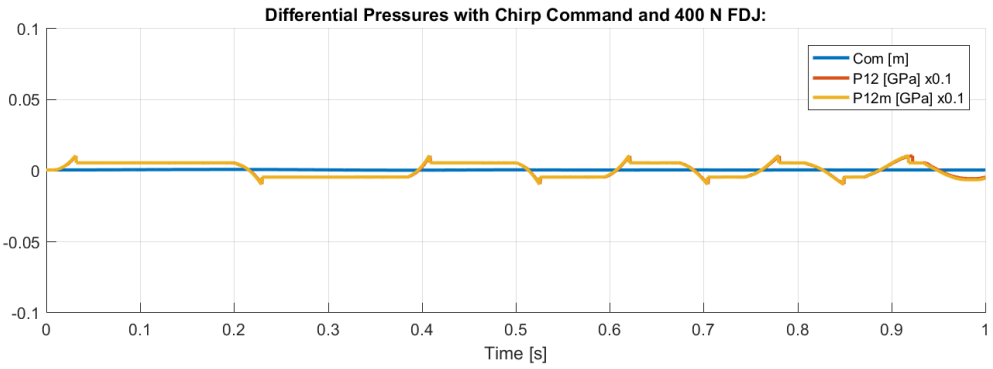


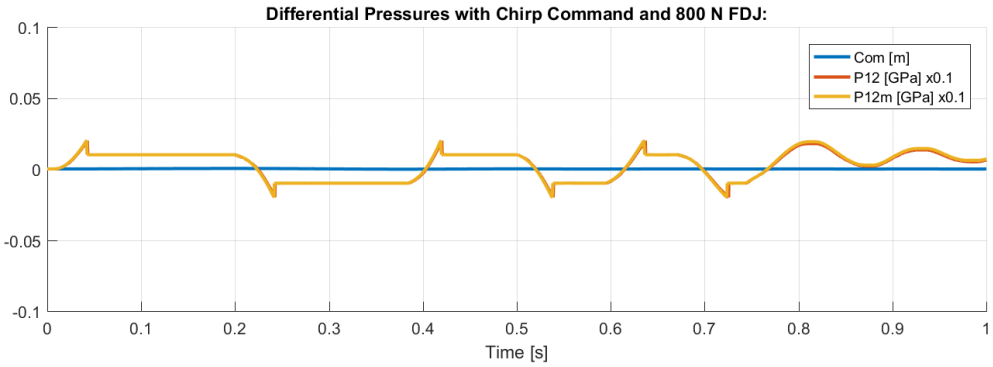
Figure 5.22: (a), (b), (c) (d)

In Figure 5.22*c* and 5.22*d* the results are different. In fact, considering the spool displacement, the deviation between the two models happen at lower frequency and higher amplitude, and the differences become gradually less noticeable with the simulation proceeding. A similar situation happens with the differential pressure curves, as shown in Figure 5.23.

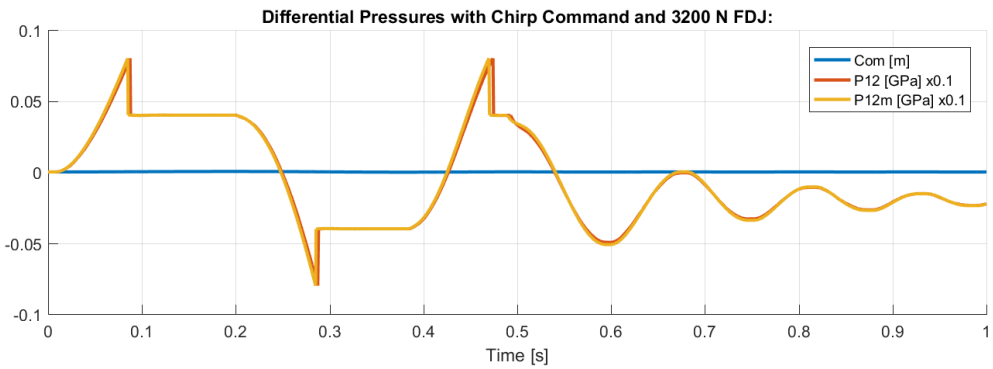
(a)



(b)



(c)



(d)

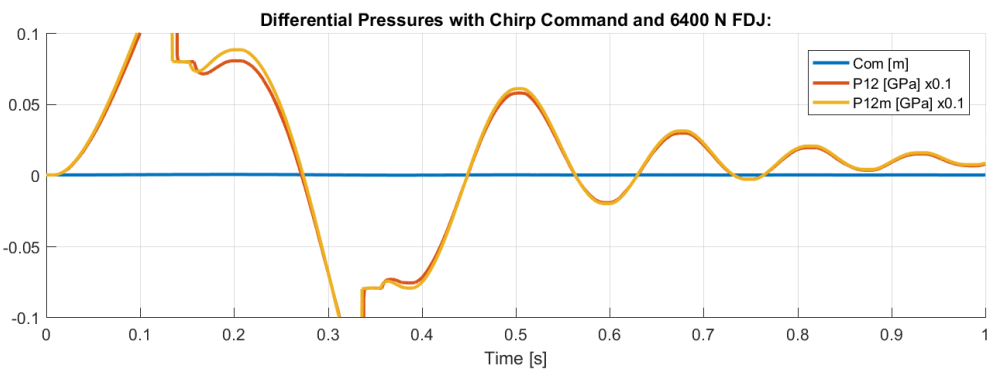
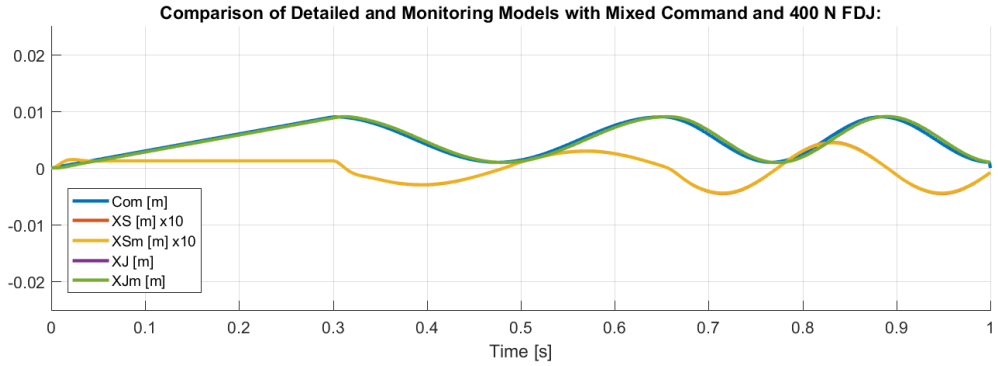


Figure 5.23: (a), (b), (c), (d)

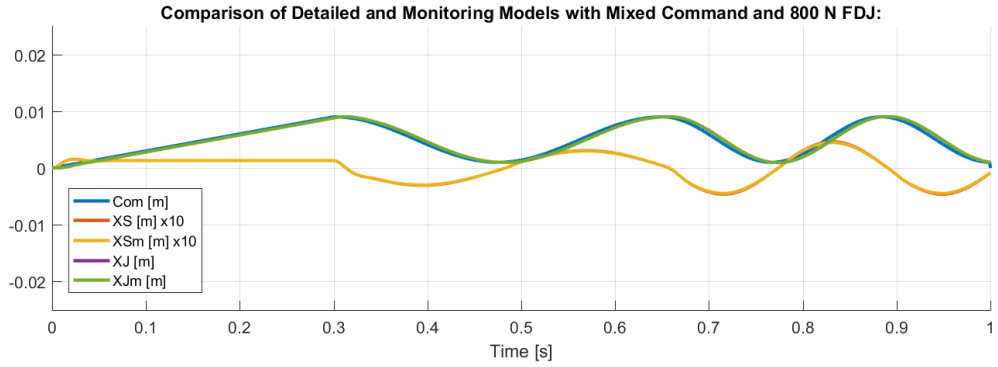
In figure 5.23*a* and 5.23*b* there are some discrepancies between the two model responses even in the higher frequencies. That emphasise an unpredictable behaviour in these conditions.

5.4.6 Dynamic friction increase with mixed command

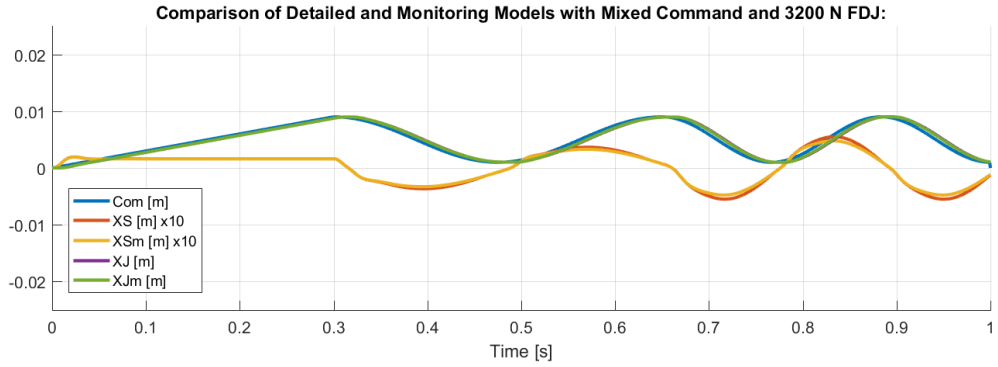
(a)



(b)



(c)



(d)

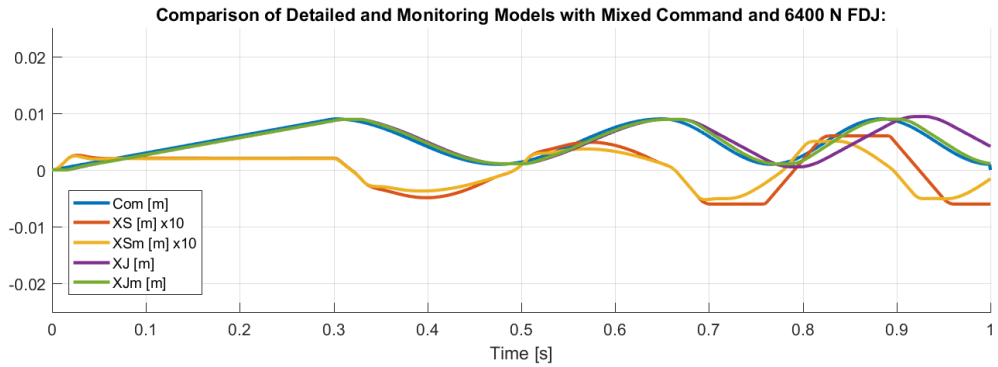


Figure 5.24: (a), (b), (c), (d)

The mixed command produces the most interesting results so far. In Figures 5.24*a*, 5.24*b* and 6.10*c* only small differences in the amount of the spool position overshoot are visible. However, in Figure 5.24*d*, it is evident how the minor overshoot prevents the spool from reaching its end-stroke, not reaching saturation condition and thus making the utilizer follow the command more closely. In reality, the utilizer has an evident delay in its response caused by the spool saturation, and this behaviour is not represented correctly in the monitoring model.

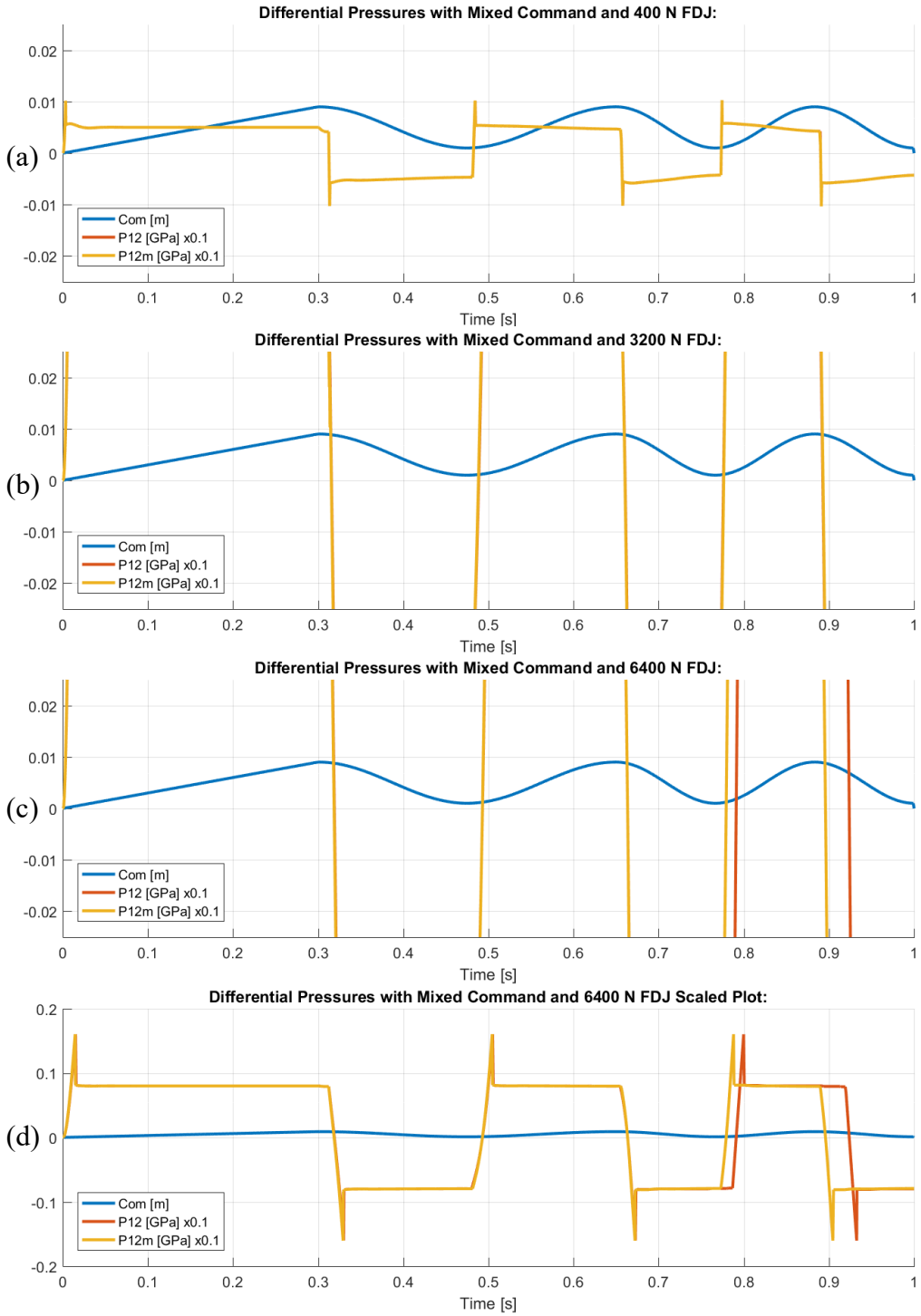
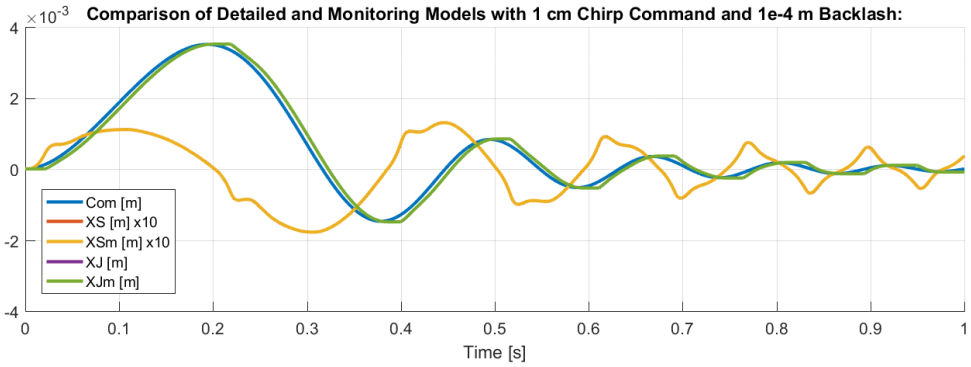


Figure 5.25: (a), (b), (c), (d)

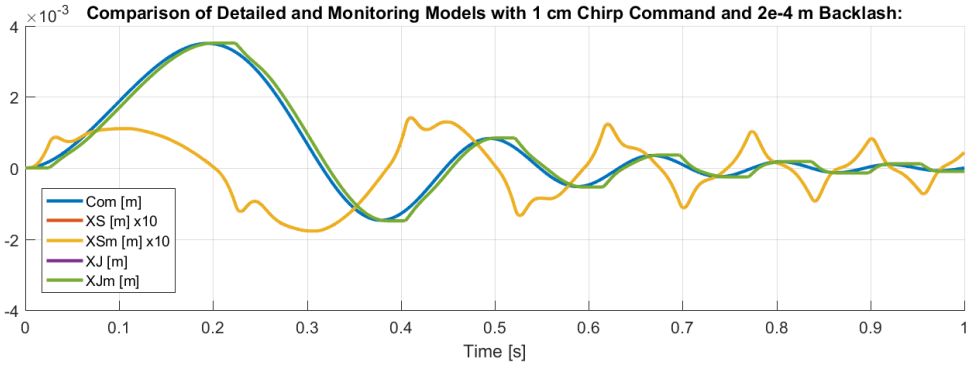
Between $400\ N < FDJ < 3200\ N$, represented with the same scale in Figures 5.25*a* and 5.25*b* respectively, no difference is perceivable. Figure 5.25*c,* instead, shows the same behaviour seen in Figure 5.25*d*, i.e. major discrepancies in differential pressures caused by 6400 *N* dynamic friction. To allow an easier interpretation of this phenomenon, a zoomed-out graph is presented in Figure 5.25*d*, which shows that the delay in the response is no longer acceptable for the detailed model.

