eman ta zabal zazu

Universidad del País Vasco
Euskal Herriko Unibertsitatea

Departamento de Ingeniería Mecánica / Ingeniaritza Mekanikoa Saila

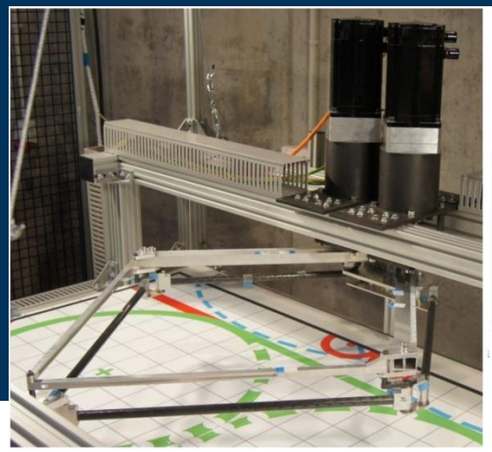
A Method for the Mechatronic Analysis of Parallel Kinematics Manipulators Based on Decoupling the Dynamics of Actuators and Mechanism

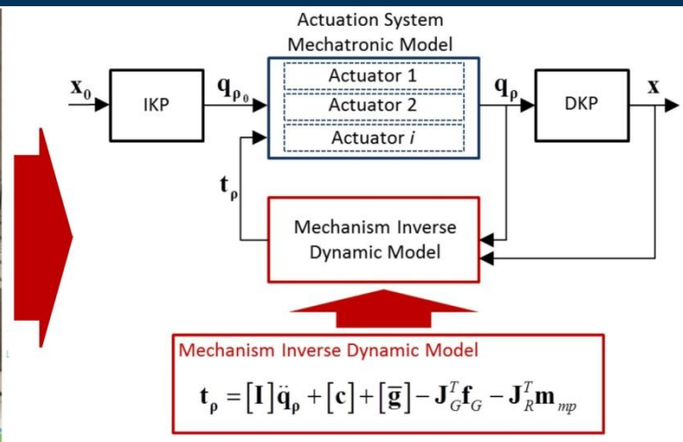
Constantino G. Roldan Paraponiaris

A Method for the Mechatronic Analysis of Parallel Kinematics Manipulators Based on Decoupling the Dynamics of Actuators and Mechanism

Ph.D. Thesis
presented to obtain the degree of
Doctor of Philosophy in Mechanical Engineering

Constantino G. Roldán Paraponiaris





Thesis Advisor
Dr. Francisco Javier Campa Gómez
Departamento de Ingeniería Mecánica / Ingeniaritza Mekanikoa Saila

February 2017, Bilbao

eman ta zabal zazu

Universidad del País Vasco
Euskal Herriko Unibertsitatea

BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

Ph.D. Thesis

A Method for the Mechatronic Analysis of Parallel Kinematics Manipulators Based on Decoupling the Dynamics of Actuators and Mechanism

Presented by:

Constantino G. Roldán Paraponiaris

At the:

Department of Mechanical Engineering
Bilbao Faculty of Engineering
University of the Basque Country (UPV/EHU)

To obtain the degree of:

Doctor of Philosophy in Mechanical Engineering

Thesis Advisor:

Ph.D. Francisco Javier Campa Gómez

February 20, 2017

Acknowledgment

Some years have passed since I left my beloved home country Venezuela and endeavored in an adventure that took me many kilometers away from the landscapes and traditions I knew, my friends with whom I use to debate and laugh, and my family which have always give me the best opportunities and love.

I arrived in this land where beautiful green mountains meet the sea, where an interesting culture sparks the curiosity of anyone because of its food, music, dances and language, where the character of its people is as strong as the iron of their mines, and their kindness makes you feel a warm welcome. All these make the Basque Country a special place, but for me, what makes it a really special place, is the people that I have come across with that helped me, in one way or another, to make this work. Without them this would have been simply impossible.

First of all, I would like to thank my supervisor Fran as he is a key person for the successful development of my PhD. thesis. His compromise with me, help, care of the details, work ethics, incredible insights, and his corrections have turned my work into a reality which have otherwise been not possible at all. He has raised the bar for me many times, for which I'm thankful.

I would also like to express my most sincere gratitude to my colleagues

of the Department of Mechanical Engineering and the Compmech research group, specially Oscar Altuzarra for answering me the many technical and some personal doubts I had during these years. I'm specially grateful to Asier for his help on Labview programming and control questions among other matters we have discussed. To Monica for her support and the nice moments with talks. To Mikel Larrea who also gave me his support at some hard times. I cannot forget to mention Lourdes Bouzas and Gorka Goikoetxea, they have been very important friends during these years, and the times we have spent have also help me to keep working and finish my work.

I'm very thankful to the IGM-RWTH for giving me the opportunity to conduct a part of my research with them, and also to the incredible people I met there. They have made my research stay a very very pleasant experience. Thanks to Professor Hüsing and my colleagues Jascha Paris and Thomas Kölling for their help and nice moments. They made me feel at home in Aachen. Moreover, my experience in Aachen would have not been complete if it wasn't for the friends I made during those 3 months. Thanks to Deliah and Anna Rotter for the nice moments I had while learning German conversation with them. Giorgos, Lena, Daniel and Jeremy made me have so much fun and laugh that I simply can't stop smiling. Also, very important to me, besides Jeremy, were Andre and Ioanna. I enjoyed the late night conversations with food spiced up with fun and friendship.

Many others I have met outside the Faculty of Engineering that should also be mentioned. I would like to express my gratitude to Lorena and Consuelo for helping me when I arrived in Bilbao and in this final part of my thesis respectively. Moreover, thanks to Gudrun, Melanie and colleagues for the nice times during german lessons. Also thanks to *Los Jueves* for the funny moments here in Bilbao. Nacho and Gradham also deserve their names in these lines. Furthermore, I would like to thank my friends Ivan, Rei, Paches, Roro and Luifer, although they are very far away from Bilbao, because of our special friendship every time we meet is as if we saw each other yesterday. A last person among my friends deserves a special mention.

Igor Olavarria has been one of the most incredible persons I have come to know during these years. Thank you very much.

Obviously, my family has also played a substantial part during these years. Thanks to my uncle Juan, ant Irene and my cousins Lua and Yanni, along with Elisa, Pepe, Alberto and Jessica for receiving me in their homes during my visits to Málaga. Thanks to my "Nona" Cristina and my grandfather Constantino which are very important figures in my life because of their support and advise. Of course my sister Dianne, mi Yole and my parents Wolfgang and Anastasia, have a fundamental part in all this. With their love, effort, education, caring and the opportunities they've provided me, I have been able to keep developing myself. I truly believe that without their enormous sacrifice and attention, perhaps I would have never come this far.

Finally, I should mention that I've been very lucky. The family that has here adopted me must be the most wonderful family I could have ever expected. With arms wide open they have allowed me to enter as just another member of their family. Arantza, Ángel and Eider have been so kind and attentive that they made me feel at home every single day. I give a big thanks to my Bihotz, as she has been the most important person for me. She always has been able to keep me smiling and with her love and support I was able to carry on.

Contents

List of Figures

List of Tables

Part I

Introduction and State of The Art

Chapter 1

Introduction

In this chapter of the industrial context for parallel mechanism will be given. Moreover, a general overview of the work carried out in this thesis will be summarized and a final outline will be presented.

1.1 Industrial context

1.1.1 Robots

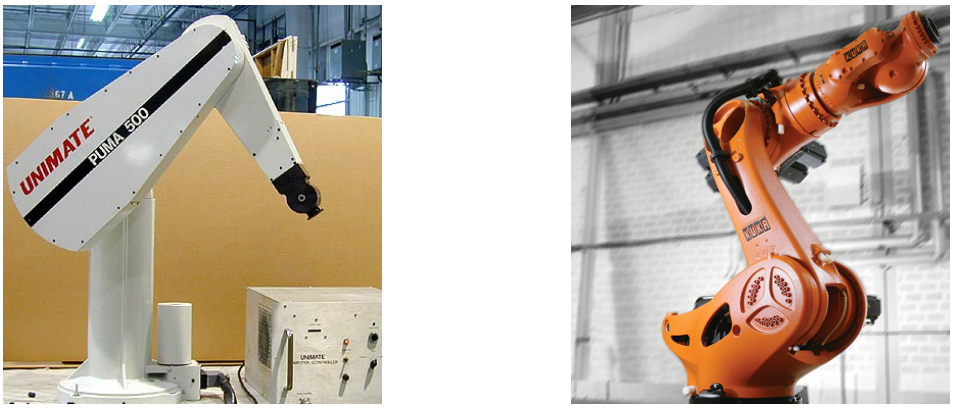
Since it conception in 1921, robotic systems have been increasingly used in several fields as their complexity and capabilities improve. This is fostered by the vast amount of research being focused on this subject from the last decades.

Robotic systems can be found in areas such as manufacturing and industry, aerospace, vehicles and medical applications. Robots have also evolve into different types to fit specific use cases. In this way, robots can generally be classified as:

- Humanoids

- Medical
- Autonomous Vehicles
- Industrial

This thesis is focused around a specific type of robotic systems found in industrial application. In this field, basically two types of robots can be found according to the architecture of their kinematic chains: serial or parallel.



(a) **(b)**

Figure 1.1. (a) picture of a Unimate PUMA 500 robot and (b) picture of a Kuka robot.

1.1.1.1 Serial mechanisms

The application for robots was firstly as manipulators to help humans move objects. The first construction type was to connect succeeding rigid bodies with joints that allow for a specific relative motion. In this way, the first robots resembled human arms and hence called *anthropomorphic robots*.

As it development advanced, these systems were increasingly being used in manufacturing facilities to assist human personnel on heavy-duty tasks. Shown in Fig. 1.1a, the *Unimate* manufactured by Unimation was the first robot to be installed in a *General Motors* plant for die-casting and welding operations. Nowadays, the KUKA robot in Fig. 1.1b and manufactured by KUKA robotics [1] is the most commonly found serial robot topology in industrial applications. The end-effector of this robot is capable of six-degree-of-freedom motions which allow for any motion.

Several other serial systems exist for industrial application such as the *Scara* robot in Fig. 1.2. Simply put, these type of robotic mechanisms have been adapted to many applications because of their large workspace and fast motions.

Figure 1.2. Picture of the Scara robot Cobra-800 by Adept [2].

As mentioned by Merlet [3], a spherical (serial) robot with 6 degrees of freedom would typically have a load-to-robot-mass ratio of between $0.035 - 0.064$. Moreover, the same ratio for a Scara type robot is in the range of $0.06846 - 0.08547$.

Also, it is mentioned that serial mechanisms have positioning accuracy limitations in terms of *absolute accuracy* due to the stiffness of the links and drive clearances. Moreover, the low load-mass ratio and poor accuracy is a result of the intrinsic serial design [3]. In this regard, each link must support the load due to the following linkages in the kinematic chain as well as for the load. As a consequence, each link must be stiffened, which implicitly increases the system's weight. What is more, due to the added mass, a serial mechanism may present significant inertia, centrifugal and Coriolis forces when high velocity motions are commanded. This may result in complex compensation techniques by control. Also the absolute accuracy is affected. As mentioned by Merlet [3], for an arm length of 1 m and a 0.06° error in a joint, an end-effector error of 1 mm results.

In summary, relatively low stiffness, limited accuracy and low weight-to-load ratio are limiting features that motivate the study of their parallel counterparts.

(a) **(b)**

Figure 1.3. (a) Picture of the Adept Quattro s800h by Omron/Adept, and (b) Picture of the Tricept-T9000 by Tricept-PKM.

1.1.1.2 Parallel mechanisms

Due to the described limitations of serial robotic systems, parallel mechanisms have been increasingly getting attention in the industrial context in recent years. As an example, the parallel manipulator Adept Quattro in Fig. 1.3a for pick and place [2] is shown. Their relatively low weight and high stiffness makes them interesting for high performance applications. Also, they are suitable for machining tasks due to their high mass-to-stiffness ratio such as the Tricept-T9000 shown in Fig. 1.3b by *Tricept PKM* [4].

In the scientific instrumentation field, hexapods like the H-840 by Physik Intrumente(PI) [5] shown in Fig. 1.4a, have been used due to their high positioning precision. Additionally, they are also used as excitation tables to simulate accelerations for vehicle developments applications as shown in Fig. 1.4b [6]. Without any doubt, parallel mechanisms have become an interesting option in the industrial robotics field.

(a) **(b)**

Figure 1.4. (a) Picture of the PI-H840 Hexapod by Physik Instrumente and (b) Picture of the Hexapod simulation table FF-1-3 by Moog Inc.

Basically, the end-effector of a parallel mechanism is connected to at least 2 separate links. In this way, the load can be effectively distributed

on each of the links. This allows for an increase in the stiffness of the system which also permits to reduce the masses of the linkages. Furthermore, the inertia, centrifugal and Coriolis forces may be lower than in the case of serial systems. Therefore, the theoretical capabilities of parallel robots make them an open area for future research and development.

Despite the potential of parallel mechanisms, much work is needed in order to place such systems in industrial contexts not only to niche markets. In fact, Merlet points out the several areas in which research must be done in order to make their capabilities become a reality[7], such as motion precision.

When compared with traditional serial systems, the dynamics of parallel mechanisms are more complex than their serial counterparts due to the closed chain kinematics. Also, the actuators and control are key elements for the overall system's performance, hence their also must be analyzed from a system integration point of view.

1.2 Motivation and scope of work

As it has been pointed out, to fully realize the capabilities of parallel mechanism, an integrated approach must be followed. Many areas, such as control, actuation, calibration, kinematic and dynamic modeling, are involved in such system. Actually, in the Fig. 1.5 Merlet [7] defines the mechanism, control and design and simulation layers present in any parallel mechanism. Furthermore, each layer involves a series of subsystems. Hence, the integrated approach, more than a design strategy, is a necessity to overcome the challenges imposed for such complex systems.

It is clear that a mechatronic approach based on reliable models are important for the conception and design of parallel robots. To serve that purpose, detailed simulation tools capable of modeling the manipulator, drives, control dynamics and their interactions in a cost efficient manner and with reliability are required. In this work, the three layers shown in

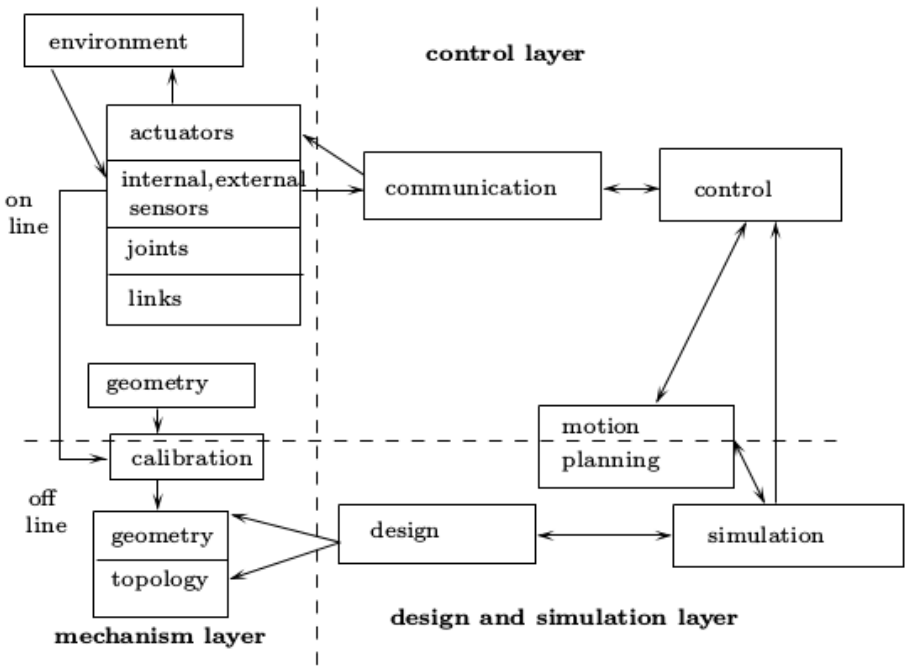


Figure 1.5. The various layers of a parallel robot. (Source: Merlet [7]).

Fig. 1.5 are addressed. However the focus has been placed in the design layer.

Furthermore, it is proposed a method to obtain a mechatronic model of parallel mechanisms, which can then be used for their analysis and design. To this end, the mechanism, actuation system, and the control are considered in a way that the mechanism dynamics is decoupled from the actuation system and its effect considered as a disturbance to the actuators.

Also, a procedure to obtain a dynamic model suitable for identification based on the principle of energy equivalence has been proposed. It is em-

ployed to obtain the actual values of the dynamic and friction parameters for parallel planar mechanism, so the mechatronic model can be adjusted.

Moreover, an experimental validation has been carried out in order to analyze and validate the proposed mechatronic modeling and the identification procedures. As for the mechatronic modeling procedure, it was applied on a planar 5R parallel manipulator and on 2PRU-1PRS *Multi-Axial Shaking Table (MAST)* mechanism, which was developed in the COMPMECH research group in the framework of a thesis [8].

Regarding the identification procedure, it was applied to two additional case studies, a 2-PRR mechanism and a *RePlaLink* Haptic mechanism developed by the Department of Mechanism Theory and Dynamics of Machines of the IGM-RWTH [9].

1.3 Thesis outline and summary of chapters

In this thesis it is proposed a method for the mechatronic analysis of parallel kinematics manipulators based on decoupling the dynamics of the actuators and mechanisms. To that end, this work has been organized as follows:

- In **chapter 2** a review of the state of the art has been done to introduce some topics on mechatronics design, motion control, parallel kinematic machines, dynamic modeling, friction and identification.
- In **chapter 3** the proposed procedure to obtain a mechatronic model of a parallel kinematic mechanism, considering the actuators, control and mechanism, is presented.
- In **chapter 4** two case studies are used to validate the proposed method to develop a mechatronic model for parallel kinematic mechanisms. The first case study is a 5R planar mechanism, and the second case study consists of a 2PRU-1PRS Multi-Axial Shaking Table (MAST).
- In **chapter 5** a procedure for the identification of the unknown dynamic parameters values of a parallel planar mechanism is described, where a rigid body model of the mechanism and actuators with friction is considered.
- In **chapter 6** two additional case studies have been used to validate the identification procedure described in the previous chapter, the 2-$\underline{P}$RR and the 5R haptic RePlaLink mechanisms.
- In **chapter 7** the contributions, conclusions and future research obtained from the development of this thesis will be presented.
- Finally, in **appendix A** the control developed in the framework of this thesis is described, and in **appendix B** the Jacobians used are shown.

Chapter 2

State Of The Art

Mechatronics is a branch of the engineering which, in an attempt to obtain better products, it integrates mechanics, electronics, and control engineering. In this chapter, a review of the state of the art on mechatronics, motion control, parallel kinematic machines, dynamic modeling and identification will be given.

2.1 Introduction to mechatronics

The term mechatronics was first conceived by Yaskawa Electric Corporation [10] in 1969. At the moment, they were looking for a concept to describe the integration of control and mechanical engineering for the design of servomechanisms they were working on. Nevertheless, a formal definition has been the subject of several authors and institutions. A popular mechatronics definition can be summoned to *"The synergistic integration of mechanical engineering with electronics and intelligent computer control in the design and manufacturing of products and processes"* [11, 12].

However, the above definition is somewhat constrained by the word

product. Perhaps, a more precise definition can be *"a technology which combines mechanics with electronics and information technology to form both functional interaction and spatial integration in components, modules, products and systems"* [13]. This definition highlights the interdisciplinary nature and components functionalities integration as a result of this design approach.

Furthermore, in the definition it is mentioned *functional interaction* as a way to describe that the functions of a system are distributed among its components [13]. Considering a machining center as an example, its main function can be regarded to shape materials with a desired precision. Then, it is easy to see that the controller strategy, sensors, actuators, software and mechanical design, all contribute to the machine functionality. Also, with *spatial integration*, it is intended to convey the notion that the components and subsystems are integrated into a single physical unit [13].

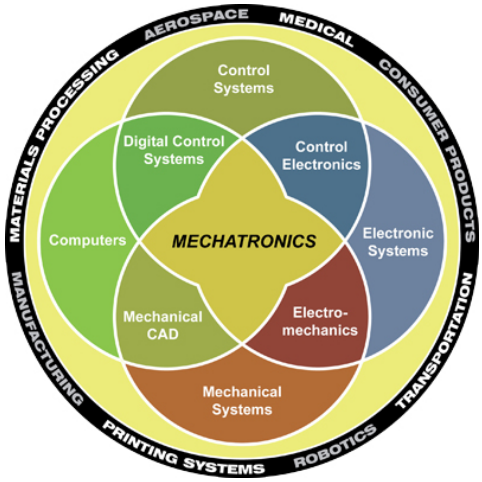


Figure 2.1. Mechatronics diagram by the Rensselaer Polytechnic Institute [14].

Whether a given definition is precise or not, a general consensus exists to regard the *mechatronics field* as an approach in engineering that looks to

integrate different disciplines in order to obtain better products, processes and in general systems. In Fig. 2.1 a diagram by the Rensselaer Polytechnic Institute represents the description of the mechatronics field [14]. As can be seen in the scheme, mechatronics is a discipline looking to combine Control Systems, Computer Science, Mechanical and Electronic Engineering. The scope of application is comprehensive ranging from consumer products to aerospace and medical systems.

A milestone in mechatronics was the approval of the norm VDI 2206 in 2004 on the *"Design Methodology for Mechatronic Systems"* created by the Association of German Engineers [15]. This norm provides a guideline for the design of mechatronic systems. It was created to extend existing guidelines with the latest findings in design research. Also, it integrates findings in procedures and tools specific to the mechatronic field to support the design of mechatronic systems [16].

In essence and following the V-model shown in Fig. 2.2 from the guideline, first the requirements of the product are established. Then, a preliminary system design phase takes place to define the required mechanical, electronic and information components. At this point, each domain-specific component is designed and developed mostly separated with the help of modeling tools. Afterwards, in the *system integration* step the obtained components are integrated together and evaluated to analyze how they affect to each other. The resulted *overall system* is continuously verified and validated against the concepts and solution obtained in the system design step. Finally, and once the design has been verified and validated, the final product is obtained.

What is important to note is that the mechatronic design approach does not follow a sequential phase design plan, where a specific component is first designed and then others are added subsequently. On the contrary, the components belonging to different engineering branches are designed in parallel, integrated and evaluated to obtain a better product. This guarantees an optimal interaction between the system's components such as in

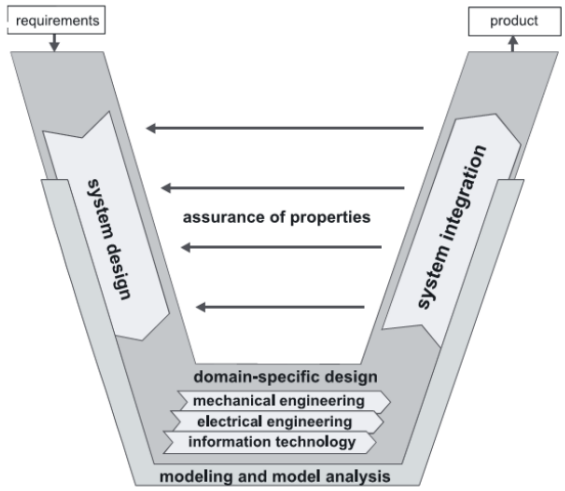


Figure 2.2. V model as a mechatronic design process.

the case for motion control.

2.1.1 Mechatronic modeling

The mechatronic modeling approach has been used before to analyze and improve the design of complex systems. In this regard, in the work of Reinhart and Weissenberger [17] it is explained the mechatronic modeling approach to obtain virtual prototypes of a machine tool, where the structural components, guideways, controllers, the dynamics of the actuators and the numeric control are considered using a multibody model and several computer-aided design tools , such as 3D-CAD and FEM tools. In this way, the overall design of the system can be improved *"... in early stages of the design process"*, which reduces the effort to obtain a final working prototype.

Moreover, in the work of Yan et al. [18] the finite element analysis is applied to a complex servo system of a hard disk drive, which is then used for

the control design. First, a finite element model was obtained and validated with experimental tests, afterwards, the resulting model is integrated into the control model and the closed-loop responses in the frequency domain of both, the simulation and experiments, are compared. It was shown that with the approach followed the model can be used for design optimization.

Also, an integrated modeling approach was applied in the work of the Huo et al. [19] to the development of an ultraprecision milling machine. To that end, the dynamics of the system, the control, and the machining process were taken into account in the model, which was afterwards used to assessed the performance of the machine at the design stage.

Another interesting application of the mechatronic modeling approach can be appreciated in the work of Brecher et al. [20]. Therein, a mechanical model of a parallel kinematics machine tool is constructed using MSC-ADAMS®, which is then used to design a Cartesian control that compensates the coupling forces of the actuators. As a result, the position error in the workspace due to the coupling forces was reduced by approximately 60 %, which also results in better manufactured products. Similarly, in the work of [21] the modeling of the feed drive of a multi-axis machine tool was addressed considering inertia, friction and the control. Then, the performance of a traditional cascaded control, typically used in numeric control, was compared with a proposed predictive control. It was found an improvement in the tracking error when a circular trajectory was executed.

From the mentioned works, it can be seen that the mechatronic modeling approach is relevant for the design of complex mechanisms, helping to improve design at early stages of development. Finally, it is appreciated the importance of considering the interactions between the different components, specially the mechanical design and the motion control, in early design phases.

2.2 Motion control

As suggested by its name, *motion control* is a field within automation that studies how to perform controlled motions of a specific system such as robots, milling machines and similar. To achieve such task several subcomponents are involved.

These components usually are a *control*, in charge of regulating the system inputs so a desired trajectory is followed; *sensors* used to measure a physical quantity, which is then employed in the control to infer the state of the system; an actuator or a group of them in charge of translating the input signals to an actual motion, which may be formed by a motor and transmissions elements; and a communication interface used for the signals transmissions between the different components. In this section, each of these elements will be addressed more in detail.

2.2.1 Control

To perform a given motion, some sort of a system must be employed to follow a desired trajectory in a controlled way and guaranteeing a required precision. In most robotic systems this is a key component that must be integrated into the design process.

The process of controlling a motion execution often involves measuring a variable, such as the position or the velocity of a specific component. This variable may be named the *measured variable*. On the other hand, the control is performed by adjusting another variable, named the *control variable* such as the current or voltage supplied to the motors. This may not necessarily be the same as the measured variable. What is more, the control is performed taking some signal as the reference for the control action. This signal is known as the *reference* or *control command*. In Fig. 2.3 a scheme is shown to illustrate a typical motion control process.

The process starts with the command signal received by the controller. Then, the controller generates a control signal adjusted to the reference and

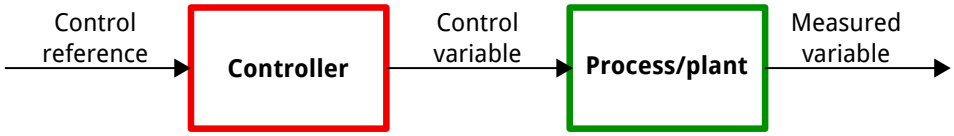


Figure 2.3. Illustration of a typical open loop control process.

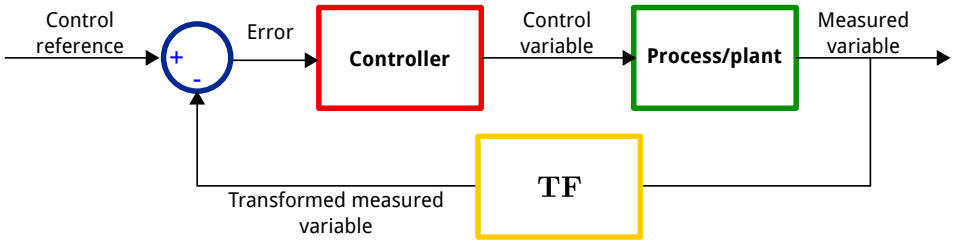


Figure 2.4. Illustration of a typical single-loop closed control process.

the state of the process or plant to be controlled. Essentially, two types of control can be used, namely, *open loop* and *closed loop* controllers. As it is, the control scheme in Fig. 2.3 corresponds to the open loop control architecture type. In Fig. 2.4 an illustration of a closed loop control is shown. There, the control action by the controller is generated from an error signal. This is obtained by substracting the measured signal from the command reference provided that both signals are referred to the same process state (e.g. position, velocity, etc). If not such case, a *transfer function* TF is used to convert the measured signal units to the command units by means of a known transformation (e.g. integration, derivation, etc). The closed loop control type is generally named *feedback control* since an output signal is used back to generate a control action.

Open loop controllers are usually employed when not so high precision output is required or when the system to be controlled presents good *open-loop* response as is the case for stepper motors [22]. On the other hand, feedback controllers are preferred for industrial applications since they provide robustness against disturbances. To explain this, let us consider the simple inertial example in Fig. 2.5.

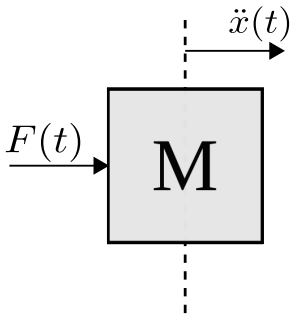


Figure 2.5. A simple mass system.

The equation of motion for the system in Fig. 2.5 are written as in 2.1. Also, by applying the Laplace transforms, Eq. 2.1 is transformed to the frequency domain as shown in 2.2.

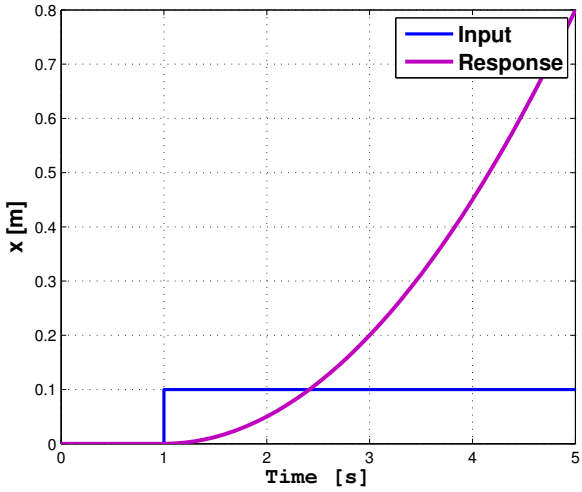
$$M\ddot{x}(t) = F(t) \tag{2.1}$$

$$Ms^2X(s) = \mathcal{F}(s) \tag{2.2}$$

$$\frac{X(s)}{\mathcal{F}(s)} = \frac{1}{Ms^2} \tag{2.3}$$

Hence, reordering the terms in Eq. 2.2, the single-input-single-output (SISO) transfer function in Eq. 2.3 is obtained. In control theory, a term $1/s$ represents an integrative behavior, which implies that the response of the system due to the inputs is accumulated. Hence, as the system in Eq. 2.3 presents a term $1/s^2$, its behaves as a *double integrator system*. In Fig. 2.6a, the response of the system in Eq. 2.3 due to a step input.

As can be seen, the response of the system deviates from the input and does not reach a constant value with time, in other words the system is *unstable*. Hence, to avoid deviations of a system from the commanded inputs in industrial applications, a controller is used to obtain an appropriate response.



(a)

Figure 2.6. (a) Input-response of the system of Eq. 2.3 without feedback.

2.2.1.1 The PID controller

In most industrial control applications, the *proportional-integral-derivative* (PID) is employed. It is a simple controller with a relatively good performance, in terms of error compensation.

Its principle of operation consists of calculating a correction action applied to the plant based on the error with respect to a command or reference. In Eq. 2.4 the mathematical representation of a PID controller is shown. The proportional gain K_p is related to the velocity of the system response, the gain K_i is used to compensate the stationary errors with respect to a reference value, and the gain K_d is used to stabilize the system.

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{2.4}$$

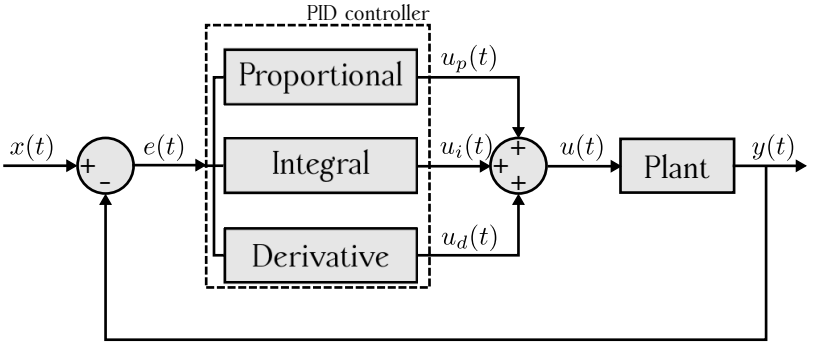


Figure 2.7. Scheme of a feedback control loop with a PID controller.

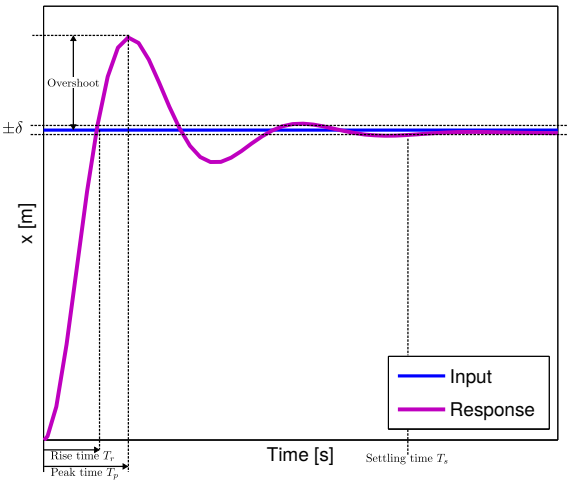


Figure 2.8. Input-response plot of a feedback control loop with a PID controller.

In Fig. 2.8 the input-response plot of a feedback loop with a PID controller is shown. As it can be seen, the response of the system approaches the input value with time, and also the system can be regarded as *stable system*. The time it takes to the response to oscillate within an error δ from a commands is known as the *settling time* T_s. Moreover the rise time

T_r is the time it takes the response to exceed the commanded signal. The overshoot is the difference between the peak value of the response and the value of the command at the peack time T_p. These parameters describe the performance of a system under the action of a PID controller. Furthermore, by increasing or decreasing the values of the proportional, integral and derivative gains, the performance of a system can be adjusted as desired for a specific application. In this regard, the tuning of a PID control is addressed in section 2.2.1.4.

2.2.1.2 Single-loop control

In Fig. 2.7 a typical feedback control process was shown. It can be seen that only a variable was measured and compared with a command reference. Hence, only one regulation loop takes place, for which they are named *single-loop* control. Basically, a single-loop control is the most basic form of feedback control that can be implemented, and serves as the building block for more advanced control systems.

What is important to mention is that, a single-loop employs only measurements from a single variable. This result in a somewhat limited control strategy for some applications. However, such controls can easily be extended to take additional and useful information into account. As a consequence, the control performance can be enhanced.

2.2.1.3 Cascade Control

In Fig. 2.9, a scheme of a position-velocity-current cascade controller is depicted. As can be seen, it consists of subsequent closed loops for different variables, where the output of one loop is used as the command for the next immediate inner loop. A cascade control is usually found in industrial applications, specially for chemical process and for motion control because it can be easily implemented [23].

Also, cascade controllers increase the performance of single loop controllers by using additional information of other process variables known as

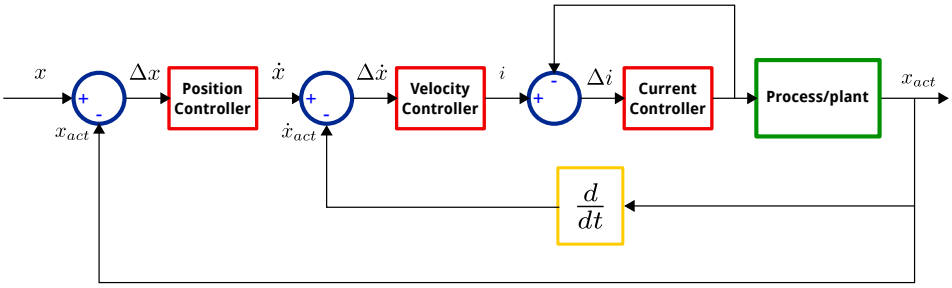


Figure 2.9. Cascade controller scheme for motion control.

secondary measured variables [23]. In the example in Fig. 2.9, the position is the primary variable respect to the velocity, which is the secondary one. As for the current, it acts as the secondary variable relative to the velocity.

It must be mentioned that to fully take advantage of the cascaded control architecture, the secondary loops must operate faster than primary loops. The main idea is to reject disturbances occurring at higher frequencies before their effects reach the process in the primary loop.

It is then easy to think that if a process can be controlled with the faster secondary loop, then there is no need for the primary one. However, often times the secondary loop can not completely reject the disturbances effects. Furthermore, other disturbances may still occur outside of the inner loops, and it is desired to control the primary variable of the process [23].

As for the controllers themselves, the family of proportional-derivative-integrative (PID) controllers are the most commonly used. For the case of cascade controllers for motion control, it is normally found that a proportional (P) controller is used for the position control loop, while a (PI) controller is used for the inner loops. Furthermore, a cascade controller can be tuned following a zone based approach, where high-frequency disturbances are first resolved.

2.2.1.4 Tuning of controllers

The tuning of a PID control is important to obtain the correct performance of the system. In the work by Ellis [24] it is explained how to tune controllers based on a *frequency zone* approach. In essence with this approach, high frequency terms are tuned first. Considering a single-loop control, the steps to tune a PID is as follows:

1. Set all gains to 0.
2. Increase the proportional gain K_p as high as possible without overshoot or losing stability.
3. Increase the value of the integral gain K_i until an overshoot of between $2 - 4\%$ is obtained.
4. Adjust the derivative gain K_d if necessary until the desired performance is reached.

The process is similar for a cascade control system. In such controls, the tuning must me performed from the inner loops first. As a result, inner loops are made to perform as low-pass filters rejecting higher frequencies disturbances.

2.2.2 Sensors

Sensors are important components for mechatronic systems since they are used to infer the state of the system. The sensors measure a physical quantity, such as an acceleration or force, and it is then converted to a more convenient signal, like voltage current or a digital signal. In this way they can be later handled appropriately by other components such as the control.

The type of sensors used depend greatly on the desired variable to measure and it is possible to use two different types of sensors for the

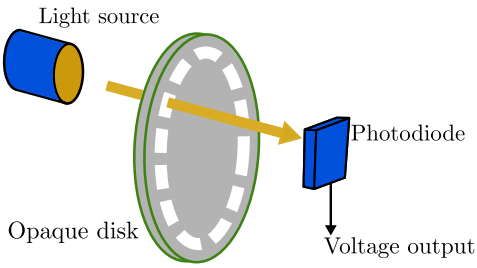


Figure 2.10. Scheme of the principle of operation of an encoder.