5.4.7 Backlash with 1 cm chirp command

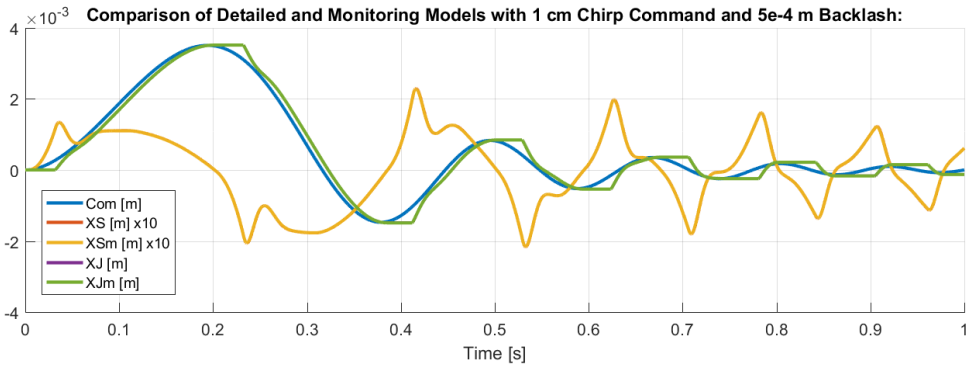
(a)



(b)



(c)



(d)

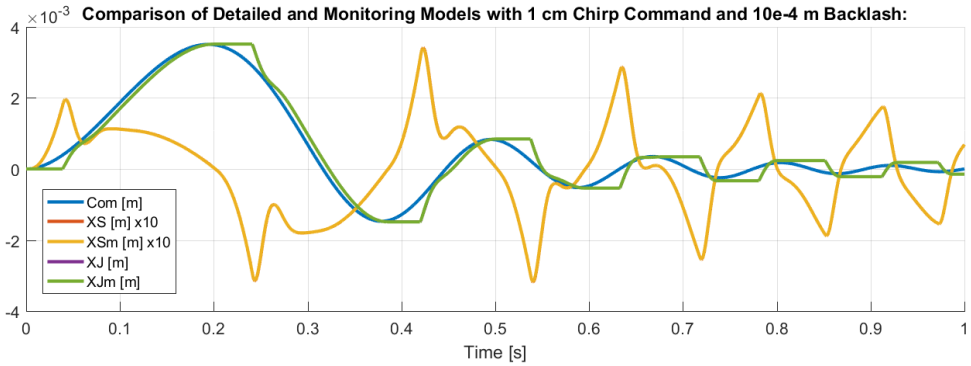
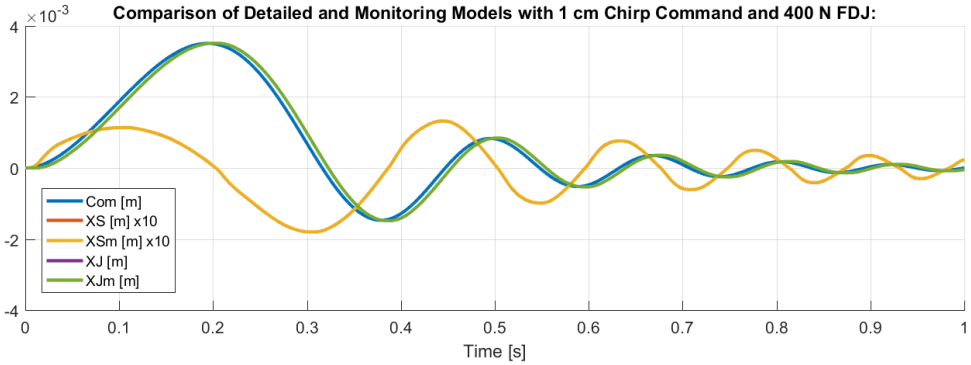


Figure 5.26: (a), (b), (c), (d)

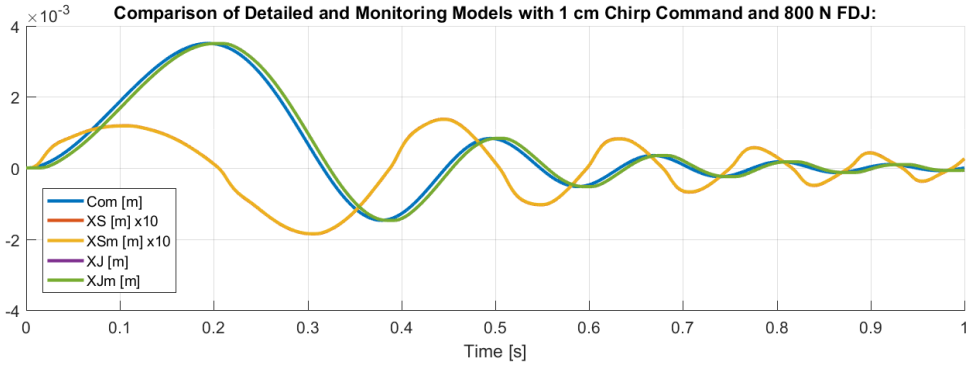
Here we compare this command with the one from Paragraph 5.4.2. The major difference between the two is the amount of initial amplitude, that was changed from the command mask. As we can see in Figure 5.26, the low fidelity model is able to perfectly reproduce the behaviour of the high fidelity model, even with high amounts of backlash involved, up to 1 *mm*.

5.4.8 Dynamic friction increase with 1 cm chirp command

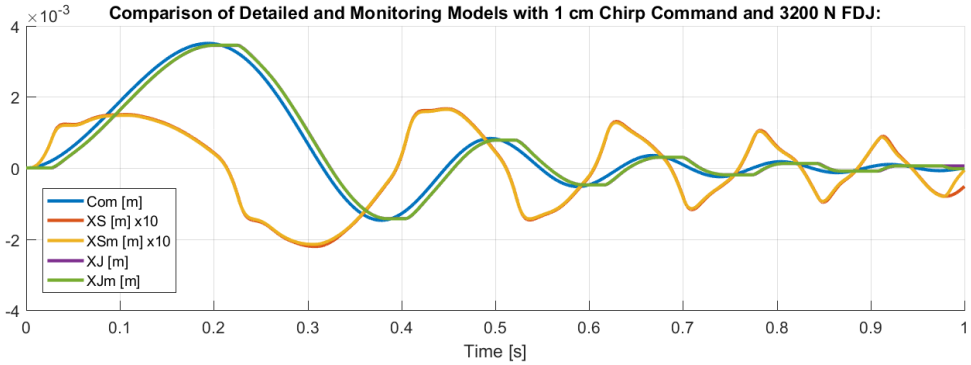
(a)



(b)



(c)



(d)

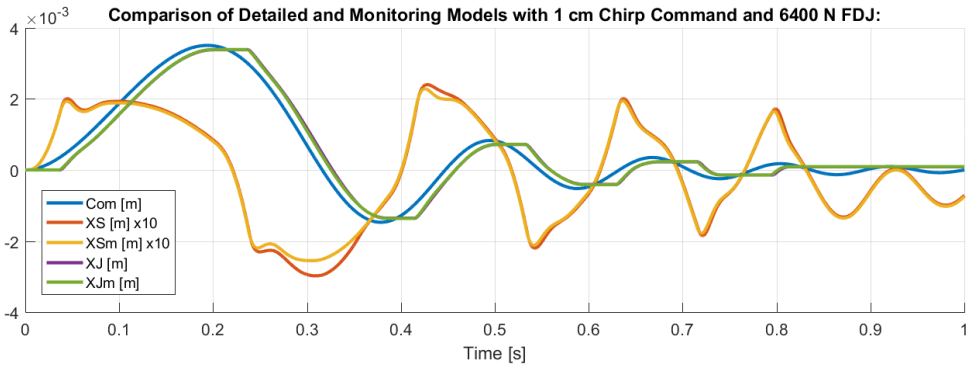


Figure 5.27: (a), (b), (c), (d)

In this set of simulations, we can see how the changed amplitude of the command has affected the response of the system, especially when compared to Figure 5.22. The utilizer is able to move without clogging even when high amounts of faults are applied, up to 3200 N of dynamic force. With 6400 N applied or more, we expect the system to behave in unforeseen ways. However, those results are not in our interest, as they are not useful for the purposes of the prognostic analysis. In Figure 5.27c we see only a little anomaly after 0.96 s on both the spool and the utilizer; we will further analyse it in Paragraph 6.3.1.

5.5 Response comparison of HF and LF models with multiple faults

Over a long operational lifespan, it is not uncommon to run into many faults simultaneously. In some cases, the effects of the faults can overlap, preventing the prognostic algorithm from recognizing them correctly. To help choosing the correct command for the algorithm and detect potential anomalies, two sets of simulations with multiple faults are run, one for each chosen input signal (mixed command and 1 cm amplitude chirp command).

Both sets of simulations are structured as a 3x5 matrix, combining the most notable amounts of faults already shown in the previous paragraphs:

- $0.1\ mm$ backlash;
- $0.5\ mm$ backlash;
- $1\ mm$ backlash;

and:

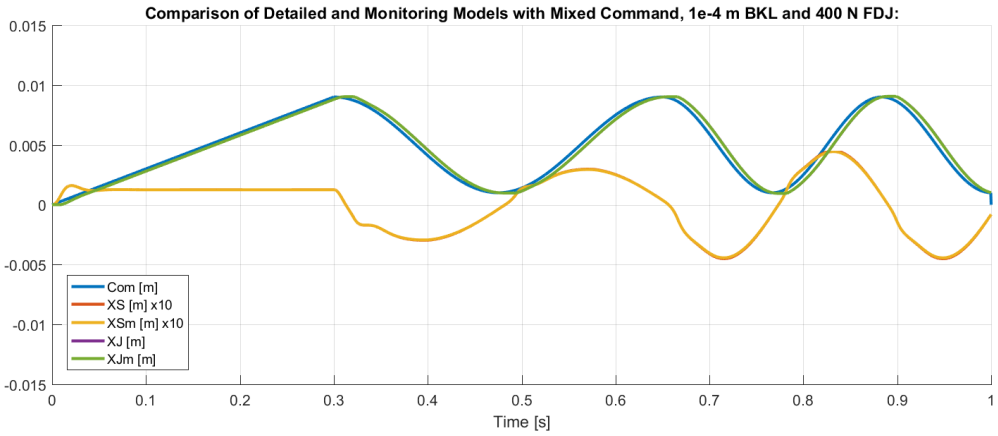
- $400\ N$ dynamic friction;
- $800\ N$ dynamic friction;
- $1600\ N$ dynamic friction;
- $3200\ N$ dynamic friction;
- $6400\ N$ dynamic friction.

Each page – corresponding to a different value of dynamic friction – will have three different graphs, one for every backlash value.

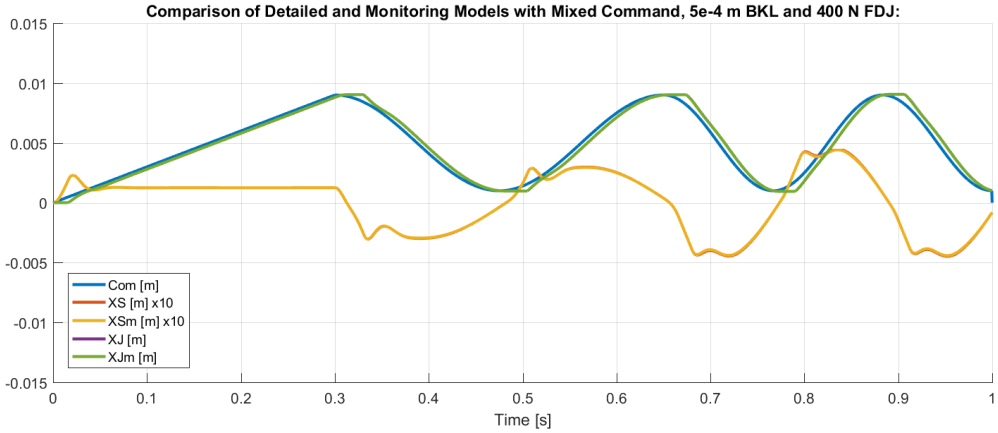
A description and comparison of the results can be found in the next the chapter (Paragraph 6.3.1).

5.5.1 Mixed command

(a)



(b)



(c)

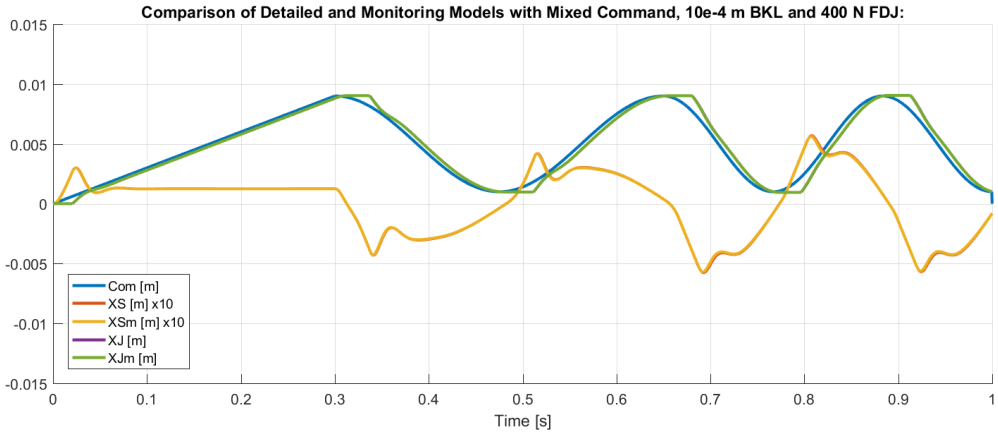
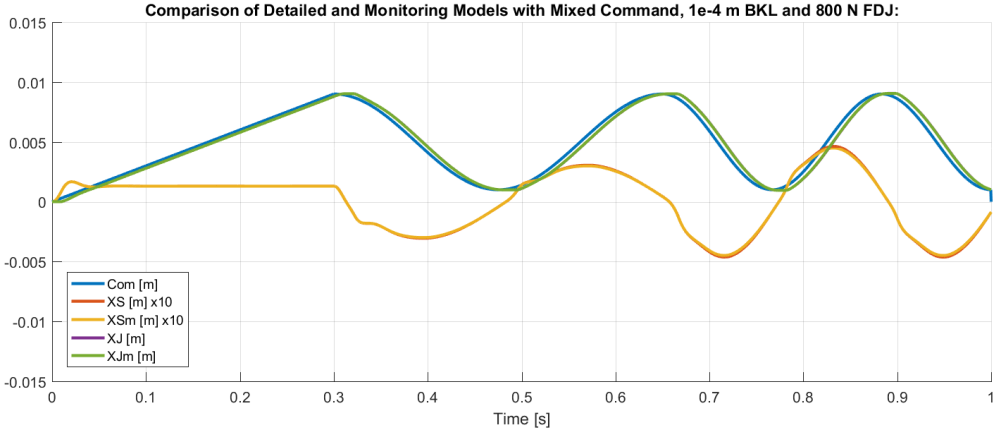
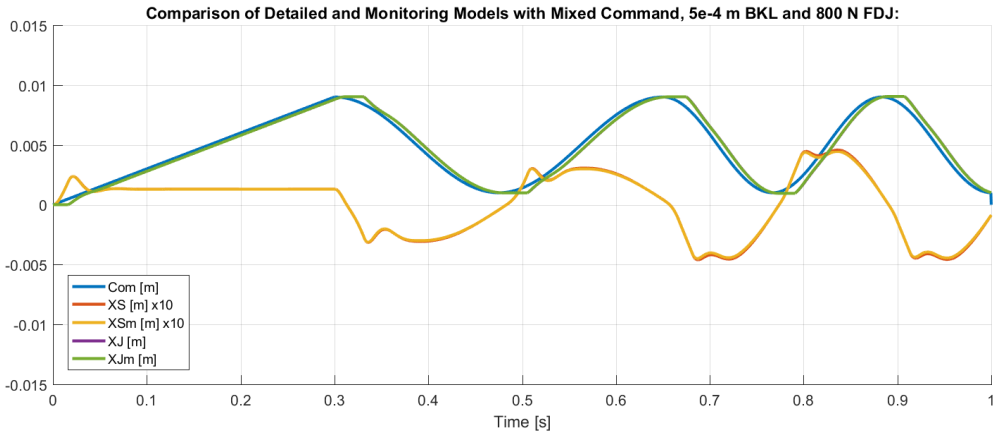


Figure 5.28: (a), (b), (c)

(a)



(b)



(c)

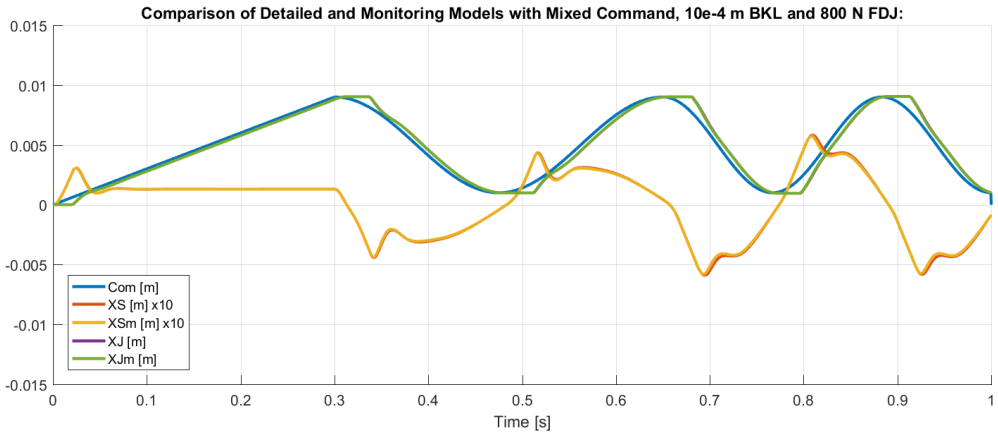
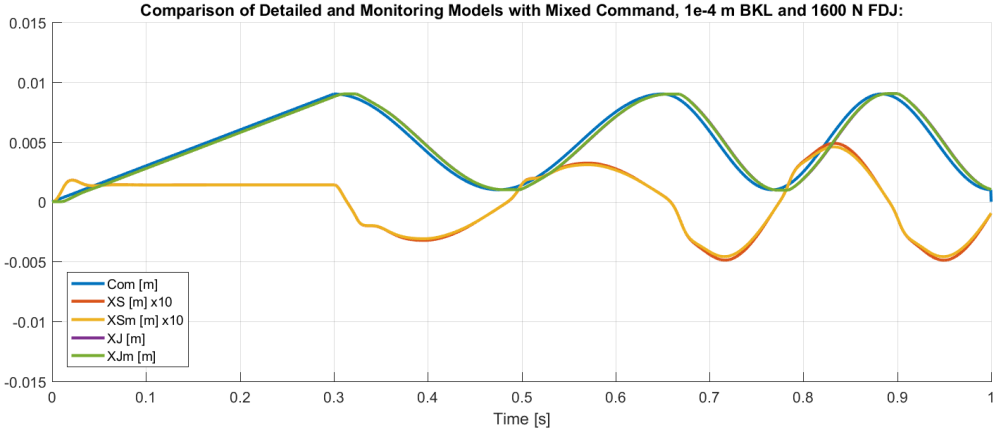
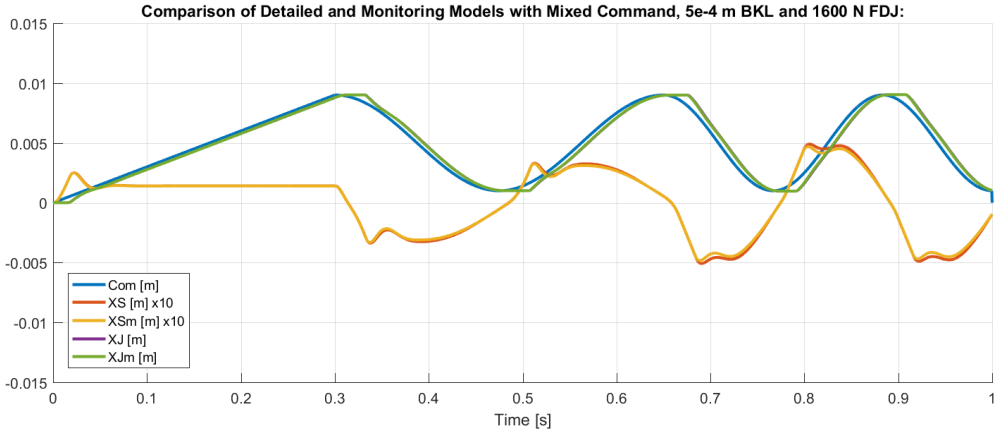


Figure 5.29: (a), (b), (c)

(a)



(b)



(c)

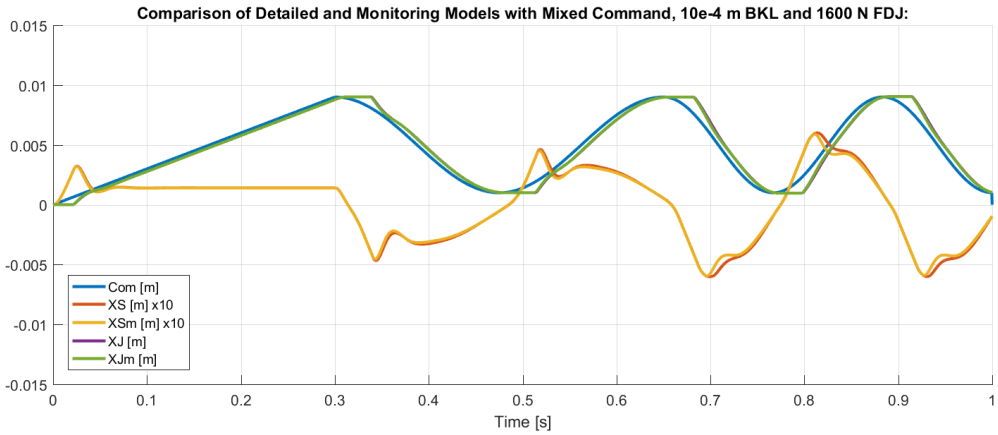
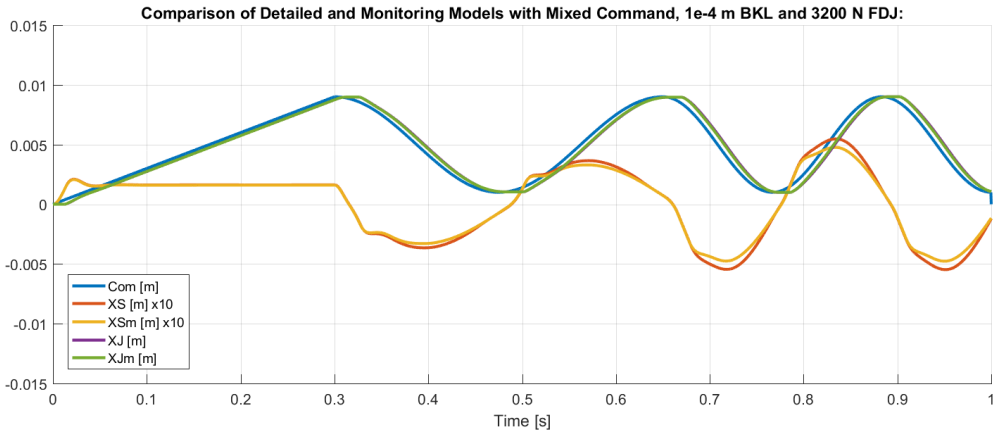
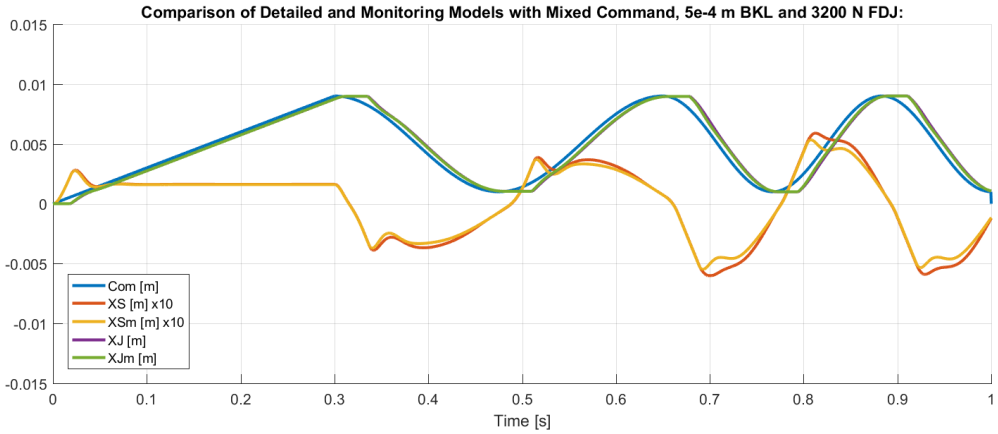


Figure 5.30: (a), (b), (c)

(a)



(b)



(c)

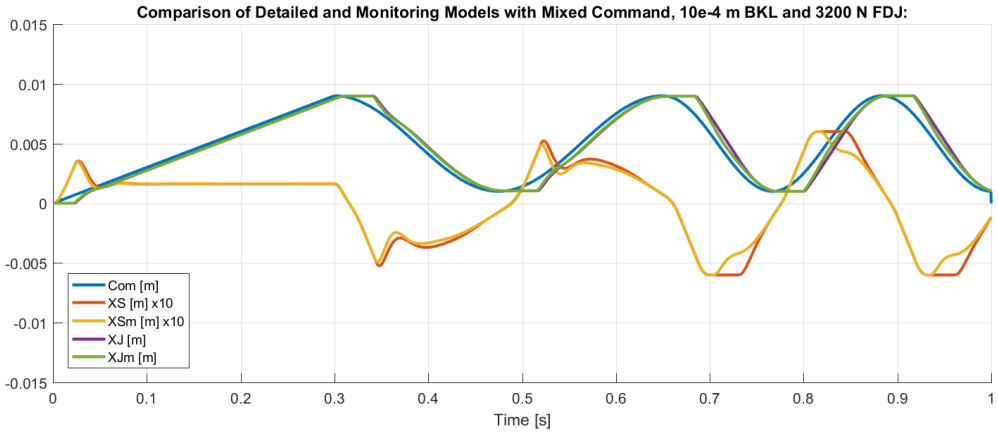
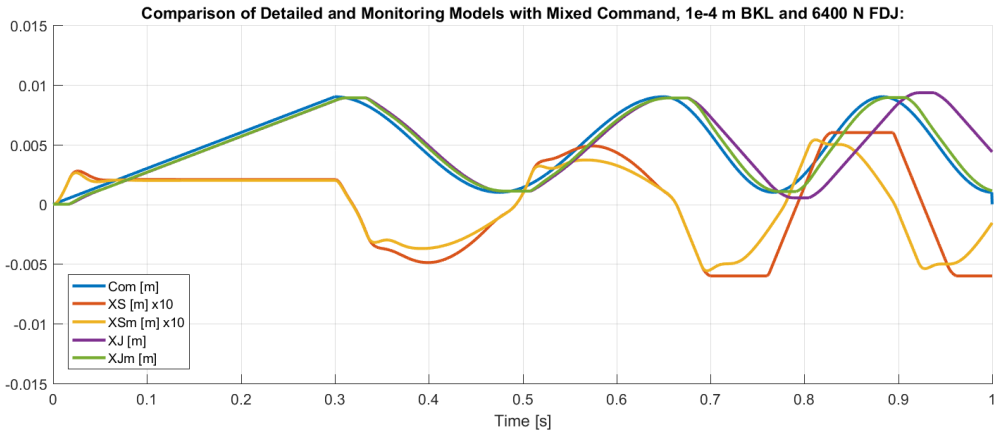
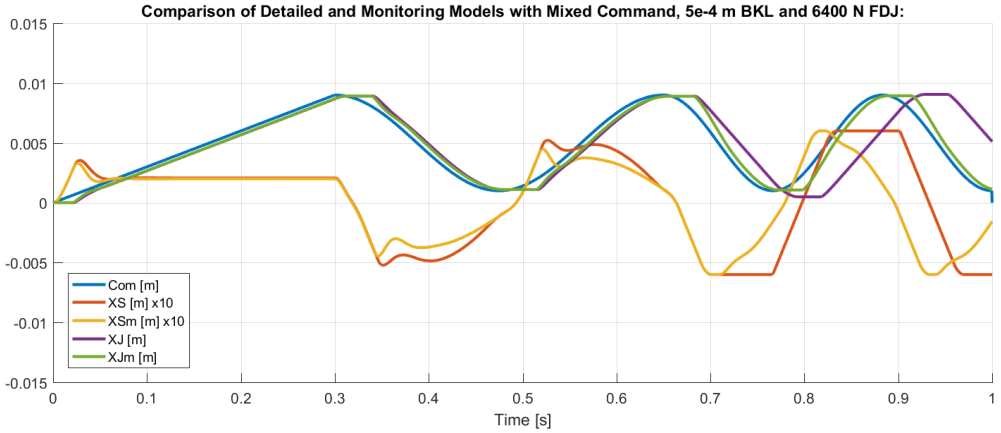


Figure 5.31: (a), (b), (c)

(a)



(b)



(c)