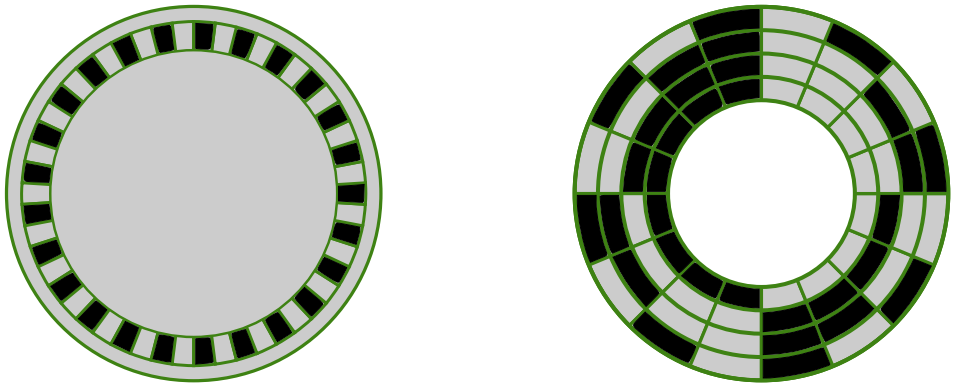
same purpose. The selection depends on economic factors or on the system design. Although a vast amount of sensors are available, in this section will be briefly explained the most important for motion control purposes: the encoders, tachometers and presence detectors.

2.2.2.1 Encoders

Encoders measure the position of an element and code the measurement into a digital output. They are usually employed in robotics to obtain the position of a shaft or arm, in machine tools for positioning systems, and in motors to control the angular position or linear displacement. The optical encoder is the most commonly used. Its principle of operation consists of a light source generating a voltage surge in a photodiode element when the light passes through a transparent window in a opaque element. In Fig. 2.10 a simple scheme of a rotary encoder is shown.

T here are two types of encoders, incremental and absolute encoders. Incremental encoders use the voltage pulses to count each time the light beam passes through the windows in the opaque material. Additionally, two diodes are placed in such a way that a phase lag of 90° in the pulse train is attained. In this way, the direction of motion can be inferred. Also, a third diode known as the *index*, can be used to count each complete turn of the disk.

On the other hand, absolute encoders make use of a printed code pattern in the opaque material, where each discrete position of the disk corresponds to one unique pattern segment. Thus, the position in absolute encoders is defined by a digital word instead of a pulse count as in incremental encoders. To this end, several windows tracks are employed where each track corresponds to a bit in the digital word. In Fig. 2.11 an incremental and absolute encoder schemes are shown.



(a) **(b)**

Figure 2.11. (a) Scheme of the front of an incremental encoder and (b) of an absolute encoder.

The absolute encoder in Fig. 2.11b is coded with a binary coding. However, ambiguous readings may occur when two or more bits are switched simultaneously in this type of encoders, such as from 1111 to 0000 digital words [25]. To solve this issue, gray coded encoders can be used, where the change from one pattern segment to the following involves switching only one bit [25].

Each of these types of encoders present advantages and drawbacks. On the one hand, incremental encoders are cheap and relatively simple. However, the do not retain the position value when the system is powered off.

As a result a reference point must be seek each time the system is turned on. On the other hand, absolute encoders do not present this issue but more tracks are used to obtain the position, and hence more photodiodes and light sources. Consequently, absolute encoders are more expensive [25].

2.2.2.2 Tachometers

To measure the rotating velocity of given device several technologies can be applied such as, variable reluctance sensors, rotating magnet sensors, stroboscopy and Wiegand effect sensors [26]. Nevertheless, incremental encoders, digital *tachometers* and or tachogenerators are commonly employed.

Incremental encoders can be used to measure the velocity if the pulse train generated is counted in a certain amount of time [27]. The advantage of this method, is that already available encoders for position measurements can be reused to obtain a velocity approximation. Digital tachometers use a similar approach to the one used by incremental encoders in that, by measuring the time between pulses the velocity can be calculated. Nevertheless, according to de Silva, "*... a digital tachometer is a device that employs a toothed wheel to measure angular velocities* ". In Fig. 2.12 a representation of a digital tachometer is shown. Two sensing elements are used to detect the tooths in the wheel. This sensing elements can be optical, when a light beam is used, or magnetic.

Moreover, tachogenerators can also be used to measure the rotation velocity of shafts. When a coil rotates inside a magnetic field, a voltage is induced in the coil. In this way, an electromotive (e.m.f.) force is generated with an amplitude proportional to the speed of the rotor [26, 27].

2.2.2.3 Presence detectors

In many motion control application it is required to detect when a moving component reaches a certain point, such as linear guide to find a reference point or stroke ends like positive or negative limits. To this end, mechanical switches can be used which generate an on-off signal when the switch

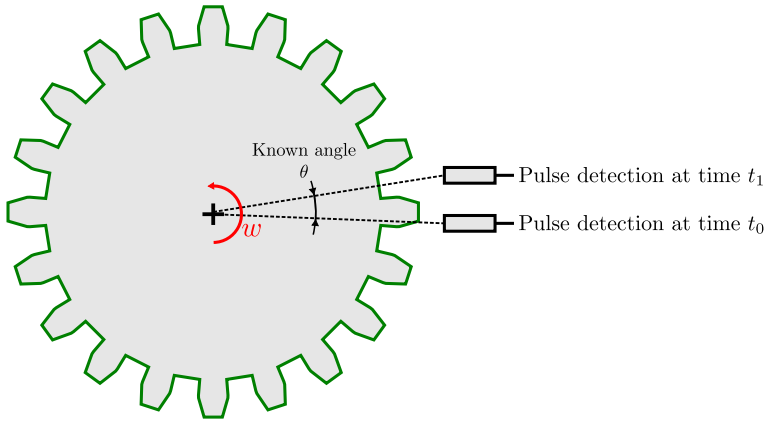


Figure 2.12. Scheme of the principle of operation of digital tachometer.

contact are closed or opened. In addition, optical, ultrasonic and capacitive sensors can be used

Also, inductive proximity sensors are commonly used in motion control to detect metallic objects. In Fig. 2.13 a representation of an inductive sensor is shown. The principle of operation is based on the electromagnetic induction between two separate coils. The first *primary* coil is connected to an AC power supply. This coil in turn induces a voltage in the *secondary* coil. When a metallic material is near the sensor, the voltage induced in the secondary coil changes, hence detecting the presence of the object.

On the other hand, capacitive sensors detect the presence of metallic or non-metallic objects by the change of capacitance C of the sensor and described in Eq. 2.5.

$$C = \frac{kA}{d} \tag{2.5}$$

Where k is the dielectric constant, A is the common area of the capacitor and d is the distance between the plates of the capacitor.

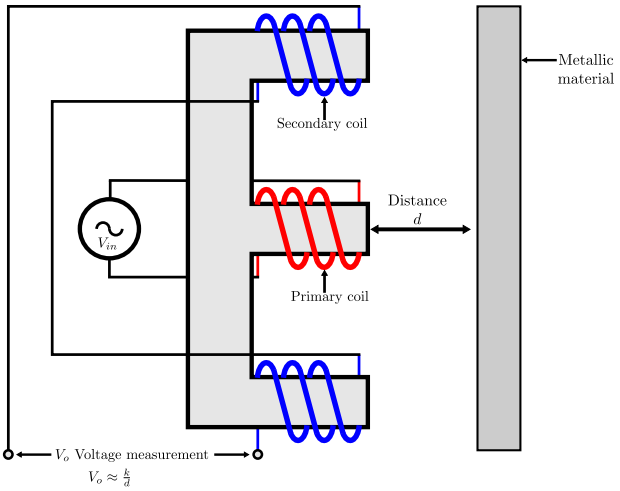


Figure 2.13. Scheme of an inductive sensor.

It can be seen in Eq. 2.5 that the proximity of an object can be measured if the distance d is modified. Also, the presence of an object between the capacitors plates changes the dielectric constant of the capacitor, thus modifying the capacitance. Finally, if the facing area between the plates is changed, the capacitance is also affected which can be used for encoders.

2.2.3 Motors for actuation systems

Motors are essential part of a mechatronic system and specially for the actuation system. They are employed to convert a source of available energy, usually electric energy, into mechanical motion. In this section, a brief introduction of the types of electric motors will be shown.

2.2.3.1 Direct Current Motors

Direct current (DC) motors are the most basic type of motors employed in industrial applications. Their operating principle is based on the Lorentz force, by which when an electric current flows through a conductor in the

presence of magnetic field, the electric charge will experience a force perpendicular to the magnetic field and current. In Fig. 2.14a a picture of a DC motor is shown.

(a) **(b)**

Figure 2.14. (a) A DC motor (Source: Maxon Motor [28]), and (b) inside of a bipolar stepper motor. (Source: Pololu Robotics & Electronics [29]).

2.2.3.2 Stepper Motors

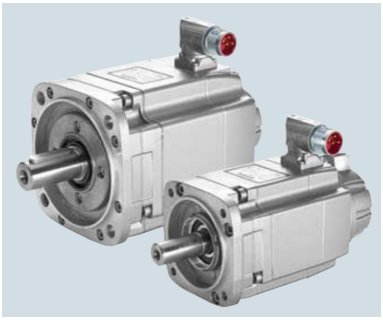
These type of motors move in subsequent fixed angular increments, also known as *steps*, when a current pulse is supplied to the coils in the armature. the coils generate a magnetic field that attracts the tooths of a toothed rotor as can be seen in Fig. 2.14b.

Moreover, stepper motors require a driver controller to properly supply the current to the coils in sequence. Also, they are know to provide good open-loop response, which implies that expensive control systems can be avoided. As a result, they are attractive for low-cost applications.

2.2.3.3 AC servo motors

Alternating current (AC) motors are an alternative to DC motors in industrial applications mainly because of their cost-effectiveness, easy maintenance and reliability. They can be classified as *induction* or *asynchronous* motors and *synchronous* motors. Their principle of operation is based on

a magnetic field rotating around the stator of the motor. In this way, the own rotor's magnetic field *follows* the stator's one, thus causing a rotation.



(a)

(b)

Figure 2.15. (a) Picture of the synchronous motor S-1FK7 by Siemens (source: Siemens [30]) and (b) picture of an induction motor Yaskawa SGM7A (source: Yaskawa [10]).

As for induction motors, the rotor is wound with a conductor material in the axial direction. Due to the rotating magnetic field, a current is induced in the conductors. Hence, a *secondary* magnetic field is created that generates a driving torque in the rotor.

Synchronous AC motors also have a rotating field at the stator. However, a constant magnetic field in the rotor is created, either by energizing the conductors using an external power source, like an external DC supply, or by employing permanent magnets. In this way, the rotor is caused to turn closed to same speed as the magnetic field rotation speed, known as *synchronous speed*.

The AC motors are also known as *constant velocity* machines since they are insensitive to variations in the load [25]. To obtain position control or speed control, advanced frequency controller drivers must be employed. In this way, a wide range of velocity-torque performance can be achieved.

2.2.3.4 Direct drive

A direct drive is a concept in which the electric motor is used to directly move system. In Fig. 2.16 a linear motor and a torque motor drive systems are shown. By employing such drive systems, intermediary transmission elements, like gearbox or linear guides, can be avoided with the consequent maintenance cost-reduction, increase in the reliability and precision, and faster response without backlash. Nevertheless, their behavior is highly non-linear and sensible to any disturbances, are expensive and a cooling system is required to dissipate the excessive heat in the air gap.

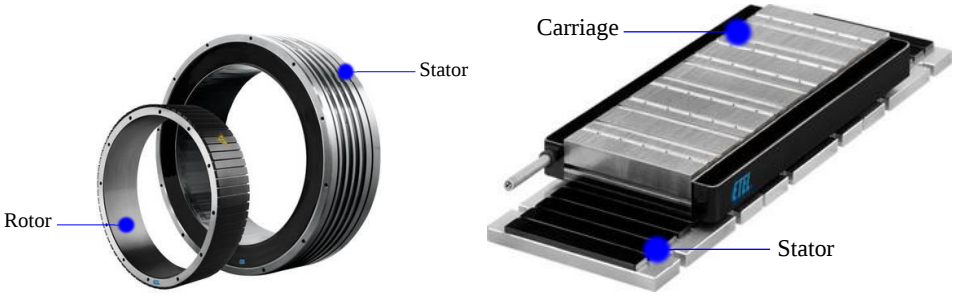


Figure 2.16. (a) Picture of a rotary torque motor TMB + c1 by Etel, and (b) picture of a linear drive LMS-91 by Etel.

2.2.4 Transmissions

Actuation systems in mechatronic applications generally use transmissions components to modified the motion, velocity and force or torque output of the motors to a more appropriate input to drive the intended load. As an example, rotary motors are used in most applications when the load moves in a linear motion. In such cases, a *conversion* from the rotation to the linear motion is required, for which a linear guide can be used.

Also, transmission components can be used in an actuation system to match the motor optimum operating point, i.e. velocity and torque, with

the driven inertia loads, as in the case of a gearbox, pulleys, planetary gearboxes and harmonic drives.

Gearboxes are the most commonly known transmission elements in mechatronic applications. In the simple case of a gearbox with two gears in Fig. 2.17, they consist of two toothed wheels with equal separation between teeth.

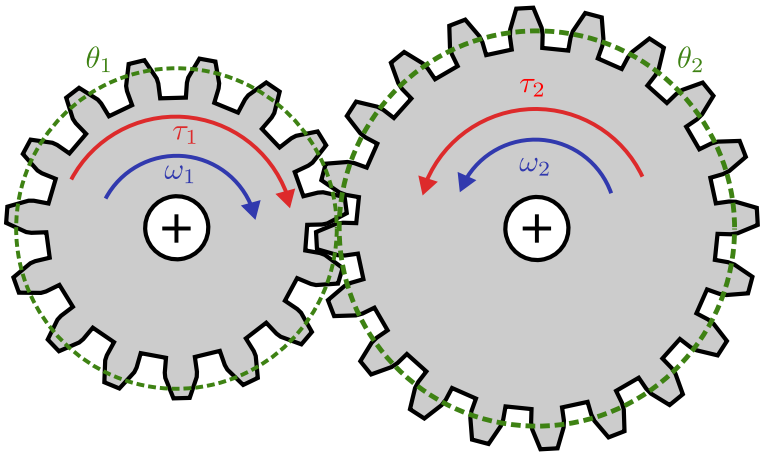


Figure 2.17. Representation of two gears with their angular velocities, and input torque τ_1 and output torque τ_2.

It can be seen that the direction of the motion of gear 1 is opposite to the direction of the gear 2. Furthermore, by using gears with different diameters a mechanical advantage is obtained. Therefore, the magnitudes of the torques τ_1 and τ_2, and the angular velocities ω_1 and ω_2 are related by the gear ratio i_r as follows:

$$\tau_2 = \frac{\theta_2}{\theta_1}\tau_1 = i_r\tau_1 \tag{2.6}$$

$$\omega_2 = \frac{\theta_1}{\theta_2}\omega_1 = \frac{1}{i_r}\omega_1 \tag{2.7}$$

Additionally, it can be noted that the equivalent of the inertia J_2 of the gear 2 at the input 1 can be found to be as in Eq. 2.8, hence reducing the inertia significantly. This allows to use a relatively simple motor controller which are more tolerant to disturbances and hence more stable than direct drive systems.

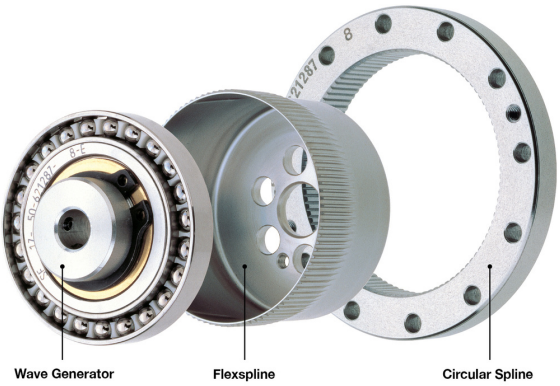
$$J_{2_1} = \left(\frac{1}{i_r}\right)^2 J_2 \tag{2.8}$$

Alternative designs of gearbox transmissions can be found for motion control. As an example, several connected gears are commonly used to obtain an output with a gear ratio, but such configurations usually take a relative large space. In contrast, planetary gearboxes can be used to obtain a certain gear ratio in less space.

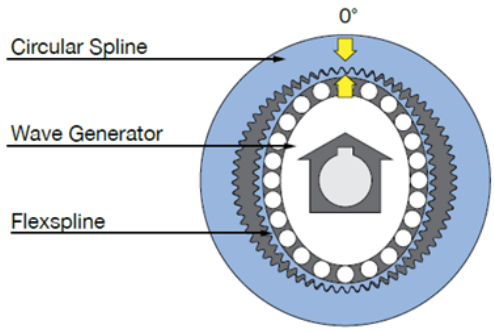
Nevertheless, gearboxes generally present *backlash* where zero torque transmission occurs [31]. As a consequence, small oscillations result during operation. Also friction and inertial torque due to the gears, and wear are common problems that reduce the performance of gear transmissions.

Harmonic drives were originally developed by C. Walton Musser as strain wave gearing in 1959 [32]. These drives can be used to obtain high output torque with low velocity zero backlash and reduced weight and space [31]. They consist of three basic elements: the wave generator, *flexispline* and the circular spline. These are shown in Fig. 2.18a and Fig. 2.18b.

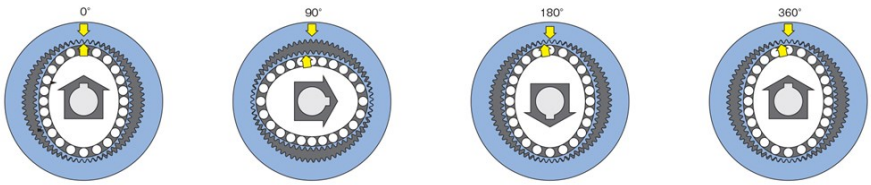
The wave generator and the flexspline are the input and output elements respectively, while the circular spline is fixed and is slighter bigger than the flexspline. As the ellipsoid wave generator turns, the flexpline (flexible element) is pushed against the circular spline engaging the teeth of both components at the major axis of the ellipsoid. Since the circular spline is bigger than the flexspline, a relative motion between both elements with a torque output is obtained. The principle of operation of an harmonic drive is described in Fig. 2.18c.



(a)



(b)



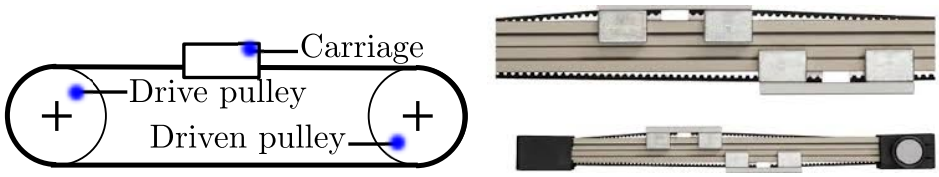
(c)

Figure 2.18. (a) Picture of the harmonic drive main components (source: www.waltmusser.org), (b) schematics of the harmonic drive main components assembly, and (c) description of the principle of operation (source: www.harmonicdrive.net).

2.2.4.1 Rotation to translation conversion

In some application is desired to have linear motions instead of a rotation. To this end, a linear guide can be used to convert the rotary motion of a motor to a linear displacement.

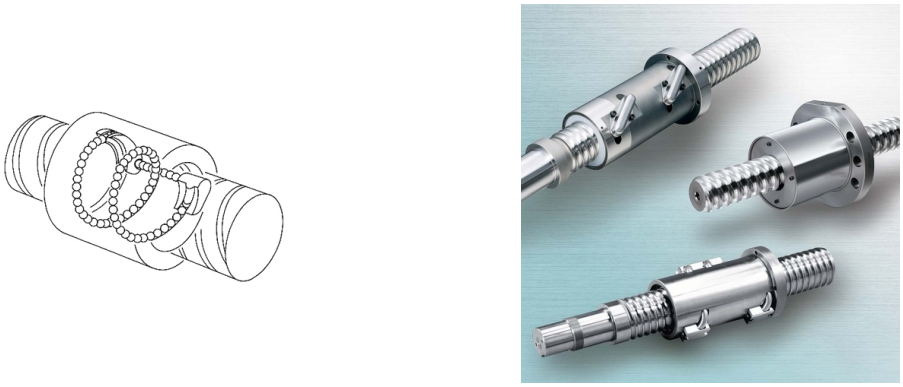
The types of linear guides which employ a set of pulleys and a belt are known as *belt drives*. In Fig. 2.19a a scheme is shown and in Fig. 2.19b an example picture is given. As a pulley on one end turns, a tension in the belt is created causing the other pulley to turn. As a result, tension on the other side of the belt is created which pulls the carriaged of the belt guide.



(a) (b)

Figure 2.19. (a) scheme of a belt drive and (b) picture of the belt drive ZLW-0630-OD by Igus [33].

Another way to obtain a linear motion from a rotary motor is by using lead screw and nut guides. In Fig. 2.19a the schematics of a ballscrew is presented, and in Fig. 2.20b three examples are shown. They consist of a bolt and nut system, and the principle of operation is similar to that of a screw. As the axis of the screw turns, the nut moves in the axis direction. Moreover, the balls are used to reduce the system friction and tolerances, thus limiting backlash.



(a) **(b)**

Figure 2.20. (a) scheme of a nut with the ball circulation system by Ipiranga Husillos [34] (b) picture the HTC-SRC ballscrew drives by NSK [35].

2.3 Introduction to parallel kinematic machines

In contrast to serial kinematic mechanisms where rigid bodies are attached one after the other, in parallel kinematic mechanisms each of the kinematic chains are connected with an *end-effector*. The end-effector is the element of the mechanism where the main tool is placed, and usually defines the mechanism's functionality such as a manipulator or machine tool.

Parallel kinematic mechanisms present several advantages when compared with serial systems. The successive placement of the elements in a serial system results in more weight that is added, which causes elastic deformation of the previous components. To counter this, elements with high stiffness are used, thus, further increasing the overall weight of the system. Hence, serial systems present lower load-to-weight ratio than parallel kinematic mechanisms.

In addition, the need to stiffen the components with heavier ones in a serial mechanism, causes that the inertial, Coriolis and gravitational dynamic effects are increased, what makes the control more complex.

On the other hand, as in parallel mechanisms the load is distributed between the links, lighter components can be used. As a result, it is possible to obtain a comparative more stiff system with a parallel mechanisms than with a serial one. Furthermore, lighter components results in smaller nonlinear dynamic effects, which could result relatively simpler control solutions, although other issues, such as forces due to the coupling of the links, should be taken car of. Another consequence of having lighter components is that faster motions can be performed, which is interesting for pick & place applications, and that they can be also used for applications requiring a high bandwidth, such as Multi-Axial Shaking Tables

2.3.1 Pick & place mechanisms

As the load/weight ratio of mechanism is lower, the *lighter* the mechanism can be, resulting in a reduction of the inertial and Coriolis dynamic forces. Obviously, this advantage becomes more clear with high velocity motions, such as in the manipulation of objects in manufacturing chains. In this regard, a pick & place mechanism is used to grasp an object and moved it to another position where the object is placed. The Adept Quattro shown in 1.2 is an example of a parallel mechanism used for pick & place operations.

2.3.2 Multi-axial simulation tables

This type of machines are typically formed by a parallel kinematic mechanism structure, which is formed by several linkages and a final platform. In this way, they are able to generate a motion with combined translations and rotations of the final platform [8].

Multi-Axial machines are common employed for industrial applications to generate motions with high accelerations or torques at their end-effector and with several degrees of freedom. Most common applications are for structural vibration testing, suspension evaluation, and in more general for applications were products with high accelerations need to be tested. In this regard, a MAST system can be used to perform dynamic tests for

structures such as buildings or bridges; or for mechanical components to evaluate their behavior under vibrations. In these cases, they are regarded as *Multi-Axial Shaking Tables* [36–38]. Also, Multi-axial machines are used as simulation platforms for flight and automotive applications. In such cases, they are regarded as Multi-Axial Simulation Tables [39, 40].

2.3.3 Control approaches for parallel kinematic machines

The most straightforward way to implement controllers on parallel mechanisms is by reusing existing controllers from serial ones, such as the linear single-axis control and the computed torque control. Indeed, this is mentioned in the work by Paccot et al. [41], where several control alternatives are reviewed.

2.3.3.1 Linear single-axis control

A common control alternative is the single-axis PID position control shown in Fig. 2.21, where the inverse kinematic problem of the mechanism is used to obtain the joints position q_d from a path in the workspace X_d. Then, a PID controller adjusts the inputs to the machine based on the error between the command q_d and the actual position of the system measured at the joints q. In addition, a feed-forward uff can be used to compensate inertial forces.

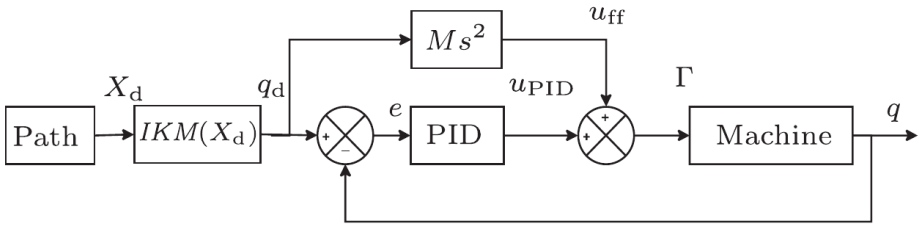


Figure 2.21. Linear single-axis control with feedforwad (Source: Paccot et al. [41]).

The main advantage of this control is that they allow to reuse existing implementations from serial systems [20, 42]. This is an attracting feature since existent controllers can be adopted and therefore easily used for industrial applications. Also, this type of control provides relatively good performance, and the tuning of such controllers is well-known from modern control theory [41]. Moreover, a linear single-axis control can also be set to compensate for the dynamics of the mechanism, specially for relatively low-speed systems [41]. However, when higher velocities are present, nonlinear dynamic effects take importance and hence the accuracy is negatively affected [42].

In the case of parallel kinematic mechanisms, the dynamics is also nonlinear as a consequence of the coupling between the mechanism's links [41]. Therefore, the same gain tuning of the controller is not guaranteed to be optimal in the whole workspace [20, 41]. Solutions to this issue have been addressed by restricting the workspace [43], path planning [44] and higher derivatives compensation with feed-forward [45].

In this regard, Wang et al. [46] investigated the use of the dynamic model of the system to compensate the non-linear dynamics of a parallel mechanism using feed-forward. It was reported in their work that the tracking error is reduced, and that the simulations compensate for the nonlinear torque caused by the parallel kinematic machine dynamics.

Another issue arise from the presence of the inverse kinematic model (IKM) and its inherent modeling errors, resulting in joints trajectory errors which are not *observed* nor compensated by the control.

2.3.3.2 Computed Torque Control in the joint-space

A control strategy commonly used in serial kinematics is the *computed torque control* (CTC) in the joint-space [47]. It uses the inverse dynamic model to calculate the torque inputs required for a motion of the mechanism [42]. Additionally, compensation of disturbances such as friction, can also be done by including a feed-forward [41]. In Fig. 2.22 a general computed

torque control is shown, where the inver kinematic model is used to obtain the position, velocity and acceleration of the joints from a path in the workspaces. Then, the gains k_p and k_v are used to calculate a compensation action from the position error e_ρ and the velocity error $e_{\dot{\rho}}$ respectively. These are then added to the demanded acceleration $\ddot{q}$ and the output is used as an input to the dynamic model. Furthermore, the actual joints positions and velocities are used as well. As a result, the required torque τ is obtained.

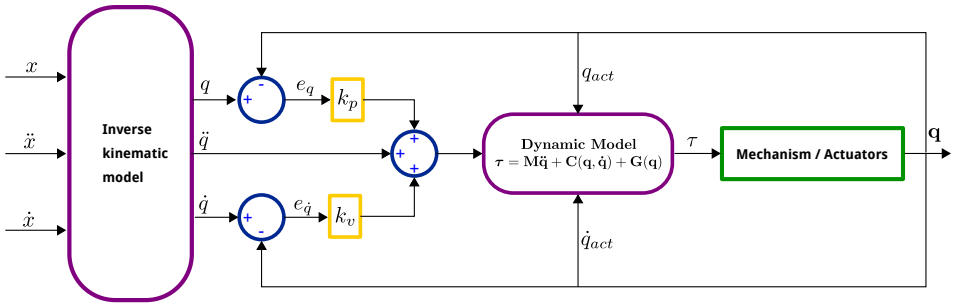


Figure 2.22. Simplified computed torque control scheme.

For computed torque control, Lagrange's equations are commonly used in order to obtain explicit expressions of the dynamics, which are separated in inertial, Coriolis and gravitational terms, depending on the actuated joints positions, velocities and accelerations. Moreover, by using the CTC in the joint-space, the issue of non-optimal tuning of the controllers is avoided because the non-linearities of the dynamics are taken into account by the inverse dynamic model (IDM). Therefore, the controllers are guaranteed to perform appropriately regardless of the workspace [41].

Nevertheless, modeling errors, which are somewhat unavoidable, affect the stability and motion accuracy of the system, thus affecting the performance of CTC controllers [41, 42]. As a solution, the common approach is to reduced the modeling errors as much as possible. One way is to consider the disturbances due to the task of the mechanism [48]. Also, a combined

flexible and rigid body model can be done in order to further reduce modeling errors [49]. Lastly, a parameter identification procedure can also be done to consider the actual dynamic parameters values of the system [50, 51].

2.3.3.3 Control in the workspace using PID

For a parallel kinematic machine, the performance is limited with a joint-space control, since the state of a PKM is completely defined by the end-effector instead of the joints as in serial systems [41, 52]. Hence, for parallel kinematic mechanisms, Cartesian control, also known as workspace control, should be used to obtain the best performance.

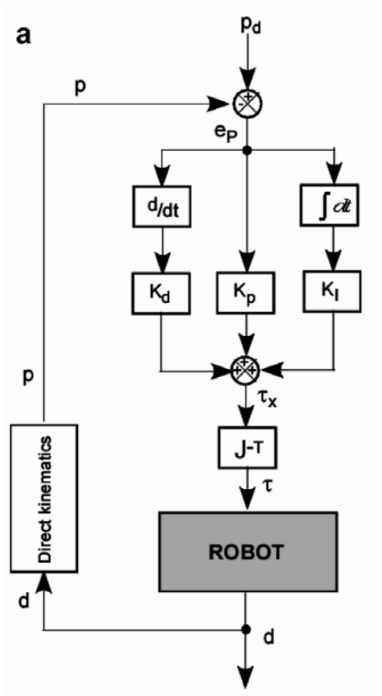


Figure 2.23. PID control in the workspace (Source: Callegari et al. [53]).

On the contrary to joint-space control, workspace control, reacts to the position and orientation error measured directly at the end-effector. Moreover, the actuators inputs are calculated from the workspace error using the inverse dynamic model of the mechanism. In Fig. 2.23 the scheme of a PID control in the workspace with gains K_p, K_I and K_d is shown, where the direct kinematics is used to obtain the position of the robot p from the actuators measurements d.

A problem with workspace control is that the end-effector position is often times inaccessible. Hence, measuring technologies like vision, indoor positioning systems and laser tracking must be used instead, thus increasing the complexity and the cost of the overall system. A solution is to used the direct kinematic problem (DKP) to estimate the end-effector position from the joints positions as it was shown in Fig. 2.23, but this approach also present some important drawbacks.

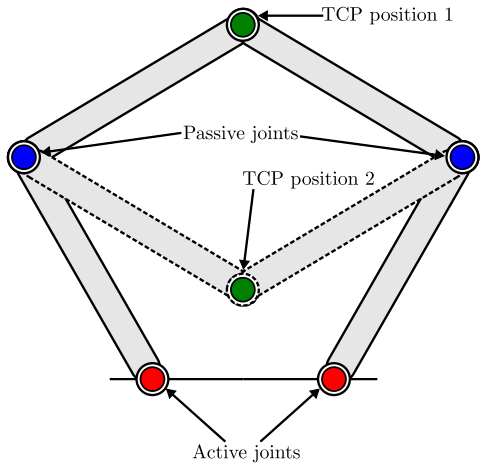


Figure 2.24. Scheme of a 5R mechanism with the two possible positions of the TCP.

A common issue seen in PKM is that a given actuated joint configuration can present different end-effector positions as can be seen in Fig. 2.24

for a 5R mechanism. Therefore, passive joints may also be measured to asses the position of the end-effector, or the workspace may be limited to avoid the singularities. Another problem, as mentioned by Paccot [41], is that the DKP results in a squared model, which is sensitive to measurement and calibration distortions and deviations in the kinematic model from the actual system.

2.3.3.4 Computed torque control in the workspace

The computed torque control can be applied in the workspace with the same advantages as described previously. In the workspace, the position and orientation of the mechanism is measured and used to calculate the forces inputs to the actuators, as shown in Fig. 2.25.

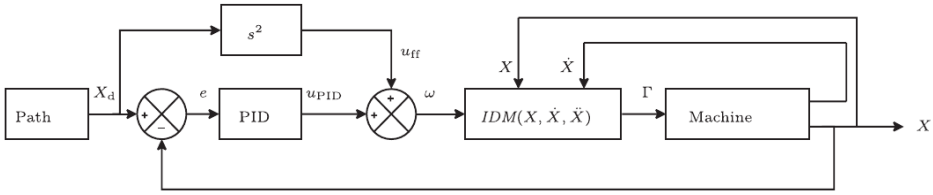


Figure 2.25. Computed torque control in the workpace (Source: Paccot et al. [41]).

In this control, the workspace position error e is obtained from the command and actual positions in the workspace, X_d and X respectively. Then, a PID controller is used to calculate the compensation action u_{PID} from the error e. Additionally, an acceleration command u_{ff} is added as a feed-forward to the compensation action u_{PID}, thus resulting in the input w to the dynamic model IDM. In addition, the position X and the velocity $\dot{X}$ are also introduced in the inverse dynamic model. Finally, the required input torque Γ is calculated and supplied to the mechanism.

As it can be appreciated, a measurement of the end-effector is needed for this control. If it cannot be measured, the direct kinematic problem

can be used to estimate its position and orientation instead. Nevertheless, the direct kinematic problem issues mentioned previously are also present in this case.

2.3.4 The problem with parallel kinematic machines

As it has been mentioned, parallel kinematics machines have a potential for improved accuracy, speed and rigidity [3] compared to serial machines. Nevertheless, their practicality on industrial settings still demands for more work on this topic. Many authors have reported a lack of good performance in their investigations. As an example, Tlusty et al. compared a *machining center* with cartesian kinematics with an Hexapod parallel kinematic machine [54]. It was found that this kind of mechanism presented lower performance to the machining center in terms of accuracy. In this regard, it was reported that the transmission of motions due to the parallel structure caused a lower stiffness of the final platform than the stiffness of a single strut. As a result, process forces (cutting, weight) cause deflection on the struts affecting the accuracy of the system. Also, in the work of Wang and Masory it was found how the tolerances in the manufacturing process, assembly errors and offsets negatively affected the accuracy of a Stewart platform [55].

Moreover, as indicated by Paccot et al. [41], two problems arise when dealing with PKM. Simplifications in the kinematics influence have a negative impact on the accuracy. Another issue reported deals with the type of control used with PKM. It was mentioned that, existent control strategies for serial mechanism are reused. This leads to errors as the state of a PKM is defined by its end-effector and not its joints.

It is clear that modeling and simulation tools are still required in order to improve the performance of parallel kinematic systems. Such models should take the actuators dynamics, mechanisms, controls and their interactions into account in order to proper simulate the behavior of parallel kinematics machines. Hence, a mechatronic modeling approach must be taken.

2.4 Dynamics modeling of parallel kinematic mechanisms

Much work is found in the kinematics and dynamics modeling for serial and parallel manipulators as well. Usually, computational tools are used to assist in the modeling of robotic systems. In this regard, in the work of Žlajpah [56] an overview of several computational tools commonly used for simulations, such as Matlab/Simulink and Dymola/Modelica, are shown. Furthermore, simulation tools allow to understand how a given system works under specific set of conditions and with different design configurations.

For the dynamics of serial mechanism, it is commonly found that Lagrange, recursive Newton-Euler and Principle of virtual work methods are employed to obtain the dynamic expressions [57]. Nevertheless, for parallel mechanism the applicability of such methods is difficult due to the closed loops kinematics.

Multibody models of the mechanism [58] or the forward dynamic problem [41] have been applied to obtain a simulation model for parallel mechanism. When high loads are present the application of multibody models is justified since they can model the flexible behavior of the components. However, costly modeling software packages may become a limiting factor. Moreover, with both approaches it is common to model the actuators dynamics assuming a rigid body behavior, as unnecessary more complex models will result. This may overlook the contribution of flexibilities of the actuators on the global dynamics of the system.

Also, with the previous mentioned formulations the forces due to the mechanism cannot be introduced into the control algorithm to perform computed or feed-forward torque control since the force in the actuators are not available. For this, the inverse dynamic model (IDM) provides the required torques or forces in the actuators to perform a given commanded motion. In fact, the inverse dynamic model has been used in the past for

model based control. In [53] the inverse dynamic model of a 3-RCC robot is used in a computed torque control and compared with a PID control. Also Lou [59] employs a model based control to reduced the contouring error of a 3 degrees of freedom translational parallel mechanism called *Orthopod*. Moreover, Codourey [60] developed a model-based control for a Delta-Robot based using the IDP. In a similar way, Yang et al. [61] developed a computed force and velocity control for a 6-DOF parallel mechanism also employing the inverse dynamics. It can be seen that the inverse dynamic problem provides to be useful for advanced control programming, however finding the inverse dynamic problem can be cumbersome specially on parallel kinematic systems.

Furthermore, it is usually found that the data given by the manufacturers of the components is not sufficient to build an accurate model. Hence an identification step is required in order to adjust the system model using the experimental signals to find estimates of the dynamic and friction parameters. The latter plays a key role in the dynamics of the system. In fact, for high precision machining and in motion control, friction is an important source of uncertainties and disturbances[62], thus it has also to be considered. The main problem with friction is its nonlinear nature and some of the parameters are unknown beforehand.

2.4.1 Parallel manipulators dynamics

To obtain the dynamics for mechanisms, formulations such as Newton-Euler, the principle of virtual work or Lagrangian methods have been applied. In the following sections, a discussion on these methods will be given. In addition, other less known methods, such as Gibbs-Appell equations, the principle of energy equivalence and Boltzmann-Hamel equations are also discussed.

2.4.1.1 Newton-Euler

Following the Newton-Euler method the dynamic equations of parallel mechanisms can be obtained. In Fig. 2.26 the free body diagram for linkage i^{th} of a robotic system is shown, where two joints are shown, joints i and $i-1$. The moment $\mathbf{M}_{i-1}$ is the moment that link $i-1$ exerts on link i. In the same way, $\mathbf{F}_{i-1}$ is the force from link $i-1$ applied to link i. In a similar way, if a link $i+1$ exist, $\mathbf{M}_i$ and $\mathbf{F}_i$ are the resultant moments and forces applied to the link $i+1$. This process can be applied to a mechanism with n links. It can be noticed that no external forces or moment are so far considered. These can be taken into account by defining $\sum \mathbf{F}_{e_i}$ and $\sum \mathbf{M}_{e_i}$ as the sum of the forces and moments respectively and externally applied to the link i.

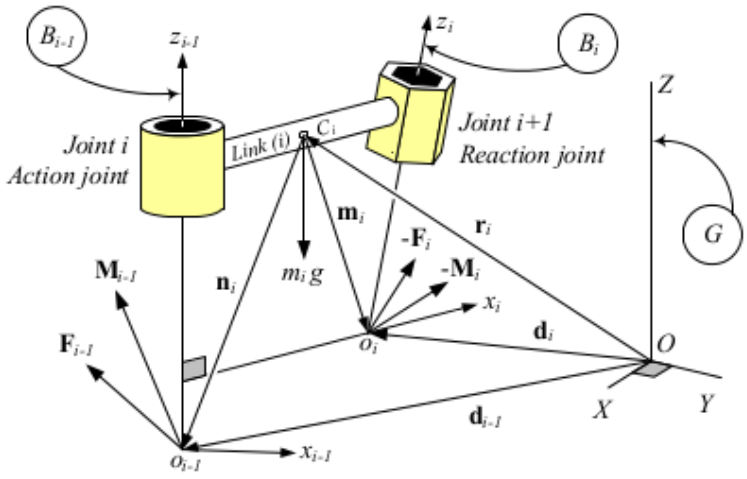


Figure 2.26. Free body diagram of link i by Jazar [63].

Moreover, the equation of forces and moments of the i^th link with respect to a reference coordinate system O can be written as follows.

$$^{O}\mathbf{F}_{i-1} - ^{O}\mathbf{F}_i + \sum ^{O}\mathbf{F}_{e_i} = m_i {}^{O}a_i \tag{2.9}$$

$$
\begin{aligned}
&{}^{O}\mathbf{M}_{i-1} - {}^{O}\mathbf{M}_{i} + \sum {}^{O}\mathbf{M}_{e_i} + \left({}^{O}d_{i-1} - {}^{O}\mathbf{r}_{i}\right) \times {}^{O}\mathbf{F}_{i-1} \\
&- \left({}^{O}d_{i} - {}^{O}\mathbf{r}_{i}\right) \times {}^{O}\mathbf{F}_{i} = I_i {}^{O}\dot{w}_i
\end{aligned}
\tag{2.10}
$$

These two equations are regarded as the equations of motion for the link i. As it can be seen, to calculate the forces or torques at a given joint (i.e. an actuated joint), all the reaction forces between joints have to be calculated. Also, this method presents a recursive nature that is advantageous for computer models. In addition, this method can be used for the dimensioning and mechanical design of joint components as it provides the loads requirements for each joint [8, 64].

It is worth mentioning that an interesting approach has been taken by Oftadeh [65] which employs intermediate variables from the joint-space and algebraic matrix manipulation tools to obtain explicit dynamic models for a Gough-Stewart platform. Nevertheless, the applicability of Newton-Euler methods on parallel mechanisms is difficult as mentioned by Ebert & Uphoff [66], as the internal forces and moments of the mechanism are also calculated, causing the *"...formulation, implementation and debugging of the algorithm extremely cumbersome and prone to errors"*.

2.4.1.2 Principle of virtual work

The principle of virtual work and the D'Alembert's principle have been used for the dynamics analysis, in which a system is regarded to be in equilibrium if the virtual work done by the forces and moments for a given virtual displacement is zero [53]. The work of Wang and Gosselin [67] can be used as a reference, where it is described in detail the application of the principle of virtual work to analyze the dynamics of parallel mechanisms. Moreover, two examples are giving to illustrate this approach, a four-bar planar mechanism and the Gough-Stewart spatial platform.

What is important of this method is that the calculation of the constraint forces is avoided, thus resulting in simpler expressions than the Newton-Euler approach [53, 64]. Furthermore, as the equations of mo-

tion obtained are simpler and computational efficient, the resulting inverse dynamic model can be included in robot control implementations.

2.4.1.3 Lagrange approach

With the Lagrange approach the forces and torques at the actuators are obtained by means of differential equations. To this end, the Lagrangian in Eq. 2.11 is defined as the kinetic (T) and potencial (V) energies difference.

$$\mathcal{L} = T - V \tag{2.11}$$

To obtain the equation of motion of a system, the *Lagrange's equations* in Eq. 2.12 is applied as follows:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}_{mech}}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial \mathcal{L}_{mech}}{\partial \mathbf{q}} = \mathbf{Q}. \tag{2.12}$$

In this equation, $\mathbf{q}$ is defined as the generalized coordinates, which are used to express the kinetic and the potential energies, and the length of $\mathbf{q}$ is the number of degrees of freedom of the system. Moreover $\mathbf{Q}$ is the vector of generalized nonconservative forces. As a result of applying the Lagrange's equation, the expression in Eq. 2.13 is obtained.

$$\mathbf{D}\left(\mathbf{q}\right)\ddot{\mathbf{q}} + \mathbf{C}\left(\mathbf{q}, \dot{\mathbf{q}}\right)\dot{\mathbf{q}} + \mathbf{G}\left(\mathbf{q}\right) = \mathbf{Q} \tag{2.13}$$

Where $\mathbf{D}$ is the inertial matrix, $\mathbf{C}$ is the *Coriolis* and centrifugal forces matrix, $\mathbf{G}$ represents the gravitational matrix and $\mathbf{Q}$ is the vector with the forces and torques for all the actuators. This expression is sometimes considered as an *explicit or detailed form* because each physical effect is obtained separately [68].

This form is important in the context of robotic systems as it is appropriate for control implementation. What is more, Eq. 2.13 can be easily

applied on serial systems, where the joint position inputs are independent and can be used as generalized coordinates. However for parallel mechanism, joint inputs are coupled due to the kinematic constraints. As a consequence, using them as generalized coordinates would imply calculating the Lagrange's multipliers $\lambda' s$, which represent the link forces between joints [68].This results in additional expressions that must be solved for parallel mechanisms.

In summary, Lagrangian analysis applied for open-chained mechanism can result in relatively simple expressions. However, when applied on closed-loop mechanism complex set of equations are commonly obtained as a result of the Lagrange's multipliers and trigonometric expressions appearing for spatial rotations. As an alternative, less known methods of analytical mechanics can be applied to overcome the difficulties of applying the methods just mentioned.

2.4.1.4 Gibbs-Appell equations

Considering a rigid body with linear acceleration $\mathbf{a}_p$, angular velocity $\mathbf{w}$ and acceleration $\boldsymbol{\alpha}_p$, and inertia matrix $\mathbf{I}_{mp}$ with respect to a reference system fixed to the body, the Gibbs-Appell function S is defined as in Eq. 2.14. This function is also known as the *energy of acceleration* for a rigid body because of its similarity with the kinetic energy.

$$S = \frac{1}{2}\left(m_p \mathbf{a}_p^T \mathbf{a}_p\right) + \frac{1}{2}\left(\boldsymbol{\alpha}_p^T \mathbf{I}_{mp} \boldsymbol{\alpha}_p\right) + \boldsymbol{\alpha}_p^T \left(\mathbf{w} \times \mathbf{I}_{mp} \mathbf{w}\right) \tag{2.14}$$

It is then possible to obtain the dynamics by differentiation of the Gibbs-Appell function for a rigid body with respect to the acceleration vector $\ddot{\boldsymbol{\gamma}}$ like in 2.15 [66], where $\boldsymbol{\tau}_g$ and $\boldsymbol{\tau}_{ext}$ represent the gravitational and external generalized forces respectively. In other words, the gradient of function S with respect to $\ddot{\boldsymbol{\gamma}}$ is:

$$\frac{\partial S}{\partial \ddot{\boldsymbol{\gamma}}} = \nabla_{\ddot{\boldsymbol{\gamma}}}(S) = \boldsymbol{\tau}_g + \boldsymbol{\tau}_{ext} = \boldsymbol{\tau} \tag{2.15}$$

Where,

$$\ddot{\boldsymbol{\gamma}} = \begin{Bmatrix} \mathbf{a}_p \\ \boldsymbol{\alpha}_p \end{Bmatrix} = \begin{Bmatrix} a_x \\ a_y \\ a_z \\ \dot{w}_x \\ \dot{w}_y \\ \dot{w}_z \end{Bmatrix} \tag{2.16}$$

Hence, the following equations of motion result after performing the differentiation:

$$\frac{\partial S}{\partial \mathbf{a}_p} = m\mathbf{a} = \begin{Bmatrix} f_{px} \\ f_{py} \\ f_{pz} \end{Bmatrix} \tag{2.17}$$

$$\frac{\partial S}{\partial \boldsymbol{\alpha}_p} = \mathbf{I}_{mp}\dot{\mathbf{w}} + \mathbf{w} \times \mathbf{I}_{mp}\mathbf{w} = \begin{Bmatrix} m_{mpx} \\ m_{mpy} \\ m_{mpz} \end{Bmatrix} \tag{2.18}$$

$$\boldsymbol{\tau} = \boldsymbol{\tau}_g + \boldsymbol{\tau}_{ext} = \begin{Bmatrix} m\mathbf{a} \\ \mathbf{I}_{mp}\dot{\mathbf{w}} + \mathbf{w} \times \mathbf{I}_{mp}\mathbf{w} \end{Bmatrix} \tag{2.19}$$

2.4.1.5 Description of the principle of energy equivalence and Boltzmann-Hamel equations

Another way to deal with the dynamic modeling of a given parallel mechanism, is by splitting the whole system into individual components and analyze them separately. Nevertheless, in order to maintain an equal relation between the assembled mechanism dynamics and the dynamics obtained for each of its components, an equivalence must be established between them.

In this regard, the principle of energy equivalence has been applied by Abdellatif and Heimann to obtain an explicit model using Lagrange's equations of a Gough-Stewart platform [68]. Basically, each component is analyzed separately as if they moved in the assembled mechanism, hence preserving the energy of the system. To this end, the Jacobian matrices for each component are used.

As a result, the dynamics of a parallel mechanism can be obtained in a *"... computational efficient manner and without simplifications"* [68]. Additionally, Abdelatiff and Heimann provide a study demonstrating that the resulting dynamic model using the principle of energy equivalence with Lagrangian formalism, is more efficient than Newton-Euler methods. Finally, they further state that they were able to implement it on commercial control boards.

For those reasons, the principle of energy equivalence approach results in a interesting candidate to develop a mechatronic model for parallel mechanisms, where the control and the performance of the system closely depends on a computationally efficient dynamic model.

Moreover, as it was mentioned in section 2.4.1.3, using the Lagrange's equation can result in complex expressions caused by the chosen generalized coordinates. This issue arises particularly in the case of rigid bodies with spatial rotation, where complex trigonometric expressions are obtained which are cumbersome to derivate for. This will be further addressed in section 3.4.2.

To alleviate this problem, quasi-coordinates can be used instead, which are already used for the Gibbs-Appell equations and for the Boltzmann-Hamel equations [69]. The latter consists of applying the Lagrange's formalism on system described by such quasi-coordinates.

In section 3.4, the principle of energy equivalence and Boltzmann-Hamel equations will be reviewed in more detail. Therein, the application of the principle of energy equivalence on parallel mechanisms will be shown. Furthermore, Boltzmann-Hamel equations will be particularized for a rigid body with spatial rotation only, since the main advantage is appreciated for such motions.

2.4.2 Friction

Although it is frequently modeled as a tangential reaction force resulting from the contact between two surfaces with relative motion [70], friction is a complex and nonlinear phenomenon dependent on factors like contact geometry, surface material, relative velocity and lubrication [70].

In some systems such as in brakes, friction may be desirable to effectively reduce the velocity of a vehicle by dissipating kinetic energy. However, for positioning or robotic systems is a source of inaccuracy. Hence, it is important to consider friction from the beginning of a design stage in order to properly select the system components, and to reduce its effect on the system performance, specially its impact on high-precision devices, by applying friction compensation control techniques.

An alternative is to employ a model-based friction compensation scheme [70], which consists of adding an estimated friction force to the control signal as in Fig. 2.27. However, a good model is required for a proper compensation. A survey of friction models and their application for control purposes has been written by Armstrong-Hélouvry [71]. Also several authors have proposed other friction models, each of them with different degrees of complexity. In this regard, friction models can be classified into two categories: *static* and *dynamic* models.

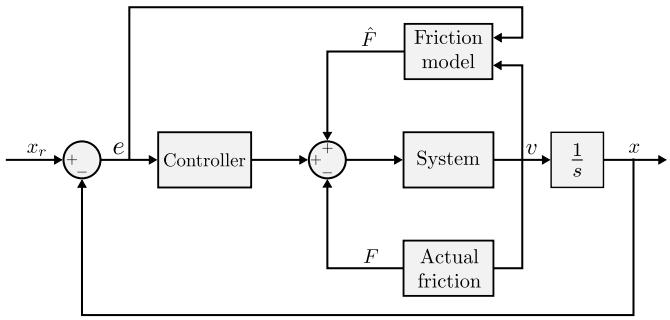


Figure 2.27. Control scheme with friction compensation. (Source: Olsson et al. [70]).

2.4.2.1 Classical friction models

Classical friction models assumes that the phenomena remain constant with time. Classical friction models are usually used for compensations because of their simplicity [70]. In Fig. 2.28, a representation of the known classical friction models are shown.

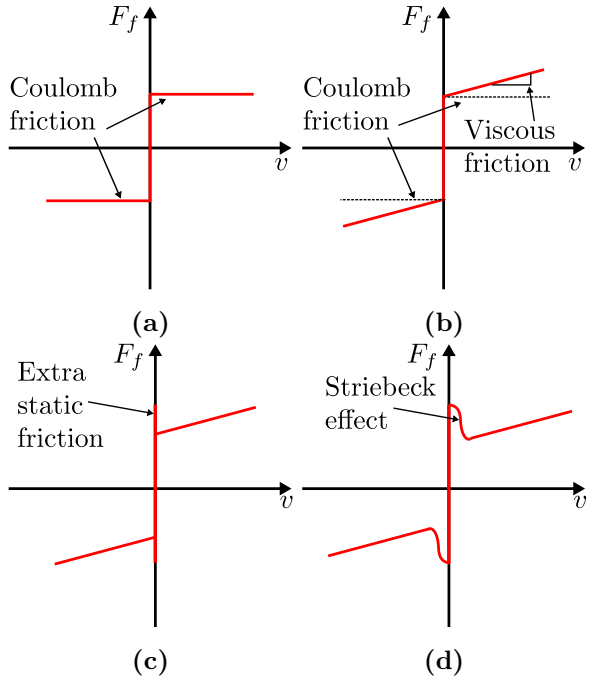


Figure 2.28. (b) Coulomb friction model, (b) Coulumb + viscous friction model, (c) Coulumb + viscous + stiction model (d) Coulomb + viscous friction model with Stribeck effect [71].

The Coulomb friction model is the most simple. It considers friction as a force that opposes the motion and is proportional to the normal load as in equations 2.20 and 2.21, where F_f is the friction force, F_c is the Coulomb friction force and v is the velocity. Additionally, μ is the coefficient of

friction and F_n is the normal force to the contact surfaces. This model has been used by Friedland for compensation in control systems [72]. Therein, a friction *observer* has been employed to estimate the value of F_c.

$$F_f = F_c sgn(v) \tag{2.20}$$

Where,

$$F_c = \mu F_n \tag{2.21}$$

However, the Coulomb friction model can be extended to take the viscous effect in Eq. 2.22 into account as in Fig. 2.28b. Therefore, F_f results as in Eq. 2.23 where c is the viscous coefficient.

$$F_v = cv \tag{2.22}$$

$$F_f = F_c + F_v = F_c sgn(v) + cv \tag{2.23}$$

Another experimental observation from Coulomb about friction is that it reaches a maximum when the relative motion is about to happen, and then it decreases up to a specific value [70, 73]. In other words, friction opposes forces applied externally preventing the motion until a value is reached. This phenomena is known as *Stiction* and is represented in Fig. 2.28c.

Lastly, the Stribeck model in Fig. 2.28d considers that friction decreases continuously within a low velocity range. This is known as the *Stribeck friction* and is dependent on the external force F_e and the static force F_s. Also, a common parameterization of the velocity dependent term, among others proposed [71], is shown in Eq. 2.25 [70]. Therein, v_σ is known as the *Stribeck velocity*.

$$F_f = \begin{cases} F_f(v) \text{ if } v \neq 0 \\ F_e \text{ if } v = 0 \text{ and } |F_e| < F_s \\ F_s sign(F_e) \text{ otherwise} \end{cases} \tag{2.24}$$

Where,

$$F_f(v) = F_c + (F_s - F_c)e^{-|v/v_\sigma|^{\delta\sigma}} + cv \tag{2.25}$$

As it is mentioned by Olsson, the above models are somewhat inconvenient for simulation and control purposes since it is not known when the velocity is zero [70]. To solve this, the Karnopp [74] model can be followed in which a velocity range is defined as the *zero velocity interval*. If the actual velocity is within this range, then the output friction force is zero. However, this may result in stick-slip behavior if the velocity range is too wide [70]. Also, the friction is not well modeled within the zero velocity interval.

Another approach to model the friction in the low velocity range is by approximating the *relay* behavior in the zero velocity crossing with the hyperbolic tangent as in Eq. 2.26 [75]. The factor β can be used to moderate the friction behavior accordingly. As a result, a continuous model at the zero velocity interval is found. This can be easily seen in Fig. 2.29a.

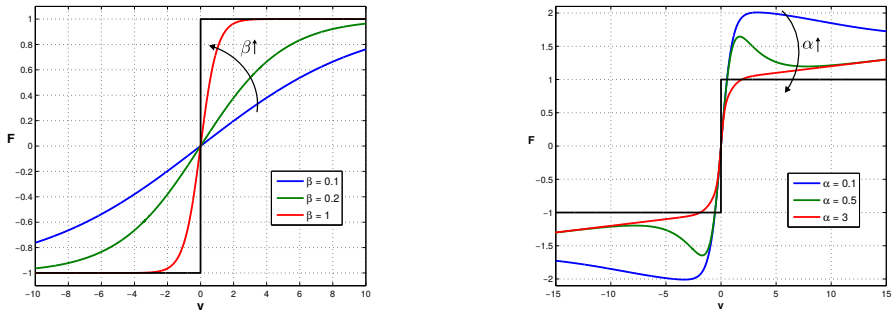
$$F_f = F_c \; tanh(\beta \; v) \tag{2.26}$$

Furthermore, the model in Fig. 2.29a can be extended to consider the viscous friction and the Stribeck effect as in Eq. 2.27. Also, the behavior of the model is shown in Fig. 2.29b.

$$F_f = c \; v + (F_c + F_{cs} \; sech(\alpha v)) \; tanh(\beta \; v) \tag{2.27}$$

2.4.2.2 Advanced friction models

The aforementioned models are regarded as classical friction models and rely of convenient simplifications to the actual more complex phenomena. Although this models may be employed in general applications, for precisions in the micrometer range more advanced models are required [76]. To



(a) **(b)**

Figure 2.29. (a) Continuous Coulomb friction model and (b) continuous Coulomb + viscous + Stribeck friction model.

improve the description of friction. For example, the model by Armstrong [71] can be employed to take the stiction into account in classical friction models. With this model, the sticking and sliding conditions are separated into two separate models and an additional parameter must be used to select between the two of them.

Also, the Dahl model [77] has been employed for friction compensation control. It is based on the condition of stress-strain curve of material and modeled as a differential equation. The first approximation of the model was not dependent of the velocity but of the displacement. Nevertheless, Dahl extended the model to include the velocity. Despite this, Olsson mentions that several extensions to the model have been proposed to include the stiction and the Stribeck effect as it was not originally considered.

Many other models have been also worked out such as the *Bristle* and *LuGre* models to find even more detailed representations. These two are based on the microscopic observations of the materials in which asperities (bristles) are present. This bristles act as springs when two surface are subjected to a tangential force. Moreover, the LuGre model introduces the damping and stiffness of the bristles, and the friction force is obtained also

as a velocity dependent function.

A relatively recent model employed is the Generalized Maxwell-Slip friction model [76, 78]. A problem with the LuGre model is that it does not take the adhesive forces into account in presliding conditions. In such cases, friction behaves as a hysteresis function dependent on the displacement, leading to complex dynamical behavior.

In essence, the generalized friction model consists of N elastic damped elements systems as in Fig. 2.30. As a result, the overall friction force is a sumation of the force on each element as described in Eq. 2.28. Therein, k_i represents the stiffness for each element, σ_i is the viscoelastic coefficient. In this way, the term $k_i\ z_i(t)$ represents the elastic behavior under sliding conditions, $\sigma_i \dot{z}_i(t)$ is the *viscous-elastic* effect and $f(v)$ represents the viscous friction. It is worth observing that the number of unknown parameters is proportional to the number of elements used. Therefore, they must be experimentally identified by employing nonlinear optimization methods [76].

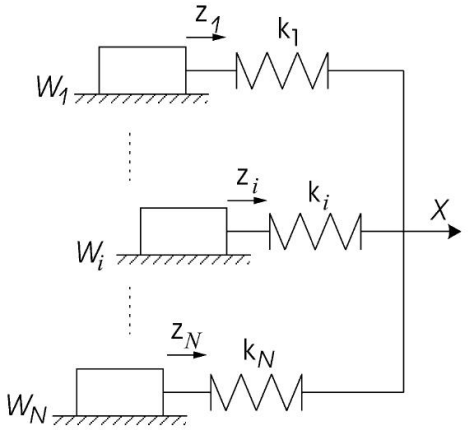


Figure 2.30. Representation of the Generalized Maxwell-slip friction model (Source: Al-Bender et al. [76]).

$$F_f(t) = \sum_{i=1}^{N} (k_i \; z_i(t) + \sigma_i \dot{z}_i(t)) + f(v) \tag{2.28}$$

Furthermore, it has been reported that the GMS model is capable of correctly representing the friction force with good accuracy [79]. Finally, because of its relatively simplicity, it is a model suitable for simulation and compensation in control, provided that the unknown parameters are correctly identified.

2.5 Identification

An essential aspect of the mechatronic design approach is to use accurate models in order to make appropriate design decisions. In mechatronics applications and robotics, also model based control is employed to improve the accuracy and performance of such systems. all these models rely on dynamic parameters that are estimated from CAD models or analytical approximations. However, such estimations are often not sufficient to build detailed models, or simply not information is available for certain phenomena, such as friction or damping. Therefore, once a first prototype has been built, or several components of the prototype are available and operative, an experimental identification procedure must be used in order to obtain accurate values of the unknown parameters, which are then used to fine tune the models.

In the identification theory, three basic concepts are applied to systems modeling. *White box* models consists of a known model structure and parameters. However, these models rely on parameter values that do not match the actual ones of the system, and it is also possible that the model structure is oversimplified. As result, the estimated behavior may differ from the performance of the actual system. Nevertheless, they are convenient for initial approximations.

On the other hand, *black box* models are built from input and output

measured data of the system. Since only data is used to infer a model of the system, their parameters may not have a relation to the actual physical properties. Also, it is required that relatively large sets of data are used to obtain a reliable model.

A third option between the above discussed type of models are the so called *grey box* models. They consist of a previously known model structure, which can be obtained by analytical methods, but still with unknown parameters. This type of identification models are key to obtaining model parameters values with physical meaning from experimental data, after an identification procedure is applied.

In this regard, a standard identification procedure consists of 6 steps according to Swevers [80], namely (1) Modeling, (2) Experiment design, (3) Data acquisition, (4) Signal processing, (5) Parameter estimation and (6) Model validation. The identification procedure is shown in Fig. 2.31. The modeling step is where the equations of motion are obtained to describe the dynamics, as function of the unknown parameters of the system. The equations of motion are written as in Eq. 2.29.

$$\mathbf{F} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{G} \tag{2.29}$$

Nevertheless, the identification procedure is greatly simplified if Eq. 2.29 is rewritten in linear form with respect to the unknown parameters in Φ like in Eq. 2.30 [80, 81]. What is important to note is that the matrix $\mathbf{K}$ depends only on the position $\mathbf{q}$, velocity $\dot{\mathbf{q}}$ and acceleration $\ddot{\mathbf{q}}$, which must be measured, as well as the forces or torques $\mathbf{F}$. Then, a linear regression can be performed to calculate the unknown parameters grouped in $\mathbf{\Phi}$. Friction can also be included into 2.30 provided that the friction model is also in linear form.

$$\mathbf{F} = \mathbf{K}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{\Phi} \tag{2.30}$$

In order to perform a linear regression, a trajectory must be prepared accordingly for the identification. The experiment design step in Fig. 2.31

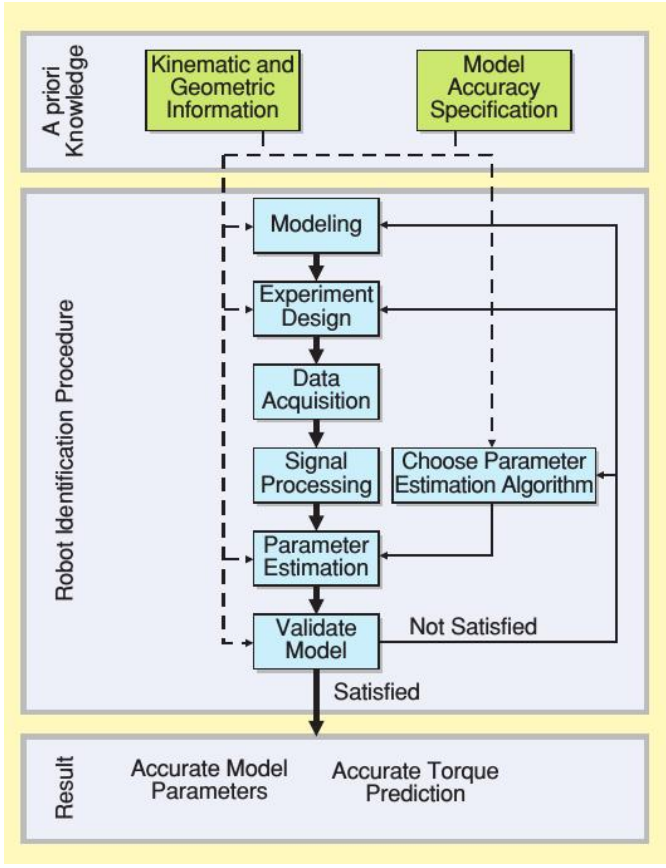


Figure 2.31. Identification procedure for robots. (Source: Swevers et al. [80]).

consists on preparing a trajectory to guarantee that the parameters can be estimated accurately. For that matter, trajectories can be parameterized as Fourier series [80], sum of sinusoids [82], or any periodic signal that results in a sufficient excitation of the dynamic parameters. In any case, the parameters of the trajectory remain unknown, thus, an optimization routine is performed to obtain them according to a previously define objective function, such as the condition number among others [51].

Once the trajectory is prepared, it is executed on the actual robotic system where the *data acquisition* takes place. Afterwards, the signals are post-processed in the *signal processing* step to remove noise or other disturbances in the signals. Finally, the unknown parameters are estimated using the experimental signals and the model is updated with the parameters values. However, to complete the identification process, the updated model must be validated with another trajectory to verify whether the estimated parameters are not biased by the signals used for the identification.

The already mentioned steps describe the general procedure used for identification of the unknown parameters of a given system. On the other hand, identification methods can be classified in two approaches as described by Wu et al. [83]: *off-line* and *on-line* identification methods. In the following, these two approaches will be briefly introduced to establish a reference for further research based on the work of Wu et al. [83] among others.

2.5.1 Off-line identification

For off-line identification the inputs and outputs signals are collected and stored previous to performing any analyzing. In other words, the data is not analyzed during normal operation of the system. Three essential methods can be performed in this regard: physical, CAD and identification.

2.5.1.1 Physical experimentation

A way to determine the value of dynamic parameters is by physical inspection of the system's components. In this regard, vibration analysis can be performed on the mechanism links to find the inertial parameters, from the *frequency response functions*(FRF). Modal analysis is another alternative to identify, the damping and stiffness matrices. Furthermore, a minimization of the error between the theoretic values and the frequency response can be done to also calculate the inertial parameters of the system.

These methods require disassembling the mechanism which is a drawback for two reasons. The first problem is that the disassembled system is not ready for normal operation. This may be assumable in academic contexts but not in industrial settings. Another problem as reported by Wu et al. is that the experimentation is a tedious process [83] and deep knowledge is required.

2.5.1.2 CAD modeling

An alternative to calculate the inertial parameters is by using CAD models of the system during the design stage, which allows for a priori estimation of the parameters and preparing of suitable control strategies. Nevertheless, the estimation of the parameters in this manner does not considers deviations due to the manufacturing and assembly processes which may have an influence. Also, friction plays an important role in robotic systems and it cannot be obtained from CAD models.

2.5.1.3 Identification procedures

Identification techniques are used to find the values of the unknown inertial and friction parameters by introducing an input signal, a trajectory in the case of robotic mechanisms, and collecting the experimental data such as the torque. Once the experimental signals are obtained and further post-processed, the parameters are estimated by minimizing an objective function previously defined.

It has been reported that this approach is easy to implement and the estimated parameters are accurate with this approach [42, 83]. Also, the same procedure has been applied on serial robots, such as the KUKA IR/361 industrial robot [84] and the PUMA 560 [85], as well as in parallel robots [51, 86, 87].

2.5.2 On-line identification

On-line approach identification methods are commonly employed for advanced control algorithms like *adaptive control*, where the parameters values are estimated from the available measurements to fit a dynamic model to the actual dynamic response of the system.

A common issue in off-line identification is that time-varying parameters cannot be taken into account. Also, constrained workspace or motion limits in position, velocity and acceleration, may restrict the capacity of identifying some of the parameters due to an improper excitation. In this regard, adaptive control has been applied to overcome such limitations. In theory, adaptive control is able to manage situations where modeling uncertainties are present. In the work by Cazalilla et al., an adaptive control is employed to improve the performance of a 3-DOF parallel manipulator [88]. Also in the work of Honegger et al., a non-linear adaptive controller is developed to identify the dynamic parameters of the inverse dynamic model and then used for feed-forward compensation [89].

Nevertheless, a drawback of adaptive controllers is that their use may lead to instability issues that may result in system failures [90]. For this reason, more advanced controllers have been developed such as *robust control* and $\mathcal{L}_1$ adaptive control theory [91].

2.5.3 Pseudo-linear regression identification with separable parameters

An off-line approach applied for system identification is by pseudo-linear regression methods. Assuming a time-varying model of the form of Eq. 2.31, where ρ contains the unknown parameters of the system and η is a vector with non-linear parameters. It is worth mentioning that ρ and η must be two disjoint sets.

$$\boldsymbol{\tau}(t \mid \boldsymbol{\rho}, \boldsymbol{\eta}) = \boldsymbol{\rho}^T \ \boldsymbol{\phi}(t, \boldsymbol{\eta}) \tag{2.31}$$

In such problems, it is usual to use a minimization of the nonlinear functional in Eq. 2.32. Additionally, the latter equation can be also written as in Eq. 2.33.

$$r(\boldsymbol{\rho}, \boldsymbol{\eta}) = \sum_{i=1}^{m} (\boldsymbol{\tau}(t_i) - \boldsymbol{\rho}^T \ \boldsymbol{\phi}(t_i, \boldsymbol{\eta}))^2 \tag{2.32}$$

$$r(\boldsymbol{\rho}, \boldsymbol{\eta}) = \|\boldsymbol{\tau}(t_i) - \boldsymbol{\Phi}(\boldsymbol{\eta})\boldsymbol{\rho}\|^2 \tag{2.33}$$

Where,

$$\boldsymbol{\Phi}(\boldsymbol{\eta}) = \begin{bmatrix} \phi_1(t_1) & \phi_2(t_1) & \dots & \phi_n(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \dots & \phi_n(t_2) \\ \vdots & \vdots & & \vdots \\ \phi_1(t_m) & \phi_2(t_m) & \dots & \phi_n(t_m) \end{bmatrix} \tag{2.34}$$

Either the parameters in $\boldsymbol{\eta}$ and in $\boldsymbol{\rho}$ are unknown. Finding the optimal parameter as in equations 2.32 and 2.33 is not straightforward. However, the minimization of Eq. 2.33 can be separated in such a way that the consideration of the parameters in $\boldsymbol{\rho}$ is deferred [92]. To this end, a new functional equation for minimization can be defined dependent only on the $\boldsymbol{\eta}$ parameters as in Eq. 2.35, and the optimal parameters of $\hat{\boldsymbol{\rho}}$ are somewhat postponed to be calculated as in Eq. 2.36

$$r_2(\boldsymbol{\eta}) = \|\left[\mathbf{I} - \boldsymbol{\Phi}(\boldsymbol{\eta}) \ \boldsymbol{\Phi}^+(\boldsymbol{\eta})\right] \boldsymbol{\tau}\|^2 \tag{2.35}$$

$$\hat{\boldsymbol{\rho}}(\hat{\boldsymbol{\eta}}) = \boldsymbol{\Phi}^+(\hat{\boldsymbol{\eta}})\boldsymbol{\tau} \tag{2.36}$$

Where $\mathbf{I}$ is the identity matrix and $\boldsymbol{\Phi}^+(\boldsymbol{\eta})$ is the Moore-Penrose pseudoinverse matrix of $\boldsymbol{\Phi}(\boldsymbol{\eta})$.

The pseudo-linear regression method here presented has been applied by Wernholt and Gunnarson [75, 82]. Therein a three step identification procedure has been used to estimate the rigid body dynamics, friction and flexibilities parameters of a industrial Robot.

Part II

Mechatronic Modeling

Chapter 3

Mechatronic Modeling Procedure

Due to the complexities in the design of parallel kinematic mechanisms, with their actuators and control, and the different design decisions involving these components, in this work it is proposed a methodology to develop a mechatronic model which helps in the design process of parallel kinematic machines. The novelty of the approach taken is that the manipulator is decoupled from the actuators, which allows for analyzing each component separately and are easily integrated to study their interactions. This methodology is specially useful to design mechanisms with high velocity motions like Pick & Place mechanisms, with large bandwidth such as in multi-axial simulating tables, or when high loads must be moved.

In Fig. 3.1 a scheme of the mechatronic analysis procedure is shown. With the mechatronic model developed considering the actuators, control and the mechanism, tests in the time and frequency domain are carried out. Regarding the time domain tests, the motor torques are calculated and the positions and velocities simulations using the model are compared with the commanded signals. As for the frequency domain tests, the bandwidth of

the actuators with and without the mechanism attached is calculated. In this way the calculated torques can be used to properly select the actuation system, and the positioning precision can be assessed. Afterwards, the actual system can be built, and its performance can be compared with the model simulations. Nevertheless, when using the theoretic values of the dynamic parameters, differences between the model and the actual system may result, specially from the effect of the friction in the system. As a consequence, an identification procedure is applied to find the actual parameters values. This is addressed in detail in chapter 5.

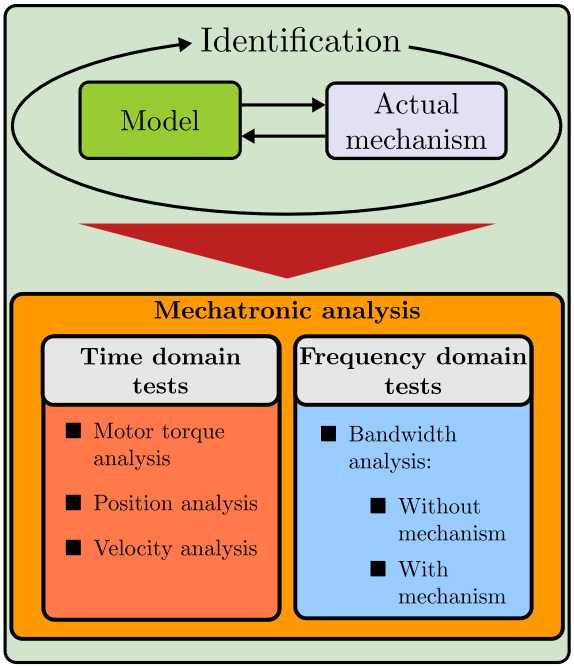


Figure 3.1. Representation of the main components of a parallel mechanism and its interactions.

In this chapter, first some general considerations for the modeling will be described. Then, the modeling for the actuators will also be consid-

ered along with the control. Finally, it will be shown the approach taken to model the dynamics of parallel mechanisms using the methods of the principle of energy equivalence and Boltzmann-Hamel equations.

3.1 Mechatronic modeling for parallel mechanisms

Three important subsystems can be identified in a parallel kinematic machine as in Fig. 3.2: (1) the actuators, (2) the control, and (3) the mechanism itself. The forces or torques of the actuators are the inputs of the mechanism that causes its motion. At the same time, the mechanism motion and external forces act in opposition to the actuators, hence they are considered as disturbances.

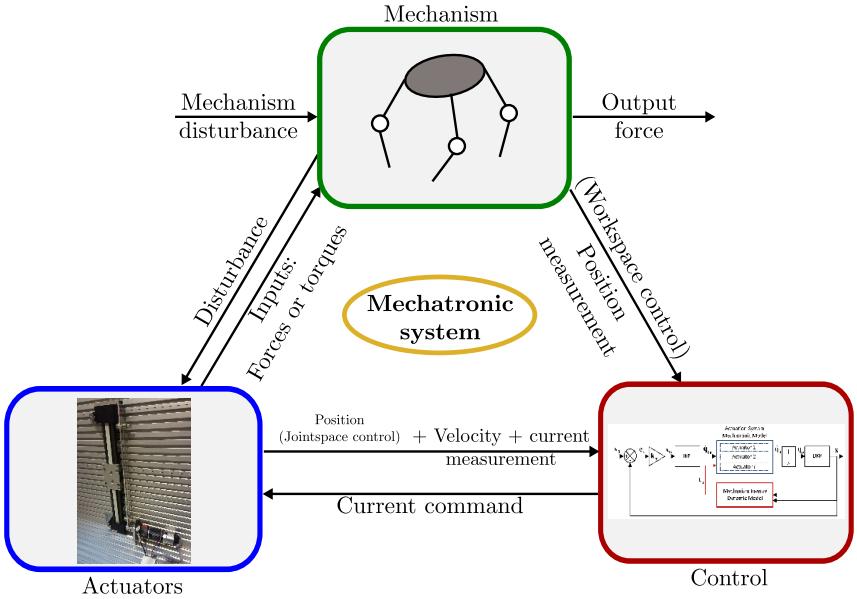


Figure 3.2. Representation of the main components of a parallel mechanism and its interactions.

Additionally, the interaction between the actuators and the control is also taken into account. By measuring the velocity, using the encoders, and the current of the actuators, the current commands required to drive the motors are calculated and supplied to the actuators. In the case that the control is performed in the joint-space, the positions of the actuators are also used and a linear single-axis position controller is considered. If a direct measurement of the position of the end-effector is possible and workspace control is used, the current commands to the actuators are calculated from the position measurements.