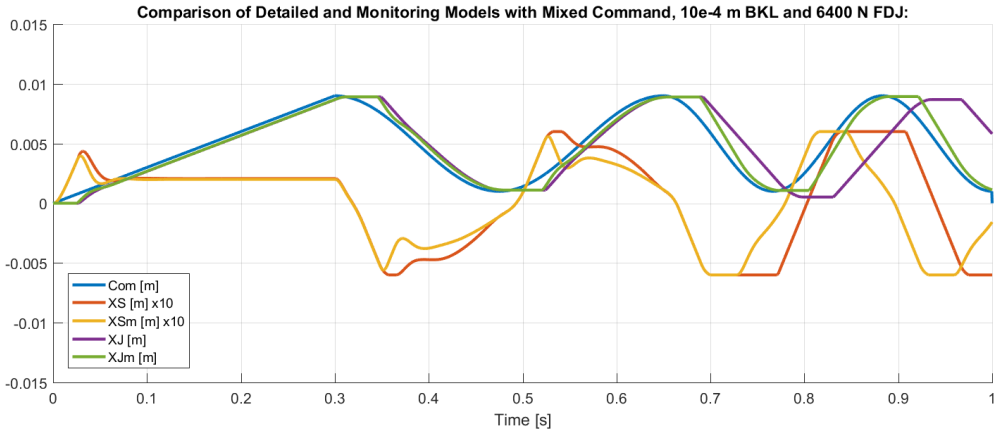
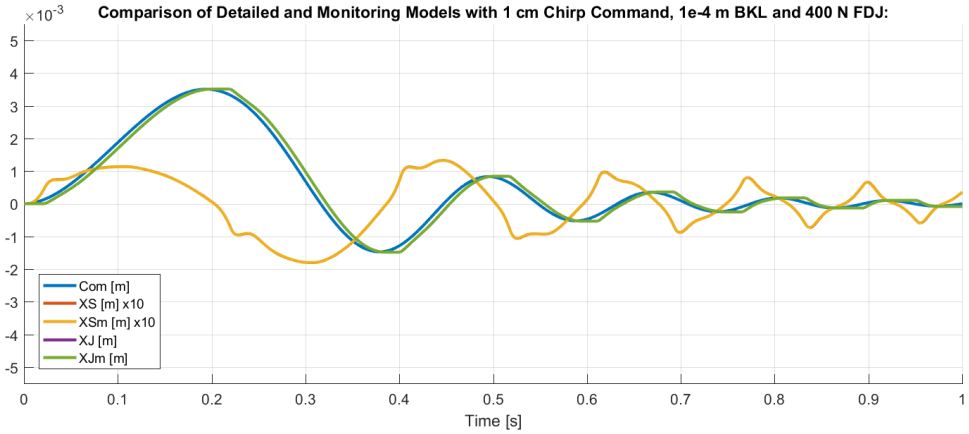


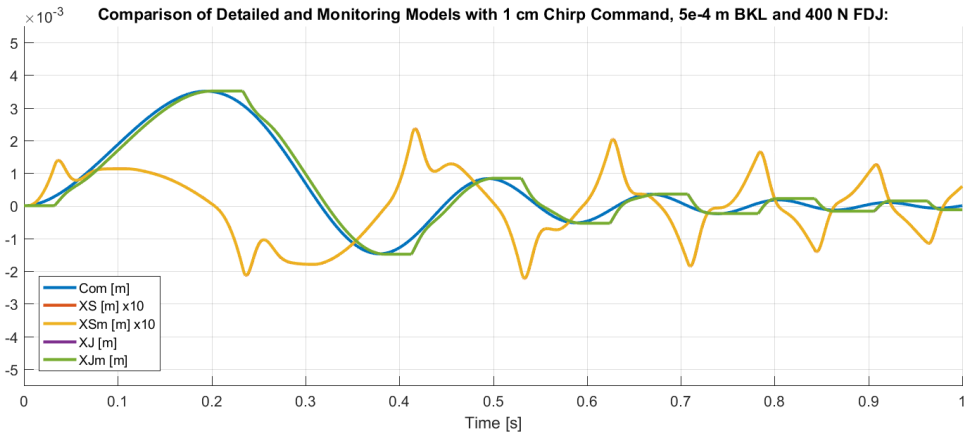
Figure 5.32: (a), (b), (c)

5.5.2 1 cm chirp command

(a)



(b)



(c)

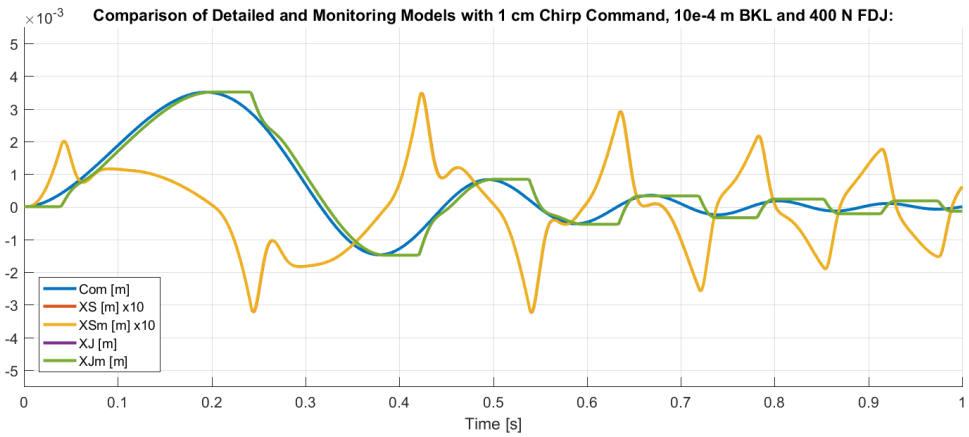
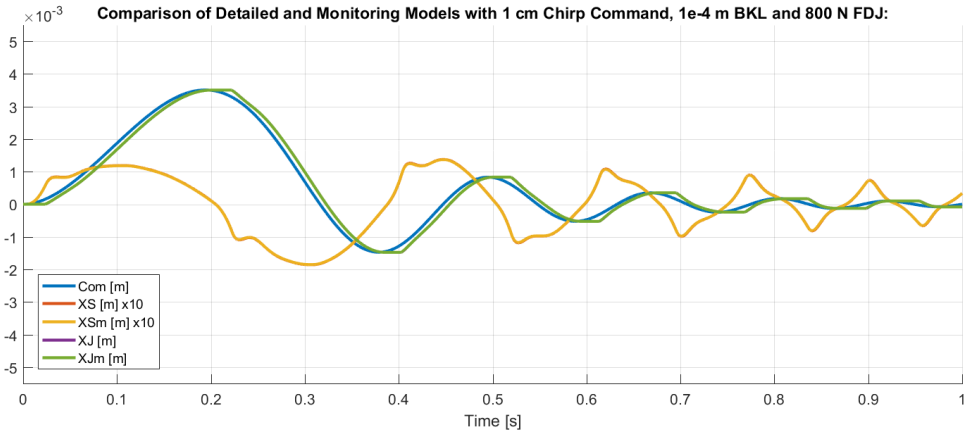
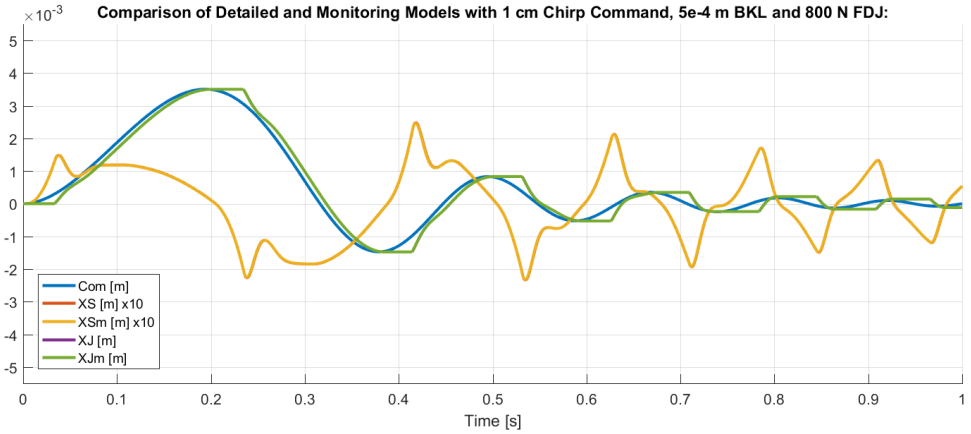


Figure 5.33: (a), (b), (c)

(a)



(b)



(c)

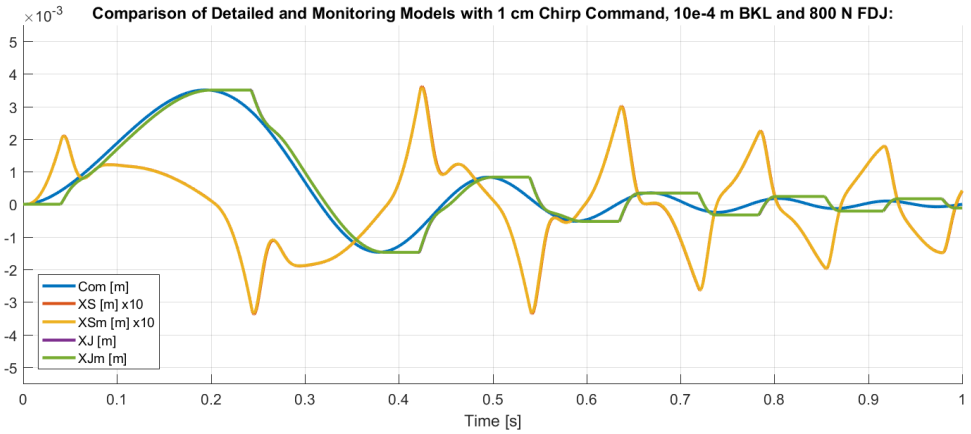
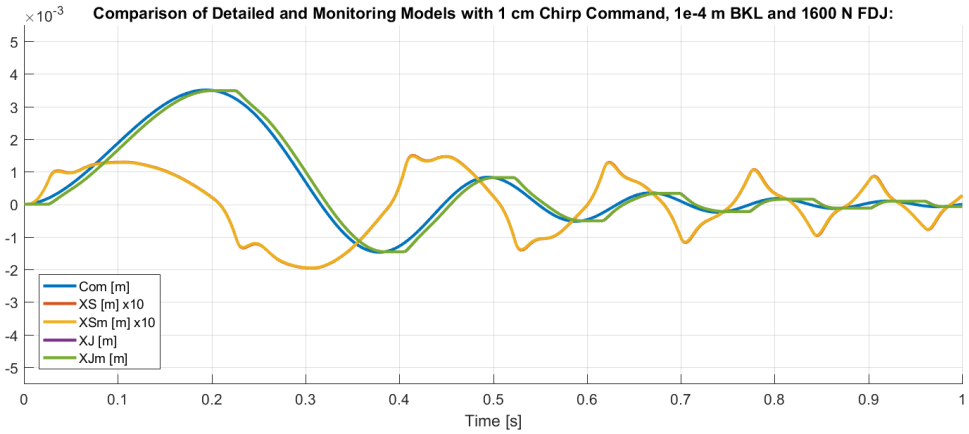
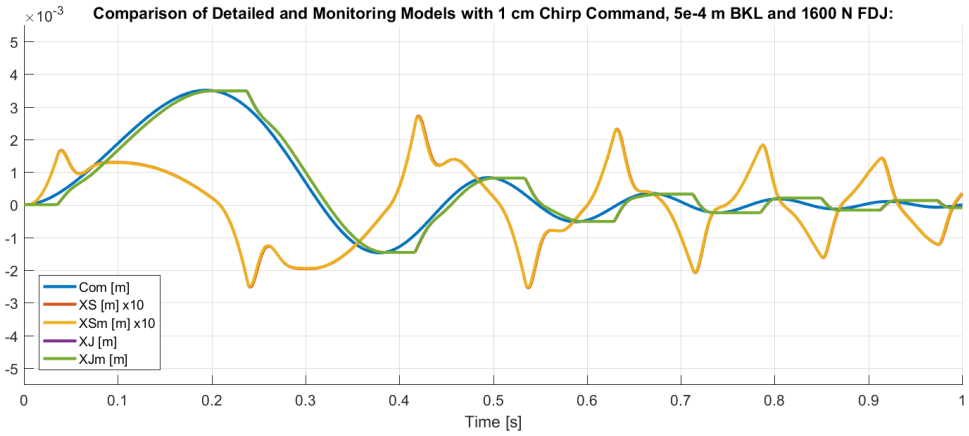


Figure 5.34: (a), (b), (c)

(a)



(b)



(c)

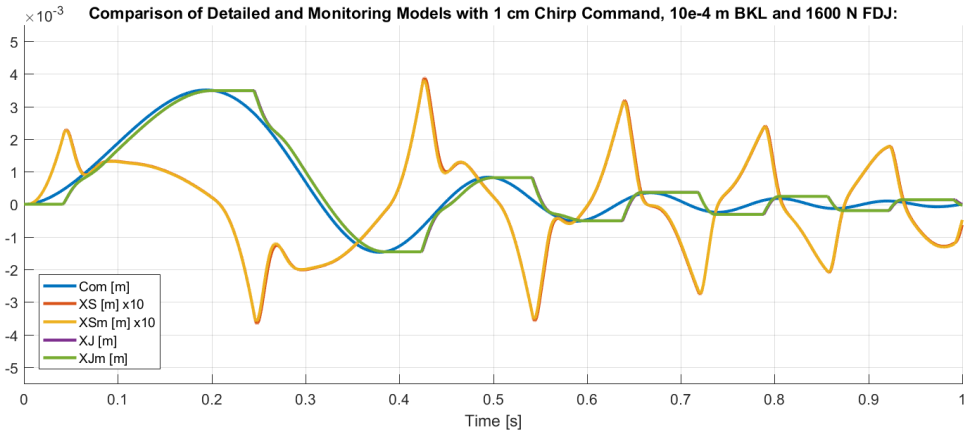
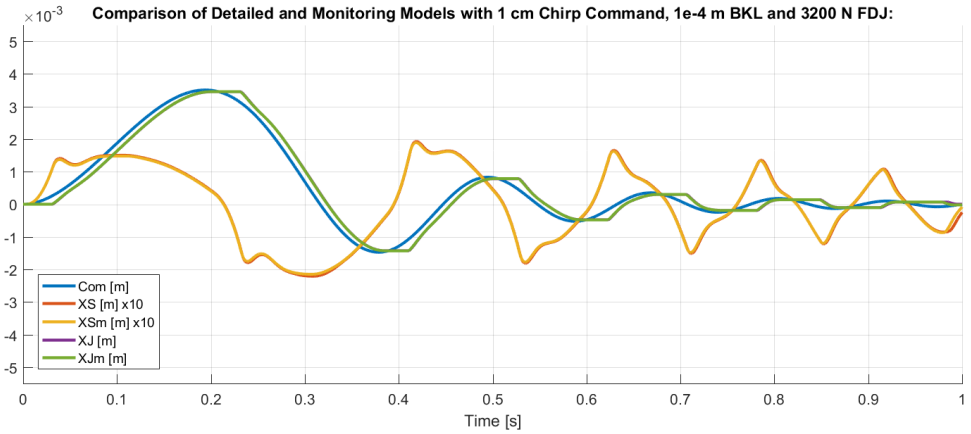
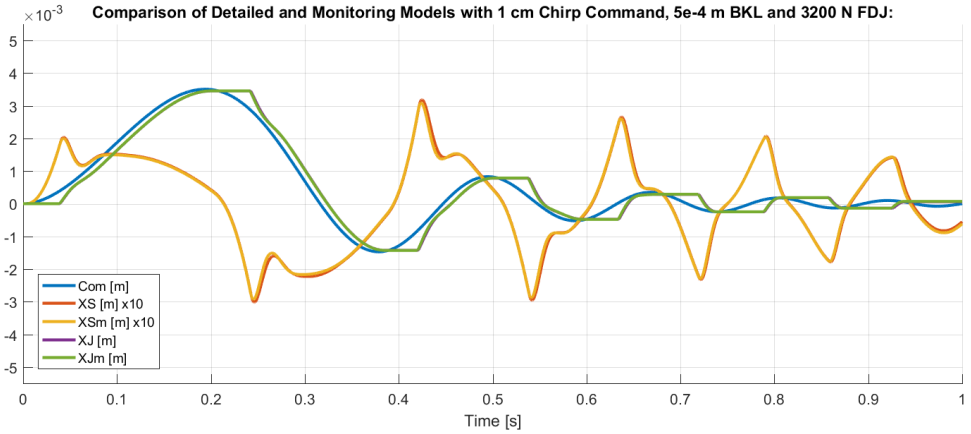


Figure 5.35: (a), (b), (c)

(a)



(b)



(c)