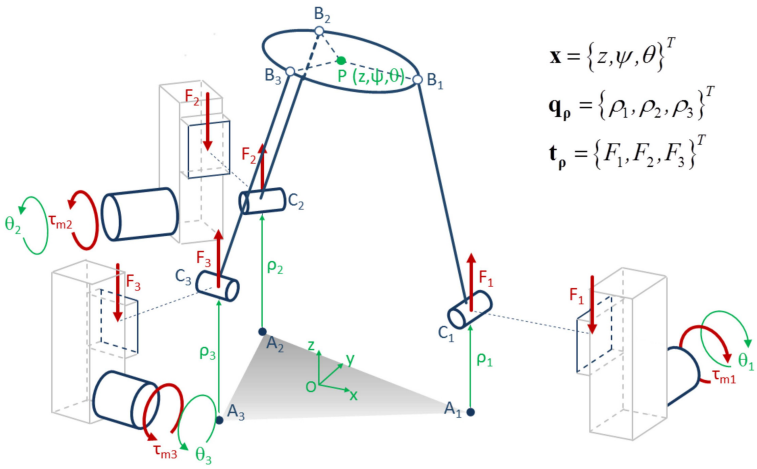


Figure 3.3. Disturbance model between the manipulator and the actuators.

To obtain the mechatronic model taking these components into account, in the present work it is proposed to consider the manipulator and the actuators as two independent subsystems in which their interactions is due to the Newton's third law. In this way, the forces required to move the mechanism are considered as disturbances to the actuators, and at the same time these same forces in opposite direction are the inputs for the mechanism. In Fig. 3.3, a 3-PRS manipulator with three prismatic actuators are shown. It can be noted that the manipulator and the actuators have been sepa-

rated. Thus, the interaction between the actuators and the mechanism is represented by the F_i forces shown in the figure. In this way, the actuators which often times may limit the system's performance and the mechanism can be modeled separately and in a more detailed manner.

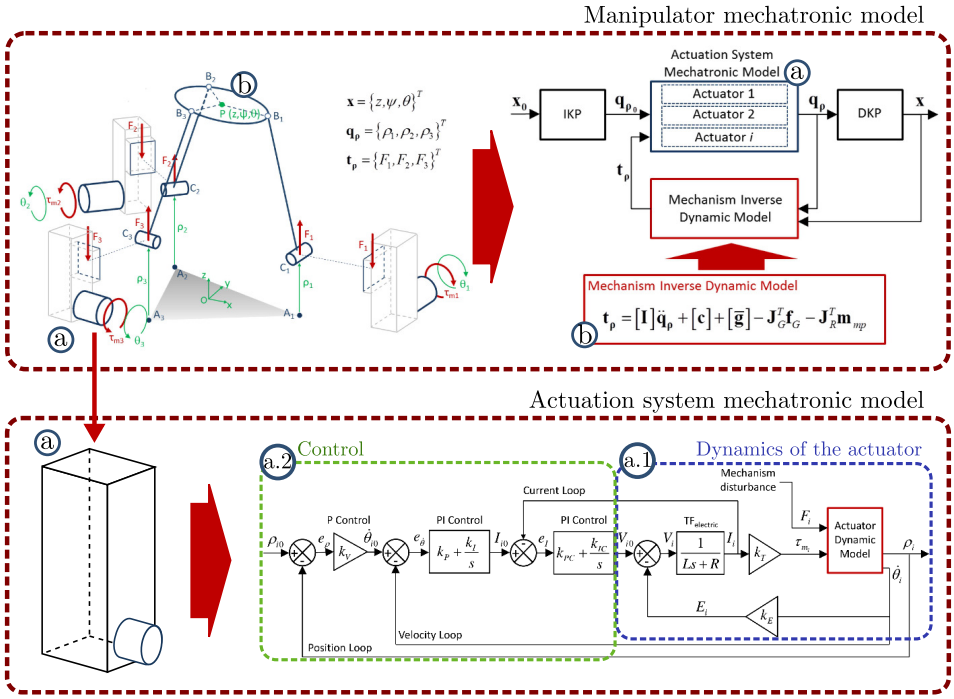


Figure 3.4. Mechatronic model.

Furthermore, the mechatronic model of the manipulator is shown in Fig. 3.4. Therein (**a**) represents the actuators, and (**b**) represents the mechanism itself. As for the actuator two subsystems are considered, (**a.1**) the dynamics of the actuator considering the electrical part and the mechanical dynamics, and (**a.2**) the position and velocity control assuming that a linear-single axis control is used. The mechanical dynamics of the actuators are modeled using transfer functions as will be shown in sections 3.2.2.1 and

3.2.2.2 and represented by the red box inside the (**a.1**) subsystem. Regarding the control algorithm, it must also be modeled because the control type and the loops cycle time affect the performance of the system in terms of trajectory tracking and disturbance rejection.

The mechatronic model of the actuation system is then integrated in the mechatronic model of the manipulator along with the inverse dynamic model of the mechanism, see the top-right part of Fig. 3.4. The inverse kinematic problem (IKP) of the mechanism is used to calculate the joint positions inputs to the actuators $\mathbf{q}_{\rho 0}$, while the direct kinematic problem (DKP) is used to estimate the actual position of the mechanism's end-effector.

Additionally, the influence of the mechanism is included in the mechatronic model with its inverse dynamic model (IDM). To this end, the IDM has to be solved in such a way that it is expressed in the joint-space coordinates. This has the advantage that the IDM resulting equations can also be used in the control algorithm.

As it has been described, in the mechatronic model of the manipulators of Fig. 3.4, the actuator, control and manipulator are considered separately and then integrated. In this way, any change in the design of any of the components can be easily modified and its effect on the overall system can be analyzed.

The following sections are focused on the detailed modeling of each subsystem. In section 3.2 the modeling of electromechanical actuators will be discussed, where lumped parameter models of the electrical and mechanical part are used. Section 3.3 is then dedicated to the modeling of the control algorithms, considering the joint-space, workspace PID control, and the computed torque control. Lastly, the modeling of the manipulator dynamics is discussed in section 3.4, where the principle of energy equivalence and Boltzmann-Hamel equations are explained and applied.

3.2 Mechatronic model of the electromechanical actuators

The mechatronic model of the actuators is composed by the actuator's dynamic model and the controller as shown in Fig. 3.5. A lumped parameters model is used to represent the dynamic behavior of the transmission components. In this way, it is possible to take the inertias, flexibilities and damping of all the components into account. The mechanism and other effects, such as friction, are considered as disturbances in such models.

Regarding the lumped parameter model, a one DOF model is suitable for actuators that could be considered as *stiff*, or that operate in a relatively low frequency range, or that move a mechanism with a low inertia. On the other hand, a 2 or N DOF model is required if the damping and flexibility of its components are not negligible, which happens when moving large inertias, heavy mechanisms and loads, or moving at high frequencies.

Also, any electromechanical actuator must have at least a current control, which supplies the required electrical power to drive the motor. For this reason, the electrical part is taken into account in the model, where its response is modeled in a transfer function considering the resistance, inductance, torque constant of the motor and the counter electromotive constant.

Moreover, depending on the control strategy used, position and velocity regulation loops can also be present such as in linear single-axis control. The cycle time is also taken into account, sampling the signals and using the z-domain for the transfer functions of the system, as the performance of the system in terms of trajectory tracking and disturbance rejections is affected by the cycle time. Also, the cycle time affects the torque output. If a small cycle time is used, smooth torque outputs are obtained. As a consequence, oscillations in the torque caused by the position error corrections are avoided.

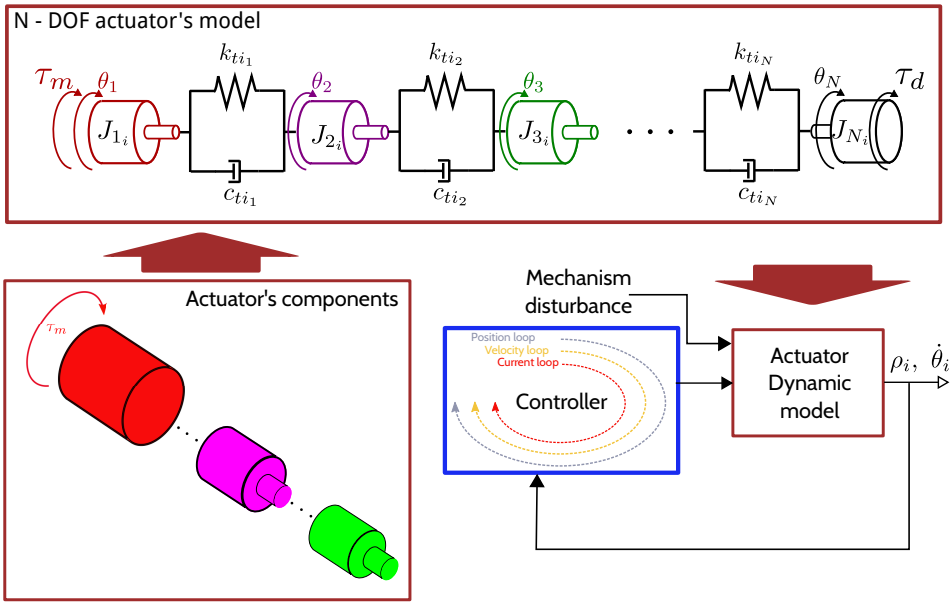


Figure 3.5. $\mathbf{i^{th}}$ actuator mechatronic model.

Finally, if a prismatic joint driven by a rotary motor configuration is used, two encoders can be employed to assess the position of the system, a rotary one for the motor and a linear one for the actuator table. In such cases it is possible to control directly the linear position ρ, however sometimes it is only possible to use the information from the motors for which the linear position must be calculated with the gear ratio, if any, and the linear guide pitch.

3.2.1 Electrical dynamics of DC motors

Most actuation systems make use of electric motors to convert the electric energy to mechanical movement. Regarding a DC motor, an electric model of the actuator can be built with an equivalent RL circuit as shown in Fig. 3.6 [93–95], where L is the inductance of the motor coils, R is the resistance of the windings, V is the voltage input from the controller and E is the

back electromotive force due to the spinning of the motor as described by the Lorentz's Law.

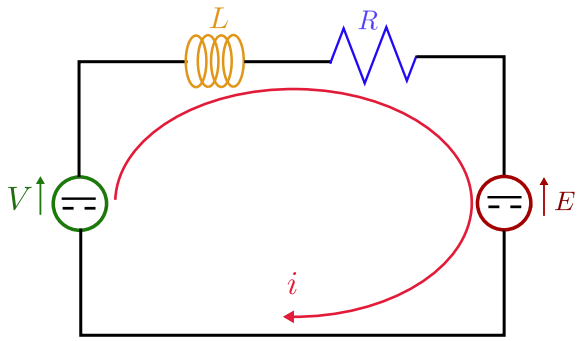


Figure 3.6. Electric circuit of the actuator.

Using Ohm's law for resistors, Faraday's law of induction and Lenz's Law for the motor coil, the dynamics in the circuit of Fig. 3.6 are expressed as in Eq. 3.1.

$$V(t) = Ri(t) + L\frac{di(t)}{dt} + E \tag{3.1}$$

Moreover, the voltage E is proportional to the velocity of the motor and is expressed in Eq. 3.2 where, K_E, is the back electromotive constant of the motor given by the manufacturer.

$$E = K_E\dot{\theta}_m \tag{3.2}$$

Applying the Laplace transform and reorganizing the resulting terms, the transfer function of the electrical part is obtained as in Eq. 3.3.

$$TF_{electric} = \frac{I(s)}{V(s) - E(s)} = \frac{1}{Ls + R} \tag{3.3}$$

A similar approach can be followed to analyze alternating current (AC) motors [21], where an equivalent DC electric circuit can be used taking into

account the root mean squared (RMS) of the voltage and current magnitudes.

3.2.2 Dynamics of the motor and the transmission

3.2.2.1 One degree of freedom model

A one degree of freedom or inertial model is suitable to describe an actuator if its flexibilities can be neglected. For an actuator with a motor inertia J_1 and transmission inertia J_2, it can be modeled as a rigid body as shown in Fig. 3.7.

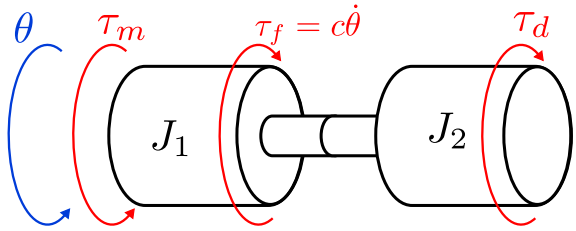


Figure 3.7. One degree of freedom model of the actuator.

The dynamics of this model is described by Eq. 3.4, where τ_m is the motor input torque, J_t is the total inertia which is the sum of J_1 and J_2, c is the viscous friction and τ_d represents all the disturbances that may be present, like the torque due to the mechanism, Coulomb friction and other effects.

$$\tau_m = J_t\ddot{\theta} + c\dot{\theta} + \tau_d \tag{3.4}$$

To obtain the transfer function of the mechanical part of the actuator, the Laplace transform is applied to Eq. 3.4, thus obtaining Eq. 3.5. If disturbances are present the effective torque τ_{ef} is used as input to the transfer function.

$$TF_{mechanical} = \frac{\theta(s)}{\tau_{ef}(s)} = \frac{\theta(s)}{\tau(s) - \tau_d(s)} = \frac{1}{J_t s^2 + cs} \tag{3.5}$$

Note that the above transfer function outputs the position of the motor. However, the velocity output is more convenient since it will be embedded into the velocity regulation loop, see Fig. 3.4. For that, by multiplying by s the velocity transfer function in Eq. 3.6 is obtained.

$$\frac{\dot{\theta}(s)}{\tau_{ef}(s)} = \frac{\dot{\theta}(s)}{\tau(s) - \tau_d(s)} = \frac{1}{J_t s + c} \tag{3.6}$$

This transfer function is then included in the electromechanical model of the actuator represented in Fig. 3.4 by the subsystem (**a.1**) as in Fig. 3.8, where the voltage command V_{i0} is used with the back electromotive voltage E to calculate the voltage input V to the transfer function of the electrical part $TF_{electric}$. The output current I is then passed through the torque constant k_t of the motor to calculate the torque input τ_m. It is used with the disturbance torque τ_d to calculate the input to the transfer function $TF_{mechanical}$ representing the mechanical dynamics of the actuator.

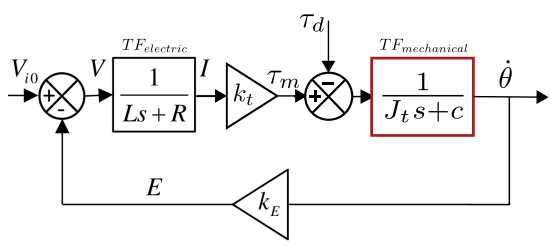


Figure 3.8. Electromechanical model of the actuator with a one degree of freedom mechanical model.

3.2.2.2 Two degrees of freedom model

As mentioned before, if the flexibilities of the system cannot be neglected, a two degrees of freedom model as shown in Fig. 3.9 can be used, where the motor torque τ_m and the disturbance τ_d are the inputs, whereas the angular position θ_ρ is the output. If a linear guide is used to drive the mechanisms,

a conversion from the rotation angle to the linear displacements is done taking into account the pitch of the guide, as will be shown in section 3.2.3 Eq. 3.28.

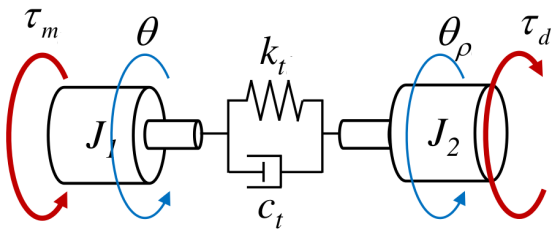


Figure 3.9. Two degrees of freedom model.

Furthermore, the actual angular position in the output θ_ρ is obtained from the ideal position θ_{ρ_m}, and the position variation due to the disturbance θ_{ρ_d} as in Eq. 3.7. Similarly, the actual velocity $\dot{\theta}$ is obtained from the motor's ideal velocity $\dot{\theta}_m$ and the velocity variation $\dot{\theta}_d$ also caused by the disturbance as in Eq. 3.8.

$$\theta_\rho = \theta_{\rho_m} + \theta_{\rho_d} \tag{3.7}$$

$$\dot{\theta} = \dot{\theta}_m + \dot{\theta}_d \tag{3.8}$$

The aforementioned magnitudes are related by four transfer functions. The primary transfer function TF_1 in Eq. 3.9, relates the angular position of the motor without disturbance θ_m and the input torque τ_m. The actuator's flexible dynamics are represented by the damping c_t and stiffness k_t. Moreover, the inertia of the driving part is represented by J_1 whereas J_2 represents the inertia of the transmission. The secondary transfer function, TF_2 in Eq. 3.10, relates the ideal angular position θ_{ρ_m} with the motor's position θ_m. The remaining transfer functions are used to model the disturbances. On one hand, TF_{1d} in Eq. 3.11 relates the disturbance torque τ_d with the angular position disturbance θ_{ρ_d}. On the other hand, TF_{2d} in Eq. 3.12 relates θ_{ρ_d} with the position variation at the motor shaft due to

the disturbance θ_d.

$$TF_1 = \frac{\theta_m}{\tau_m} = \frac{J_2 s^2 + c_t s + k_t}{s^2 \left(J_1 J_2 s^2 + (J_1 + J_2)\, c_t s + (J_1 + J_2)\, k_t\right)} \tag{3.9}$$

$$TF_2 = \frac{\theta_{\rho m}}{\theta_m} = \frac{c_t s + k_t}{(J_2 s^2 + c_t s + k_t)} \tag{3.10}$$

$$TF_{1d} = \frac{\theta_{\rho d}}{\tau_d} = \frac{J_1 s^2 + c_t s + k_t}{s^2 \left(J_1 J_2 s^2 + (J_1 + J_2)\, c_t s + (J_1 + J_2)\, k_t\right)} \tag{3.11}$$

$$TF_{2d} = \frac{\theta_d}{\theta_{\rho_d}} = \frac{c_t s + k_t}{(J_1 s^2 + c_t s + k_t)} \tag{3.12}$$

These transfer functions representing the dynamics of the actuators, are then included in the electromechanical model represented by the subsystem (**a.1**) in Fig. 3.4 as follows:

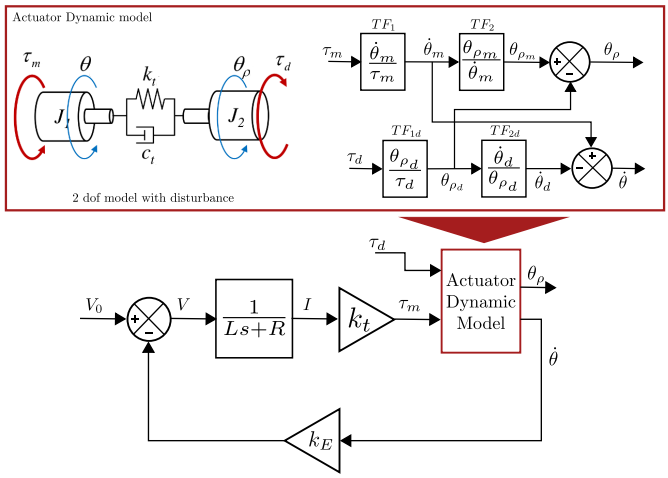


Figure 3.10. Electromechanical model of the actuator with a two degrees of freedom mechanical model.

3.2.2.3 N degrees of freedom

A two degrees of freedom model is often sufficient to describe the compliant dynamics of most transmissions. Nevertheless, if less stiff systems are involved, their flexibilities must be taking into account in the dynamics by employing higher order models [82]. Moreover, a N degrees of freedom model like in Fig. 3.11 can also be used to describe the mechanics of the actuation system if a more precise model is desired, or when information of intermediary components is wanted.

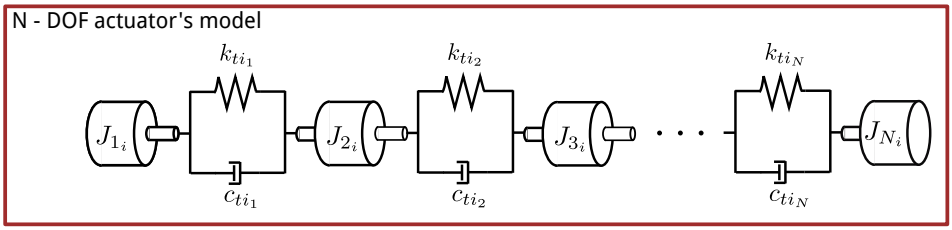


Figure 3.11. Scheme of a N degrees of freedom model.

In a N degrees of freedom model, the dynamic behavior can be described in matrix form as follows:

$$\mathbf{J}\ddot{\mathbf{x}} + \mathbf{c}\dot{\mathbf{x}} + \mathbf{k}\mathbf{x} = \mathbf{F} \tag{3.13}$$

Where,

$$\mathbf{J} = \begin{bmatrix} J_1 & 0 & 0 & 0 \\ 0 & J_2 & 0 & \vdots \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & J_n \end{bmatrix} \tag{3.14}$$

$$\mathbf{c} = \begin{bmatrix} c_1 & -c_1 & 0 & 0 & & 0 \\ -c_1 & c_1 + c_2 & -c_2 & 0 & & \vdots \\ 0 & -c_2 & c_2 + c_3 & -c_3 & & 0 \\ \vdots & & \ddots & \ddots & \ddots & -c_{n-1} \\ 0 & & \cdots & 0 & -c_{n-1} & c_{n-1} \end{bmatrix} \tag{3.15}$$

$$\mathbf{k} = \begin{bmatrix} k_1 & -k_1 & 0 & 0 & & 0 \\ -k_1 & k_1+k_2 & -k_2 & 0 & & \vdots \\ 0 & -k_2 & k_2+k_3 & -k_3 & \ddots & 0 \\ \vdots & & & \ddots & \ddots & -k_{n-1} \\ 0 & & \cdots & 0 & -k_{n-1} & k_{n-1} \end{bmatrix} \tag{3.16}$$

$$\mathbf{x} = \begin{Bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{Bmatrix} \tag{3.17}$$

$$\mathbf{F} = \begin{Bmatrix} \tau_m \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \tag{3.18}$$

Moreover, it is convenient to write Eq. 3.13 in the frequency domain by using the Laplace operator $\mathcal{L}$, thus obtaining the system in Eq. 3.19

$$\mathbf{M}(\mathbf{s}) \begin{Bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{Bmatrix} = \begin{Bmatrix} \tau \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \tag{3.19}$$

Where,

$$\mathbf{M}(\mathbf{s}) = \begin{bmatrix} m_1 s^2 + c_1 s + k_1 & -c_1 s - k_1 & 0 & & \\ -c_1 s - k_1 & m_2 s^2 + (c_1 + c_2) s + (k_1 + k_2) & -c_2 s - k_2 & & \\ & -c_2 s - k_2 & \ddots & \ddots & \\ & & \ddots & m_N s^2 + c_{N-1} s + k_{N-1} \end{bmatrix} \tag{3.20}$$

From the above system, the following transfer functions are obtained.

$$TF_1 = \frac{x_1}{F} = \frac{1}{m_1 s^2 + c_1 s + k_1 - (c_1 s + k_1) TF_{1-2}} \tag{3.21}$$

$$TF_2 = \frac{x_1}{F} = \frac{1}{m_1 s^2 + c_1 s + k_1 - (c_1 s + k_1) TF_{1-2}} \tag{3.22}$$

$$TF_N = \frac{x_N}{x_{N-1}} = \frac{c_{n-1} s + k_{n-1}}{m_n s^2 + c_{n-1} s + k_{n-1}} \tag{3.23}$$

These expression are difficult to handle in natural coordinates. However, by applying modal analysis, the transfer functions can be obtained in a straightforward way as show in Eq. 3.24 and Eq. 3.25.

$$TF_1 = \frac{x_1}{F} = \sum_{i=1}^{n} \left(\frac{\phi_{i1}^2}{s^2 + 2\xi_i \omega_i s + \omega_i^2} \right) \tag{3.24}$$

$$TF_2 = \frac{x_n}{x_1} = \frac{\sum_{i=1}^{n} \left(\frac{\phi_{in}\phi_{i1}}{s^2 + 2\xi_i \omega_i s + \omega_i^2} \right)}{\sum_{i=1}^{n} \left(\frac{\phi_{i1}^2}{s^2 + 2\xi_i \omega_i s + \omega_i^2} \right)} \tag{3.25}$$

Where ϕ_{ij} represents the i^{th} vibration mode of the j^{th} degree of freedom. Also, ω_i and ξ_i are the natural frequency and relative damping of the i^{th} vibration mode.

the modal vector can be written in matrix form as follows.

$$\mathbf{\Phi} = \begin{bmatrix} \phi_{11} & \phi_{21} & \phi_{31} & \cdots & \phi_{n1} \\ \phi_{12} & \phi_{22} & \phi_{32} & \cdots & \phi_{n2} \\ \phi_{13} & \phi_{23} & \phi_{33} & \cdots & \phi_{n3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{1n} & \phi_{2n} & \phi_{3n} & \cdots & \phi_{nn} \end{bmatrix} \tag{3.26}$$

An alternative to model the dynamics of the actuator with N degrees of freedom is by using state-space models. Such models are then included in the mechatronic model as in the case of the 2 degrees of freedom model.

3.2.3 Conversion due to the transmission

It is usual that an actuator is composed by en electric motor and a transmission to drive a load. Transmissions are used to match the power output of the actuator to the working range of the electric motor by reducing the speed and increasing the torque output. Also they may be used to change from a rotation to a linear motion. In any case, they conversion factor should be considered.

For actuators with a gearbox and rotary output, Eq. 3.27 is used to convert the angle of the motor to the angle at the output shaft of the gearbox with a i_r gear ratio.

$$i_R = \frac{1}{i_r} \tag{3.27}$$

For actuators with linear guides, Eq. 3.28 can be used to convert the motor angle to linear position, where p is the pitch of the linear guide.

$$i_R = \frac{p}{2\pi i_r} \tag{3.28}$$

3.3 Modeling of the control

In Fig. 3.5 a representation of a general controller is shown. Depending on the system's performance requirements and complexity, different types of controllers can be implemented.

As mentioned in the state of the art, the most common approach regarding the control for parallel mechanism, is to try to use methods from the serial robotics field [41]. Two main control approaches can be followed, *joint-space* or *workspace* control. Each of them presents advantages and limitations

3.3.1 Joint-space position control

As shown in Fig. 3.12, in the mechatronic model with joint-space or linear single-axis control, the position of the manipulator $\mathbf{x}_0$ is passed through the inverse kinematics of the mechanism to obtain the joint commands $\mathbf{q}_{\boldsymbol{\rho}_0}$ for each actuator. The actual position of the joints $\mathbf{q}_{\boldsymbol{\rho}}$ is used in the direct kinematics to calculate the actual positions of the manipulator $\mathbf{x}$. The actual position is then used in the inverse dynamic model to calculate the disturbances $\mathbf{t}_{\boldsymbol{\rho}}$ to the actuators.

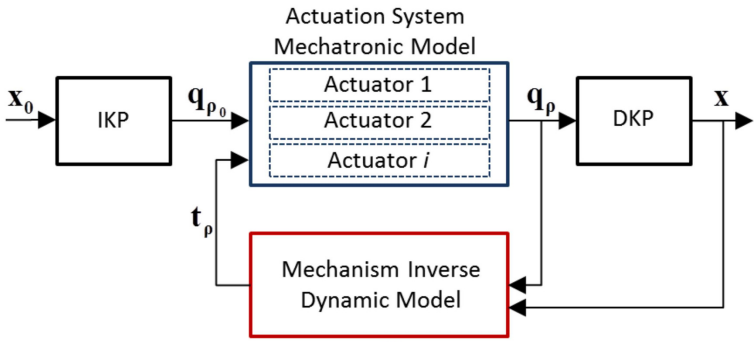


Figure 3.12. Mechatronic model with joint-space control scheme.

Regarding each actuator block in Fig. 3.12, the joint-space control shown in Fig. 3.13 is used, where the position error e_ρ is calculated from the joint command $q_{\rho_{0i}}$ and the actual joint position q_{ρ_i}. A proportional control with gain k_v is used to obtain the motor velocity command $\dot{\theta}_{i0}$ to the velocity control loop. In this loop, the velocity error $e_{\dot{\theta}}$ is calculated using the actual velocity of the motor $\dot{\theta}_i$, and then passed through a proportional-integral (PI) control with gains k_P and k_I to calculate the current command I_{i0}. In the current control loop, the current error e_I is calculated with the actual current I_i. Additionally, a PI control with gains k_{PC} and k_{IC} is used to obtain the voltage input V_{i0} to the electromechanical model of the actuator.

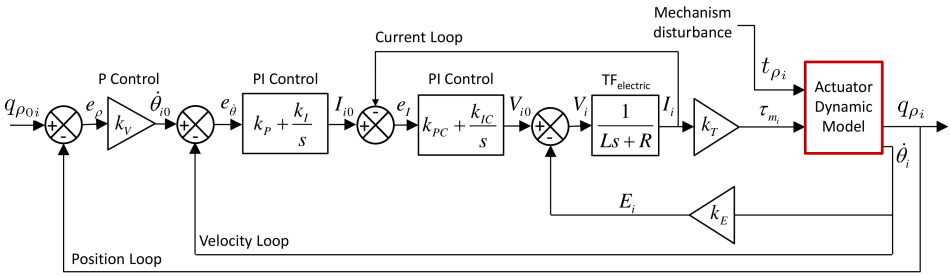


Figure 3.13. Joint-space control scheme.

3.3.2 Workspace control

As for the workspace control, it is modeled as shown in Fig. 3.14. The end-effector position error $\mathbf{e}_x$ is calculated from the position command $\mathbf{x}_0$ and the actual position $\mathbf{x}$. A proportional gain $\mathbf{k}_v$ is used to calculate the velocity command of the end-effector $\dot{\mathbf{x}}_0$, which is then passed through the inverse kinematic problem to obtain the joints velocity commands $\dot{\mathbf{q}}_{\boldsymbol{\rho_0}}$. Moreover, the actual joint velocities are integrated, and the direct kinematic problem is used to estimate the position $\mathbf{x}$, which is also used in the inverse dynamic model to calculate the disturbances to the actuators $\mathbf{t}_{\boldsymbol{\rho}}$.

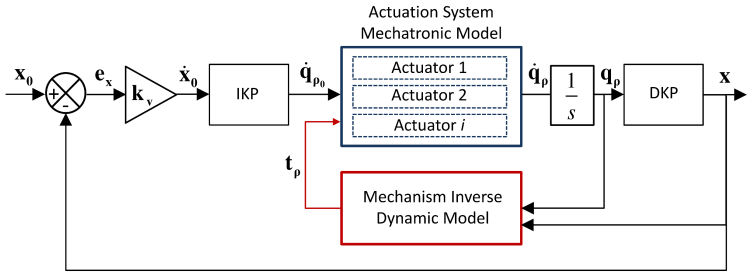


Figure 3.14. Mechatronic model with workspace control.

For the workspace control, the control scheme shown in Fig. 3.15 is included in each actuator block. In this case, the joint velocity command is

converted to the motor velocity $\dot{\theta}_{i0}$ using the factor i_R described in section 3.2.3. This factor is used again to obtain the actual joint velocity.

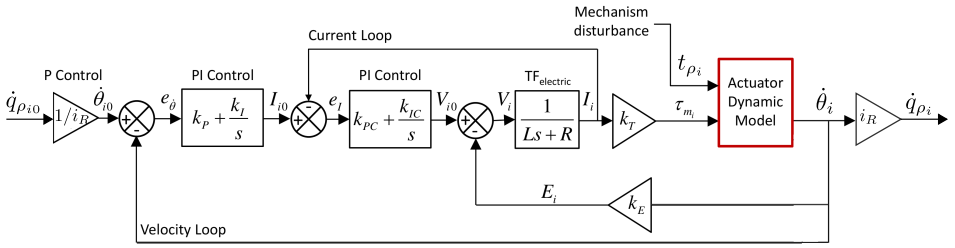


Figure 3.15. Control of the i^{th} actuator for the workspace control.

3.3.3 Computed torque control

In Fig. 3.16 a joint-space CTC is shown, where the calculated joint commands using the IKP, are then employed with the actual joints positions and velocities to calculate the position errors $\mathbf{e}_{\mathbf{q}_\rho}$ and velocity errors $\mathbf{e}_{\dot{\mathbf{q}}_\rho}$. These errors are passed through the position and the velocity gains $\mathbf{k_p}$ and $\mathbf{k_v}$ respectively, to calculate the compensation accelerations which are summed to the commanded joint accelerations $\ddot{\mathbf{q}}_{\boldsymbol{\rho}_0}$. The inverse dynamic model is then used to calculate the torque inputs $\boldsymbol{\tau}_\mathbf{m}$ for the motors. Also, the inverse dynamic model for simulation is used to calculate the disturbances $\mathbf{t}_\rho$ to the actuators.

Another approach is to use the CTC to control in the workspace as shown in Fig. 3.17. To this end, the direct kinematic problem is used to estimate the position of the end-effector.

With the computed torque control in the joint-space or in the workspace, the actuator control shown in Fig. 3.18 is used, where the torque constant k_t of the motors are employed to convert the torque command τ_{m_0} to the current command I_{i0}, which is then used as the input to the current control of the motor.

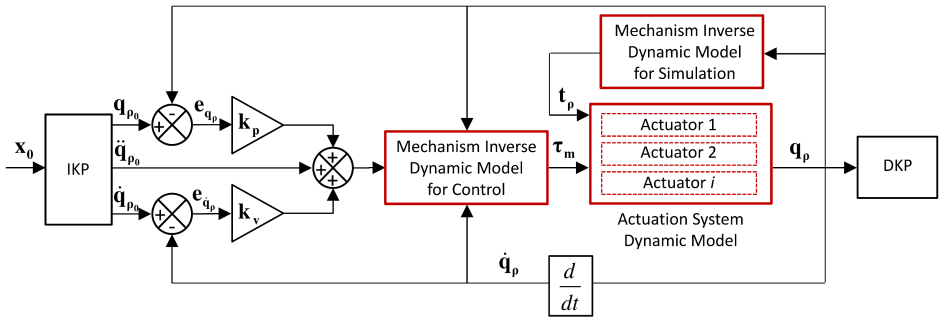


Figure 3.16. Mechatronic model with the computed torque control in the joint-space.

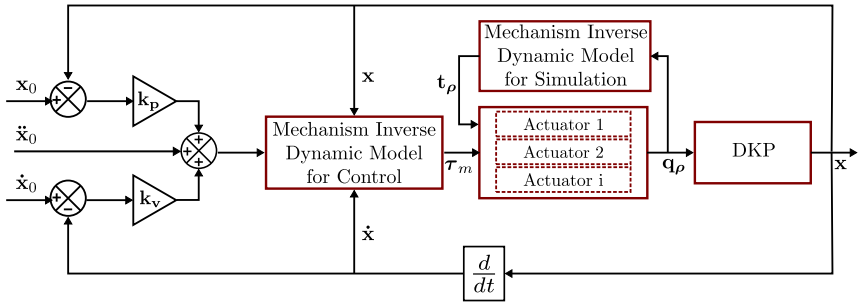


Figure 3.17. Mechatronic model with the computed torque control in the workspace.

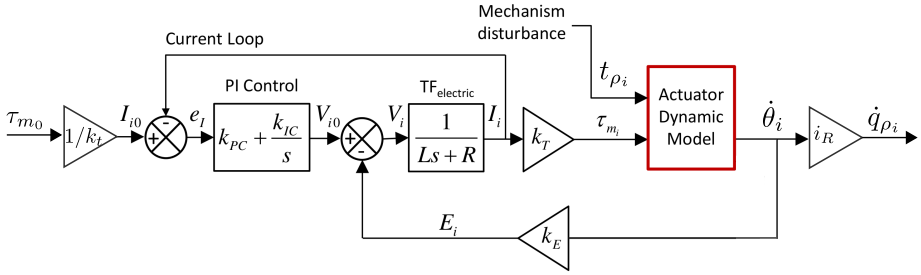


Figure 3.18. Control of the i^{th} actuator for the computed torque control.

3.4 Modeling of the dynamics of the parallel mechanism

3.4.1 Principle of energy equivalence

To obtain the dynamic model to be used for the mechatronic models with the joint-space, workspace or computed torque control, in this work it is proposed to use the principle of energy equivalence based on the work by Abdellatiff and Heimann [68]. With this approach, a given mechanism can be "virtually" split and each subsystem can be analyzed independently by using the Lagrangian formalism. For this to work, it is required that the motion conditions for each subsystem must be the same as that of the assembled system. In summary, the principle of energy equivalence states that the sum of each subsystem's energy must be equal to the energy of the complete system.

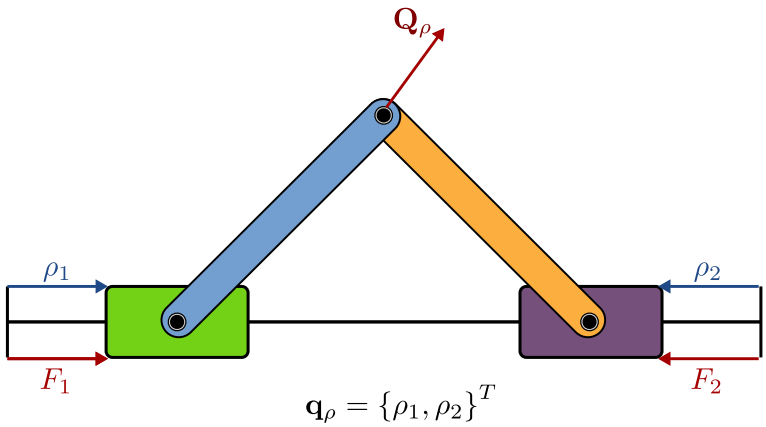


Figure 3.19. Scheme of a simple parallel mechanism.