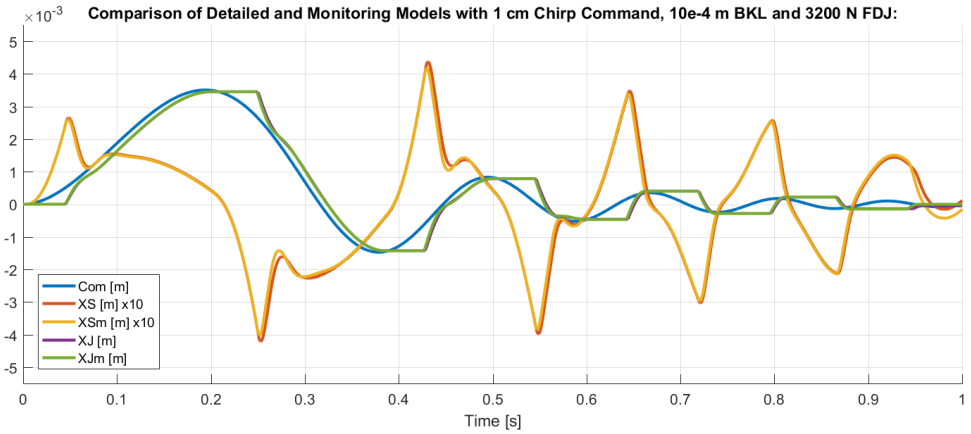
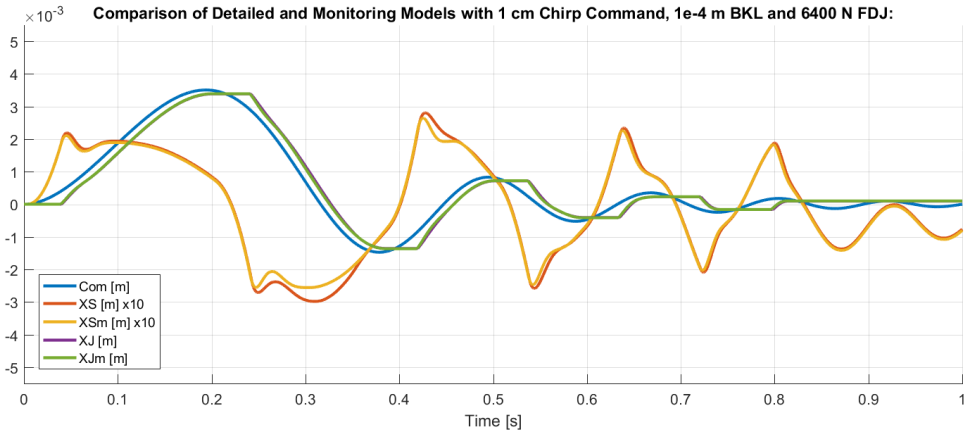
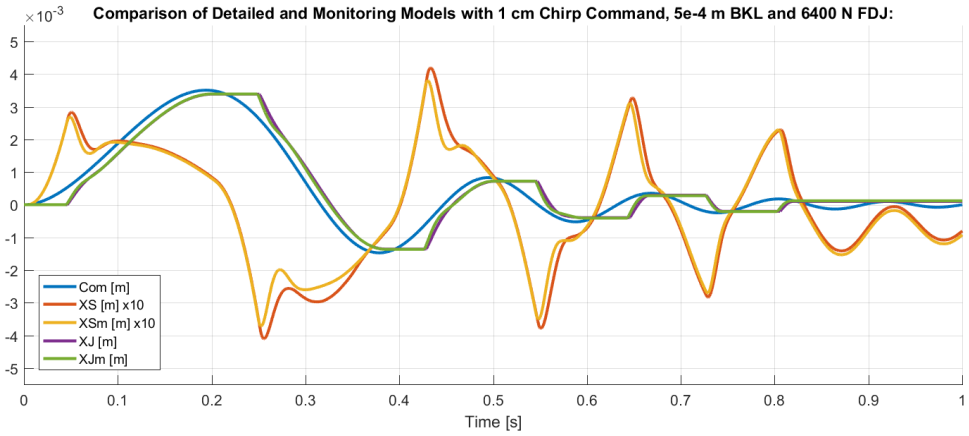


Figure 5.36: (a), (b), (c)

(a)



(b)



(c)

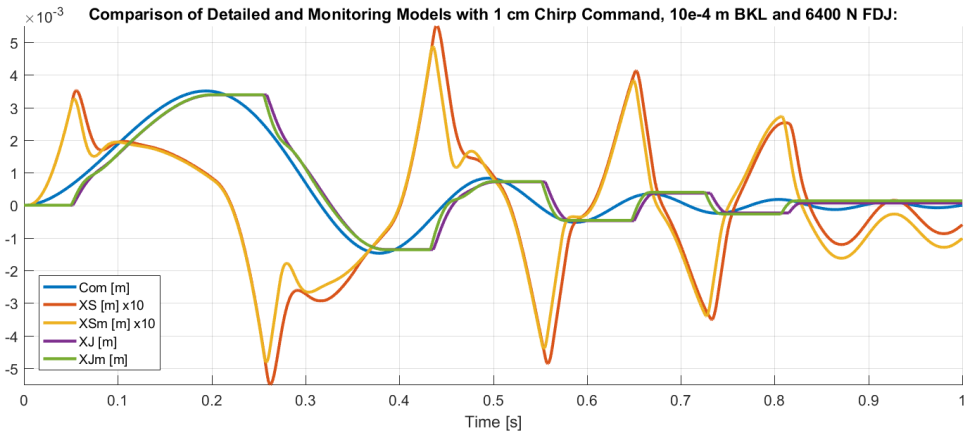


Figure 5.37: (a), (b), (c)

6 PROGNOSTIC ANALYSIS

Prognostics is an advanced scientific discipline whose purpose is to predict critical failures of an operational component or system and to compute its remaining lifespan through advanced algorithms after a detailed monitoring of its dynamic response and performances over time, detecting evidences of incipient faults and allowing predictive maintenance.

The prognostic design philosophy, as opposed to the diagnostic one, is a relatively new area of interest in the aerospace environment. Its origin, like many other scientific processes, lies in the medical field of study and is used as a means to reduce maintenance costs and to raise the reliability, the safety and the performances of a system.

Fault detection and identification is the basis of prognostic analysis ([3],[4]). Through this process, the incipient faults can be recognized by comparing the two models' responses. The comparison is driven by an objective function, whose choice is very critical, as it can vary the results drastically. Then, the results can be used when computing the remaining useful life (RUL) ([5],[6]) – the time span of operational life residue before maintenance or replacement – of the system or component.

This is classified as an optimization problem. It is resolved by means of diverse and complex algorithms, specifically chosen depending on the problem and the researched solution.

Many algorithms for fault identification have been developed, e.g. neural networks[7].

The one that was chosen for this work is a genetic algorithm (GA) [8], which is part of the family of evolutionary algorithms (EA), a metaheuristic influenced by the biological process of natural selection.

6.1 Genetic algorithms

The natural selection is an evolution mechanism, whose concept was introduced by Charles Darwin in the XIX century, which suggests that organisms excogitate methods for survival transmitting only the best fitting characteristics for the environment in which they live.

John Henry Holland, professor of electrical engineering and computer science at the University of Michigan, thought that these evolution mechanisms could be used in programming to resolve problems not yet completely understood, generating populations of results ever-improving with each generation. These were called genetic algorithms.

A generic structure of a genetic algorithm can be seen in Figure 6.1.

An initial population of solutions (also known as individuals or phenotypes) is chosen randomly within a determined range. Each solution will have a set of characteristics, analogue to the genome – a complete set of genetic information which tells an organism how to function – and represented by a fault value. The objective function (or fitness function) tells if those solutions fit the reference model, and if they are valuable to the problem resolution. If so, they can likely become part of the next generation by reproduction and recombination of the genotypes. The objective function we chose is a Root Mean Squared Error (RMSE):

$$RMSE = err = \sqrt{\frac{\sum_{i=1}^{N}(X_N - ref)^2}{N}}$$

When another generation is being produced, some mutations are introduced. If they satisfy the objective function requirements, and if the fitness functions identifies them as possible solutions, they can be passed on to the next generations. If not, they are discarded. This process is repeated until convergence is reached.

The elapsed time of the process depends on many factors, some of which are:

- the chosen objective function;
- the number of solutions of the population;
- the range in which the solutions can change;
- the complexity of the simulated model (i.e. the low fidelity model);
- the input command chosen for the model;
- the initial faults chosen for the simulation;
- the presence of discontinuities or cycle limits in the simulation;
- the integration interval (if fixed-type);
- the time of the simulation of the model.

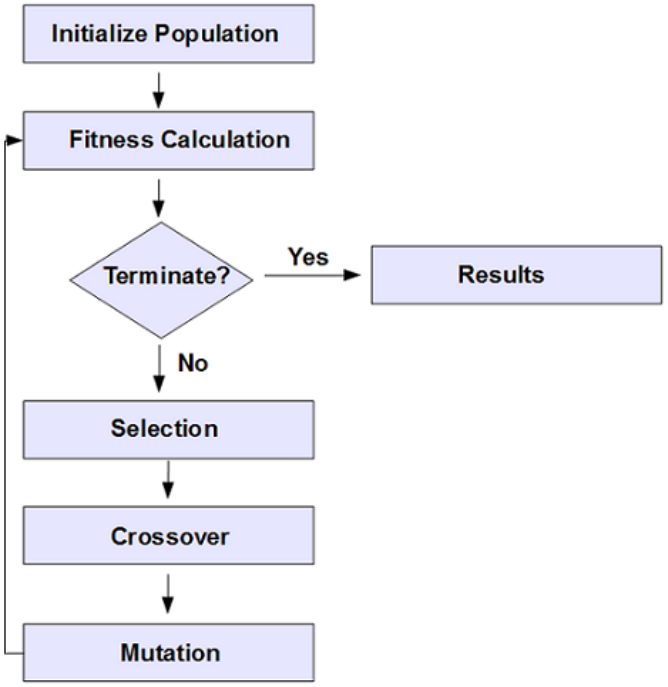


Figure 6.1: Typical genetic algorithm flow chart

6.2 Scripts description

The scripts used for the simulations have been developed by Pier Carlo Berri for his thesis work about servo mechanical actuators [9] using the Matlab Optimtool, a GUI designed to solve optimization problems, and adapted for the purposes of this work.

They are two:

- $main.m$, which contains the core functions and parameters for the simulation of both models (cf. Appendix B);
- $objfcn.m$, which is an external function that collects the solutions and contains the objective function used for their selection (cf. Appendix C).

The script $EHA\_Dat.m$ contains all the parameters and the constants already used in the fault analysis.

The script $main.m$ (Appendix B) loads the workspace by running the file $EHA\_Dat.m$. Then, an initial set of faults is entered by changing their relative variables BKL and FDJ, depending on the needs. The HF model is run and then used as reference by the algorithm. The second section of the script is dedicated to the fault detection and identification, and it implements the algorithm parameters (e.g. number of variables, population size, boundaries, etc.), the objective function handle and the Optimtool functions. This way, the toolbox GUI does not need to be opened.

For the purpose of the simulation, the two Simulink models have been subdivided into two different files ($Mod\_EHA\_LF.slx$ and $Mod\_EHA\_HF.slx$), making sure that the same command is used and the output variable XJ of the first is being sent to the Matlab workspace for each step.

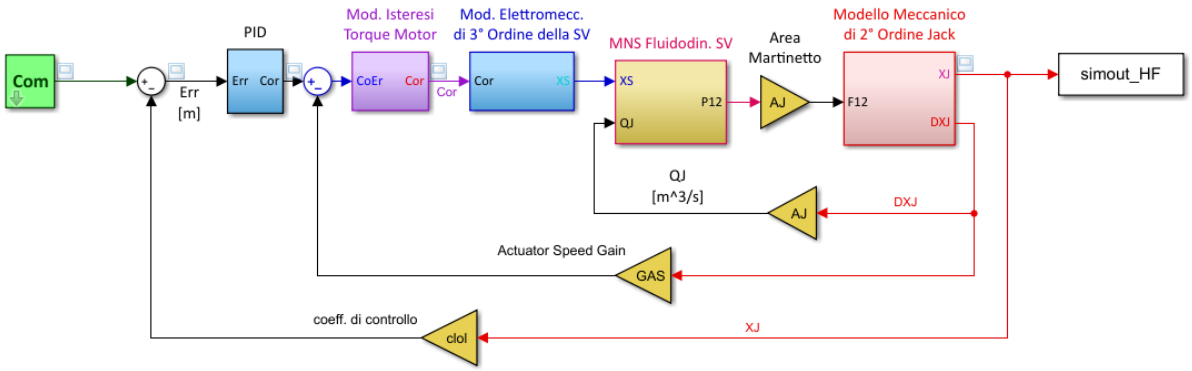


Figure 6.2: High-fidelity model (cf. §4)

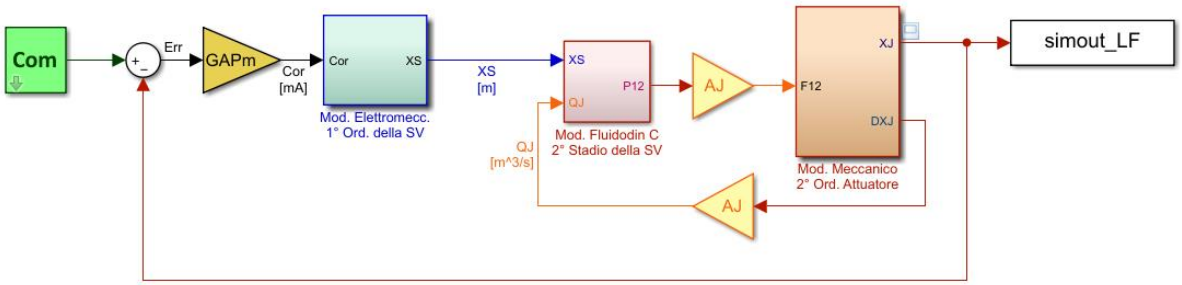


Figure 6.3: Low-fidelity model (cf §4.1)

The fundamental sample time (also called integration step) is the same as the one used for the single and multiple fault analysis, i.e. $10^{-5}\ s$ over $1\ s$ of simulation time.

The output of the simulation is the value of the faults, decided by the fitness function, that can be reintroduced in the LF model to obtain a response as similar to the HF model as possible. If these are of comparable scale to the initial amounts, and differences in the two curves plotted are not recognizable, the fault identification problem has found its solutions. If not, changes have to be made to the simulation parameters. They can be the population size, the fault amount, their boundaries, or the input command.

6.3 Simulations and results

The simulations have been divided into two sections according to the command we used.

6.3.1 Mixed command

The mixed command was the first choice, because of its apparently better results in comparison with the other commands (e.g. sinusoidal, step, low amplitude chirp, etc.).

Below, the first simulations are introduced and commented.