Considering the mechanism in Fig. 3.19, differential equations in terms of the actuation geometrical kinematic variables ($\mathbf{q}_\rho$, $\dot{\mathbf{q}}_\rho$, $\ddot{\mathbf{q}}_\rho$), inertial properties and applied forces by the actuators are required for simulations and control purposes. In this way, the dynamic problem results in inertial (**I**),

gravitational ($\mathbf{G}$) and Coriolis ($\mathbf{C}$) terms that are obtained separately as in Eq. 3.29.

$$\mathbf{f}_\rho = \mathbf{I}\ddot{\mathbf{q}}_\rho + \mathbf{C}(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho) + \mathbf{G}(\mathbf{q}_\rho) \tag{3.29}$$

To obtain the dynamics of a mechanism in this way, the Euler operator $\mathcal{E}_{\mathbf{q}_\rho}$ [69] can be applied to the Lagrange's equations of the mechanism $\mathcal{L}_{mech}$ as shown in Eq. 3.30, where $\mathbf{q}_\rho$ are regarded as the generalized coordinates of the system, and $\mathbf{f}_\rho$ is the generalized forces vector which include input torques $\mathbf{t}_\rho$ in terms of the input variables $\mathbf{q}_\rho$, and $\mathbf{Q}_\rho$ represents the generalized output forces.

$$\mathcal{E}_{\mathbf{q}_\rho}\left(\mathcal{L}_{mech}\right) = \frac{d}{dt}\frac{\partial \mathcal{L}_{mech}}{\partial \dot{\mathbf{q}}_\rho} - \frac{\partial \mathcal{L}_{mech}}{\partial \mathbf{q}_\rho} = \mathbf{f}_\rho = \mathbf{t}_\rho + \mathbf{Q}_\rho \tag{3.30}$$

This equation can be used to simulate the forces or torques depending on the position, velocity and accelerations variables of the actuators (i.e. $\mathbf{q}_\rho$, $\dot{\mathbf{q}}_\rho$, $\ddot{\mathbf{q}}_\rho$). Such equation can easily be obtained for serial systems but it is a difficult task on parallel mechanisms due to the constraints imposed by the closed-chain kinematics.

To apply the principle of energy equivalence on the mechanism in Fig. 3.19, the first step is to separate the mechanism assembly into N free-bodies subsystem as shown in Fig. 3.20, four bodies in this case, $b = 1, 2, 3, 4$. Secondly, each body is positioned in space by defining its generalized coordinates $\mathbf{q}_b$, and grouped in the set $\mathbf{q}_B$ as shown in Eq. 3.31.

$$\begin{aligned} \mathbf{q}_B &= \{\mathbf{q}_1\ \mathbf{q}_2\ \mathbf{q}_3\ \mathbf{q}_4\}^T \\ &= \{\{x_1\}\ \{x_2\ y_2\ \alpha_2\}\ \{x_3\ y_3\ \alpha_3\}\ \{x_4\}\}^T \end{aligned} \tag{3.31}$$

To maintain the energy equivalence with the original system, all subsystems have to move as if they were assembled. This implies that the virtual displacement $\delta\mathbf{q}_b$ can be written as a function of the generalized coordinates of the assembled mechanism $\delta\mathbf{q}_\rho$. Thus, the virtual displacements

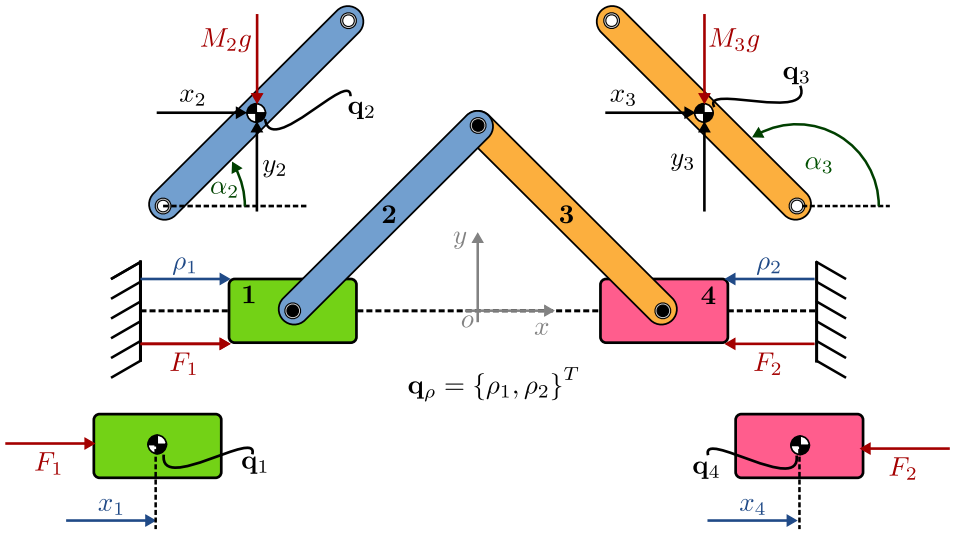


Figure 3.20. Representation of a general mechanism and its coordinates.

are related as in Eq. 3.32 by the Jacobian matrix of the assembled system $\mathbf{J}$.

$$\delta\mathbf{q}_b = \frac{\partial \mathbf{q}_b}{\partial \mathbf{q}_\rho}\ \delta\mathbf{q}_\rho = \mathbf{J}\ \delta\mathbf{q}_\rho \tag{3.32}$$

Moreover, a virtual displacement of a body within the assembly is the same virtual displacement for that body when it is separated. This is a result of the assembled motion condition imposed on it. As a consequence, the virtual displacement for a given body b, can also be related to the generalized coordinates of the assembled mechanism $\mathbf{q}_\rho$ with its Jacobian matrix $\mathbf{J}_b$ as it is shown in Eq. 3.33, where $b = 1, 2, \ldots, N$ and $N = 4$.

$$\delta\mathbf{q}_b = \mathbf{J}_b\ \delta\mathbf{q}_\rho \tag{3.33}$$

Furthermore, the virtual work for the assembled system is defined as the product of the virtual displacement and the actuation forces as shown

in Eq. 3.34. Also, the total virtual work δW_B of the separated bodies is the sum of the virtual work for each body b like shown in Eq. 3.35.

$$\delta W_\rho = \delta \mathbf{q}_\rho^T \mathbf{f}_\rho \tag{3.34}$$

$$\delta W_B = \sum_{b=1}^{N} \delta W_b = \sum_{b=1}^{N} \delta \mathbf{q}_b^T \mathbf{f}_b \tag{3.35}$$

Since the virtual displacement for each body is the same as for the assembled system and under the same forces, then it follows that the sum of the virtual work for all the bodies is the same to the virtual work of the assembled system like shown in Eq. 3.36. It is worth mentioning that, by "virtually disassembling" the system, the joint forces between the solids also produce work. However, when all the contributions are summed up, the reaction force on adjacent subsystems get canceled by the virtual work of the previous one, and thus is not necessary to take them into account.

$$\begin{aligned} \delta W_\rho &= \sum_{b=1}^{N} \delta W_b \\ \delta \mathbf{q}_\rho^T \mathbf{f}_\rho &= \sum_{b=1}^{N} \delta \mathbf{q}_b^T \mathbf{f}_b \end{aligned} \tag{3.36}$$

Finally, the expression of the input forces in $\mathbf{f}_\rho$ is obtained by substituting $\mathbf{J}_b$ from Eq. 3.33 into 3.36, resulting in

$$\mathbf{f}_\rho = \sum_{b=1}^{N} \frac{\delta \mathbf{q}_b^T}{\delta \mathbf{q}_\rho^T} \mathbf{f}_b = \sum_{b=1}^{N} \mathbf{J}_b^T \mathbf{f}_b \tag{3.37}$$

The forces $\mathbf{f}_b$ due to each body are calculated using the Lagrange's equations, and then projected into the joint-space using the Jacobian $\mathbf{J}_b$. The contribution of separate solid are then summed up as it was shown in Eq. 3.37, thus obtaining $\mathbf{f}_\rho$.

3.4.2 Modeling the dynamics of components with spatial rotation

For bodies with 2 or more spatial rotation degrees of freedom, complex expressions result when Lagrange's equations in standard form are applied. Let us consider the platform with spatial rotation in Fig. 3.21. The three angular velocity components of $\mathbf{w}$ in Eq. 3.38, and its inertia $\mathbf{I}_{mp}$ are written with respect to a reference system attached to the mobile platform $X_bY_bZ_b$. The mass of the platform is represented by m_p.

$$\mathbf{w} = [\omega_x, \omega_y, \omega_z]^T \tag{3.38}$$

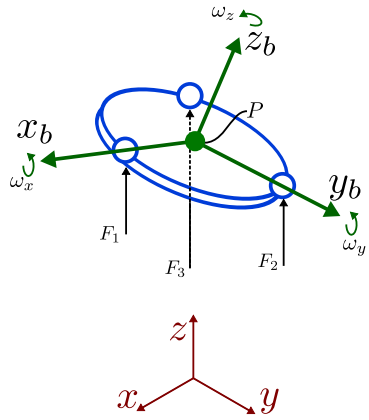


Figure 3.21. Representation of a platform in the space.

The expression of the kinetic energy with respect to a fixed coordinate system XYZ is written as follows:

$$T = \frac{1}{2} m_p \left(v_{px}^2 + v_{py}^2 + v_{pz}^2\right) + \frac{1}{2} \mathbf{w}_{p_{xyz}}^T \mathbf{I}_{p_{xyz}} \mathbf{w}_{p_{xyz}} \tag{3.39}$$

Where $\mathbf{w}_{p_{xyz}}$ is the angular velocity vector relative to the coordinate system XYZ, which can be obtained from the skew-symmetric matrix $\mathbf{S}$ in

Eq. 3.40. Also, the inertia $\mathbf{I}_{p_{xyz}}$ is written relative to the to the coordinate system XYZ, and is calculated as in Eq. 3.41 using the rotation matrix $\mathbf{R}$.

$$\mathbf{w}_{p_{xyz}} = \begin{Bmatrix} w_{p_x} \\ w_{p_y} \\ w_{p_z} \end{Bmatrix} \leftarrow \mathbf{S} = \dot{\mathbf{R}}\mathbf{R}^{-1} = \begin{bmatrix} 0 & -w_{p_z} & w_{p_y} \\ w_{p_z} & 0 & -w_{p_x} \\ -w_{p_y} & w_{p_x} & 0 \end{bmatrix} \tag{3.40}$$

$$\mathbf{I}_{p_{xyz}} = \mathbf{R}\ \mathbf{I}_{mp}\ \mathbf{R}^T \tag{3.41}$$

It can be seen that the rotation matrix $\mathbf{R}$, which relates the coordinate system $X_bY_bZ_b$ to XYZ is obtained from Eq. 3.42, where $\psi,\ \theta,\ \phi$ are the rotation angles in $x,\ y,\ z$ respectively.

$$\begin{aligned} &\mathbf{R} = \mathbf{R}_y(\theta)\mathbf{R}_x(\psi)\mathbf{R}_z(\phi) = \\ &= \begin{bmatrix} c(\theta)c(\phi) + s(\psi)s(\theta)s(\phi) & -c(\theta)s(\phi) + s(\psi)s(\theta)c(\phi) & c(\psi)s(\theta) \\ c(\psi)s(\phi) & c(\psi)s(\phi) & -s(\psi) \\ -s(\theta)c(\phi) + s(\psi)c(\theta)s(\phi) & s(\theta)s(\phi) + s(\psi)c(\theta)c(\phi) & c(\psi)c(\theta) \end{bmatrix} \end{aligned} \tag{3.42}$$

Where, $c(\alpha)$ and $s(\alpha)$ denotes the cosine and sine of the angle α respectively.

As it can be observed, derivation of matrix $\mathbf{R}$ will result in too complex expressions when the Lagrange's equations in standard form are used. A similar problem arises as a result of using the position and the Euler's angles as generalized coordinates. This issue was demonstrated for a Gough-Stewart platform by Ebert-Uphoff and Kozak [66].

As a solution, it is proposed here to decouple the translation and rotation motions of a component. On one hand, the mass m_p at the center of gravity can be considered to analyze the translation motion with Lagrange's equations in standard form. On the other hand, the inertia $\mathbf{I}_{mp}$ of the platform is considered to analyze the rotation dynamics around the

fixed point in the mass center using the angular velocities $\mathbf{w}$, which are quasi-coordinates.

In simple terms, a quasi-coordinate $\tilde{\mathbf{q}}$ is a coordinate without physical meaning. In contrast, its derivative $\frac{d}{dt}\tilde{\mathbf{q}}$ does have a physical meaning [66]. Moreover, in the context of analytical mechanics a quasi-velocity is that from which a position cannot be obtained by integrating the term. The importance of using the quasi-coordinates, is that it allows representing the rotational dynamics of a body using the inertia term $\mathbf{I}_{mp}$ relative to a coordinate system fixed to the body itself, e.g. its center of gravity [68] as it will be shown.

Moreover the Euler's angles in Eq. 3.43 (i.e. precession, nutation and spin) must be used to define the orientation of the platform if energy methods, such as Lagrange's equations are used.

$$\mathbf{q_e} = \{\theta_e, \psi_e, \phi_e\}^T \tag{3.43}$$

The relation between the quasi-velocities $\dot{\tilde{\mathbf{q}}} = \mathbf{w}$ and the Euler's angles $\mathbf{q_e}$ is shown in equations 3.44 and 3.45. What is important to note is that the quasi-velocities $\mathbf{w}$ are a function of the Euler's angles and its velocities.

$$\mathbf{w}\left(\dot{\mathbf{q}}_e, \mathbf{q}_e\right) = \mathbf{D}_\phi{}^T \dot{\mathbf{q}}_e = \begin{bmatrix} \cos\phi_e & \sin\theta_e \sin\phi_e & 0 \\ -\sin\phi_e & \sin\theta_e \cos\phi_e & 0 \\ 0 & \cos\theta_e & 1 \end{bmatrix} \begin{Bmatrix} \dot{\theta}_e \\ \dot{\psi}_e \\ \dot{\phi}_e \end{Bmatrix} \tag{3.44}$$

$$\dot{\mathbf{q}}_e = \mathbf{H}\mathbf{w};\ \mathbf{H}\mathbf{D}_\phi{}^T = \mathbf{I} \tag{3.45}$$

However, using quasi-coordinates to obtain the rotation dynamics imply that Lagrange's equations in standard form cannot be employed. To that end, Lagrange's equations for quasi-coordinates, also known as Boltzmann-Hamel equations, must be used instead. Despite the advantages of using Boltzmann-Hamel equations for spatial rotations, they are seldom used and less known than other analytical dynamics methods. In the next section,

the Boltzmann-Hamel equations are particularized for the case of spatial rotation dynamics following the procedure described by Meirovitch [96].

3.4.2.1 Particularization of the Boltzmann-Hamel equations for spatial rotation dynamics

Boltzmann-Hamel equations are obtained from the Lagrange's equations in standard form. In this regard, the dynamics of the platform can be expressed as in Eq. 3.46, where T and V are recalled to be the kinetic and potential energy respectively, $\mathbf{q}_e$ is the vector with the generalized coordinates, in this case the Euler's angles, and $\tau_{\mathbf{q_e}}$ is the vector with the generalized forces.

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}_e}\right) - \left(\frac{\partial T}{\partial \mathbf{q}_e}\right) + \left(\frac{\partial V}{\partial \mathbf{q}_e}\right) = \tau_{\mathbf{q_e}} \tag{3.46}$$

It should be mentioned that a change in notation must be introduced to make a clear distinction between when *true coordinates* or when quasi-coordinates are involved. On one hand, the kinetic energy $T(\mathbf{q_e}, \dot{\mathbf{q}}_\mathbf{e})$ is expressed with respect of the true coordinates. On the other hand, $\bar{T}(\mathbf{q}_e, \dot{\tilde{\mathbf{q}}})$ is expressed with respect to the quasi-coordinates, where $\dot{\tilde{\mathbf{q}}} = \mathbf{w}$.

Eq. 3.46 can be expanded to explicitly show the derivatives with respect to each generalized coordinate. Therefore, Eq. can be written in vector form as follows.

$$\frac{d}{dt}\left(\begin{Bmatrix} \frac{\partial T}{\partial \dot{\mathbf{q}}_{e_1}} \\ \\ \frac{\partial T}{\partial \dot{\mathbf{q}}_{e_2}} \\ \\ \frac{\partial T}{\partial \dot{\mathbf{q}}_{e_3}} \end{Bmatrix}\right) - \begin{Bmatrix} \frac{\partial T}{\partial \mathbf{q}_{e_1}} \\ \\ \frac{\partial T}{\partial \mathbf{q}_{e_2}} \\ \\ \frac{\partial T}{\partial \mathbf{q}_{e_3}} \end{Bmatrix} + \begin{Bmatrix} \frac{\partial V}{\partial \mathbf{q}_{e_1}} \\ \\ \frac{\partial V}{\partial \mathbf{q}_{e_2}} \\ \\ \frac{\partial V}{\partial \mathbf{q}_{e_3}} \end{Bmatrix} = \begin{Bmatrix} \tau_{e_1} \\ \\ \tau_{e_2} \\ \\ \tau_{e_3} \end{Bmatrix} \tag{3.47}$$

Considering only the first term, the partial derivative of the kinetic energy with respect to the j^{th} velocity of the generalized coordinate can be

written as in Eq. 3.48.

$$\frac{\partial T}{\partial \dot{\mathbf{q}}_{e_j}} = \sum_{i=1}^{3} \frac{\partial \bar{T}}{\partial \mathbf{w}_i} \frac{\partial \mathbf{w}_i}{\partial \dot{\mathbf{q}}_{e_j}} = \sum_{i=1}^{3} d_{ji} \frac{\partial \bar{T}}{\partial \mathbf{w}_i} \tag{3.48}$$

The term $\frac{\partial \mathbf{w}_i}{\partial \dot{\mathbf{q}}_{e_j}}$ corresponds to the element in the i^{th} row and j^{th} column of matrix ${\mathbf{D}_\phi}^T$. In other words the term d_{ji} of matrix $\mathbf{D}_\phi$ (the transpose of matrix ${\mathbf{D}_\phi}^T$). The above equation can then be written in vector form to account for all the generalized coordinates as follows:

$$\frac{\partial T}{\partial \dot{\mathbf{q}}_e} = \mathbf{D}_\phi \frac{\partial \bar{T}}{\partial \mathbf{w}} \tag{3.49}$$

Moreover, the time derivative can be calculated to be:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}_e}\right) = \mathbf{D}_\phi \left(\frac{d}{dt}\left(\frac{\partial \bar{T}}{\partial \mathbf{w}}\right)\right) + \dot{\mathbf{D}}_\phi \left(\frac{\partial \bar{T}}{\partial \mathbf{w}}\right) \tag{3.50}$$

Perhaps, the clearest way to express the derivative $\dot{\mathbf{D}}_\phi$ is by considering a general element d_{ji} of the matrix $\mathbf{D}_\phi$ and calculating its derivative as follows.

$$\begin{aligned}
\dot{d}_{ji}(q_{e1}, q_{e2}, q_{e3}) &= \frac{\partial d_{ji}}{\partial q_{e1}}\dot{q}_{e1} + \frac{\partial d_{ji}}{\partial q_{e2}}\dot{q}_{e2} + \frac{\partial d_{ji}}{\partial q_{e3}}\dot{q}_{e3} \\
&= \begin{bmatrix} \frac{\partial d_{ji}}{\partial q_{e1}} & \frac{\partial d_{ji}}{\partial q_{e2}} & \frac{\partial d_{ji}}{\partial q_{e3}} \end{bmatrix} \begin{Bmatrix} \dot{q}_{e1} \\ \dot{q}_{e2} \\ \dot{q}_{e3} \end{Bmatrix} \\
&= \left\{ \frac{\partial d_{ji}}{\partial \mathbf{q}_e} \right\}^T \dot{\mathbf{q}}_e = \left(\left\{ \frac{\partial d_{ji}}{\partial \mathbf{q}_e} \right\}^T \dot{\mathbf{q}}_e \right)^T \\
&= \dot{\mathbf{q}}_e^T \left\{ \frac{\partial d_{ji}}{\partial \mathbf{q}_e} \right\} = \dot{\mathbf{q}}_e^T \left(\nabla_{\mathbf{q}_e} d_{ji} \right)
\end{aligned} \tag{3.51}$$

Where the symbol $\nabla_{\mathbf{q}_e}$ denotes the gradient of the term d_{ji} with respect the generalized coordinates $\mathbf{q}_e$. In other words, the partial derivatives of the element with respect to each generalized coordinate written in vector form. Therefore, $\dot{\mathbf{D}}_\phi$ is the matrix whose elements are calculated as in Eq. 3.51, which is briefly written in Eq. 3.52.

$$\dot{\mathbf{D}}_\phi = \left[\dot{\mathbf{q}}_e^T \left\{\nabla_{\mathbf{q}_e} d\right\}\right] = \left[\mathbf{w}^T \mathbf{H}^T \left\{\nabla_{\mathbf{q}_e} d\right\}\right] \tag{3.52}$$

In this way, the first term of the Boltzmann-Hamel equations can be calculated. Nevertheless, the partial derivative of the kinetic energy with respect to the generalized coordinates $\left(\frac{\partial T}{\partial \mathbf{q}_e}\right)$ must be addressed. In this regard, for a body with its mass and inertia considered together, this term would result as in the first line in Eq. 3.53. The first term in the right hand side of the equation correspond to the elements of $\bar{T}$ that depend explicitly on $\mathbf{q}$. However, on account of considering only the rotation motion, that term does not exist. As a consequence, the partial derivative results as in Eq. 3.53.

$$\frac{\partial T}{\partial \mathbf{q}} = \frac{\partial \bar{T}}{\partial \mathbf{q}} + \frac{\partial \bar{T}}{\partial \dot{\tilde{\mathbf{q}}}} \frac{\partial \dot{\tilde{\mathbf{q}}}}{\partial \mathbf{q}} = \mathbf{0} + \frac{\partial \bar{T}}{\partial \dot{\tilde{\mathbf{q}}}} \frac{\partial \dot{\tilde{\mathbf{q}}}}{\partial \mathbf{q}} = \frac{\partial \bar{T}}{\partial \dot{\tilde{\mathbf{q}}}} \frac{\partial \dot{\tilde{\mathbf{q}}}}{\partial \mathbf{q}} \tag{3.53}$$

Moreover, to obtain the expression of the partial derivative in Eq. 3.53 easily, lets consider the partial derivative with respect to one generalized coordinate q_{e_j}. It can be seen that the derivative results as follows.

$$\frac{\partial T}{\partial q_{e_j}} = \sum_{i=1}^{3} \frac{\partial \bar{T}}{\partial w_i} \frac{\partial w_i}{\partial q_{e_j}} \tag{3.54}$$

Where,

$$\begin{aligned}\frac{\partial w_i}{\partial q_{e_j}} &= \sum_{k=1}^{3} \frac{\partial d_{ki}}{\partial q_{e_j}} \dot{q}_k = \frac{\partial d_{1i}}{\partial q_{e_j}} \dot{q}_{e_1} + \frac{\partial d_{2i}}{\partial q_{e_j}} \dot{q}_{e_2} + \frac{\partial d_{3i}}{\partial q_{e_j}} \dot{q}_{e_3} \\ &= \left[\frac{\partial d_{1i}}{\partial q_{e_j}} \; \frac{\partial d_{2i}}{\partial q_{e_j}} \; \frac{\partial d_{3i}}{\partial q_{e_j}} \right] \dot{\mathbf{q}}_e \\ &= \left\{ \frac{\partial \mathbf{D}_i}{\partial q_{e_j}} \right\}^T \dot{\mathbf{q}}_e \end{aligned} \tag{3.55}$$

It is worth clarifying that the term $\mathbf{D}_i$ in Eq. 3.55 refers to the vector of the i^{th} column of matrix $\mathbf{D}$. Hence Eq. 3.54 can be rewritten as follows.

$$\begin{aligned}\frac{\partial T}{\partial q_{e_j}} &= \sum_{i=1}^{3} \frac{\partial \bar{T}}{\partial w_i} \frac{\partial w_i}{\partial q_{e_j}} = \left\{ \frac{\partial \bar{T}}{\partial \mathbf{w}} \right\}^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_j}} \right]^T \dot{\mathbf{q}}_e \\ &= \left(\left\{ \frac{\partial \bar{T}}{\partial \mathbf{w}} \right\}^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_j}} \right]^T \dot{\mathbf{q}}_e \right)^T \\ &= \left(\left[\frac{\partial \mathbf{D}}{\partial q_{e_j}} \right]^T \dot{\mathbf{q}}_e \right)^T \left\{ \frac{\partial \bar{T}}{\partial \mathbf{w}} \right\} \\ &= \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_j}} \right] \left\{ \frac{\partial \bar{T}}{\partial \mathbf{w}} \right\} \end{aligned} \tag{3.56}$$

It should be noted that,

$$\left[\frac{\partial \mathbf{D}}{\partial q_{e_j}} \right]_{3\times 3} \neq \left\{ \frac{\partial \mathbf{D}_i}{\partial q_{e_j}} \right\}_{3\times 1} \tag{3.57}$$

Also, it is worth observing that the triple matrix product in Eq. 3.56 results in a scalar, hence it can be expanded to take all the generalized coordinates into account resulting in a vector as in Eq. 3.58.

$$\frac{\partial T}{\partial \mathbf{q}_e} = \begin{Bmatrix} \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_1}}\right] \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} \\ \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_2}}\right] \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} \\ \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_3}}\right] \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} \end{Bmatrix} = \begin{bmatrix} \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_1}}\right] \\ \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_2}}\right] \\ \dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial q_{e_3}}\right] \end{bmatrix} \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} = \left[\dot{\mathbf{q}}_e^T \left[\frac{\partial \mathbf{D}}{\partial \mathbf{q}_e}\right]\right] \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} \tag{3.58}$$

Furthermore, substituting Eq. 3.45 into Eq. 3.58 results in

$$\frac{\partial T}{\partial \mathbf{q}_e} = \left[\mathbf{w}^T \mathbf{H}^T \left[\frac{\partial \mathbf{D}}{\partial \mathbf{q}_e}\right]\right] \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} \tag{3.59}$$

Finally, it remains to substitute Eq. 3.50 and 3.59 into the Lagrange's equations in standard form of Eq. 3.46 to obtain the first version of the Boltzmann-Hamel equations. As for the potential energy, it has been neglected since only the rotation motion has been considered.

$$\mathbf{D}_\phi \left(\frac{d}{dt} \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\}\right) + \mathbf{\Omega} \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} = \tau_{\mathbf{q_e}} \tag{3.60}$$

Where

$$\mathbf{\Omega} = \dot{\mathbf{D}}_\phi - \left[\mathbf{w}^T \mathbf{H}^T \left[\frac{\partial \mathbf{D}}{\partial \mathbf{q}_e}\right]\right] = \left[\mathbf{w}^T \mathbf{H}^T \left\{\nabla_{\mathbf{q}_e} d\right\}\right] - \left[\mathbf{w}^T \mathbf{H}^T \left[\frac{\partial \mathbf{D}}{\partial \mathbf{q}_e}\right]\right] \tag{3.61}$$

Also, it is worth observing that if Eq. 3.60 is pre-multiplied by $\mathbf{H}^T$, the moments applied with respect to the coordinate system fixed to the body are obtained as follows.

$$\frac{d}{dt} \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} + \mathbf{H}^T \mathbf{\Omega} \left\{\frac{\partial \bar{T}}{\partial \mathbf{w}}\right\} = \mathbf{H}^T \tau_{\mathbf{q_e}} = \mathbf{m}_{mp} = \begin{Bmatrix} m_{mpx} \\ m_{mpy} \\ m_{mpz} \end{Bmatrix} \tag{3.62}$$

It is important to stress that matrix ${\mathbf{D}_\phi}^T$ may be rank deficient depending on the orientation of the rigid body relative to the defined coordinate system. Such a problem is commonly found with every parametrization of the orientation and is specially present in the case of rotation angles above 180 deg. Nevertheless, rank deficiency can be avoided by properly choosing an orientation of the coordinate system that ensures that the singularities are beyond the actual orientation workspace. This guarantees a full rank matrix and hence good numerical results. Another approach is to employ quaternions to express the rotation of the platform. Yet, this would result in unnecessary more complex dynamic equations for mechanisms with smaller rotation angles.

3.4.2.2 Projection of the moments onto the joint-space

Boltzmann-Hamel equations have been applied on the Gough-Stewart platform by Abdelatiff [68] and Ebert-Uphoff [66]. In this work, they have been considered only in the case of rotation motion, which result in the kinetic energy expression of Eq. 3.63. In this equation, $\mathbf{I}_{mp}$ is recalled to be the inertia matrix with respect to a coordinate system fixed to the platform.

$$\bar{T} = \frac{1}{2}\mathbf{w}^T \, \mathbf{I}_{mp} \, \mathbf{w} \tag{3.63}$$

After applying the Boltzmann-Hamel equations as described, the moments $\mathbf{m}_{mp}$ with respect to a fixed coordinate system in the platform are obtained as shown in Eq. 3.64.

$$\mathbf{m}_{mp} = \mathbf{I}_{mp}\dot{\mathbf{w}} + \mathbf{w} \times (\mathbf{I}_{mp}\mathbf{w}) \tag{3.64}$$

However, the above equation must be expressed in terms of the input variables $\mathbf{q}_\rho$ of a parallel mechanism to be used for control or simulation. Therefore, the Jacobian matrix $\mathbf{J}_R$, which relates the inputs $\mathbf{q}_\rho$ with the quasi-velocities $\mathbf{w}$, is used to *project* the rotation dynamics onto the actu-

ation system. Thus, Eq. 3.64 is rewritten as follows.

$$\mathbf{J}_R^T \mathbf{m}_{mp} = \mathbf{J}_R^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \dot{\mathbf{w}} + 2\dot{\mathbf{D}}_\phi \mathbf{I}_{mp} \mathbf{w} - \mathbf{A}^T \mathbf{I}_{mp} \mathbf{w} \right] \tag{3.65}$$

Where

$$\mathbf{A} = \left(\frac{\partial \dot{\mathbf{w}}}{\partial \dot{\mathbf{q}}_e} \right) \tag{3.66}$$

Moreover, the time derivatives of $\mathbf{w}$ and $\dot{\mathbf{q}}_e$ are defined as in equations 3.67 and 3.68. Substituting these into Eq. 3.65 when appropriate, the rotation dynamics in Eq. 3.69 is obtained with the inertial term $\bar{\mathbf{I}}_{mp}$ and the quadratic velocity term $\mathbf{c}_{mp}$ clearly separated.

$$\dot{\mathbf{w}} = \dot{\mathbf{D}}_\phi^T \dot{\mathbf{q}}_e + \mathbf{D}_\phi^T \ddot{\mathbf{q}}_e \tag{3.67}$$

$$\ddot{\mathbf{q}}_e = \mathbf{J}\ddot{\mathbf{q}}_\rho + \dot{\mathbf{J}}\dot{\mathbf{q}}_\rho \tag{3.68}$$

$$\mathbf{J}_R^T \mathbf{m}_{mp} = \mathbf{J}_R^T \mathbf{I}_{mp} \mathbf{J}_R \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} = \bar{\mathbf{I}}_{mp} \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} \tag{3.69}$$

Where,

$$\begin{aligned} \mathbf{c}_{mp} = & \Big[\mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T \right] \dot{\mathbf{J}} + \\ & \mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \dot{\mathbf{D}}_\phi^T + 2\dot{\mathbf{D}}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T - \mathbf{A}^T \mathbf{I}_{mp} \mathbf{D}_\phi{}^T \right] \mathbf{J} \Big] \dot{\mathbf{q}}_\rho \end{aligned} \tag{3.70}$$

It is important to mention that equations 3.69 and 3.70 are general and can be applied for any component with spatial rotations. In this way, only the inertia matrix $\mathbf{I}_{mp}$ and matrix $\mathbf{J}_R$ are required. As a result, the approach taken is independent of the solids of the system which yields in a systematic and less error prone method than when the dynamics have to be derived for each particular system. This means that any component can be analyzed in this way, provided that the inertia matrix and Jacobians are appropriately updated.

3.5 Mechatronic analysis of a parallel kinematics manipulator

The mechatronic analysis of parallel manipulators can be done in the time and in the frequency domain. In the time domain, the joint-space and workspace positions, velocities and accelerations can be analyzed, and by comparing the simulated and commanded positions, an estimation of the tracking error can be made. In addition, the simulated total torque can be decomposed into the required torque to move the mechanism, the required torque to move only the actuators, and the friction torque once it is identified. In this way, it can be appreciated if the selected actuators and transmission are appropriate for a given application.

The mechatronic model can also be used to analyze the performance of the control. On the one hand, the gains of the position, velocity and current controllers can be tuned using the model. On the other hand, it can also be analyzed the effect of the cycle times of each control loop on the performance of a parallel kinematic manipulator. As a result, a control can be properly selected, which is important to avoid choosing an overdimensioned control as lower cycle times increase its cost.

Regarding the frequency domain, what is important to assess is if the bandwidth of the manipulator, in other words the maximum frequency at which a controlled motion can be performed, suits the design and application requirements. Moreover, the bandwidth of an actuator can be easily found as its dynamic parameters, such as inertia, stiffness and damping, are constant. Following the theory of modern control, the transfer function relating the commanded position and the response of an actuator can be obtained, which is known as the *closed-loop transfer function*.

The same approach cannot be applied to parallel kinematics manipulator, since the inertial, Coriolis and gravitational terms are not constant in the workspace, in other words, nonlinearities are present. For this reason, to obtain the bandwidth of a manipulator from the workspace variables,

simulations in the time domain with sinusoidal motions in the workspace at different frequencies are carried out, and the time delay and amplitude of the response of the actuators are measured and compared with the commanded signals. This approach allows to obtain an estimation of the bandwidth of the manipulator at different positions in the workspace. However, this approach has two drawbacks:

- In general, a sinusoidal motion in the workspace does not correspond to a sinusoidal motion in the joint-space due to the nonlinearities of the inverse kinematics.
- The nonlinearities in the inverse kinematics also lead to more complex harmonic motions than the originally intended in the workspace.

Nevertheless, these drawbacks can be limited by using motions with reduced amplitudes. If a sinusoidal signal with a small amplitude is commanded, the effect of the nonlinearities of the kinematics and dynamics will also result small. Therefore, to obtain the bandwidth or the closed-loop transfer function based on tests at several frequencies, the amplitude of the motions should be taken into account, which depends on the manipulator design.

3.6 Summary

The mechatronic modeling procedure proposed in this thesis have been described through out this chapter. Using this procedure, the mechatronic model of a parallel kinematic mechanism is obtained, where, the actuators, the control and the mechanism are taken into account. In this way, the performance of the system can be analyzed using time and frequency domain tests.

Regarding the actuators, an electromechanical model has been used. On one hand, the electrical dynamics has been modeled with an RL circuit

to obtain the transfer function. On the other hand, the dynamics of the motor and the transmission were modeled by using 1, 2 or N degrees of freedom models.

Also, the control of a parallel kinematic mechanism has been modeled, considering joint-space, workspace and computed torque control. It was shown how the electromechanical model of the actuator is included in these control strategies.

As for the mechanism, the principle of energy equivalence has been used to separate the mechanism into several bodies, thus allowing to analyze each one separately. Furthermore, Lagrange's equations in standard form can be used for components with translation and one degrees of freedom rotation motion. For components with two or more spatial rotation degrees of freedom, it was shown that complex expressions result when Lagrange's equations in standard form are applied. To solve this, quasi-coordinates were used to describe the rotational dynamics, for which Boltzmann-Hamel equations were particularized.

Finally, the resulting mechatronic model can then be included in Matlab /Simulink® to carry simulations and evaluate the performance of a parallel mechanism. In the following chapter, a experimental validation of the mechatronic models obtained using the proposed procedure is done using two case studies.

Chapter 4

Experimental Validation of the Mechatronic Model

To validate the mechatronic modeling approach proposed in chapter 3, two case studies are here presented. The first case study is based on a planar 5R mechanism and the second is a 2-PRU-1PRS spatial Multi-Axial Simulating Table. The validation has been carried out comparing the experimental signals of position, velocity and torque obtained with the simulation results. Furthermore, key findings are further analyzed and discussed.

4.1 Case study: 5R mechanism

The case study mechanism shown in Fig. 4.1a is a 5R mechanism with two degrees of freedom and motion in the horizontal plane. It is composed by four main bars and five revolute joints, two of them used to drive the system. These active joints located in the O_A and O_B joints, are separated by a distance $H = 200\,\text{mm}$ as shown in Fig. 4.1b. Furthermore, they are driven by identical rotary servo-actuators composed by an electric motor

Schneider BSH-1401P coupled to a gearbox Neugart PLE-160 with a gear ratio of 40:1.

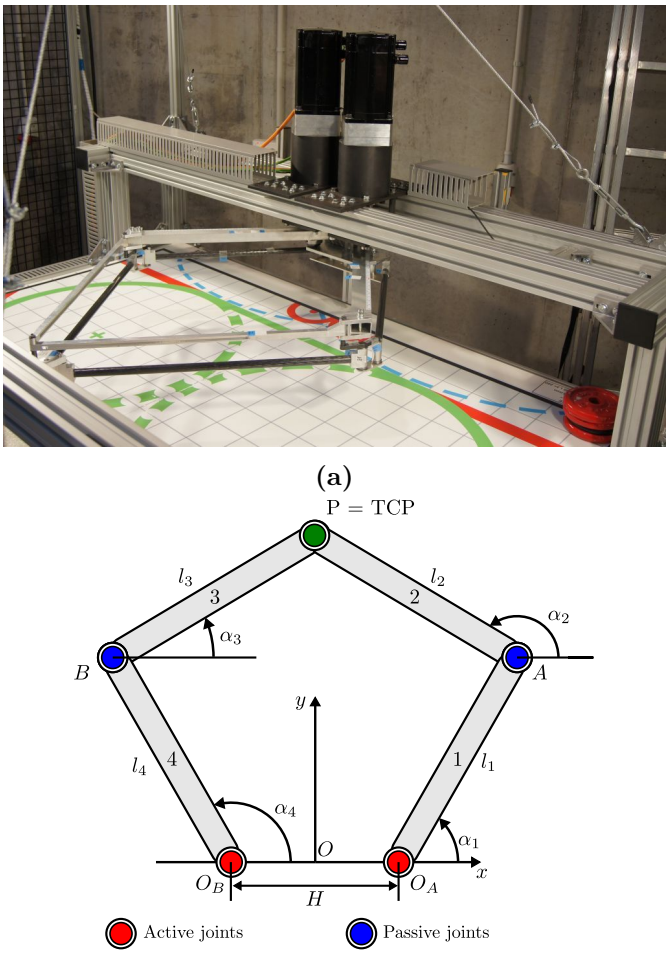


Figure 4.1. (a) Picture of the 5R mechanism and (b) schematics of the 5R mechanism.

The four main bars of the mechanism are manufactured with square aluminum profiles, with lengths l_1 and l_4 of 0.6 m and lengths l_2 and l_3 of 0.7 m. These dimensions result in the workspace shown in Fig. 4.2, where the green lines represents the permissible working borders and the red lines remarks the feasible workspace limits.