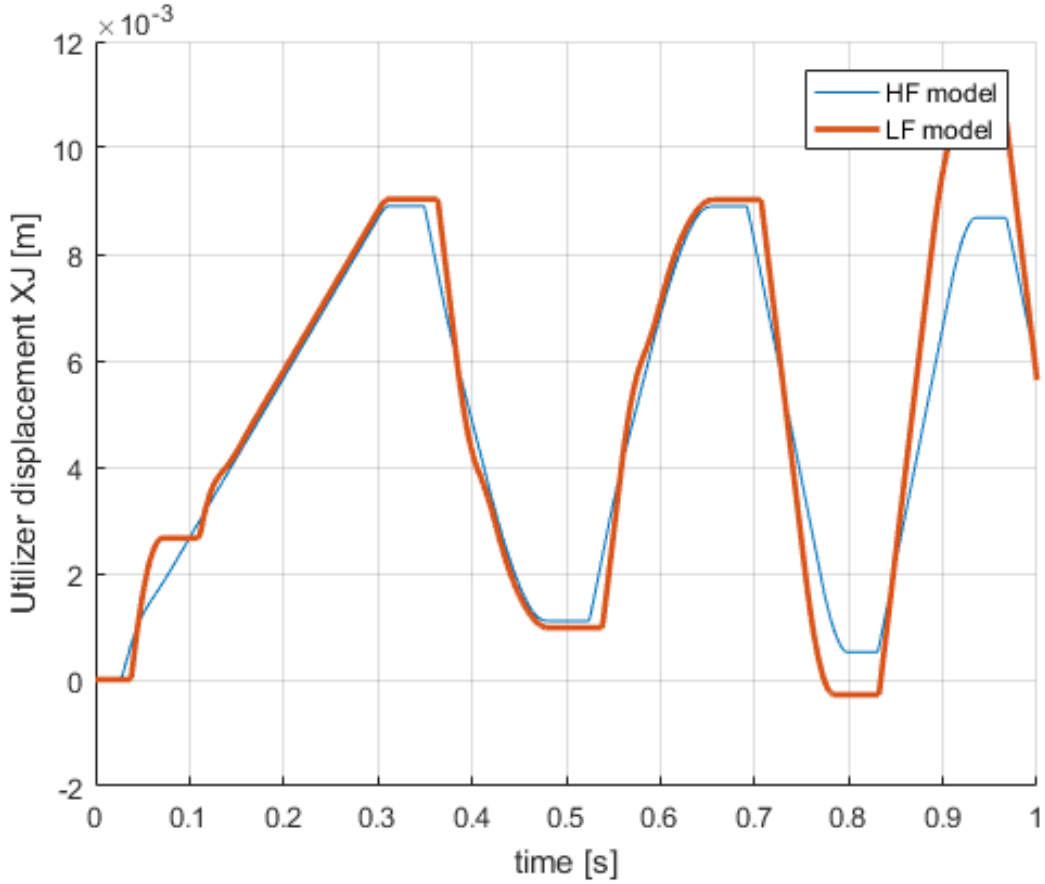


Figure 6.4:Case 1 – 10e-4 m backlash and 6400 N FDJ

The simulation shown in Figure 6.4 clearly shows major deficiencies between the two curves. The LF model is not able to correctly reproduce the HF model behaviour, as the fitness function cannot find the right solutions. The fault values detected by the algorithm are:

- Backlash length: $0.0046792\ m$
- Dynamic friction: $2885.5684\ N$

The causes of the massive discrepancies can be several, but the most notable one is certainly the choice of the amounts of faults applied: in fact, this result was expected. Looking at Figure 5.32*c,* it becomes evident that the response was contradictory from the beginning and better results were not obtainable. The simulation time was also relatively high: $3532\ s$ (approximately one hour). Increasing the population number from 20 to 50, no difference in the result is found, except for the simulation time, which goes up to $10208\ s$ ($3\ h$). In conclusion, the HF and the LF models need comparable dynamic responses for the optimization algorithm to work effectively. To test how much discrepancy is tolerable by the algorithm, the fault amounts were lowered.

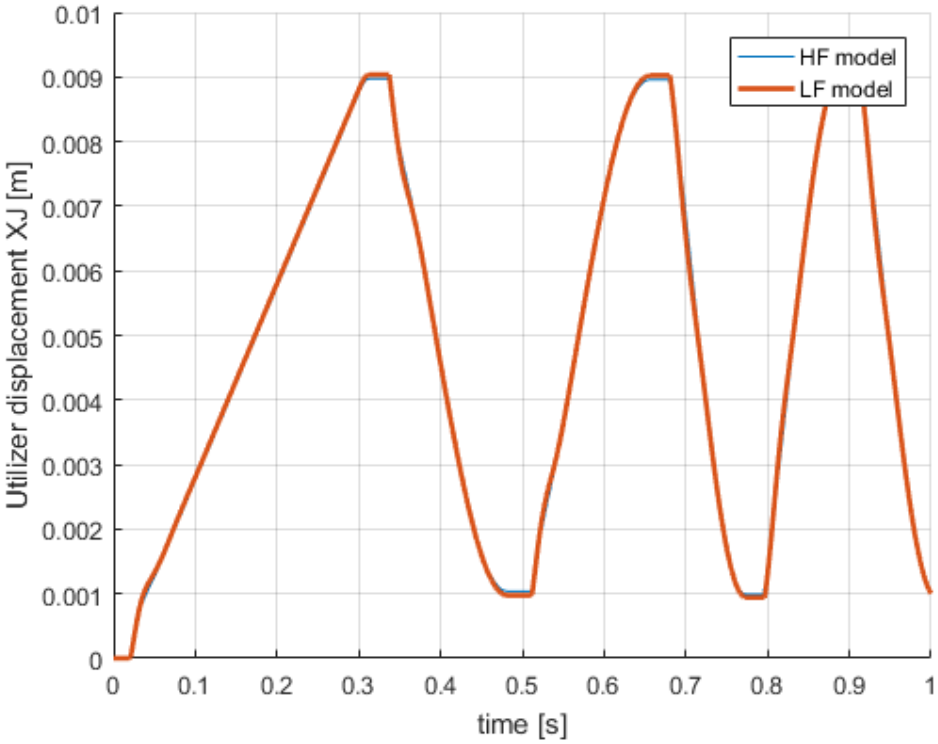


Figure 6.6: 5e-4 m backlash and 3200 N FDJ

The result in Figure 6.5 is more acceptable than the previous one, but there are still some discrepancies near the peaks, and the fault amounts detected by the fitness function were not close to the ones introduced in the HF model.

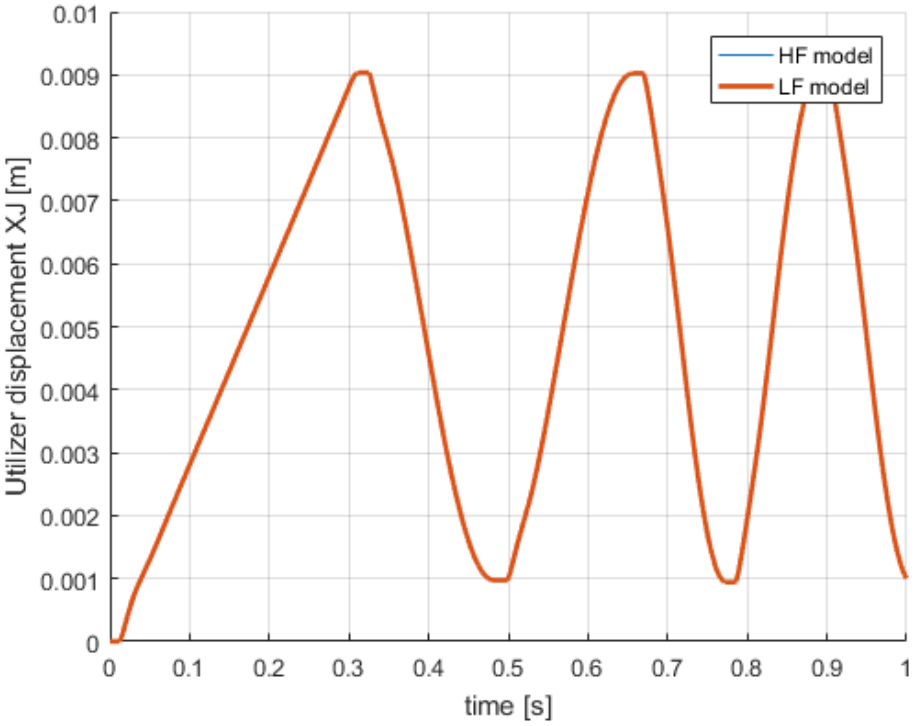


Figure 6.5:Case 2 – 2e-4 m backlash and 800 N FDJ

In Figure 6.6, the fault amounts were lowered again, and the curves match almost perfectly. However, the fault amounts detected by the algorithms are:

- Backlash length: $0.00045249\ m$
- Dynamic friction: $1070.98\ N$

both very different from the fault values entered in the HF model.

The algorithm is not able to recognize accurately enough the faults and thus considered not suited for mixed faults. For this reason, the 1 cm amplitude chirp command has been used instead.

6.3.2 1 cm chirp command

Judging from the mixed fault simulations reported in Paragraph 5.5.2, the command is able to simulate the behaviour of the HF model better. However, in Figure 5.36c, after $0.95\ s$, an anomaly caused by the low amplitude of the command is shown. For this reason, a small change to the command has been made. From the command mask in both models, the final frequency that the chirp reaches at the end of the simulation ($1\ s$) is set to $1.5\ Hz$, thus producing a shift in the dynamic response (unaltered) but omitting the part of the signal that shows the anomaly.

To verify the quality of the chosen command, a set of single faults simulations is paired with the simulation results and then discussed.