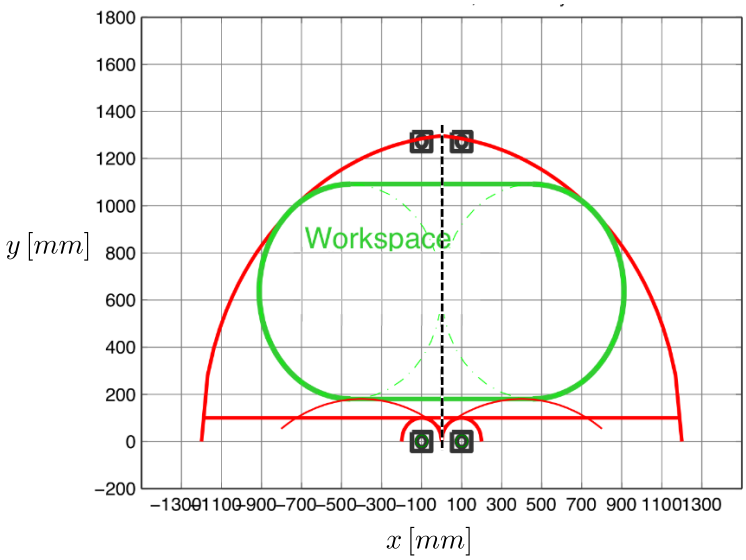


Figure 4.2. Workspace of the 5R mechanism.

Additionally, since the tool center point of the 5R mechanism is a point, little space is left for tools or sensors. In order to accommodate a platform to which elements can be added, another 5R mechanism has been fixed parallel to the main 5R mechanism to form parallelogram mechanisms at each bar. In this way, the rotation motion of the platform is constrained, thus capable only of translation motions. The bars of this mechanism have been built with carbon fiber tubes to keep a lightweight design.

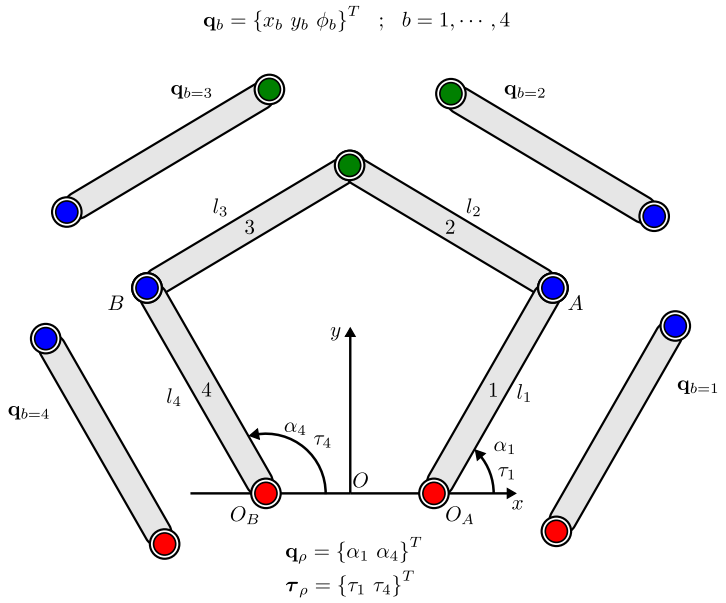


Figure 4.3. Kinematic scheme of the 5R mechanism.

4.1.1 Dynamic model of the manipulator

In Fig. 4.3 a kinematic scheme of the mechanism with its main variables is shown. The actuated joints positions and torques have been grouped into the vector $\mathbf{q}_\rho$ and $\boldsymbol{\tau}_\rho$. To avoid using the actuated joints positions as generalized coordinates, the position and orientation of a body b is defined in terms of its center of mass coordinates $\mathbf{q}_b$, where, x_b and y_b are the x and y position with respect to the reference O, and $\phi_b = \alpha_b$ its orientation.

In this way, the principle of energy equivalence can be effectively applied, thus simplifying the process to obtain the mechanism's dynamics. For this reason, the mechanism is separated into b open-chained subsystems where the Lagrange's equations for each body is defined with their

specific generalized coordinates $\mathbf{q}_b$ as follows:

$$\mathcal{L}_b = \frac{m_b}{2}\left(\dot{x}_b^2 + \dot{y}_b^2\right) + \frac{I_b}{2}\dot{\phi}_b^2 \tag{4.1}$$

The potential energy has been neglected as the mechanism is set to move in the horizontal plane. Also, by using Euler's operator [69] the dynamic equations are obtained as a function of the known Lagrangian function $\mathcal{L}_b$ and the unknown generalized forces $\boldsymbol{\tau}_b$

$$\boldsymbol{\tau}_b = \mathcal{E}\left(\mathcal{L}_b\right) = \frac{d}{dt}\frac{\partial \mathcal{L}_b}{\partial \dot{\mathbf{q}}_b} - \frac{\partial \mathcal{L}_b}{\partial \mathbf{q}_b} = \begin{bmatrix} m_b & 0 & 0 \\ 0 & m_b & 0 \\ 0 & 0 & I_b \end{bmatrix} \ddot{\mathbf{q}}_b = \mathbf{M}_b \ddot{\mathbf{q}}_b \tag{4.2}$$

Moreover, since the Jacobian matrix $\mathbf{J}_b$ relates the actuated joint velocities with the velocities of each body $\dot{\mathbf{q}}_b$ as in Eq. 4.3, it follows that Eq. 4.2 can be rewritten as in Eq. 4.4.

$$\dot{\mathbf{q}}_b = \mathbf{J}_b \dot{\mathbf{q}}_\rho \tag{4.3}$$

$$\boldsymbol{\tau}_b = \mathbf{M}_b \mathbf{J}_b \ddot{\mathbf{q}}_a + \mathbf{M}_b \dot{\mathbf{J}}_b \dot{\mathbf{q}}_\rho \tag{4.4}$$

In order to keep the energy equivalence with the assembled system, each of the b subsystems must move as if they were part of the mechanism assembly. Therefore, the virtual work of the assembled system is equal to the sum of the virtual work of each body as was shown in section 3.4.1. Consequently, the actuation torques can be written as follows:

$$\boldsymbol{\tau}_\rho = \sum_{b=1}^{4} \mathbf{J}_b^T \boldsymbol{\tau}_b \tag{4.5}$$

Where the contribution of the actuation torques due to the b body $\boldsymbol{\tau}_{\rho_b}$ can then be written as in Eq. 4.6.

$$\boldsymbol{\tau}_{\rho_b} = \mathbf{J}_b^T \boldsymbol{\tau}_b = \mathbf{J}_b^T \mathbf{M}_b \mathbf{J}_b \ddot{\mathbf{q}}_\rho + \mathbf{J}_b^T \mathbf{M}_b \dot{\mathbf{J}}_b \dot{\mathbf{q}}_\rho \tag{4.6}$$

Expression 4.6 can be reorganized to explicitly show the inertial $\mathbf{I}_b$, centrifugal $\mathbf{C}_b$ and Coriolis $\mathbf{c}_b$ terms.

$$\boldsymbol{\tau}_{\rho_b} = \mathbf{I}_b \ddot{\mathbf{q}}_\rho + \mathbf{C}_b \dot{\mathbf{q}}_\rho^2 + \mathbf{c}_b \tag{4.7}$$

To obtain the total torque, the individual torques for each subsystem are simply added resulting in Eq. 4.8, where matrices $\mathbf{I}$, $\mathbf{C}$ and $\mathbf{c}$ are function only of the position of the mechanism.

$$\boldsymbol{\tau}_\rho = \sum_{b=1}^{4} \boldsymbol{\tau}_{\rho_b} = \sum_{b=1}^{4} \left(\mathbf{I}_b \ddot{\mathbf{q}}_\rho + \mathbf{C}_b \dot{\mathbf{q}}_\rho^2 + \mathbf{c}_b \right) = \mathbf{I} \ddot{\mathbf{q}}_\rho + \mathbf{C} \dot{\mathbf{q}}_\rho^2 + \mathbf{c} \tag{4.8}$$

4.1.2 Dynamic model of the actuation system

A one degree of freedom inertial model has been used to model the actuators. It was found in preliminary tests that the gear ratio is sufficiently high, which results in a small inertia load and that the flexibilities of the transmission do not affect the motors. Therefore, the model shown in Eq. 4.9 has been used, where τ is the torque of the motor generated by the supplied current and τ_f is the load torque caused by the friction in the actuators. Moreover, J_t represents the total inertia of the actuator, where J_1 and J_2 are the inertia of the motor and the gearbox respectively.

$$\tau - \tau_f = J_t \ddot{\theta} = (J_1 + J_2)\, \ddot{\theta} \tag{4.9}$$

During preliminary tests it was found that the friction torque should be considered in the model. This is essential since other authors have proven that the friction contribution to the resulting torque is significant [70, 71]. Therefore, the non-linear friction model in Eq. 4.10 has been considered and takes into account the viscous and the Coulomb friction torques. The hyperbolic tangent dependent on the β factor has been used to make the friction model continuous regardless of the motor's direction of motion.

$$\tau_f = c\dot{\theta} + F_c \cdot tanh(\beta \dot{\theta}) \tag{4.10}$$

4.1.2.1 Experimental identification of the friction parameters

Friction parameters c, F_c and β are unknown and dependent on each system and even may change with time. Thus, a system identification procedure was applied to identify them, based on minimizing the least-square error between the experimental data and the model of the system. To this end, the pseudo-linear regression shown in Eq. 4.11 has been used, where $\hat{\boldsymbol{\tau}}$ is the estimated torque, $\boldsymbol{\rho}$ is defined as the vector of unknown parameters, $\boldsymbol{\varphi}$ is the matrix containing the experimental data measured at t samples, and $\boldsymbol{\eta}$ is the vector containing the non-linear parameters

$$\hat{\boldsymbol{\tau}}(t|\boldsymbol{\rho},\boldsymbol{\eta}) = \boldsymbol{\rho}^T \cdot \boldsymbol{\varphi}(\boldsymbol{t},\boldsymbol{\eta}) \tag{4.11}$$

Where,

$$\boldsymbol{\rho} = \begin{Bmatrix} J_t \\ c \\ F_c \end{Bmatrix} \quad \boldsymbol{\varphi}(t,\boldsymbol{\eta}) = \begin{bmatrix} \ddot{\boldsymbol{\theta}}(t) \\ \dot{\boldsymbol{\theta}}(t) \\ tanh(\beta\dot{\boldsymbol{\theta}}(t)) \end{bmatrix} \quad \boldsymbol{\eta} = \{\beta\} \tag{4.12}$$

The minimization function is written as Eq. 4.13.

$$V_N(\boldsymbol{\rho},\boldsymbol{\eta}) = \sum_{t=1}^{N} |\boldsymbol{\tau}(t_i) - \boldsymbol{\rho}^T \cdot \boldsymbol{\varphi}(t_i,\boldsymbol{\eta})|^2 = \|\boldsymbol{\tau} - \boldsymbol{\Phi}(\boldsymbol{\eta})\boldsymbol{\rho}\|^2 \tag{4.13}$$

Eq. 4.13 can be separated into a two step optimization process using the pseudo-inverse of the matrix $\boldsymbol{\Phi}$ [75, 97]. The first step is finding the optimum values of $\boldsymbol{\eta}$ using the optimization algorithm *fminunc* from Matlab®. Afterwards, the optimum values of the vector $\boldsymbol{\rho}$ are calculated as follows.

$$\hat{\boldsymbol{\rho}}(\hat{\boldsymbol{\eta}}) \equiv \boldsymbol{\Phi}^{+}(\hat{\boldsymbol{\eta}})\tau \tag{4.14}$$

On the other hand, the correct identification of the unknown parameters depends of the data used to perform the linear regression and hence, the

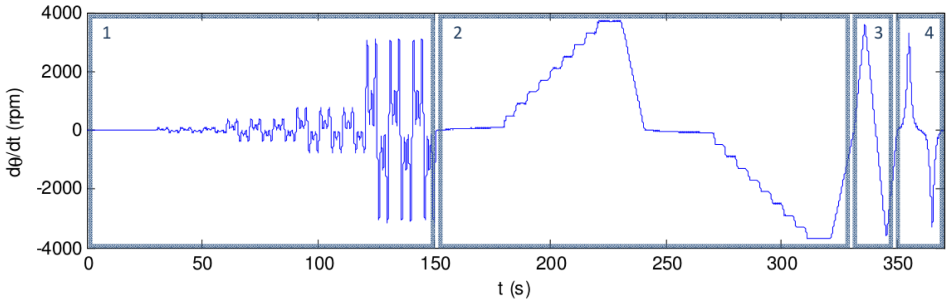


Figure 4.4. Velocity multiprofile test signal.

trajectory must be used to excite all the dynamic phenomena. For this reason, the velocity profile shown in Fig. 4.4 has been used. It consists of the following motions:

1. Sum of sinusoids with frequencies 0.1, 0.3, 0.5 Hz and increasing amplitude from 10 rpm to 3700 rpm.
2. Two stages of ascending and descending velocity steps from 10 rpm to 3700 rpm.
3. Two stages with constant acceleration and deceleration.
4. Two stages of exponentially increasing and decreasing velocity.

The profile velocity was executed on the actual motors and the experimental signals were obtained from the control system programmed in Labview with a 7 ms sampling rate. Moreover, the experimental signals were then post-processed with a Butterworth third order low-pass filter with a cutoff frequency of 3 Hz.

Finally, after applying the identification procedure, the following parameters were obtained:

Table 4.1. Identified parameters values of the 5R actuators.

Parameter	Value	Units
J_t	13×10^{-4}	$[\mathrm{kg\,m^2}]$
c	1.7×10^{-3}	[Nm s/rad]
F_c	0.478	[Nm]
β	10.78	–

4.1.3 Mechatronic model of the manipulator

The control arquitecture is a linear single-axis control in the joint-space where a cascaded PID is used to control the position, velocity and current supplied to the actuators. The control loops take place in two different hardware components. On one hand, the position loop is controlled in the NI-PXI real-time platform shown in Fig. 4.5a with a 20 ms cycle time. The output from this loop is a velocity reference that is sent via the CANopen protocol to the actuator's drivers LXM05A by Schneider Electric shown in Fig. 4.5b, where the velocity and current control is done. The cycle time are 250 µs and 62.5 µs for the velocity and current loops respectively. The position is measured by the encoder of the actuators, which implies that an indirect method is herein used to measure the position of the end-effector.

(a)

(b)

Figure 4.5. (a) Picture of the PXIe-1062 and (b) picture of the motor driver LXM05A by Schneider Electric.

The model of the system has been programmed in Matlab/Simulink® and is shown in Fig. 4.6. First, the commanded path of the end-effector is converted to the joint-space by solving the Inverse Kinematics Problem (IKP). Then, the joint-space position commands are fed to the actuators models where the control loops have been considered, see Fig. 4.7. The outputs from the control models are then fed into the direct kinematics to obtain the simulated position of the end-effector. On the other hand, the simulated joint positions are passed through the $5R$ dynamic problem where the torques required to move the mechanism are obtained. These torques are at the same time the disturbances to the actuators which must be then input to the actuators models.

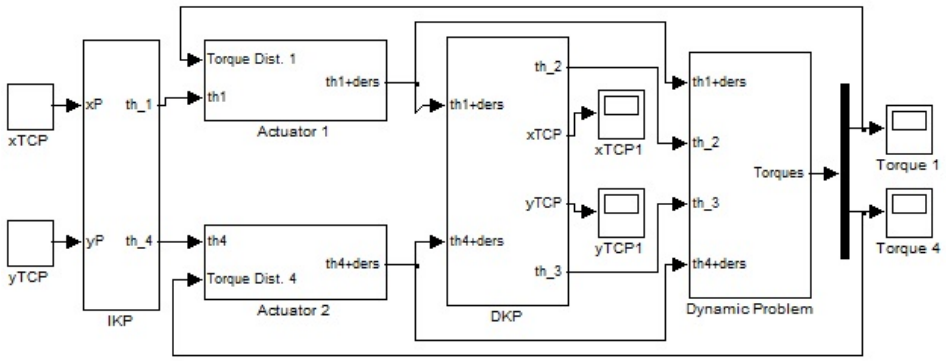


Figure 4.6. Mechatronic model of the 5R manipulator.

The actuators mechatronic model, dynamics and control are shown in Fig. 4.7 for the second motor. The outer loop is the position loop, it has a proportional controller with gain k_v with a velocity feed-forward and a low-pass filter. The output of this loop is then fed into the velocity loop which has a PI controller and a low-pass filter. Since the current loop is significantly much faster than the other loops, it is replaced by the motor's torque constant $K_t = 1.43\,\mathrm{Nm/A}$, assuming that the response of the current loop is instantaneous.

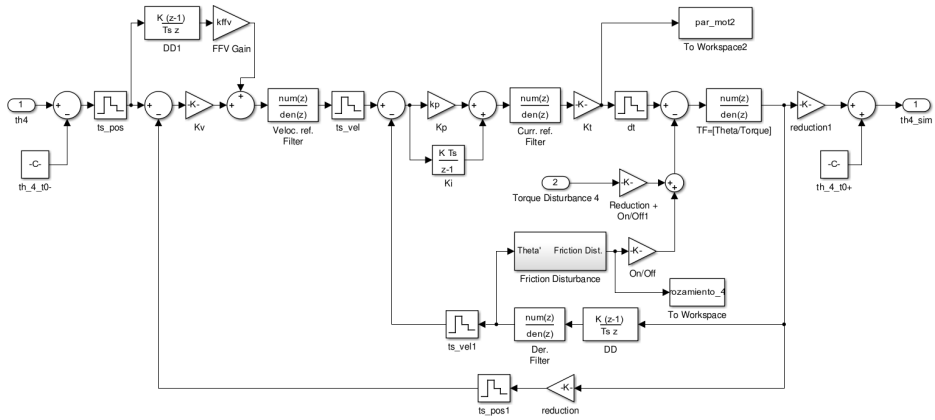


Figure 4.7. Mechatronic model of the Actuator 2 of Fig. 4.6.

Also, two sources of disturbance are considered, the torque disturbance from the mechanism *Torque disturbance 4*, and the friction torque identified for the actuator in block *Friction Disturbance*. The plant *TF* is the transfer function that represents the identified inertia of the actuator. Additionally, for the velocity controller the gains were set to $k_p = 0.14\,\mathrm{As/rad}$ and $k_i = 11.45\,\mathrm{A/rad}$ for the proportional and integral parts respectively. Finally, the different cycle times of all the loops have also taken into account, where the position cycle time ts_pos is 5 ms, and the velocity cycle time ts_vel is 250 µs. Additionally, the discretization time dt is used to model the electromechanical behavior of the motor and should be set lower than the velocity cycle time. In Matlab/Simulink® this must be defined as a multiple of the velocity cycle time. For this reason, a value of 125 µs, half the velocity cycle time, was used since a lower discretization innecessary increases the simulation time.

4.1.4 Experimental tests

Two type of trajectories have been programmed and executed to validate the mechatronic model, a displacement in X direction and two counterclockwise circular trajectory with different feed speeds. The executed trajectories

in the workspace have been estimated using the direct kinematic problem to the angular positions measured at the encoders. Also, for the position loop a low gain value of $k_v = 3.54\,1/\mathrm{s}$ has been used. In this way, apparent position errors are induced through the controller, allowing to compare the performance of the actual system with the model simulations.

Regarding the displacement in X direction, it has been programmed with a trapezoidal velocity profile. The path length is of 1.2 m, the feed speed is 1 m/s and the acceleration $1\,\mathrm{m/s^2}$. The commanded, simulated and measured TCP trajectory is shown in Fig. 4.8. It can be seen how the end effector path deviates from the programmed path up to 22 mm due to the low gain, which is adequately predicted by the simulation.

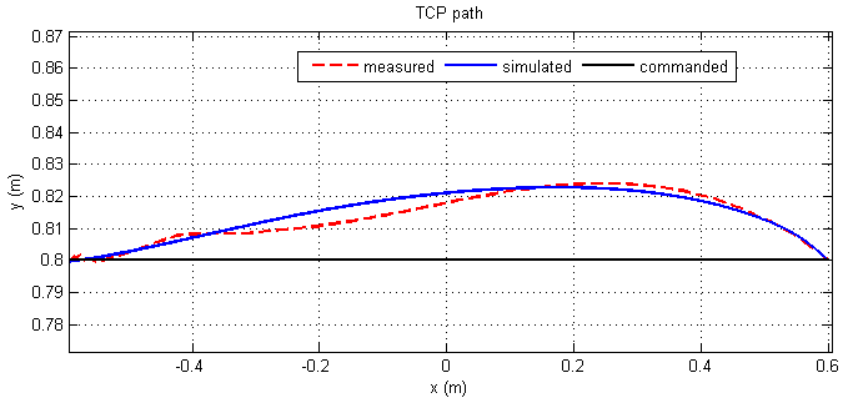


Figure 4.8. Comparison of the programmed, experimental and simulated paths in a X displacement.

The deviation of the simulated from the measured path may be due to the joints clearances, mechanism disturbances, unmodeled friction effects, and numerical errors in the direct kinematics. Also, differences between the actual controller in the motor drivers and the modeled control, such as saturators, filters and feed-forwards, may be a cause for the observed deviations.

Regarding the circular displacements, a radius of 0.2 m was programmed with a constant feed speed of 0.628 m/s for the first circle and a constant feed speed of 1.256 m/s for the second, thus resulting in half and one circle per second respectively. Also, the position loop gain was changed to the optimum value of $k_v = 19.2\,1/\mathrm{s}$.

In Fig. 4.9 and Fig. 4.10 the position and velocity in the joint-space for both circles are shown. In the case of the first circular trajectory, it is observed that the magnitudes of the model present a position deviation of about 0.25 rev and the experimental of approximately 0.5 rev from the command. Additionally, the velocities of both actuators for the model present a time lag of around 0.1 s while the time lag of the experimental signal is of 0.3 s approximately.

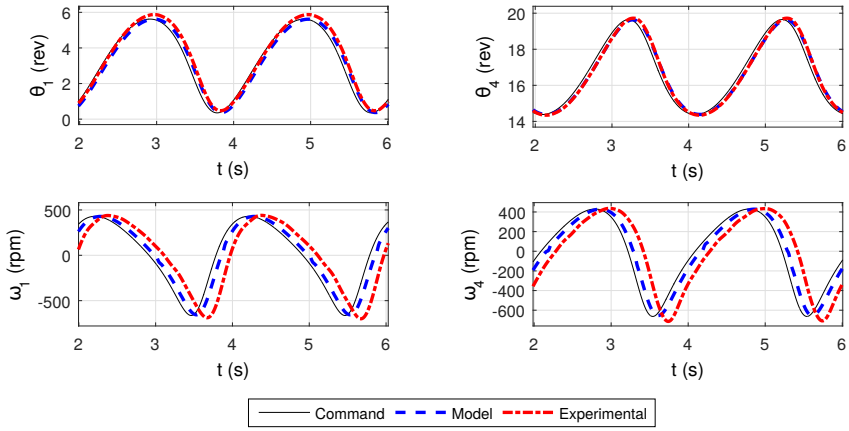


Figure 4.9. Position and velocity in the joint-space for the circular trajectory with $v_f = 0.628\,\mathrm{m/s}$.

As for the circular trajectory with $v_f = 1.256\,\mathrm{m/s}$, a similar pattern is observed. The simulated signals present less deviations than the experimental signals. Regarding the position in Fig. 4.10, deviations of approximately 0.3 rev and 0.7 rev were observed for the simulated and experimental signals

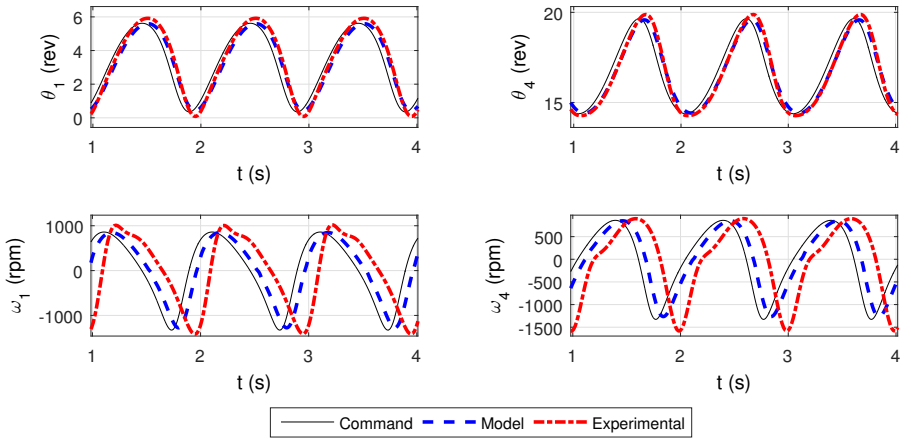


Figure 4.10. Position and velocity in the joint-space for the circular trajectory with $v_f = 1.256\,\mathrm{m/s}$.

respectively. As for the velocity, the time lags observed are of approximately 0.25 s and 0.6 s, which is about the double to the circular trajectory with $v_f = 0.628\,\mathrm{m/s}$.

Moreover, in Fig. 4.11 the joint and workspace simulated position, velocity and acceleration are compared with the measured signals from the actual system. It can be observed that the simulated position presents a 0.2 rev difference with the experimental in the joint-space, while no major errors are seen in the x direction of the workspace. Also, a time lag of about 0.5 s is observed in the velocity and acceleration signals of both, the joint-space and the workspace.

In figures 4.12a and 4.12b the two circle trajectories in the workspace are shown. It is observed that the deviations of the experimental signals are higher than the obtained with the model. In fact, up to 40 mm of radial deviations are appreciated at some points when the circle trajectory with $v_f = 1.256\,\mathrm{m/s}$ in Fig. 4.12b was executed.

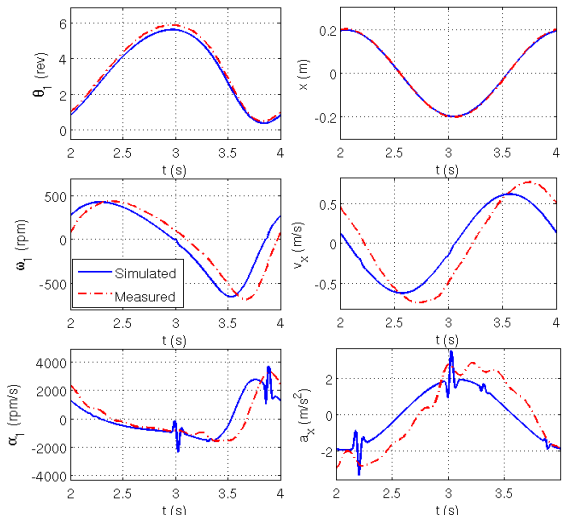
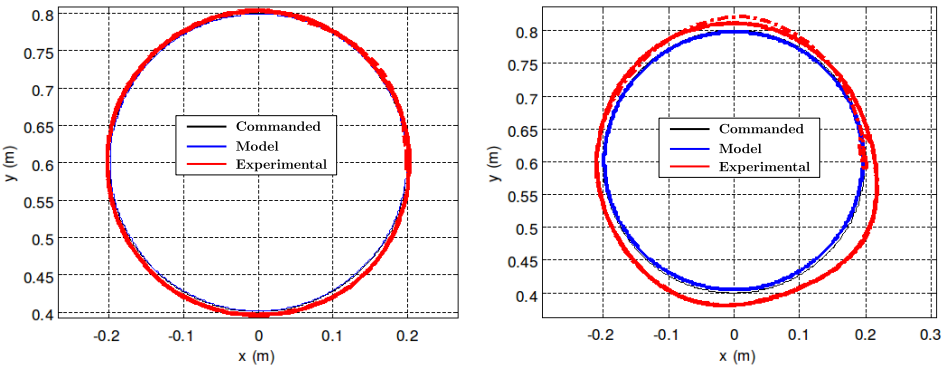


Figure 4.11. Position, Velocity and Acceleration in the joint-space and in the workspace.



(a) **(b)**

Figure 4.12. (a) Position in the workspace for the circular trajectory with $v_f = 0.628\,\mathrm{m/s}$ and (b) for $v_f = 1.256\,\mathrm{m/s}$.

In addition, the torque for both circular trajectories was obtained and shown in figures 4.13 and 4.14. It can be seen that the shape of the simulated torques is similar to the experimental ones, with maximum deviations of approximately 0.25 Nm.

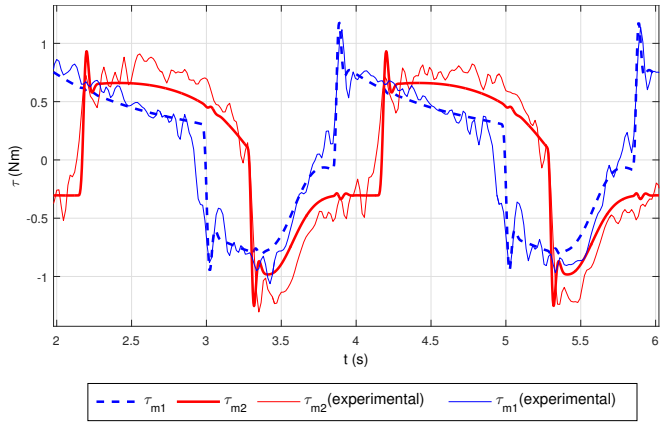


Figure 4.13. Torque signals for the circular trajectory with $v_f = 0.628\,\mathrm{m/s}$.

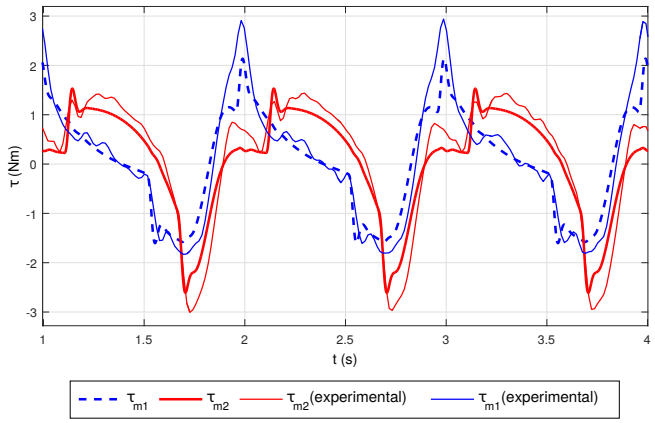


Figure 4.14. Torque signals for the circular trajectory with $v_f = 1.256\,\mathrm{m/s}$.

4.2 Case study: 2PRU-1PRS MAST mechanism

A MAST parallel mechanism was also considered as a mechatronic design case study. This mechanism is based on a 2PRU-1PRS parallel mechanism as shown in Fig. 4.15 and was developed in the framework of the thesis of Herrero [8]. In Fig. 4.15 the main geometric variables are shown. As it can be seen, the MAST mechanism is formed by 3 actuated prismatic joints. The actuated joint ρ_2 is located at a distance $H = 0.4\,\text{m}$ in the Y^+ direction from the reference coordinate system $X_0Y_0Z_0$. The remainder actuated joints, ρ_1 and ρ_3, are located symmetrically on the plane XZ separated by a distance $2H$. Moreover, a rotation joint is located on each prismatic joint allowing the bars of length $L = 0.26\,\text{m}$ and mass $0.07\,\text{kg}$ to rotate only in their respective plane. An end-platform with a mass of $2.06\,\text{kg}$ is located at the end of the bars and connected with spherical joints. In this configuration, the platform is capable of one translation motion in the Z direction and two rotations in the space, thus the mechanism presents 3 degrees of freedom.

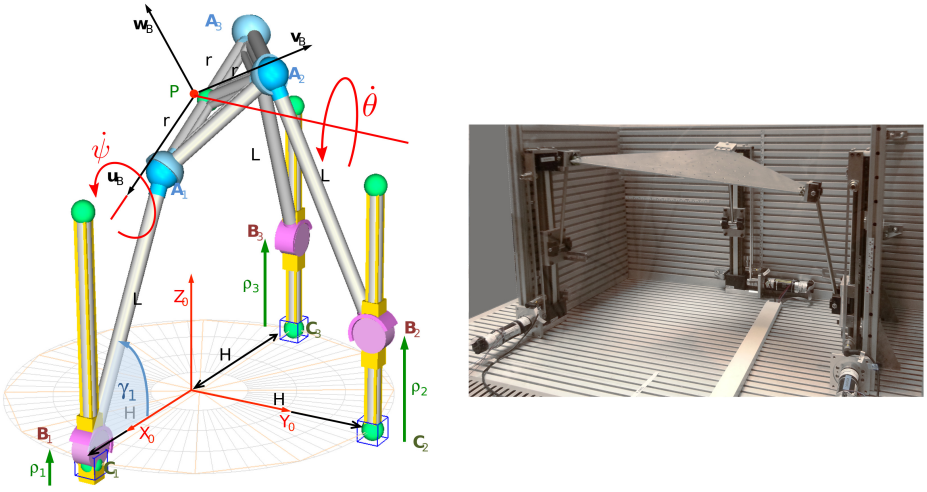


Figure 4.15. 2PRU-1PRS MAST prototype.

4.2.1 Dynamic model of the manipulator

To model the mechanism dynamics, the principle of energy equivalence described in section 3.4.1 was used. The bars, which move in a vertical plane, and the translation motion of the platform, were analyzed using Lagrange's equations in standard form. On the other hand, The Boltzmann-Hamel equations were used to study the dynamics of the spatial rotation of the platform.

4.2.1.1 Bars modeling

To model the bars, first the Lagrangian for each bar is written as in Eq. 4.15, and the Euler's operator is applied to obtain the equations of motion.

$$\mathcal{L}_b = \frac{m_b}{2}\left(\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2\right) + \frac{I_b}{2}\dot{\phi}_b^2 - m_b g z_b \tag{4.15}$$

$$\boldsymbol{\tau}_b = \begin{Bmatrix} 0 \\ F_b \\ 0 \end{Bmatrix} = \mathcal{E}\left(\mathcal{L}_b\right) = \begin{bmatrix} m_b & 0 & 0 \\ 0 & m_b & 0 \\ 0 & 0 & I_b \end{bmatrix} \ddot{\mathbf{q}}_b + \begin{Bmatrix} 0 \\ m_b g \\ 0 \end{Bmatrix} = \mathbf{M}_b \ddot{\mathbf{q}}_b + \mathbf{g}_b \tag{4.16}$$

Where,

$$\begin{aligned} \mathbf{q}_b &= \{x_b \ z_b \ \phi_b\}^T \ \text{ for bars 1 and 3} \\ \mathbf{q}_b &= \{y_b \ z_b \ \phi_b\}^T \ \text{ for bar 2} \end{aligned} \tag{4.17}$$

In Eq. 4.17, x_b, y_b and z_b are the mass center coordinates of the respective bars, and ϕ_b their angular position on the plane. By using the kinematic relationship $\dot{\mathbf{q}}_b = \mathbf{J}_b \dot{\mathbf{q}}_\rho$ the expression 4.16 can be obtained with respect of the prismatic actuated joints coordinates $\mathbf{q}_\rho$ as in Eq. 4.18, where $\mathbf{J}_b$ is the Jacobian of each bar as shown in the appendix B.2.

$$\boldsymbol{\tau}_b = \mathbf{M}_b \mathbf{J}_b \ddot{\mathbf{q}}_\rho + \mathbf{M}_b \dot{\mathbf{J}}_b \dot{\mathbf{q}}_\rho + \mathbf{g}_b \tag{4.18}$$

$$\mathbf{q}_\rho = \{\rho_1 \ \rho_2 \ \rho_3\}^T \tag{4.19}$$

The contribution of the solid b to the actuators torque $\boldsymbol{\tau}_{\rho_b}$ is found by using Eq. 3.36 which results in:

$$\boldsymbol{\tau}_{\rho_b} = \mathbf{J}_b^T \boldsymbol{\tau}_b = \left[\mathbf{J}_b^T \mathbf{M}_b \mathbf{J}_b\right] \ddot{\mathbf{q}}_\rho + \left[\mathbf{J}_b^T \mathbf{M}_b \dot{\mathbf{J}}_b \dot{\mathbf{q}}_\rho\right] + \left[\mathbf{J}_b^T \mathbf{g}_b\right] \tag{4.20}$$

Eq. 4.20 can be simplified by operating the matrices and renaming them. In this way, Eq. 4.21 results.

$$\boldsymbol{\tau}_{\rho_b} = \mathbf{I}_b \ddot{\mathbf{q}}_\rho + \mathbf{c}_b + \bar{\mathbf{g}}_b \tag{4.21}$$

Where $\mathbf{I}_b$ and $\bar{\mathbf{g}}_b$ are the inertia matrix and gravitational vector respectively, and $\mathbf{c}_b$ is the matrix containing the centrifugal and Coriolis terms. These terms depend on the position of the mechanism.