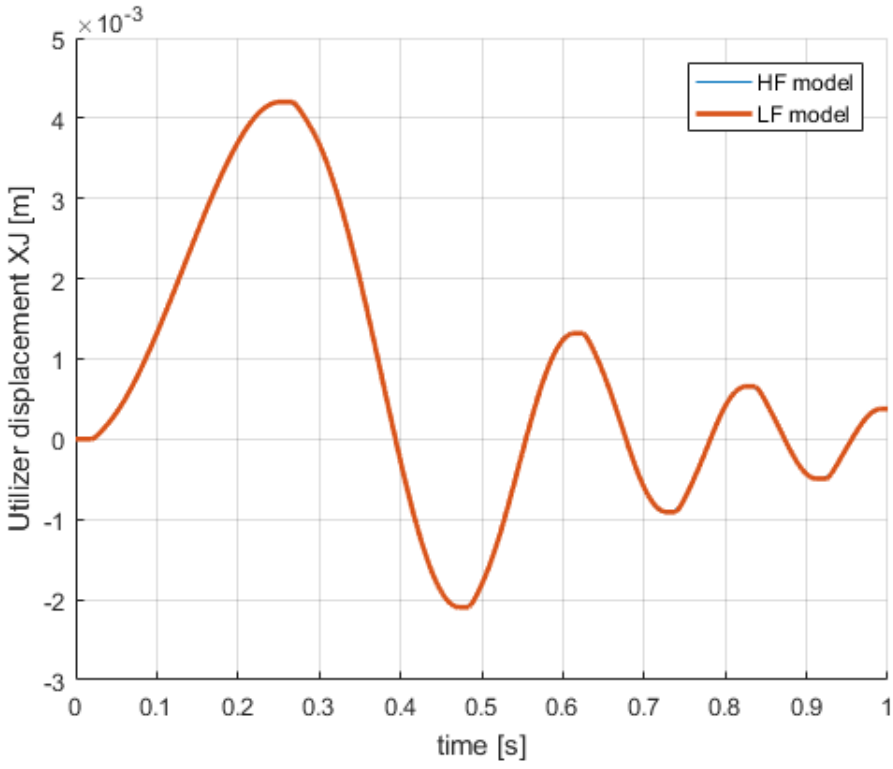


Figure 6.7: Case 3 – 0 m backlash and 400 N FDJ

- Backlash length: $3.3477e-05\ m$
- Dynamic friction: $554.223\ N$

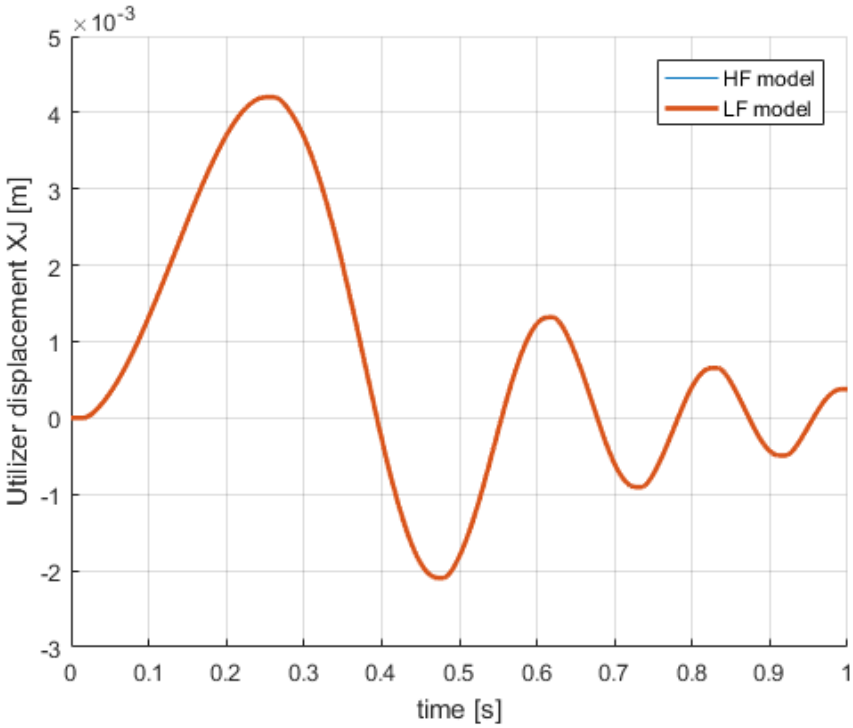


Figure 6.8: Case 4 – 0 m backlash and 800 N FDJ

- Backlash length: $8.5831e-06\ m$
- Dynamic friction: $596.9125\ N$

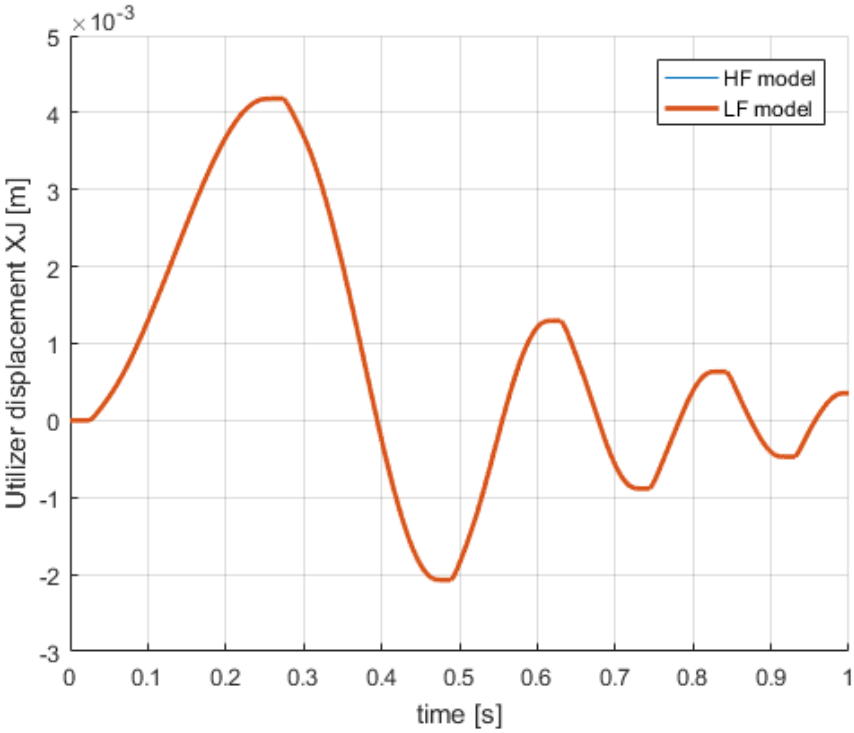


Figure 6.9: Case 5 – 0 m backlash and 1600 N FDJ

- Backlash length: $3.634e-05\ m$
- Dynamic friction: $1799.0696\ N$

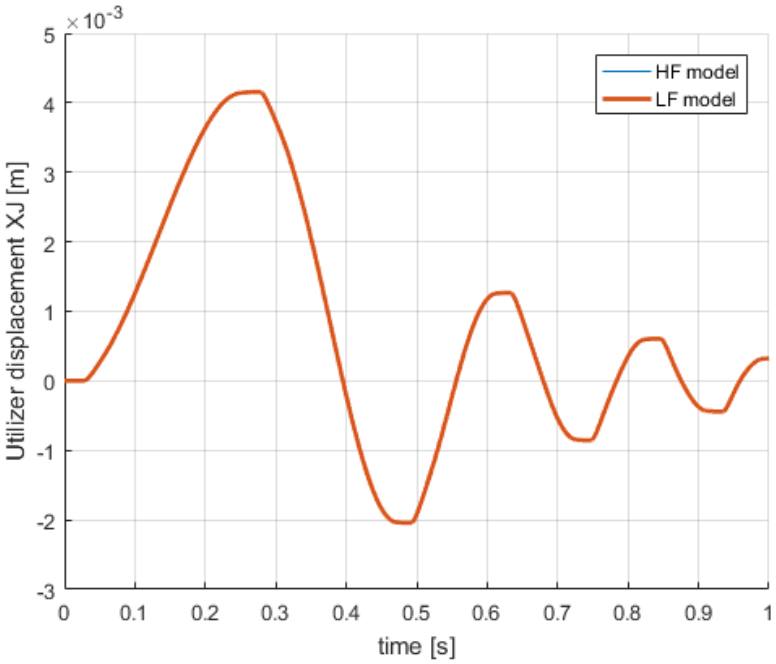


Figure 6.10: Case 6 – 0 backlash and 3200 N FDJ

- Backlash length: $3.0162e-05\ m$
- Dynamic friction: $3567.2466\ N$

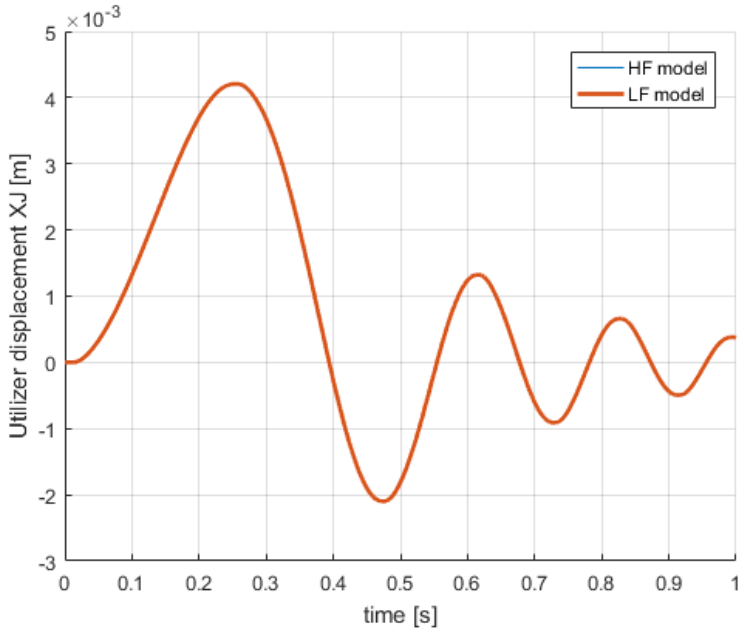


Figure 6.11: Case 7 – 0 m backlash and 400 N FDJ with lower boundaries

- Backlash length: $0\ m$
- Dynamic friction: $421.8742\ N$

The above-mentioned single faults are able to recognize the lack of backlash, resulting in extremely small backlash values. However, the algorithm tends to overestimate the amount of dynamic friction of the HF model. This is probably caused by the boundaries set for the simulations – between 200 and $4000\ N$ for the dynamic friction. More simulations were conducted setting a lower amount for FDJ and null for BKL, producing better results, as shown in Figure 6.11. The elapsed time of each simulation is around $3000\ s$ ($50\ min$).

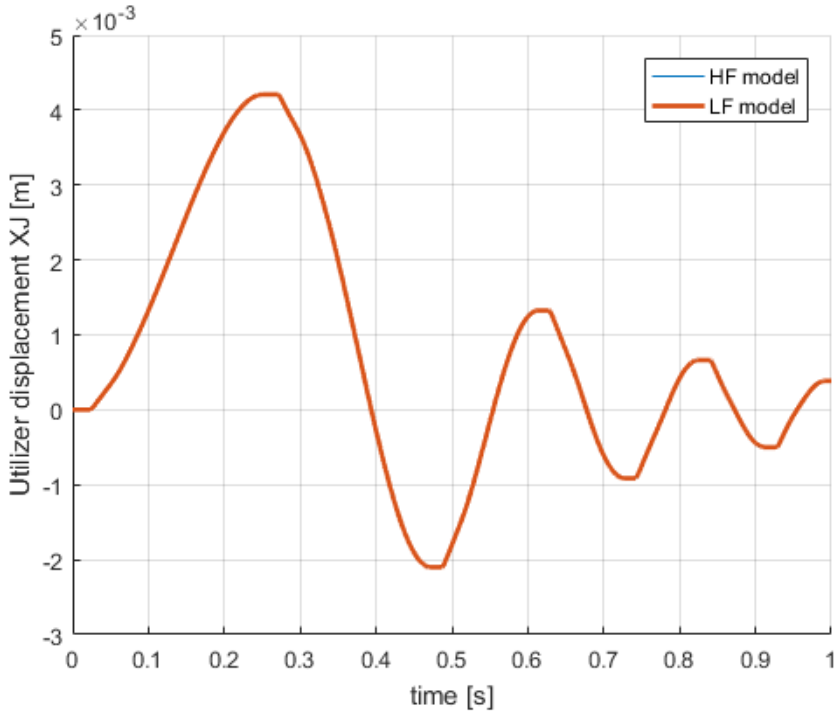


Figure 6.12: Case 8 – 1e-4 m backlash and 200 N FDJ

- Backlash length: $0.00010113\ m$
- Dynamic friction: $200\ N$

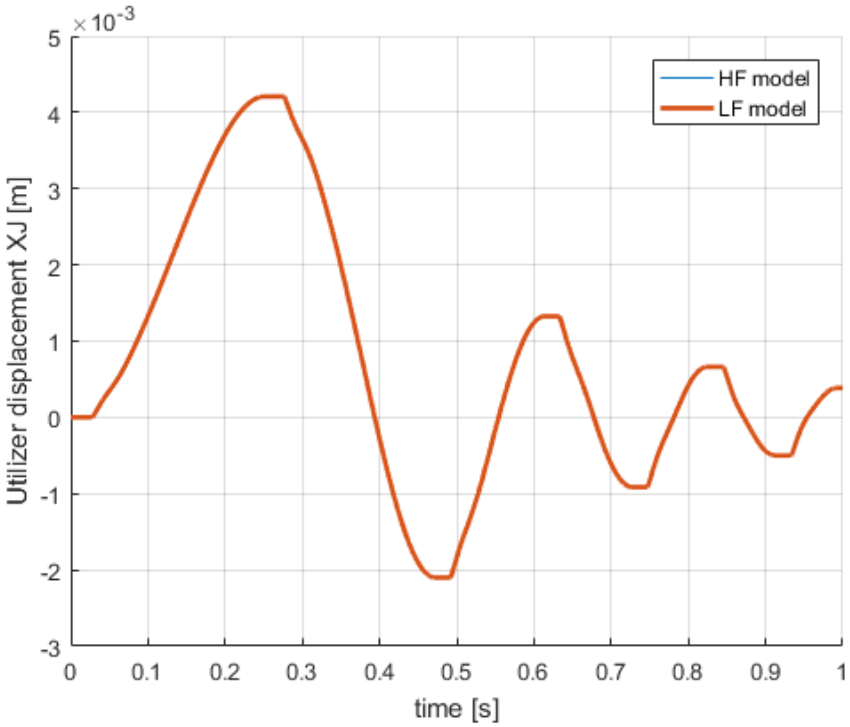


Figure 6.13: Case 9 – 2e-4 m backlash and 200 N FDJ

- Backlash length: $0.00020156\ m$
- Dynamic friction: $200\ N$

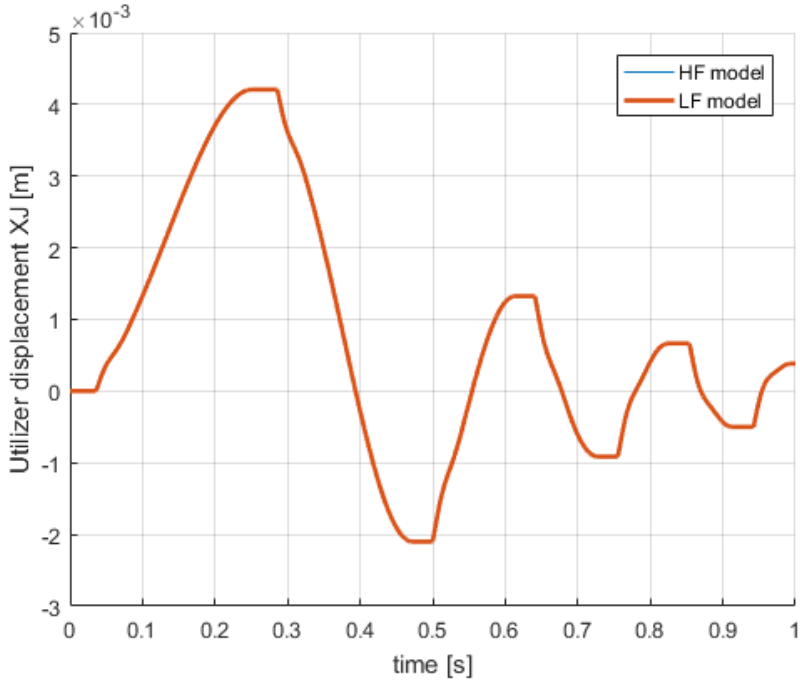


Figure 6.14: Case 10 – 5e-4 m backlash and 200 N FDJ

- Backlash length: $0.00050238\ m$
- Dynamic friction: $200\ N$

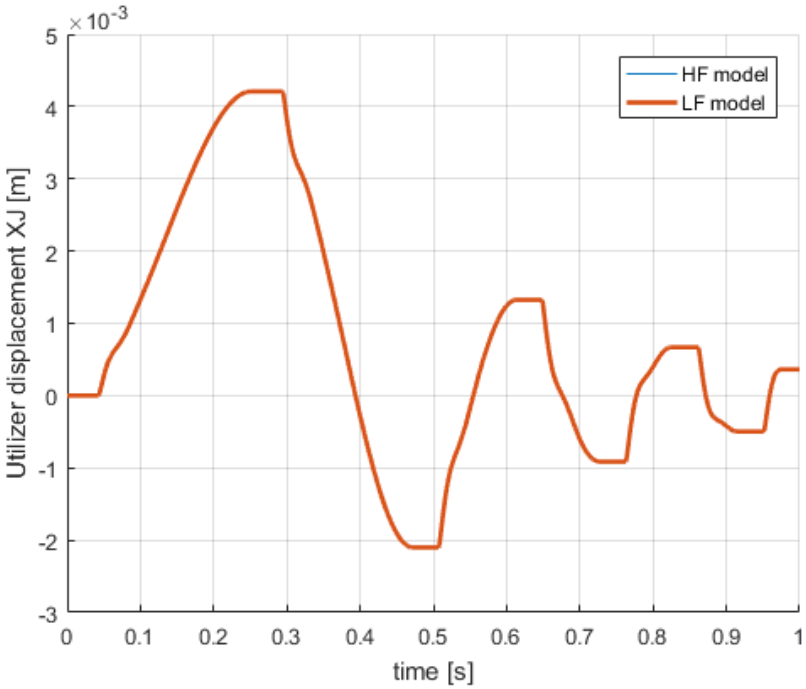


Figure 6.15: Case 11 – 10e-4 m backlash and 200 N FDJ

- Backlash length: $0.0010054\ m$
- Dynamic friction: $200\ N$

The algorithm is able to recognize the backlash fault perfectly, even when high boundaries are applied (between 0 and $0.00125\ m$ for the backlash fault).

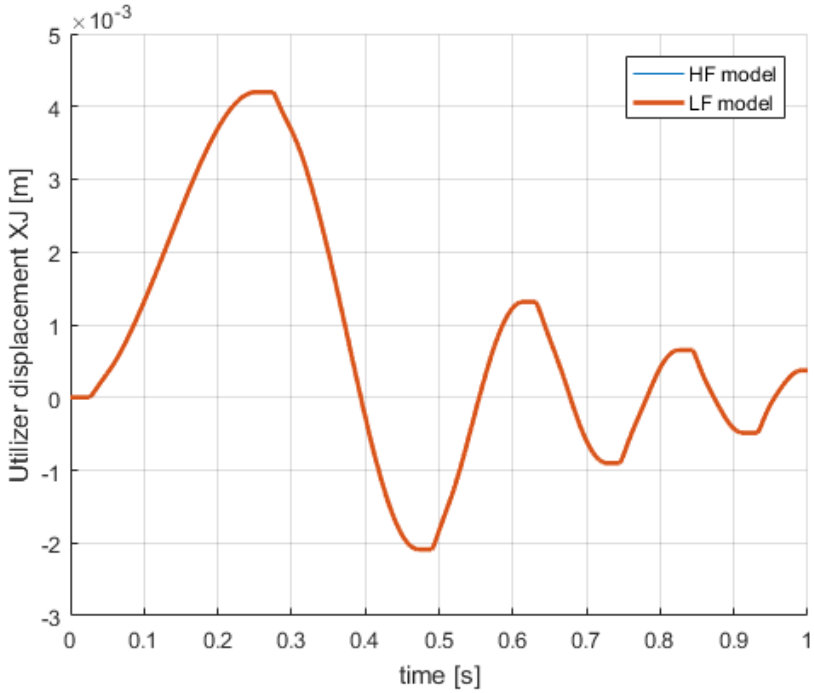


Figure 6.16: Case 12 – 1e-4 m backlash and 800 N FDJ

- Backlash length: $0.00010593\ m$
- Dynamic friction: $832.5374\ N$

Finally, a mixed fault simulation with incipient faults was conducted, and its result is shown in Figure 6.16. The population number was increased to 40, which caused the elapsed time to grow to $4873\ s$ (about $82\ min$).

In order to better analise the results of the simulations, we calculated the percent and mean errors for both backlash and dynamic friction values, reported them in a Excel data table, and then plotted them.

The percent error were calculated using the following expression [8]:

$$\%err = \frac{(input - output)}{input} \cdot 100$$

The mean error, instead, was calculated through the following expression [8]:

$$mean\ \%err = \frac{|BKL\ \%err| + |FDJ\ \%err|}{2}$$

The results can be seen in the following table:

Command type	Case	I - BKL [m]	I - FDJ [N]	O - BKL [m]	O - FDJ [N]	Population size	ERR%BKL	ERR%FDJ	MEAN_ERR
Mixed	1	1,00E-03	6400	0,0046792	2885,5684	20	-367,920	54,913	211,42
	2	2,00E-04	800	0,00045249	1070,98	20	-126,245	-33,873	80,06
1 cm chirp	3	0	400	3,35E-05	554,223	20	0	-38,556	19,28
	4	0	800	8,58E-06	596,9125	20	0	25,386	12,69
	5	0	1600	3,63E-05	1799,0696	20	0	-12,442	6,22
	6	0	3200	3,02E-05	3567,2466	20	0	-11,476	5,74
	7	0	400	0,00E+00	421,8742	20	0	-5,469	2,73
	8	1,00E-04	200	1,01E-04	200	20	-1,130	0	0,56
	9	2,00E-04	200	2,02E-04	200	20	-0,780	0	0,39
	10	5,00E-04	200	5,02E-04	200	20	-0,476	0	0,24
	11	1,00E-03	200	1,01E-03	200	20	-0,540	0	0,27
	12	1,00E-04	800	1,06E-04	832,5374	40	-5,930	-4,067	5,00

Figure 6.17: Percent and mean error data table

The case numbers refer to the cases as they appear in this dissertation. The abbreviations *I* and *O* refer to the inputs and outputs of the simulations.