4.2.1.2 Platform modeling

Regarding the end platform, the MAST mechanism is capable of constrained translation and rotation motions. The translational dynamics has been analyzed by considering the coordinates of the mass center $\mathbf{q}_G$, shown in Eq. 4.22, as generalized coordinates, an applied external force $\mathbf{f}_G$ also at the mass center, and the kinematic relationship with the input variables through the corresponding Jacobian $\mathbf{J}_G$. In this way, equations 4.23 and 4.24 are obtained, where $\mathbf{M}_G$ is the mass matrix and $\mathbf{I}_G$, $\bar{\mathbf{g}}_G$ and $\mathbf{c}_G$ are recalled to be the matrices of inertial, gravitational and Coriolis terms.

$$\mathbf{q}_G = [x_G,\ y_G,\ z_G]^T \tag{4.22}$$

$$\mathbf{J}_G^T \mathbf{f}_G = \mathbf{J}_G^T \mathbf{M}_G \mathbf{J}_G \ddot{\mathbf{q}}_\rho + \mathbf{J}_G^T \mathbf{M}_G \dot{\mathbf{J}}_G \dot{\mathbf{q}}_\rho + \mathbf{J}_G^T \mathbf{g}_G \tag{4.23}$$

$$\mathbf{J}_G^T \mathbf{f}_G = \mathbf{I}_G \ddot{\mathbf{q}}_\rho + \mathbf{c}_G + \bar{\mathbf{g}}_G \tag{4.24}$$

On the other hand, the rotational dynamics around a fixed point is analyzed by considering an inertial body and applying Boltzmann-Hamel

equations. A mobile coordinate system is placed at the point P as shown in Fig. 4.15. Furthermore, the angular velocity $\mathbf{w}$ in the moving frame is used as *quasi-velocities* for the application of the Boltzmann-Hamel equations. Since only the rotation motion of the platform around the mass center is being analyzed, the translation and potential energies are neglected. Hence, the Lagrangian function of the moving platform results in:

$$\mathcal{L}_{mp} = \overline{\mathbf{T}} = \frac{1}{2}\mathbf{w}^T\mathbf{I}_{mp}\mathbf{w} \tag{4.25}$$

Where $\mathbf{w}$ is the angular velocity, shown in Eq. 4.26, and $\mathbf{I}_{mp}$ is the inertia tensor of the platform in the local reference system.

$$\mathbf{w} = [\omega_u, \omega_v, \omega_w]^T \tag{4.26}$$

Also, the assembled condition of the platform with the rest of the mechanism is provided by the loop-closure and constrained equations. These relate the angular velocity $\mathbf{w}$ with the input variables velocities in $\dot{\mathbf{q}}_\rho$ as follows.

$$\mathbf{w} = \mathbf{J}_R\dot{\mathbf{q}}_\rho \tag{4.27}$$

Where the expression of $\mathbf{J}_R$ is also shown in appendix B.2. Moreover, from Eq. 3.44 in section 3.4.2, the relation between the Euler's angle $\mathbf{q}_e$ and the angular velocity $\mathbf{w}$ is recalled to be as follows:

$$\mathbf{w}\left(\dot{\mathbf{q}}_e, \mathbf{q}_e\right) = {\mathbf{D}_\phi}^T\dot{\mathbf{q}}_e = \begin{bmatrix} \cos\phi_e & \sin\theta_e\sin\phi_e & 0 \\ -\sin\phi_e & \sin\theta_e\cos\phi_e & 0 \\ 0 & \cos\theta_e & 1 \end{bmatrix} \begin{Bmatrix} \dot{\theta}_e \\ \dot{\psi}_e \\ \dot{\phi}_e \end{Bmatrix} \tag{4.28}$$

By substituting Eq. 4.28 into Eq. 4.27, the relationship between the Euler angles and the inputs can be found to be:

$$\dot{\mathbf{q}}_e = \left(\mathbf{D}^T\right)^{-1}\mathbf{J}_R\dot{\mathbf{q}}_\rho = \mathbf{J}\dot{\mathbf{q}}_\rho \tag{4.29}$$

It was also shown in section 3.4.2 that, once the Boltzmann-Hamel equations are applied, the expression of the rotation dynamics results in:

$$\mathbf{J}_R^T \mathbf{m}_{mp} = \mathbf{J}_R^T \mathbf{I}_{mp} \mathbf{J}_R \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} = \bar{\mathbf{I}}_{mp} \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} \tag{4.30}$$

Where,

$$\begin{aligned} \mathbf{c}_{mp} = & \Big[\mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T \right] \dot{\mathbf{J}} + \\ & \mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \dot{\mathbf{D}}_\phi^T + 2 \dot{\mathbf{D}}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T - \mathbf{A}^T \mathbf{I}_{mp} \mathbf{D}_\phi{}^T \right] \mathbf{J} \Big] \dot{\mathbf{q}}_\rho \end{aligned} \tag{4.31}$$

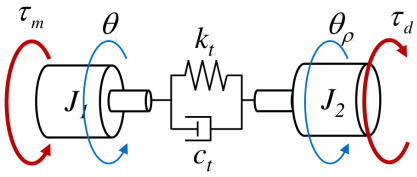
Finally, the dynamic contribution of the legs in Eq. 4.21, and the translational and rotational dynamic contributions of the platform, Eq. 4.24 and Eq. 4.30 respectively, can be added to obtain the explicit dynamic equations of the MAST shown in Eq. 4.32, where $\mathbf{t}_\rho$ are the input forces from the actuators.

$$\begin{aligned} & \mathbf{t}_\rho + \mathbf{J}_G^T \mathbf{f}_G + \mathbf{J}_R^T \mathbf{m}_{mp} = \\ & \quad \left[\mathbf{I}_{Ls} + \mathbf{I}_G + \bar{\mathbf{I}}_{mp} \right] \ddot{\mathbf{q}}_\rho + \left[\mathbf{c}_{Ls} + \mathbf{c}_G + \mathbf{c}_{mp} \right] + \left[\bar{\mathbf{g}}_{Ls} + \bar{\mathbf{g}}_G \right] \end{aligned} \tag{4.32}$$

4.2.2 Dynamic model of the actuation system

The prismatic joints of the MAST mechanism are formed by a linear belt guide Igus® ZLW-1040-02-S-100-L/R with a 300 mm stroke [33]. The DC motors model RE-40 by Maxon® are used and a GP42C gearbox with a 15:1 gear ratio [28] was selected. The linear guide is connected to the gearbox with a flexible coupling. A picture of the actuation system is shown in Fig. 4.16a.

A two degrees of freedom model, as shown in Fig. 4.16b, was used to represent the dynamics of each actuator, where J_1 and J_2 were calculated as in equations 4.33 and 4.34, considering that the belt is the most flexible



(a) **(b)**

Figure 4.16. (a) Picture of the actuator and (b) the two degrees of freedom model of the actuator.

element of the actuator. Moreover, θ is the angle of the motor and θ_ρ is the equivalent output angle to the linear displacement ρ as in Eq. 4.35.

$$J_1 = J_{motor} + J_{gearbox} + J_{coupling} \tag{4.33}$$

$$J_2 = J_{guide}\left(\frac{1}{i_r}\right)^2 + M_{load}\left(\frac{p}{2\ \pi\ i_r}\right)^2 \tag{4.34}$$

$$\theta_\rho = \rho\frac{2\pi i_r}{p} \tag{4.35}$$

This model results in four transfer functions. TF_1 relates the angular position of the motor without disturbance θ_m and the input torque τ_m, TF_2 relates the equivalent ideal angular position of the linear guide θ_{ρ_m} with the motor's position θ_m, TF_{1d} relates the disturbance torque τ_d with the angular equivalent of the position disturbance in the linear guide θ_{ρ_d}, and TF_{2d} relates θ_{ρ_d} with the position variation at the motor shaft due to the

disturbance θ_d.

$$TF_1 = \frac{\theta_m}{\tau_m} = \frac{J_2 s^2 + c_t s + k_t}{s^2 \left(J_1 J_2 s^2 + (J_1 + J_2)\, c_t s + (J_1 + J_2)\, k_t\right)} \tag{4.36}$$

$$TF_2 = \frac{\theta_{\rho_m}}{\theta_m} = \frac{c_t s + k_t}{(J_2 s^2 + c_t s + k_t)} \tag{4.37}$$

$$TF_{1d} = \frac{\theta_{\rho_d}}{\tau_d} = \frac{J_1 s^2 + c_t s + k_t}{s^2 \left(J_1 J_2 s^2 + (J_1 + J_2)\, c_t s + (J_1 + J_2)\, k_t\right)} \tag{4.38}$$

$$TF_{2d} = \frac{\theta_d}{\theta_{\rho_d}} = \frac{c_t s + k_t}{(J_1 s^2 + c_t s + k_t)} \tag{4.39}$$

The parameters values used in this case are shown in Table 4.2. Since the stiffness k_t and damping c_t are unknown, they were experimentally identified from the frequency response after a modal analysis was carried out.

Table 4.2. Parameter values of the actuator model.

Parameter	Value	Units
f_n *	986.5	[Hz]
k_t *	67.702	[Nm/rad]
c_t *	3.00×10^{-08}	[Nms/rad]
M *	0.076	[kg]
J_{motor} **	1.42×10^{-5}	$[\mathrm{kgm}^2]$
$J_{gearbox}$ **	1.4×10^{-6}	$[\mathrm{kgm}^2]$
$J_{coupling}$ **	4.68×10^{-6}	$[\mathrm{kgm}^2]$
J_{guide} **	4.524×10^{-4}	$[\mathrm{kgm}^2]$
J_{brake} **	1.00×10^{-6}	$[\mathrm{kgm}^2]$

*Experimentally identified

**From manufacturer

For the modal analysis, an accelerometer was placed at the drive carriage and an impact test was performed with a PCB-086-C03 modal hammer. The input signals from the hammer and the vibrations registered by the accelerometer were passed through the OROS-OR35 signal analyzer to obtain the parameters values. As for the rest of parameters in table 4.2, they were taken from the manufacturers datasheet [28].

4.2.3 Mechatronic model of the manipulator

A monoarticular local control have been employed for the actuation system. In such control, each actuator is independently controlled in the joint-space after the demanded pose of the mechanism is passed through the inverse kinematic problem where the actuated joint positions commands are obtained. These are then fed into the joint controller, each one with a cascaded in position, velocity and current architecture control. In Fig. 4.17 a scheme with the actual control implementation is shown.

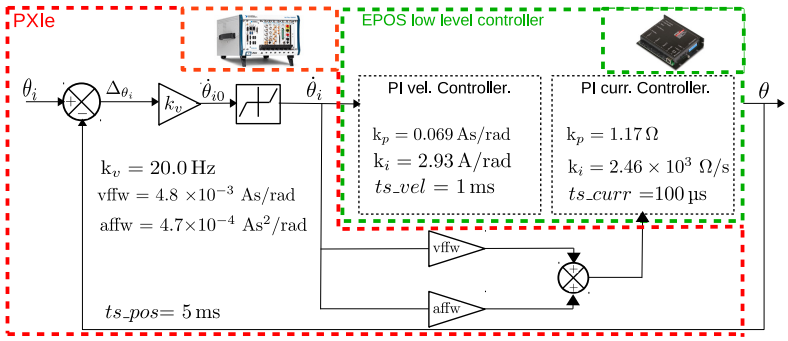


Figure 4.17. Actual controller.

Moreover, the mechatronic model of Fig. 4.18 has been programmed in Matlab/Simulink, where the inverse kinematics is used to obtain the joint commands and the inverse dynamics is used to calculate the disturbances to the actuators. Also, in Fig. 4.19 the mechatronic model of the actuators is shown, where the 2 DOF model of the actuator is included.

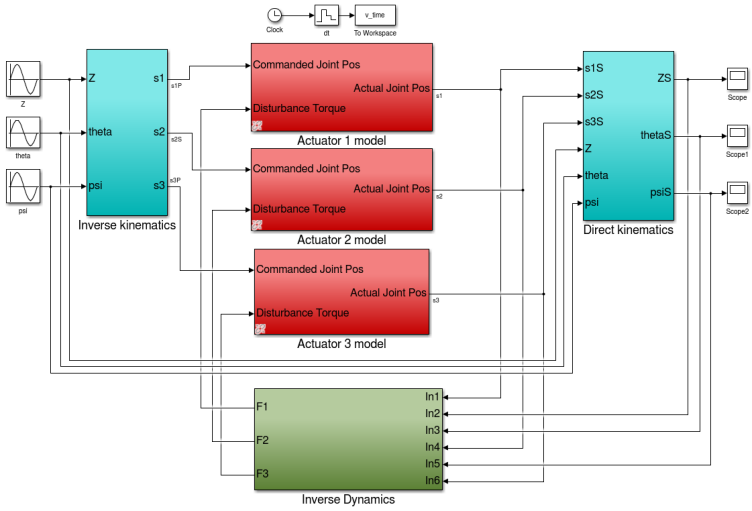


Figure 4.18. Mechatronic model of the 2PRU-1PRS MAST mechanism.

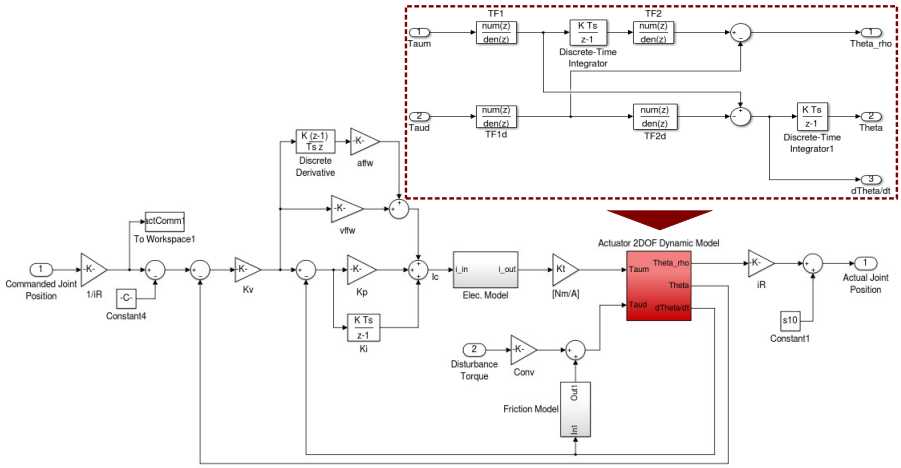


Figure 4.19. Mechatronic model of the actuator of the 2PRU-1PRS MAST mechanism.

4.2.3.1 Frequency domain testing

The magnitude and phase response of the transfer functions in TF_1 and TF_2 are show in Fig. 4.20. It was found that the resonant frequency of the TF_2 is approximately located at 900 Hz. Since the intended operating frequency is set to be in 20 Hz, the system can be consider as *stiff*.

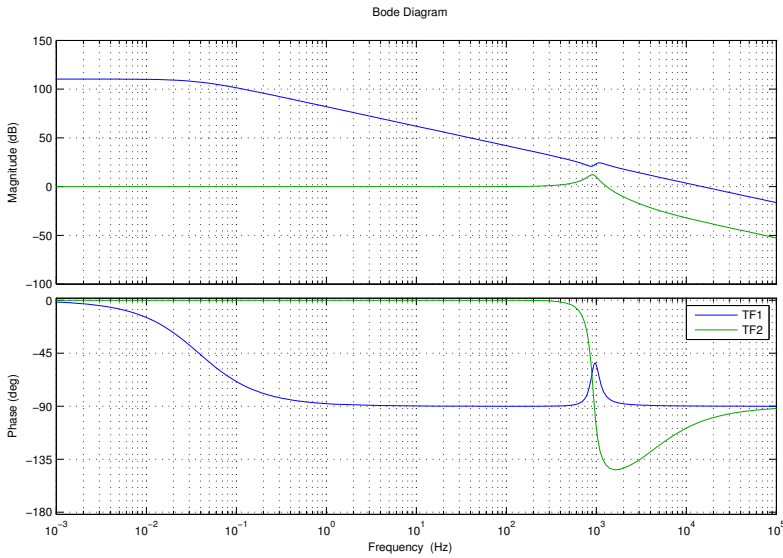


Figure 4.20. Bode plot of the two transfer functions of the actuators.

4.2.4 Experimental tests

To validate the model of the MAST mechanism, the bandwidth of the simulation model was compared with the experimental results obtained from the prototype, and the modeled and measured torque outputs were compared.

4.2.4.1 Bandwidth of the system

The bandwidth of the system has been studied to know up to which frequency the MAST can be controlled to generate excitations. To this end, pure sinusoidal motions in z, ψ and θ were simulated in the $0.5 - 20\,\text{Hz}$ frequency range using the mechatronic model. For the experimental tests, the same motions have been commanded to the control using the PXIe with frequencies set to 0.4, 1, 5, 10 and 20 Hz. A comparison between the simulation and the experimental tests is shown in figures 4.21-4.22. Moreover, to study the interaction between the actuation system and the mechanism, the bandwidth with and without the mechanism is compared in Fig. 4.23.

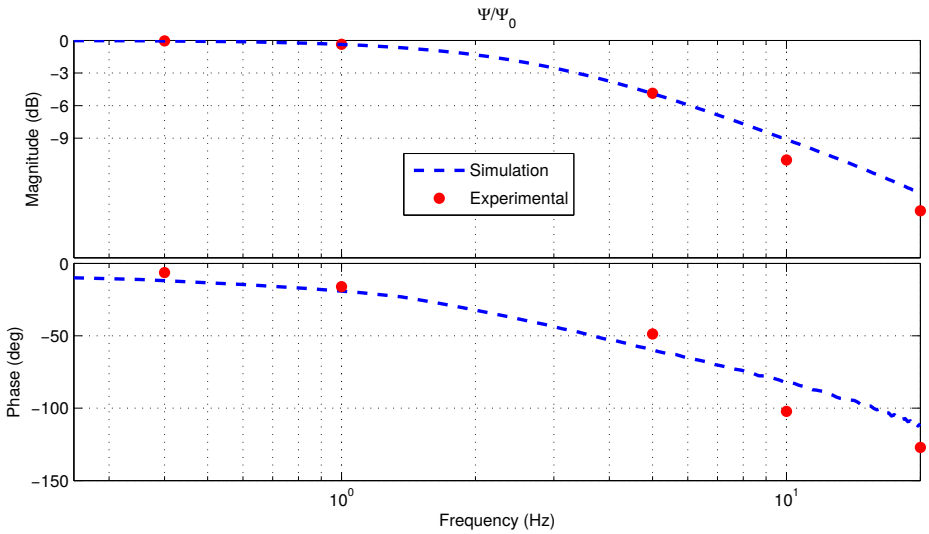


Figure 4.21. Manipulator closed position loop Bode diagram for the ψ motion.

From Fig. 4.21 and Fig. 4.22 it can be seen that the simulations are in close correspondence with the experimental results obtained for each motion. Furthermore, when the two motions are compared, no significant differences are present in the system's performance despite the fact that both motions are different. This leads to the idea that the mechanism does

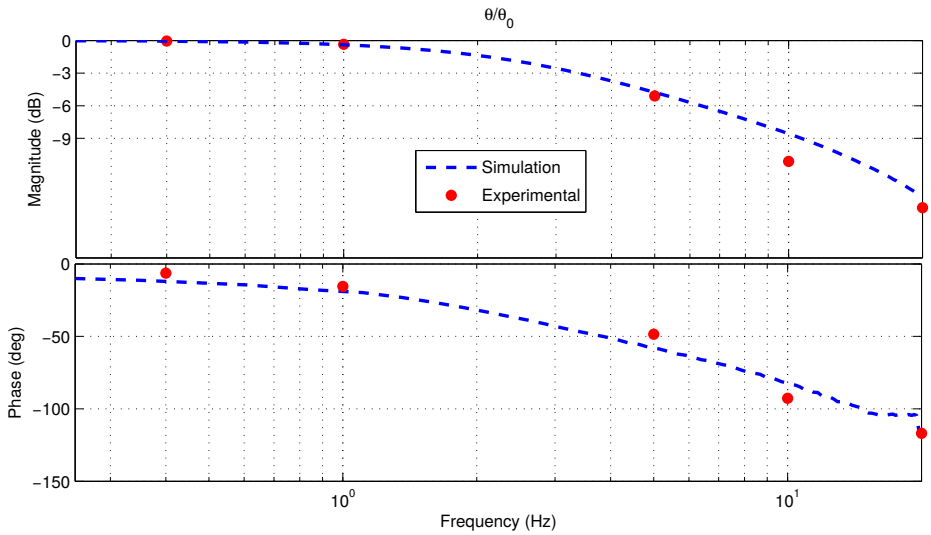


Figure 4.22. Manipulator closed position loop Bode diagram for the θ motion.

not exerts a significant influence on the system's performance. This can be further analyzed with the z motion.

In Fig. 4.23 the Bode diagram of the closed position loop for the z motion is shown. The same motion is compared with the MAST attached and with only the actuators. In this way, it can be easily analyzed the effect of the mechanism in the system's response. It is appreciated that up to 2 Hz both systems behave similarly. However at 2 Hz the magnitude of their response start diverging. What is more, it has been found that the actuators without the MAST have a bandwidth about 4 Hz, whereas with the mechanism the bandwidth is reduced to 3.5 Hz. Thus, it can be noted that the actuators are the most limiting components and that the mechanism itself has only a 0.5 Hz effect on the system's performance.

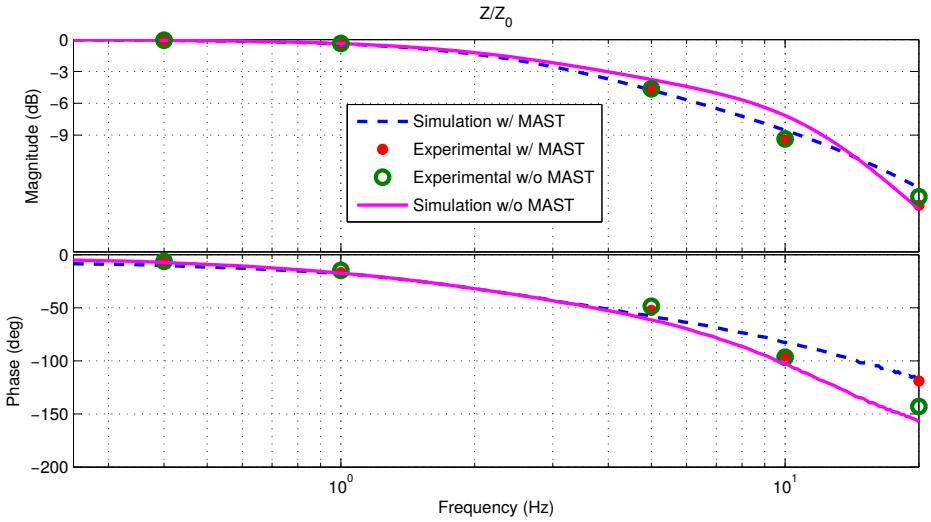


Figure 4.23. Manipulator closed position loop Bode diagram for the actuator model with and without the MAST attached.

4.2.4.2 Time domain tests

Also, in figures 4.24-4.26 the position and the torques are shown for the three pure motions executed at frequencies 0.4, 5 and 10 Hz . Two set of model parameters were used to simulate the model and compare the results with the measurements. The simulation with the parameter set *Sim. Manufac.* was constructed with the data as obtained from the manufacturers. On the other hand, it was observed that the manufacturer parameters did not provided good simulation results. For this reason, the parameters set *Sim. Ident.* was constructed by modifying the inertia J_1 to $J_1/1.56$ and the Coulomb friction to 7.18×10^{-2} Nm by an adjustment using the optimization algorithm *fminunc* from Matlab®. In this way, it is possible to compare the behavior of the model with the parameters as given by the manufacturers and with the parameters adjusted. The purpose is to understand if the manufacturers values provide a sufficiently good estimates.

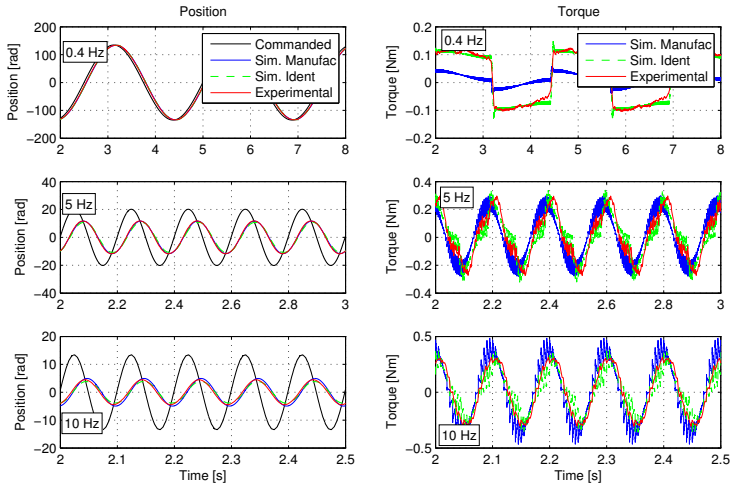


Figure 4.24. Position and torque signals comparison for the z motion.

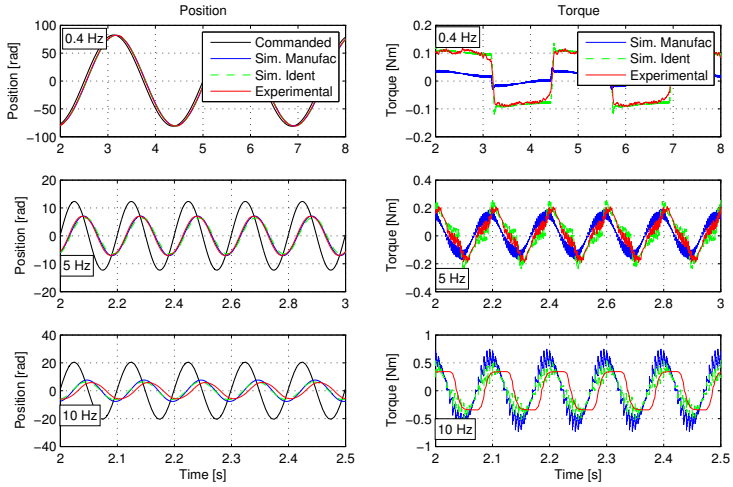


Figure 4.25. Position and torque signals comparison for the ψ motion.

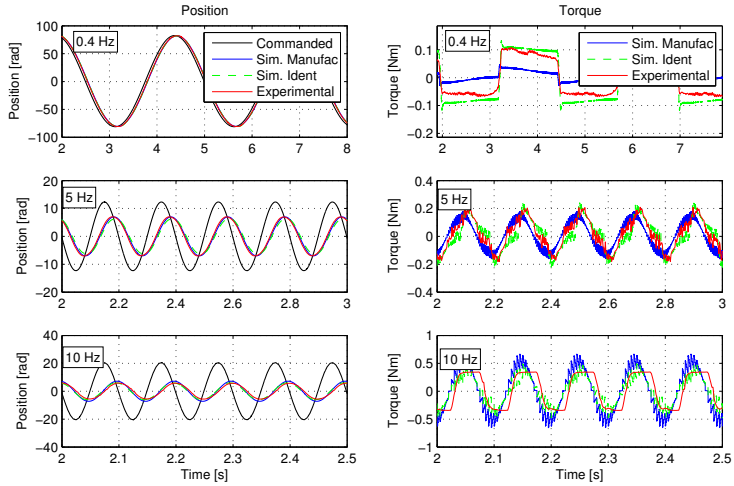


Figure 4.26. Position and torque signals comparison for the θ motion.

From the figures, the simulated positions present and maximum deviation of approximately 11% in magnitude and 13% in phase respect to the experimental signals for the frequencies shown and motions executed. Also, it can be observed that as the frequencies of the motions are increased, the magnitude of the response is reduced in accordance with the Bode diagrams shown previously. Indeed, at 0.4 Hz the magnitude of the response was found to be approximately 99.4%, whereas at 5 Hz and 10 Hz the obtained magnitudes were 65.4% and 43.2% respectively. Likewise, the phase lag between the commands and the actual signals are appropriately estimated by the model when compared with the experimental data. It can be seen that at 0.4 Hz, 5 Hz and 10 Hz, the phase lags were found to be $-6.3°$, $-62.5°$ and $-103.7°$ respectively.

Nevertheless and to validate the model, the position response must not be used alone as the controller's action may be masquerading dynamic deviations with the appearance of good positioning results. This effect can be clearly seen by comparing the position and the torque of the *Sim.*

Manufac. model at 0.4 Hz. It is observed that, despite an apparent good positioning with respect to the experimental data, the simulated torque differs from the experimental torque obtained by approximately 70%. This indicates that the dynamics are not correctly represented by this model.

Moreover, the torques obtained from the simulation with the two parameters set are different. When the data as given by the manufacturer was introduced, a difference in the torque of approximately 0.10 Nm at 0.4 Hz can be appreciated as it was pointed out before. This implies that the Coulomb friction force in the actual system is higher. Also, at a frequency of 10 Hz, the model estimated 0.1 Nm higher torque than the experimental signals. Since the acceleration forces increase with the frequencies, it follows that the actual inertia should be less than the initially estimated with the manufacturer data. For this reason, the Coulomb friction and the inertia J_1 were adjusted as commented before. In this way, the simulation provided better results than when the parameters set with the original manufacturer data was used.

On top of that, it is worth observing that even when the inputs are pure sinusoids, the resulted torques are not sinusoids as well. This effect is caused by the non-linearities of the dynamics and the friction present in the actuation system. The latter can be easily identified at lower frequencies specially at 0.4 Hz, where an almost square signal is appreciated. Finally, it can also be seen that at 10 Hz the experimental torque are *trimmed* by the action of a low-pass filter attributed to the low level controller.

4.3 Conclusions

In this chapter the mechatronic model proposed has been developed for a 5R mechanism which has a planar motion, and for a 2PRU-1PRS MAST mechanism with spatial motion. In both cases, the mechatronic model has been developed addressing the actuators and their control, as well as the mechanisms dynamics.

As for the actuators of the 5R mechanism, a one degree of freedom model was used considering the high gear ratio and inertia of the elements. On the other hand, for the MAST mechanism a two degrees of freedom model with disturbance was considered for the actuators, resulting in two direct transfer functions and two additional transfer functions for the disturbances.

Moreover, a experimental identification was carried out to obtain the unknown values of relevant dynamic parameters, such as friction. In the case of the $5R$ mechanism, a pseudo-linear regression was used to minimize the least-square error between the experimental data and the model to identify the parameters. As for the MAST, a modal analysis was carried out to obtain the parameters values.

Regarding the dynamic modeling of the mechanisms, the principle of energy equivalence, described in section 3.4, was applied. For elements with only planar motion such as in the 5R and the bars of the MAST, Lagrange's equations in standard form were used. In contrast, quasi-coordinates and Boltzmann-Hamel equations were used to obtain the dynamics of the MAST's end platform rotational motion.

In either case, the resulting mechanism dynamics and the actuator dynamic model were used to construct a mechatronic model of the 5R and the MAST, which were further compared with the actual systems. It was observed that for both mechanisms the simulations in the time domain presented a similar behavior to the experimental signals, as can be appreciated in figures 4.11-4.14 in the case of the 5R mechanism, and figures 4.24-4.26 in the case of the MAST mechanism.

Nevertheless, some deviations were still observed. In the case of the circular trajectory of the 5R in Fig. 4.9, when the joint positions were compared with the commanded signals, a deviation of 0.25 rev was obtained from the model while the actual system resulted in 0.5 rev. Also, a time lag in the velocity signal of 0.1 s and 0.3 s were obtained from the simulation and the experimental test respectively. Similarly, for the signals shown in

Fig. 4.10, deviations from the commanded signal of approximately 0.3 rev and 0.7 rev can be appreciated in the joint positions of the model and the experimental tests respectively, while a time lag of 0.25 s and 0.6 s is appreciated for the model and actual system respectively in the velocity signal. As can be noted, the actual system errors is about twice the error performed by the model. Such differences in the *following error* between both signals are attributed to unmodeled characteristics of the controller and the parallel mechanism.

In the case of the MAST mechanism, the results of the joint position simulations at different frequencies show that the mechatronic model performed similar to the actual system, where deviations on 11% in magnitude and 13% in phase lag were observed for both sets of parameters values used. The differences in position between the model and the actual system are more clearly seen at 10 Hz, where an error of 1.2 rad in position and 0.18 s in time is appreciated between them.

Also, the simulated and measured torques of the two mechanism evaluated were compared. In the case of the 5R mechanism up to approximately 0.25 Nm of error between the model and the experimental torques are appreciated. On the other hand, for the MAST mechanism, a 70% of error is observed between the model and the experimental torques when the manufacturer parameters *Sim. Manufac.* were used. This presumably indicated that a significant deviation existed between the used parameters values in the model and the physical ones. For this reason, an optimization was carried to obtain the better set of parameters values *Sim. ident.* compared to the initially used.

From the results obtained, it can be appreciated that despite the obtained deviations the developed mechatronic models were capable of simulating the behavior of the actual mechanisms. The differences observed can be attributed to three main possible sources of uncertainties: the controller, unmodeled dynamic effects and errors in the dynamic parameters values.

Regarding the controllers, their actual implementation result in relatively complex systems due to the presence of hardware and software functions such as filters and filters parameters, saturators, feed-forward, oscillators, analog-to-digital converters and similar. In this work, simplifications of such complex system has been used to model their behavior, since the actual manufacturer schematics would be required to model them in detail, which is often not available.

Also, unmodeled mechanical effects, such as clearances, and dynamic model errors may cause deviations of the performance of the mechatronic model with respect to an actual system.

Finally, it was shown that the dynamic parameters values from the manufacturer may present uncertainties that can result in significant simulation errors. This can lead to an erroneous mecatronic analysis since the required torques can be underestimated, as it was shown in MAST case study. Furthermore, the value of some of the dynamic parameters can change with time. For this reason, an identification procedure, where dynamic parameters are experimentally identified, is addressed in the following part of this work.

Part III

Identification

Chapter 5

Identification of the dynamic parameters of planar parallel mechanisms

In simulation and advanced control, having accurate dynamic parameters values is as important as having a proper model of the system. In mechatronics applications and robotics, model based design and control is employed to improve the accuracy and performance of such systems. Usually, these models rely on dynamic parameters that are often obtained or estimated from the manufacturers data sheets, from CAD models or analytical approximations. However, such estimations are often not sufficient to build detailed models, or simply not information is available for certain phenomena, such as friction. Therefore, an experimental identification procedure must be used in order to obtain accurate values of the unknown or uncertain parameters, which are then used to adjust the models.

5.1 Overview of the identification process

The identification process starts by developing an analytical dynamic model of the mechanism and actuators in such a way that a matrix $\mathbf{K}$, dependent on gravitational terms, velocities and accelerations, is multiplied by a vector with the unknown dynamic parameters $\mathbf{\Phi}$ such as inertias, masses and friction, hence resulting in the forces and torques of the system as in Eq. 5.1.

$$\mathbf{K\Phi} = \mathbf{F} \tag{5.1}$$

By using the described linear model with several measurements of an identification trajectory, it is possible to perform a linear regression to solve the system of equations and find the values of the unknown parameters in $\mathbf{\Phi}$.

Once the dynamic parameters are found, a more realistic model of the system can be built. However, a final validation step is required in order to verify that the identified model is not biased by the identification trajectory previously used. The main idea of this step is to evaluate how the identified model differs from the measured data when a trajectory, other than the one used for the identification, is executed. The completed process is depicted in the scheme of Fig. 5.1.

In this regard, a standard identification procedure consists of 6 steps according to Swevers [80], namely (1) Modeling, (2) Experiment design, (3) Data acquisition, (4) Signal processing, (5) Parameter estimation and (6) Model validation. In section 5.2, the modeling step is explained in which the equations of motions describing the dynamics of the system are obtained. To that end, a rigid body model of the mechanism and the actuators is assumed and friction is also considered. Moreover, in that section the principle of energy equivalence described in 3.4.1 was again used to obtain the dynamics of the system in a simple and systematic manner, which can be applied to other mechanisms.

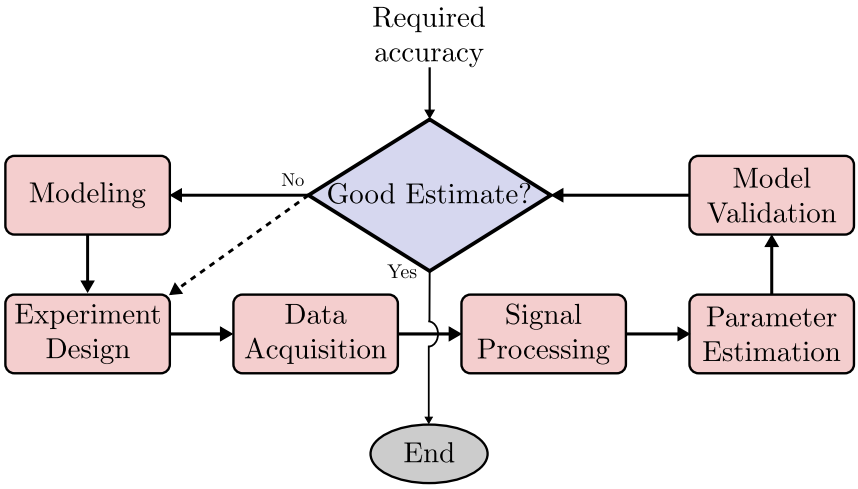


Figure 5.1. Scheme of the identification process.

The experiment design step is then described in section 5.3, where the identification trajectory are parameterized as Fourier series, and an optimization is used to obtain a suitable trajectory. Afterwards, the methods to remove distortions in the experimental signals will be addressed in the post-processing step in section 5.4. Finally, the method followed to identify the parameters will be shown in section 5.5.

5.2 Dynamic model for identification in a planar mechanism using the principle of energy equivalence

As it was stated before, the objective of the dynamic modeling step consists of expressing the equations of motions of the system as in Eq. 5.1. Herein it will be considered a rigid body model of the mechanism and the actuators with friction.

5.2.1 Rigid body parameters

It is recalled that the principle of energy equivalence considers that the sum of the energy of all the subsystems is equivalent to the energy of the whole system, if all subsystems move as if they were assembled together. This approach allows analyzing each linkage separately without taking into account the reaction forces.

The first step consists on splitting the mechanism into N subsystems like in Fig. 5.2, assuming a planar mechanism in a vertical plane. Solving the kinematic problem for each subsystem, the velocity and acceleration relations in equations 5.2 and 5.3 are obtained, where x_b and y_b are the coordinates of the mass center, and α_b is the orientation of the solid b in the plane.