Below, the plot containing the mean errors in correlation to the differet cases:

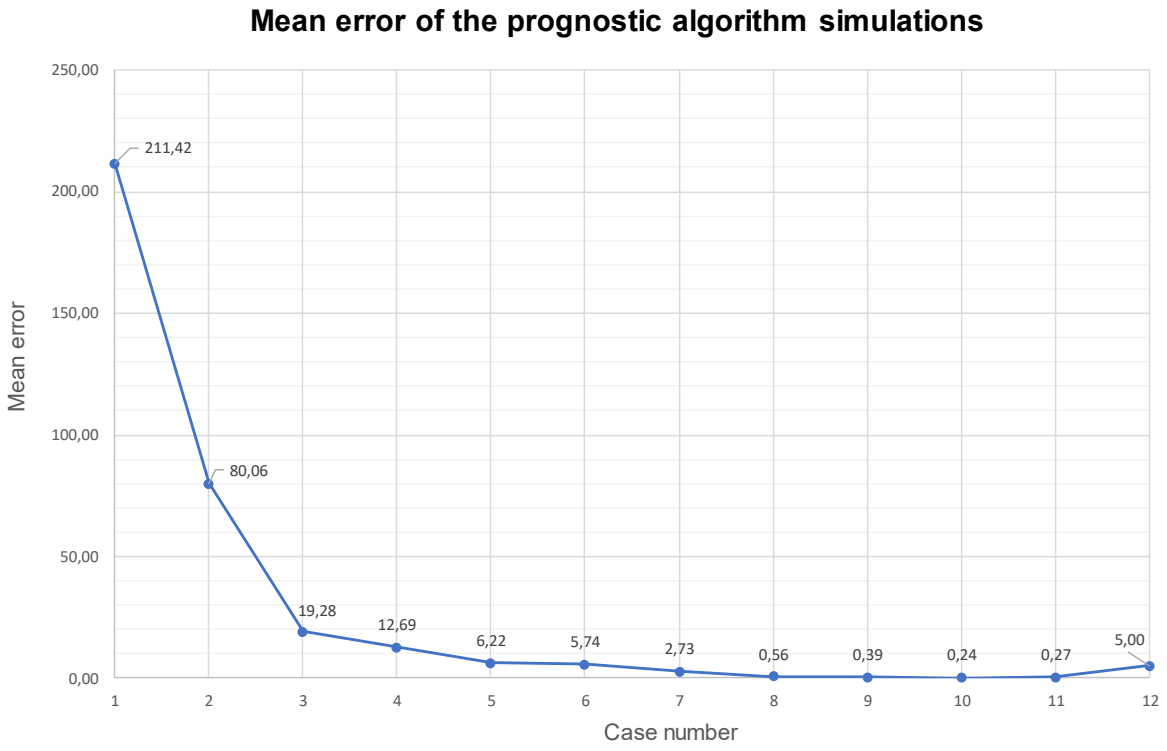


Figure 6.18: Mean error plot

The data shows that the single fault simulations with backlash give acceptable results (cases 8-11). Single faults simulation with dynamic friction give higher amounts of error instead (circa 3% or higher; cases 3-7). Moreover, mixed faults simulations with mixed command give, as expected, completely unacceptaple results (cases 1-2). Finally, the mixed fault simulation with the 1 cm chirp command gives a better result than the mixed command simulation, although the mean error value is still too high, considering that the population size was doubled.

Therefore, the fault detection and identification (FDI) algorithm gives the best results when treating the single fault simulations with backlash, while it need to be perfected when dealing with mixed faults simulations and single faults simulations with dynamic friction.

7 CONCLUSIONS AND FUTURE DEVELOPMENTS

In this thesis, a mathematical optimization task called fault detection and isolation – a fundamental task in prognostic analysis – has been successfully completed on the model of an electrohydraulic actuator (EHA) by means of a genetic algorithm provided by a Matlab toolbox called Optimtools.

The fault analysis we operated enabled the conduction of a detailed set of tests on the models, verifying their suitability for the simulations, and allowed the choice of the correct command for the algorithm.

However, it is important to remember that genetic algorithms – as they are non-deterministic methods – could not reach the same results in every simulation. In fact, a small difference in the parameters led to substantial differences in results, especially when changing fault boundaries or population number.

The elapsed time of the simulations can change drastically as these parameters vary. Ranging between 50 and $180\,min$, it makes the possibility of a real-time processing very difficult.

The monitoring model design, in particular, plays the main role – second only to population size and fundamental sample time – in determining the elapsed time. One possible solution to speed up its simulation time would be the reduction of the order of the servo valve to a 1^{st} order subsystem, or the employment of parallel processing.

Another point of interest is the selection of the right fault boundaries in the genetic algorithm. As the simulations were run, it seemed like the result converged faster in the backlash than in the dynamic friction because of the different numeric scale between the two. So, it could be useful trying to normalize those values between 0 and 1 directly on the model and studying how the simulation results change. This method could also affect the percent error, as the algorithm could have behaved unexpectedly with unnormalized parameters.

Other steps of the prognostic analysis could be the estimation of the remaining useful life (RUL) of the actuator, trying to predict the evolution of the faults after their initial appearance. This field requires a lot of empiric data, but the upsides are many and well received by the aerospace industry.

APPENDIX A

EHA_Dat.m

```
clear all
clc
%%  Simulation data

DT=1e-5; % Passo d'integrazione adottato [s]
TiBr=1.0; % Intervallo di simulazione

% TimeCom=0:DT:TiBr;
% TiBr1=0.01;
% Com1=0*(0:DT:TiBr1);
% TiBr2=0.2;
% Com2=0.003+0*((TiBr1+DT):DT:TiBr2);
% TiBr3=0.4;
% Com3=0*((TiBr2+DT):DT:TiBr3);
% TiBr4=0.6;
% Com4=0.01*(((TiBr3+DT):DT:TiBr4)-(TiBr3+DT));
% TiBr5=0.6;
% Com5=0.003+0*((TiBr4+DT):DT:TiBr5);
% Time=(TiBr5+DT):DT:TiBr;
% Com6=0.003*sin(10*pi*(Time-TiBr5+DT)+pi/2);
% Com=[Com1,Com2,Com3,Com4,Com5,Com6];

clol=1; %1 for close loop, 0 for open loop tests
TimeCom=0:DT:TiBr;
TiBr1=0.3;
Com1=0.03*(0:DT:TiBr1);
TiBr2=0.3+0.35;
Com2=0.005+0.004*sin(2*pi/0.35*(DT:DT:TiBr2-TiBr1)+pi/2);
Com3=0.005+0.004*sin(3*pi/0.35*(DT:DT:TiBr-TiBr2)+pi/2);

Com=[Com1,Com2,Com3];

frq=0

%% Controller
global Kintm XJref XJmod FSSm  XSref XSmod DXJref ...
    DXJmod P12ref P12mod FDJm


ErIM=1e-3;        % Saturazione dell'integratore del PID [m*s]
GAP=1e4;          % Guadagno proporzionale del controllore [mA/m]
GAI=0;            % Guadagno integrativo del controllore [mA/m*s]
GAD=0.0;          % Guadagno derivativo del controllore [mA*s/m]
GAS=0.0;          % Guadagno dell'anello di retroazione [mA*s/m]


%% Electro-mechanical model of the servo valve

ASV=0.7e-4; % Area del 2° st. della SV [m^2]
CSV=1000;
CorM=10.0; % Massima corrente in ingresso alla SV [mA]

His=0.0; % Semi-amp. isteresi TM [mA] (0.04)

GM=0.01; % Guadagno del motore SV [N/mA]
```

```
GPF=1e11; % GP 1° st. di SV a PSR=PSR0 [m^2/s]
GQF=0.014; % GP 1° st. di SV a PSR=PSR0 [m^2/s]

KF=40;  % Rigidezza traslaz. molla 1° st. SV [N/m]
KSF=160; % Rigidezza molla di Feed Back 1°-2° st. SV [N/m]

Ofs=0.0; % Corrente di off set SV [mA]

SNF=2513; % Pulsazione propria 1° st. della SV [rad/s]
XFM=0.1e-3; % Posizione di fine-corsa del 1° st. di SV [m]
XSM=0.6e-3; % Finecorsa spool servovalvola [m]
ZF=0.6; % Smorzamento adimensionale 1° st. della SV [#]

gioco=0.000; % Semiampiezza backlash sfera feedback spring [m]

MF=(KSF+KF)/SNF^2; % Massa Flapper [kg]
CF=2*ZF*sqrt(MF*(KSF+KF));  % Coeff. smorz. v. flapper [N*s/m]


%% Fluid dynamic model of the servo valve:

PSR=2e7; % Press. diff. Supply-Retourn SV effettiva [Pa]
PSR0=2e7; % Press. diff. S-R SV a cui sono def. GP e GQ [Pa]

PRV=1e6; % Pressione di serbatoio [Pa]
Pi=(PSR+2*PRV)/2; % Press. iniz. nel cilindro [Pa]
PVap=1e3; % Valore della tensione di vapore dell'olio [Pa]

    Clk=1.0e-13; % C. di trafilamento di J+SV [m^3/(Pa*s)]

    GP=1e11; % Guadagno in pressione SV [Pa/m]

GQ=0.2; % Guadagno in portata SV [m^2/s]
GPQ=GP/GQ;

XSS=PSR0/GP; % Apertura di saturazione dello spool[m]


%% External loads

FRC=0*8000; % Carico esterno agente sull'attuatore [N]
FRR=0; % Rampa di carico esterno [N/s]
FR_time=0; % Istante di aplicazione del carico esterno
FRSA=0; % Ampiezza Carico Sinusoidale [N]
FRSF=0; % Frequenza Carico Sinusoidale [rad/s]


%% Jack and LVDT data

AJ=8e-4; % Area martinetto [m^2]
CJ=50; % Coefficiente di smorzamento viscoso [N*s/m]
MJ=20; % Massa martinetto [kg]
XJM=0.2; % Posizione fine corsa del martinetto [m]
XJ0=0; % Posizione iniziale martinetto [m]

    BKL=0*1e-4; %2*XJM/1000;


%%  Linear Hinge-Moment model

Kaer=0*0.25*AJ*PSR/XJM; % Coeff. Hinge Moment [N*m/m]
```

```
%% Coulomb friction

FSS=1*ASV*GPF*XFM/100; % Val.Nom. FSS=0.01*FSSmax
FSDS=1.2; % Rapp. FF secco stat/dinam sullo spool
FDS=FSS/FSDS; % Attrito Coulombiano Din. spool [N]


%% Karnopp Friction model

eps=1e-5; % Banda morta di Karnopp (2e-6) (3.2)

    FDJ=200; % Forza Attrito Dinamico sul jack [N]

FSD=2; % Rapp. Forze Attrito Stat/Dinam sul jack
%FSJ=FDJ*FSD


%% LF model parameters

GAPm=1e4; % Guadagno proporzionale del controllore [mA/m]

    GMm=0.01; % Guadagno del motore elettrico SV [N/mA]

KFm=40; % Rigidezza traslaz. molla 1° st. SV [N/m]
KSFm=160; % R.Tr. fb. spring 1°-2° st. SV [N/m]

    giocom=0; % Backlash sfera feedback spring [m]

GPFm=1e11; % GP 1° st. di SV a PSR=PSR0 [m^2/s]
GQFm=0.014; % GP 1° st. di SV a PSR=PSR0 [m^2/s]


Kintm=0; % C. intasam. filtro 1° St. SV (0=OK - 1=KO)

CSVm=1000;
ASVm=0.7e-4; % Area del 2° st. della SV [m^2]
CSV=1000;

FSSm=7; % Attrito Coulombiano Statico sullo Spool [N]

FDJm=200;

PSRm=2e7; % Press. diff. Supply-Retourn SV effettiva [Pa]
GPm=1e11; % Guadagno in pressione SV [Pa/m]
GQm=0.2; % Guadagno in portata SV [m^2/s]
CJm=50; % Coefficiente di smorzamento viscoso [N*s/m]

    BKLm=0*1e-4;


%%  Filter clogging parameters

Kintas=0; % Fattore intasam. filtro 1° St. SV

%%  Fault parameters check

clc
disp('Fault parameters')
disp('==============================')
disp(' ')
disp(['1 - His = ',num2str(His),' [mA] - (N.C. = 0.04)'])
disp(' ')
```

```
disp(['2 - GM = ',num2str(GM),' [N/mA] - (N.C. = 0.01)'])
disp(' ')
disp(['3 - Ofs = ',num2str(Ofs),' [N/mA] - (N.C. = 0.0)'])
disp(' ')
disp(['4 - gioco = ',num2str(gioco),' [m] - (N.C. = 0.0)'])
disp(' ')
disp(['5 - Kintas = ',num2str(Kintas),' [N] - (N.C. = 0)'])
disp(' ')
disp(['6 - GP = ',num2str(GP/1e11),'e11 [Pa/m] - (N.C. = 1e11)'])
disp(' ')
disp(['7 - Clk = ',num2str(Clk/1e-13)...
      ,'e-13 [m^3/(Pa*s)] - (N.C. = 1e-13)'])
disp(' ')
disp(['8 - FSS = ',num2str(FSS),' [N] - (N.C. = 0.01*FSSmax = 7)'])
disp(' ')
disp(['9 - FSJ = ',num2str(FDJ*FSD),' [N] - (N.C. = 400)'])
disp(' ')
disp(['10 - BKL = ',num2str(1000*BKL),' [mm] - (N.C. = 0)'])
disp(' ')
```

APPENDIX B

main.m

```
clear
close all
clc

%% HF model simulation
diary('log.txt') %save the command window in a log file
tic %start counting simulation time

EHA_Dat %loads the workspace parameters of the HF model

BKL = 1*1e-4; % backlash width [rad]      min 0 max 0.1
FDJ = 800; % dynamic friction on user [N] min 0 max 800

disp('Actual parameters:')
disp(['Backlash length: ' num2str(BKL) ...
    ' m; Dynamic friction: ' num2str(FDJ) ' N'])

sim('Mod_EHA_HF')

%% LF model simulation
time = 0:100*DT:TiBr;
figure, hold on, grid on
plot(time,simout_HF)

EHA_Dat
sim('Mod_EHA_LF')
h1 = plot(time,simout_LF,'linewidth',2);
xlabel('time [s]')
ylabel('Utilizer displacement XJ [m]')
legend('HF model','LF model')

%% Fault detection & Identification (FDI)
nvars = 2; % 2 number of variables
lb    = [0 200]; % [0 200] NC - lower bounds
ub    = [0.0015 900]; % [0.1 10] NC - upper bounds
PopulationSize_Data = 40; % 20 NC - Population size

fcn = @(x) objfcn(x, simout_HF, h1);

options = optimoptions('ga');
options = optimoptions(options,'PopulationSize',...
    PopulationSize_Data);
options = optimoptions(options,'Display', 'off');

[x,fval,exitflag,output,population,score] = ...
ga(fcn,nvars,[],[],[],[],lb,ub,[],[],options);

fcn(x);

disp('Parameters estimated by FDI:')
disp(['Backlash length: ' num2str(x(1)) ...
    ' m; Dynamic friction: ' num2str(x(2)) ' N'])

toc %end simulation time count
diary OFF %stop writing log file
```

APPENDIX C

objfcn.m

```
function [err] = objfcn(faults, ref, h1)

    EHA_Dat

    BKL = faults(1)
    FDJ = faults(2)

    options = simset('SrcWorkspace','current');

    sim('Mod_EHA_LF',[],options)

    set(h1, 'YData', simout_LF)
    drawnow limitrate

    err = mean(sqrt((simout_LF - ref).^2));

end
```

REFERENCES

[1] P. Maggiore, *Modellazione, Simulazione e Sperimentazione dei Sistemi Aerospaziali,* slides from the course

[2] F. Marino, *Analisi Parametrica di Modelli a Parametri Concentrati di Servo-Azionamenti Aerospaziali,* Master's Degree Thesis, March 2015

[3] M. D. L. Dalla Vedova, P. Maggiore, L. Pace, "Proposal of Prognostic Parametric Method Applied to an Electrohydraulic Servomechanism Affected by Multiple Failures", *WSEAS Transactions on Environment and Development,* no.10 – January 2014, E-ISSN: 2224–3496

[4] A. Romano, *Modelli per la Prognostica delle Avarie Multiple di Comandi di Volo Primari,* Master's Degree Thesis, April 2011

[5] L. Borello, M. Dalla Vedova, G. Jacazio, M. Sorli, "A Prognostic Model for Electrohydraulic Servovalves", *Annual Conference of the Prognostics and Health Management Society,* 2009

[6] M. Dalla Vedova, G. Jacazio, P. Maggiore, M. Sorli., "Identification of Precursors of Servovalves Failures for Implementation of an Effective Prognostics", *Recent Advances in Aerospace Actuation Systems and Components,* May 5-7 2010, Toulouse, France

[7] G. Cerqua, *Prognostic Neural Approach for Electrohydraulic Actuators Affected by Progressive Failures,* Master's Degree Thesis, April 2017

[8] M. Borghetto, *Study of Optimization Algorithms for the Prognostic of Electro-Hydraulic Servomechanisms for Aeronautical Applications,* Master's Degree Thesis, October 2016

[9] P. C. Berri, *Genetic Algorithms for Prognostics of Electromechanical Actuators,* Master's Degree Thesis, October 2016