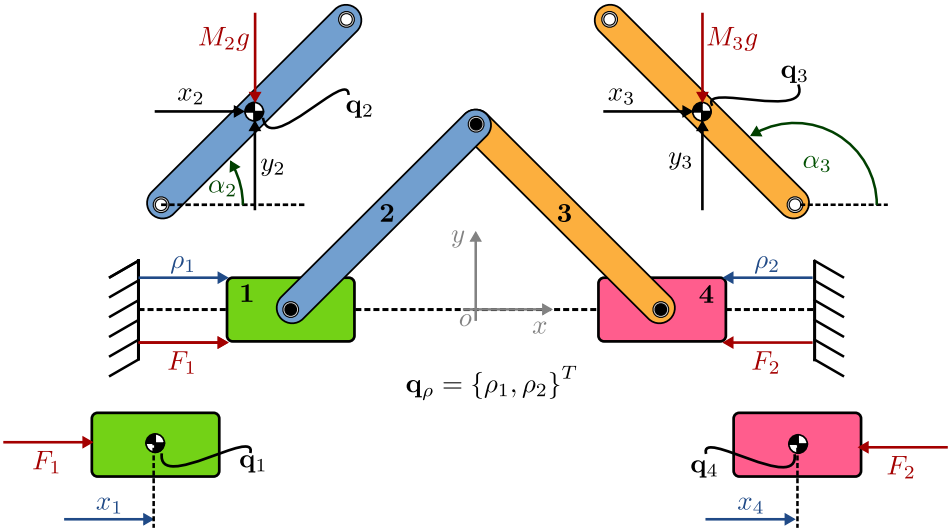


Figure 5.2. Representation of a general mechanism and its coordinates.

$$\begin{Bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{\alpha}_b \end{Bmatrix} = \mathbf{J}_b \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} \tag{5.2}$$

$$\begin{Bmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{\alpha}_b \end{Bmatrix} = \dot{\mathbf{J}}_b \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} + \mathbf{J}_b \begin{Bmatrix} \ddot{\rho}_1 \\ \vdots \\ \ddot{\rho}_j \end{Bmatrix} \tag{5.3}$$

Looking in detail equations 5.2 and 5.3, the Jacobian matrix $\mathbf{J}_b$ can be separated in a matrix ${}^t\mathbf{J}_b$, that accounts for the transformation of the linear velocities, and a matrix ${}^r\mathbf{J}_b$ that accounts for the transformation of the angular velocities. Hence, equations 5.2 and 5.3 can be rewritten as follows:

$$\begin{Bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{\alpha}_b \end{Bmatrix} = \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} \tag{5.4}$$

$$\begin{Bmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{\alpha}_b \end{Bmatrix} = \begin{bmatrix} {}^t\dot{\mathbf{J}}_b \\ {}^r\dot{\mathbf{J}}_b \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} + \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \ddot{\rho}_1 \\ \vdots \\ \ddot{\rho}_j \end{Bmatrix} \tag{5.5}$$

The equations of motion of each b subsystem can be obtained using the Lagrange's equations as in Eq. 5.6-5.8, where g is the acceleration of gravity assuming it in direction y.

$$m_b \ddot{x}_b = F_{x_b} \tag{5.6}$$

$$m_b(\ddot{y}_b + g) = F_{y_b} \tag{5.7}$$

$$I_b \ddot{\alpha}_b = \tau_b \tag{5.8}$$

For each b subsystem, the equations of motion can be written as follows:

$$\left[\begin{bmatrix} \ddot{x}_b & 0 \\ \ddot{y}_b & 0 \\ 0 & \ddot{\alpha}_b \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ g & 0 \\ 0 & 0 \end{bmatrix} \right] \begin{Bmatrix} m_b \\ \\ I_b \end{Bmatrix} = \begin{Bmatrix} F_{x_b} \\ F_{y_b} \\ \tau_b \end{Bmatrix} \tag{5.9}$$

This expression can be applied to any component of any planar mechanism in a vertical plane under the action of gravity, actuation forces or

reaction with other solids, as long as springs or dampers are not present. Furthermore, calling $\mathbf{\Omega}_b$ the acceleration matrix, $\mathbf{G}_b$ the gravitational matrix and $\mathbf{\Gamma}_b$ the vector of forces, equation 5.9 can be rewritten as in Eq. 5.10 for each component.

$$\left[\mathbf{\Omega}_b + \mathbf{G}_b\right] \begin{Bmatrix} m_b \\ I_b \end{Bmatrix} = \mathbf{\Gamma}_b \tag{5.10}$$

It is worth noting that matrix $\mathbf{\Omega}_i$ is dependent on the mass center acceleration and angular acceleration of each component b. This implies that those magnitudes have to be measured in order to perform the identification of the dynamic parameters. This is somewhat inconvenient and often unfeasible. The alternative approach is to calculate them from the actuated joints accelerations using the direct kinematics as in Eq. 5.5. In this way, the $\mathbf{\Omega}_b$ matrix in Eq. 5.10 can be expressed as follows:

$$\mathbf{\Omega}_b = \begin{bmatrix} {}^t\dot{\mathbf{J}}_b \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} + {}^t\mathbf{J}_b \begin{Bmatrix} \ddot{\rho}_1 \\ \vdots \\ \ddot{\rho}_j \end{Bmatrix} & 0 \\ 0 & {}^r\dot{\mathbf{J}}_b \begin{Bmatrix} \dot{\rho}_1 \\ \vdots \\ \dot{\rho}_j \end{Bmatrix} + {}^r\mathbf{J}_b \begin{Bmatrix} \ddot{\rho}_1 \\ \vdots \\ \ddot{\rho}_j \end{Bmatrix} \end{bmatrix} \tag{5.11}$$

Applying the principle of energy equivalence, by pre-multiplying Eq. 5.10 with the transpose of the Jacobian matrix $\mathbf{J}_b$, the torque contribution of the component b to the actuation torques in the assembled mechanism results. Thus, Eq. 5.12 is obtained.

$$\left[\mathbf{J}_b^T\mathbf{\Omega}_b + \mathbf{J}_b^T\mathbf{G}_b\right] \begin{Bmatrix} m_b \\ I_b \end{Bmatrix} = \mathbf{J}_b^T\mathbf{\Gamma}_b = \begin{Bmatrix} F_{1_b} \\ \vdots \\ F_{j_b} \end{Bmatrix} \tag{5.12}$$

It is worth remembering that $\mathbf{J}_i^T \neq {}^t\mathbf{J}_i$. Moreover, to obtain the overall dynamics of the system, it only remains to expand Eq. 5.12 to add all the subsystems of the mechanism. In this way, Eq. 5.13 represents the rigid solid dynamics for the *complete* system as a function of the dynamic parameters.

$$\begin{aligned}
&\left[\mathbf{J}_1^t\boldsymbol{\Omega}_1 + \mathbf{J}_1^t\mathbf{G}_1, \mathbf{J}_2^t\boldsymbol{\Omega}_2 + \mathbf{J}_2^t\mathbf{G}_2, \ldots, \mathbf{J}_N^t\boldsymbol{\Omega}_N + \mathbf{J}_N^t\mathbf{G}_N\right]
\begin{Bmatrix} m_1 \\ I_1 \\ m_2 \\ I_2 \\ \vdots \\ m_N \\ I_N \end{Bmatrix} = \\
&\left\{\mathbf{J}_1^t\boldsymbol{\Gamma}_1 + \mathbf{J}_2^t\boldsymbol{\Gamma}_2 + \cdots + \mathbf{J}_N^t\boldsymbol{\Gamma}_N\right\} = \sum_{b=1}^{N} \begin{Bmatrix} F_{1_b} \\ \vdots \\ F_{j_b} \end{Bmatrix} = \begin{Bmatrix} F_1 \\ \vdots \\ F_j \end{Bmatrix}
\end{aligned} \tag{5.13}$$

Calling the matrix in Eq. 5.13 $\mathbf{K}_c$, the vector with N_c unknown parameters $\boldsymbol{\Phi}_c$ and the vector of forces $\boldsymbol{\Gamma}_c$, the previous equation can be expressed as:

$$\mathbf{K}_c\boldsymbol{\Phi}_c = \boldsymbol{\Gamma}_c \tag{5.14}$$

It is worth mentioning that some of the parameters in $\boldsymbol{\Phi}_c$ may not have any significant influence on the system's dynamics. As an example, solids with only pure translation motion do not present a torque contribution due to rotation inertias and likewise, pure rotating components do not have a direct torque contribution of their mass. Hence, it is unnecessary to consider the respective dynamic parameters, the mass m_b or inertia I_b, and the linear system in equations 5.13 and 5.14 can be simplified. Moreover, in such cases its respective column in matrix $\mathbf{K}_c$ will be a zero column. Reordering the parameters in $\boldsymbol{\Phi}_c$ and once the non-significant parameters and their respective columns are eliminated, the following linear system

results.

$$\mathbf{K_{rb}}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{[j\times N_{rb}]}\mathbf{\Phi}_{rb_{[N_{rb}\times 1]}} = \mathbf{\Gamma}_{rb_{[j\times 1]}} \tag{5.15}$$

Where, N_{rb} is the number of significant rigid body dynamic parameters in $\mathbf{\Phi}_{rb}$, such that $N_{rb} \leq N_c$. Furthermore, the $\mathbf{q}_\rho$ is defined as follows:

$$\mathbf{q}_\rho = \begin{Bmatrix} \rho_1 \\ \vdots \\ \rho_j \end{Bmatrix} \tag{5.16}$$

5.2.2 Friction parameters

As it was mentioned before, Eq. 5.15 only represents the rigid body dynamics of the system. Nevertheless, friction generally plays an important role in the system dynamics as it has been proven in chapter 4. For this reason, it must be included in the identification model. Moreover, friction is a complex non-linear and random process, specially in the low-velocity range, with many factors involved [62, 70]. For this reason, several authors have studied it and different friction models have been proposed [71, 76]. Nevertheless, a simple model with Coulomb and viscous friction shown in Eq. 5.17, will be considered as it has been used before in the work of Díaz-Rodríguez et al. [51] with a good agreement, and since it can compensate for most frictional effects at high amplitude excitations [75].

$$F_{f_j} = sign(\dot{\rho}_j)\ (F_{c_j} + c_j\ \dot{\rho}_j) \tag{5.17}$$

In Eq. 5.17, F_{c_j} represents the Coulomb friction, c_j is the viscous friction coefficient and F_{f_j} is the friction force on the j^{th} actuator. In this way, it is assumed that all the friction occurs at the actuators. Although this assumption is not completely accurate, is from the actuators torque that the friction influence can be measured for the mechanism.

Also, by considering j actuators, Eq. 5.17 can be written in linear form to include the friction as follows:

$$\left[sign\left(\dot{\mathbf{q}}_\rho^T\ \mathbf{I}_{j\times j}\right),\ \dot{\mathbf{q}}_\rho^T\ \mathbf{I}_{j\times j}\right]\begin{Bmatrix}\mathbf{F_c}\\ \mathbf{c}\end{Bmatrix} = \mathbf{K}_f\ \mathbf{\Phi}_f = \mathbf{\Gamma}_f \tag{5.18}$$

Where, $\mathbf{I}$ is an identity matrix, $\mathbf{q}_\rho$ is the position vector, $\mathbf{F_c}$ and $\mathbf{c}$ are the vectors with the Coulomb and viscous friction parameters respectively, and $\mathbf{\Gamma}_f$ is the vector with the friction and inertial forces for all the actuators as shown in Eq. 5.19.

$$\mathbf{F_c} = \begin{Bmatrix}F_{c_1}\\ \vdots\\ F_{c_j}\end{Bmatrix};\ \mathbf{c} = \begin{Bmatrix}c_1\\ \vdots\\ c_j\end{Bmatrix};\ \mathbf{\Gamma}_f = \begin{Bmatrix}F_{f_1}\\ \vdots\\ F_{f_j}\end{Bmatrix} \tag{5.19}$$

Furthermore, the number of unknown friction parameters in $\mathbf{\Phi}_f$ to identify can be defined as $N_f = 2j$. To include the friction effects in the identification model, the matrix $\mathbf{K}_f$ and the unknown friction parameters vector $\mathbf{\Phi}_f$ should be assembled to the rigid body model as will be shown.

5.2.3 Assembly of the rigid body and friction models

To obtain an identification model including the rigid body and friction models, matrix $\mathbf{K}_f$ and and the unknown friction parameters vector $\mathbf{\Phi}_f$ are simply appended to matrix $\mathbf{K}_{rb}$ and $\mathbf{\Phi}_{rb}$ respectively. In this way, the identification model results as in Eq. 5.20, where the subindex rf indicates that the rigid body and friction parameters are considered.

$$\begin{aligned}&\left[\mathbf{K}_{rb}\mathbf{K}_f\right]\begin{Bmatrix}\mathbf{\Phi}_{rb}\\ \mathbf{\Phi}_f\end{Bmatrix} =\\ &\mathbf{K}(\mathbf{q}_\rho,\dot{\mathbf{q}}_\rho,\ddot{\mathbf{q}}_\rho)_{j\times N_\Phi}\mathbf{\Phi}_{N_\Phi\times 1} = \begin{Bmatrix}F_{rb_1}+F_{f_1}\\ \vdots\\ F_{rb_j}+F_{f_j}\end{Bmatrix} = \begin{Bmatrix}F_{rf_1}\\ \vdots\\ F_{rf_j}\end{Bmatrix} = \mathbf{\Gamma}_{rf}\end{aligned} \tag{5.20}$$

Where,

$$N_\Phi = N_{rb} + N_f \tag{5.21}$$

It is worth observing that Eq. 5.20 usually represents an undetermined linear system because $j < N_\Phi$. However, bearing in mind that matrix $\mathbf{K}$ contains the kinematic data and $\mathbf{\Gamma}_{rf}$ contains the force of a time instant for a given motion trajectory, if more samples n_p are used instead, an overdetermined system results which must be used to perform a linear regression. In this way, Eq. 5.20 can be rewritten to account for n_p points of an identification trajectory as follows:

$$\mathbf{W}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{(j \cdot n_p) \times N_\Phi} \mathbf{\Phi}_{N_\Phi \times 1} = \mathbf{\Gamma}_{(j \cdot n_p) \times 1} \tag{5.22}$$

5.2.4 Procedure to obtain the *base parameters* of a rank deficient linear system dynamic model

The model shown in Eq. 5.22 presents all the significant dynamic parameters of the system. However, due to the coupled kinematics loops in parallel mechanisms, some of the dynamic parameters can not be identified individually. The reason is that usually matrix $\mathbf{W}$ is not full rank. In such cases, the unknown parameters can only be identified as N_{base} linear combinations, such that $N_{base} < N_\Phi$ and N_{base} is the rank of matrix $\mathbf{W}$. As a result, the linear system in Eq. 5.23 results.

$$\mathbf{W}_{base}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right) \mathbf{\Phi}_{base} = \mathbf{\Gamma} \tag{5.23}$$

Where the elements in vector $\mathbf{\Phi}_{base}$ of size $N_{base} \times 1$ are known as the *base parameters*. Furthermore, $\mathbf{W}_{base}$ would result in a matrix of size $(n_p\ j) \times N_{base}$.

The procedure to calculate those linear combinations is the following. When matrix $\mathbf{W}$ in Eq. 5.22 is rank deficient, its number of non-null singular values is less than the number of its columns, which indicates

linear dependencies between them. In this case, the number of dependencies between columns of $\mathbf{W}$ are simply calculated as:

$$N_{lindep} = N_{\Phi} - N_{base} \tag{5.24}$$

Because of the existing linear dependencies, a N_{Lindep} number of columns of matrix $\mathbf{W}$ must be eliminated and the same number of unknown parameters in $\mathbf{\Phi}$ have to be combined with the rest. As a consequence, when $\mathbf{W}$ is rank deficient the dynamic parameters cannot be identified separately. A solution to this issue is proposed in this thesis and will be appropriately addressed in section 5.5.

Following the work of Díaz-Rodríguez et al. [51] and Gautier [85], a procedure to obtain the base parameters based on the *singular value decomposition* method (SVD) is herein explained and applied. It is worth mentioning that the Matlab® command *svd* has been used in this work to perform the singular value decomposition.

Defining the SVD as a matrix operator, when it is applied to matrix $\mathbf{W}$ of size $m \times n$, it is decomposed into three matrices as in Eq. 5.26. Matrix $\mathbf{U}$ is a unitary matrix of size $m \times m$, while matrix $\mathbf{V}$ is also an unitary matrix but of size $n \times n$. An square unitary matrix $\mathbf{A}$ is remembered to be defined as:

$$\mathbf{A}\ \mathbf{A}^H = \mathbf{A}^H\ \mathbf{A} = \mathbf{I} \tag{5.25}$$

Where, matrix $\mathbf{A}^H$ is the conjugate transpose of $\mathbf{A}$ and $\mathbf{I}$ is the identity matrix.

Additionally, $\mathbf{S}$ is a diagonal matrix where the elements of its diagonal are non-negative and known as the singular values σ_k of matrix $\mathbf{W}$. When matrices $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{S}$ are multiplied as shown in Eq. 5.27, matrix $\mathbf{W}$ is obtained back.

$$[\mathbf{U}, \mathbf{S}, \mathbf{V}] = svd(\mathbf{W}) \tag{5.26}$$

$$\mathbf{W} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V} \tag{5.27}$$

It is important to mention that if $\mathbf{W}$ is full rank, then all $\sigma_k > 0$ for $1 \leq k \leq n$, otherwise, null singular values are present indicating rank deficiency and thus linear combination between columns of $\mathbf{W}$. Furthermore, by multiplying Eq. 5.27 by $\mathbf{V}^H$, Eq. 5.28 results, where matrix $\mathbf{V}$ is separated into matrix $\mathbf{V_1}$ and $\mathbf{V_2}$. What is important in this manipulation is that the columns in $\mathbf{V}_2$ define the linear relations of the columns of $\mathbf{W}$. However, in cases when matrix $\mathbf{W}$ is full rank, then $\mathbf{V}_2$ is empty and $\mathbf{V} = \mathbf{V}_1$.

$$\mathbf{W}\left[\mathbf{V}_{1_{N_{rb}\times N_{base}}}\mathbf{V}_{2_{N_{rb}\times N_{lindep}}}\right] = \mathbf{U}\cdot\mathbf{S} \tag{5.28}$$

The objective is to find a solution of the form of Eq. 5.29 in such a way that $\mathbf{V}_{22}$ is a full rank matrix. When $\mathbf{V}_{22}$ is found the permutation matrix $\mathbf{P}$ results. This process is iterative and starts with the last row of $\mathbf{V}_2$ seeking to construct a matrix $\mathbf{V}_{22}$. Then, the rank of the matrix $\mathbf{V}_{22}$ is evaluated. When the first regular matrix $\mathbf{V}_{22}$ is found, the permutation matrix $\mathbf{P}$ results and the indexes of the columns of $\mathbf{W}$ to be eliminated are also obtained.

$$\mathbf{P}^T\mathbf{V}_2 = \begin{bmatrix} \mathbf{V}_{21_{N_{base}\times N_{lindep}}} \\ \mathbf{V}_{22_{N_{lindep}\times N_{lindep}}} \end{bmatrix} \tag{5.29}$$

Finally, Eq. 5.30 is solved to find the base parameters. In it, $\mathbf{I}$ is a square identity matrix of size $N_{base} \times N_{base}$ to which matrix $\mathbf{B}$ in Eq. 5.31 is appended column-wise, thus resulting in matrix $\mathbf{IB}$.

$$\mathbf{\Phi}_{base} = (\mathbf{IB}\cdot\mathbf{P}^T)\mathbf{\Phi} \tag{5.30}$$

$$\mathbf{B}_{N_{base}\times N_{lindep}} = -\mathbf{V}_{21}\mathbf{V}_{22}^{-1} \tag{5.31}$$

It can be appreciated that identifying the parameters depends on the trajectory used for the identification. Therefore, an optimum trajectory has to be generated in order to accurately identify the dynamic parameters.

5.3 Trajectory optimization for dynamic parameters identification

The correct identification of the base parameters in $\mathbf{\Phi}_{base}$ using Eq. 5.23 depends on the trajectory used. In this regard, matrix $\mathbf{W}_{base}$ must have a sufficiently low condition number. The latter, is a measure of the worst case lost of precision of the estimates [98]. In other words, a large condition number would result in inaccurate parameters estimates.

As proposed by Swevers et al. [84], Fourier series were chosen to parameterize the identification trajectory in position, velocity and acceleration. In this way, expressions of the position, velocity and acceleration can be written explicitly. Also, the bandwidth can be limited and a periodic signal can be used, which is interesting to perform several cycles [99].

The following equations represent the position, velocity and acceleration of the joints trajectory in Fourier series form. There, n_h represent the number of harmonics of the trajectory, C_{ji} and D_{ji} the amplitude of the harmonics of the velocity, ρ_{a_0} the initial position and f the fundamental frequency of the Fourier series for each actuator ρ_j, where $j = 1, \cdots, length(\mathbf{q}_\rho)$.

$$\rho_j(t) = \rho_{j_0} + \sum_{i=1}^{n_h} \left[\frac{C_{ji}}{2\pi f\ i}\ sin(2\pi f\ i\ t) - \frac{D_{ji}}{2\pi f\ i}\ cos(2\pi f\ i\ t) \right] \tag{5.32}$$

$$\dot{\rho}_j(t) = \sum_{i=1}^{n_h} \left[C_{ji}\ cos(2\pi f\ i\ t) + D_{ji}\ sin(2\pi f\ i\ t) \right] \tag{5.33}$$

$$\ddot{\rho}_j(t) = \sum_{i=1}^{n_h} \left[-C_{ji}\ (2\pi f\ i)\ sin(2\pi f\ i\ t) + D_{ji}\ (2\pi f\ i)\ cos(2\pi f\ i\ t) \right] \tag{5.34}$$

5.3.1 Optimization of the trajectory

From the equations above, C_{ji}, D_{ji}, ρ_{j_0}, n_h and f are initially unknown. Thus, the optimization procedure herein applied is used to find the values of the amplitudes and the initial position ρ_{j_0} that minimize the condition

number of matrix $\mathbf{W}_{base}$, given that the fundamental frequency f and the number of harmonics n_h of the Fourier series are supplied in advance.

Also, since the range of positions is limited to the joint-space of the mechanism, and the velocities and acceleration are also limited by the actuators specification, the optimization process has to be *constrained* to these limits. For this reason, a constrained optimization is performed.

Furthermore, it is remembered that the condition number is calculated as the highest to lowest ratio of the singular values of matrix $\mathbf{W}_{base}$. Even if matrix $\mathbf{W}_{base}$ is full rank its condition number can be high. In such cases, the rank of the matrix is forced to be a lower value if an acceptable condition number is not achieved.

5.3.2 Local and global optimization

In general, local or global optimization procedures can be used to find the parameters C_{ji}, D_{ji} and ρ_{j_0} of the trajectory. A local optimization algorithm, such as the Matlab® function *fmincon* can be used to find the values of the parameters that minimize the objective function, defined as the condition number of matrix $\mathbf{W}_{base}$. An initial set of parameters values must be used as a seed to start the optimization process, around which the optimum parameters values are searched for. However, if a local minimum of the objective function is reached, the resulted optimum parameters values are not the best. On the other hand, a *global optimization* algorithm, such as the Matlab® function *patternsearch*, can be used instead, as it is explained in the following section.

5.3.3 Direct search method optimization

Direct search methods of optimization such as the Matlab function *patternsearch*, are a global optimization algorithm used to solve problems when the objective function is not differentiable or is not continuous [100]. Considering the condition number of matrix $\mathbf{W}_{base}$ as the objective function to minimize, the direct search method works as follows. With an initial set

of parameters values as a seed, the algorithm looks for a set of parameters values by building a mesh around the current set that minimize the objective function, which is then used as the starting point for the next iteration. This process stops if a predefined number of iterations are reached, or if the variation of the value returned by the objective function in consecutive iterations is lower than a priori defined tolerance.

5.4 Post-processing

Often times, the experimental signals may present noise or distortions such as time offsets between the commanded and the measured variables or transient effects. These have to be eliminated before performing the identification because they may significantly influence the condition number of the matrix $\mathbf{W}_b$ and, hence, worsen the robustness of the value of the parameters to be identified.

In this step, any time offset is eliminated by trimming the signals first. Also, if the acceleration of the actuators cannot be measured directly, it has to be calculated using the velocity signal from the encoders using numerical differentiation. This further introduces noise, the calculated acceleration signal has to be filtered with a low-pass filter setup accordingly to the noise in the signal.

5.5 Identification of each individual parameter

As a result of performing the identification using Eq. 5.23, the base parameters are obtained which can be written as a sum of the actual system's parameters like in Eq. 5.35. Therein, $\mathbf{C}_{base}$ is the matrix with the coefficients multiplying each dynamic parameter in $\mathbf{\Phi}$ within each linear combination of $\mathbf{\Phi}_{base}$. Similarly, $\mathbf{V}_{base}$ is the vector with the resulting values of each

base parameters after the identification.

$$\mathbf{\Phi}_{base} = \mathbf{C}_{base}\mathbf{\Phi} = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N_\Phi} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N_\Phi} \\ \vdots & \vdots & & \vdots \\ c_{N_{base},1} & c_{N_{base},2} & \cdots & c_{N_{base},N_\Phi} \end{bmatrix} \begin{Bmatrix} \mathbf{\Phi}_1 \\ \mathbf{\Phi}_2 \\ \vdots \\ \mathbf{\Phi}_{N_\Phi} \end{Bmatrix} = \begin{Bmatrix} \mathbf{V}_{base_1} \\ \mathbf{V}_{base_2} \\ \vdots \\ \mathbf{V}_{base_{N_{base}}} \end{Bmatrix} \tag{5.35}$$

The number of rows in matrix $\mathbf{C}_{base}$, N_{base}, is less than the number of elements in $\mathbf{\Phi}$, hence it is an undetermined linear system. As a result, a linear regression can not be used to obtain each individual dynamic parameter. However, if a set $\mathbf{\Phi}_t$ of enough parameters can be estimated or are known beforehand accurately, then it is possible to extend the system and solve it to find the values for each parameter. In this way, an estimation of the parameters $\hat{\mathbf{\Phi}}$ can be obtained by solving Eq. 5.36.

$$\mathbf{C}\hat{\mathbf{\Phi}} = \mathbf{V} = \begin{bmatrix} \mathbf{C}_{base} \\ \mathbf{C}_x \end{bmatrix} \hat{\mathbf{\Phi}} = \begin{bmatrix} \mathbf{V}_{base} \\ \mathbf{V}_x \end{bmatrix} \tag{5.36}$$

Where $\mathbf{\Phi}_t$ is defined as the set of N_t *trusted parameters* and N_x is the number of required known parameters to make the linear system in Eq. 5.35 determined. Furthermore, N_x is calculated as follows:

$$N_x = N_\Phi - N_{base} \tag{5.37}$$

Also, let $\mathbf{x}$ to be defined as a subset of $\mathbf{\Phi}_t$ like in Eq. 5.38.

$$\mathbf{x} = \{x_1, x_2, \cdots, x_{N_x}\} \text{ such that } \mathbf{x} \subseteq \mathbf{\Phi}_t \subseteq \mathbf{\Phi} \tag{5.38}$$

It follows that the set $\mathbf{x}$, chosen among $\mathbf{\Phi}_t$, can be used to solve Eq. 5.36 as in Eq. 5.39, where $\mathbf{C}_x$ is defined as in Eq. 5.40. Moreover, the term $c_{x_{ij}}$ equals 1 if the chosen trusted parameter x_j is the dynamic parameter

$\mathbf{\Phi}(k)$, otherwise $c_{x_{ij}}$ is equal to 0. Also, the elements $\mathbf{V}_{x_j}$ of matrix $\mathbf{V}$ are the values of the chosen *trusted parameter*.

$$\hat{\mathbf{\Phi}} = \mathbf{C}^{-1}\mathbf{V} \tag{5.39}$$

$$\mathbf{C}_x = \begin{bmatrix} c_{x_{11}} & c_{x_{12}} & \cdots & c_{x_{1N_\Phi}} \\ c_{x_{21}} & c_{x_{22}} & \cdots & c_{x_{2N_\Phi}} \\ \vdots & \vdots & & \vdots \\ c_{x_{N_x1}} & c_{x_{N_x2}} & \cdots & c_{x_{N_xN_\Phi}} \end{bmatrix} \tag{5.40}$$

Where,

$$\begin{aligned} c_{x_{ij}} &= 1 \text{ if } x_j = \mathbf{\Phi}(k) \\ c_{x_{ij}} &= 0 \text{ if } x_j \neq \mathbf{\Phi}(k) \end{aligned} \tag{5.41}$$

$$\mathbf{V}_{x_j} = \mathbf{\Phi}_{t_j}(j) \tag{5.42}$$

If the number of required parameters N_x to build matrix $\mathbf{C}$ is less than the number of trusted parameters N_t, there will be several possible combinations of the subset $\mathbf{x}$ that can be used to solve Eq. 5.39. In fact, the number of possible combinations N_{comb} can be calculated as follows:

$$N_{comb} = \frac{N_t!}{N_x!\,(N_t - N_x)!} \tag{5.43}$$

Among all these combinations, some of them may result in a matrix $\mathbf{C}$ without an inverse, in which case matrix $\mathbf{C}$ would be rank deficient. As a consequence, these combinations must be discarded. Moreover, with every possible combination of the trusted parameters, different values for the dynamic parameters are obtained. Hence, it is possible to establish a range in which the actual value for each dynamic parameter lays in. Then, the problem is to determine the exact value for each parameter from their respective range.

Also, negative values could also be obtained by solving the system in Eq. 5.39. Although these values solve the linear system mathematically, they lack of any physical meaning. Hence, the negative values can be ignored and the range of values is established to be from the minimum to maximum positive values for each parameter.

Lastly, to determine an specific value, the mean of the range can be used or, alternatively, an optimization routine can be applied to find the best set of parameters values that minimizes the error between estimated force signals with the estimated parameters and the experimental force signals, which has been done in this work.

5.6 Summary of the parameters identification process

In summary, the process to identify the dynamic parameters is described in Fig. 5.3. The process starts with the definition of the initial Fourier series parameters ρ_{j_0}, C_{ji}, D_{ji}, f, n_h. These are the used for the trajectory optimization using the *fmincon* or the *pattersearch* Matlab algorithms.

Upon successful completion of the optimization, matrix $\mathbf{W}_{base}$ is obtained as well as the *base parameters*. If the condition number $\kappa(\mathbf{W}_{base})$ is not lower or equal to a tolerance defined beforehand, then the rank of matrix $\mathbf{W}_{base}$ is reduced to find a better condition number and start again the optimization. Otherwise, the *trusted parameters* are used to find the N_{com} possible combinations of values for each parameters, which are finally used to obtain an estimate of the parameters by calculating the mean or using an optimization method.

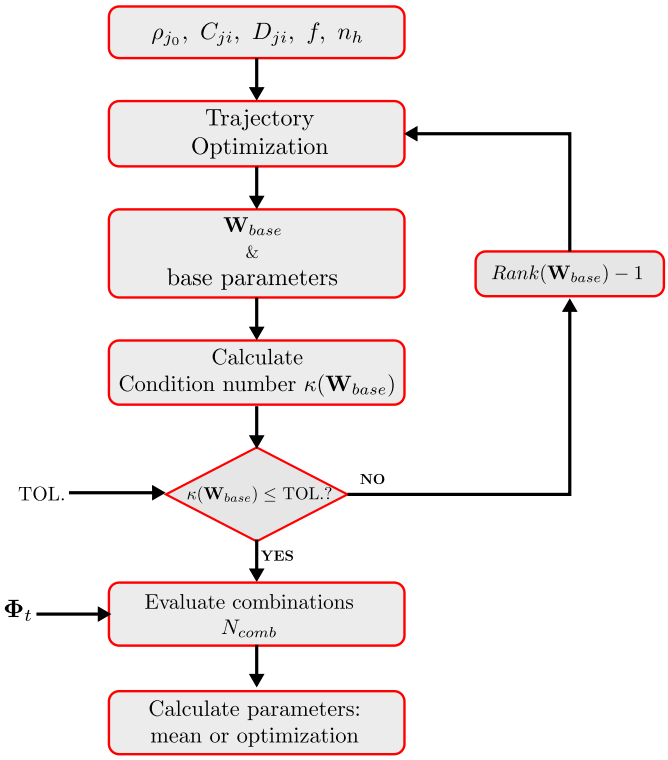


Figure 5.3. Scheme of the parameters identification process.

Chapter 6

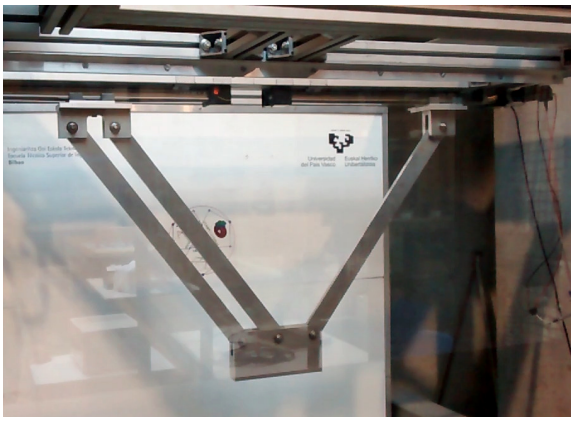
Experimental Validation of the Identification Procedure

In this chapter, an experimental validation of the identification procedure has been carried out on two case studies. The first case study is a simple 2-PRR planar mechanism for which the identification was first implemented. The second case study is a haptic 5R robotic system developed by the *IGM-RWTH* for which the identification procedure was performed in the context of a research internship.

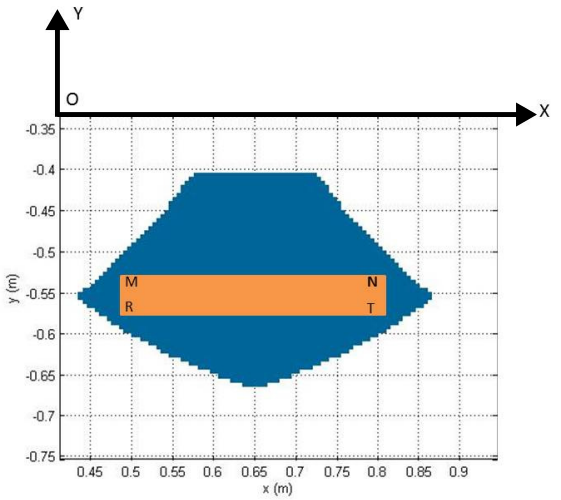
6.1 Case Study 1: 2-PRR Mechanism

The 2-PRR mechanism in Fig. 6.1a was considered as a first case study. The intended application of this mechanism is foreseen to be for pick & place purposes. Moreover, this prototype consists of a 2-PRR mechanism with linear guides IGUS-ZLW-1040 as prismatic joints actuated by a Maxon RE-40 DC motors with a gearbox GP-32A of 14:1 gear ratio. The linear guides are coupled to the gearbox with a flexible shaft coupling WAC25 from *Helical*. Also, the mechanism presents a *Four-bar linkage* on one

prismatic joint to constrain the rotation motion of the final platform, in this way the end-effector is only capable of a translation motion in the XY plane defined in Fig. 6.1.



(a)



(b)

Figure 6.1. (a) Picture and (b) workspace of the 2-PRR mechanism.

In Fig. 6.1a, it can be appreciated the two linear guides where the mechanism is attached. Additionally, the 3 bars of the mechanism as well as the end platform are shown. The three identical bars have a weight of 1.02 kg and a length of 0.654 m. As for the end platform, it has a weight of 2.5 kg. Furthermore, the bars are joined by shafts and ball bearings, and the complete mechanism has been manufactured and assembled in the facilities of the Mechanical Engineering Department.

In Fig. 6.1b the workspace of the mechanism is shown. It can be seen that the operating workspace is limited between -0.41 m and -0.68 m in the y direction, and between 0.4 m and 0.86 m in the x direction, with no singularities within the workspace. In this way, a pick & place task can be performed in a rectangle of 0.46 m $\times$ 0.27 m.

Additionally, in Fig. 6.2 a scheme of the mechanism with the main kinematics variables is shown, where $a = 1.3$ m is the distance separating the origin of the two actuators ρ_1 and ρ_2, F_1 and F_2 are the input forces, $d = 95$ mm is the distance between the revolute joints, and the orientation of the bars 3 and 4 is represented by α_1 and α_2 for the bar 5.

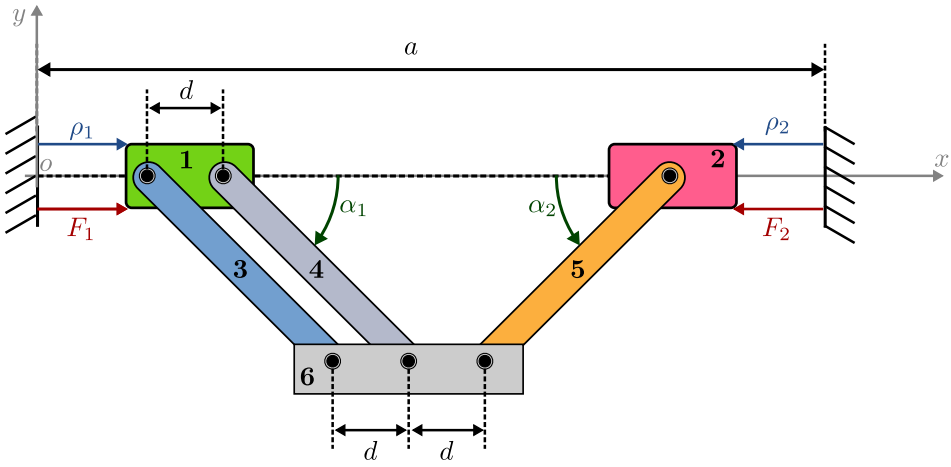


Figure 6.2. Kinematic scheme of the 2-PRR mechanism.

6.1.1 Rigid body model

To obtain a dynamic model for identification of the 2-PRR mechanism, the principle of energy equivalence was applied as described in chapter 5. To that end, the 2-PRR mechanism was separated into the following 6 solids with their respective dynamic parameters, where m and I are the mass and inertia respectively:

- Actuator ρ_1: $m_1,\ I_1$
- Actuator ρ_2: $m_2,\ I_2$
- Bar 3: $m_3,\ I_3$
- Bar 4: $m_4,\ I_4$
- Bar 5: $m_5,\ I_5$
- End platform: $m_6,\ I_6$

Afterwards, the forward kinematics is solved for each solid relating the mass center positions x_b, y_b and orientation α_b with the coordinates of the actuated joints ρ_1, ρ_2 as in equations 6.1 and 6.2. As each actuator is composed by a rotational motor, a gearbox with gear ratio i_r and a linear guide with pitch p, the force F_1 and F_2 are calculated from the torque supplied by the motors τ_j as in Eq. 6.3. In this way, the masses m_1 and m_2 include the inertia of the whole actuators.

$$\begin{Bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{\alpha}_b \end{Bmatrix} = \mathbf{J}_b \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{6.1}$$

$$\begin{Bmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{\alpha}_b \end{Bmatrix} = \dot{\mathbf{J}}_b \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} + \mathbf{J}_b \begin{Bmatrix} \ddot{\rho}_1 \\ \\ \ddot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} {}^t\dot{\mathbf{J}}_b \\ {}^r\dot{\mathbf{J}}_b \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} + \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \ddot{\rho}_1 \\ \\ \ddot{\rho}_2 \end{Bmatrix} \tag{6.2}$$

$$F_j = \tau_j \; i_r \frac{2\pi}{p}; \; j = 1, 2 \tag{6.3}$$

In equations 6.1 and 6.2, the Jacobians $\mathbf{J}_b$ were separated into ${}^t\mathbf{J}_b$ and ${}^r\mathbf{J}_b$, To account for the transformations of the linear and angular terms respectively for each solid b. Regarding the dynamics, by considering the rigid body diagram of the components shown in Fig. 6.3, the equations of motion are obtained for each solid after applying the Lagrange's equations as mentioned in chapter 5, resulting in Eq. 6.4. It is worth remembering that the joint forces are not taken into account as mentioned in section 3.4.1.

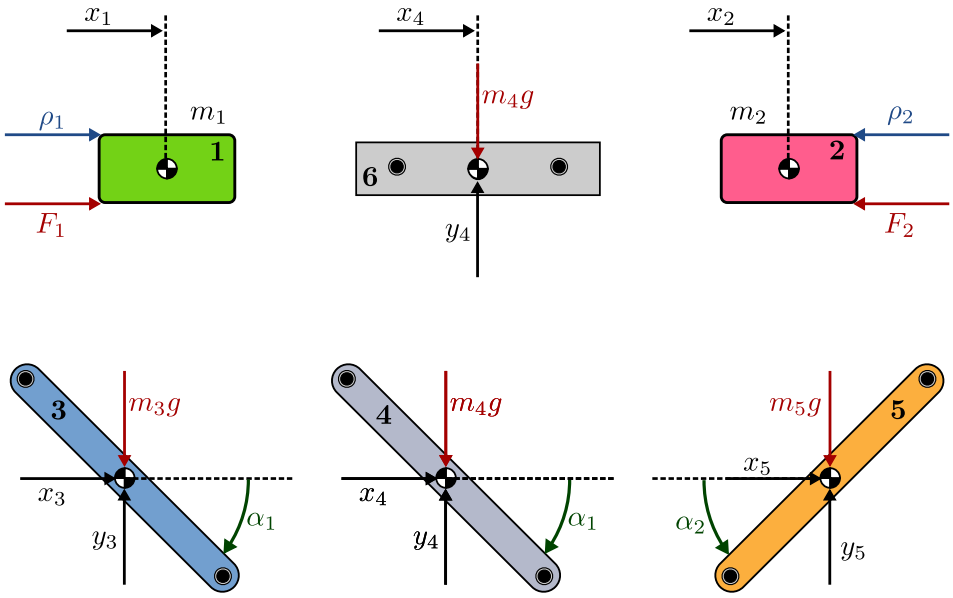


Figure 6.3. Rigid body diagrams of the components of the 2-PRR mechanism.

$$\left[\begin{bmatrix} \ddot{x}_b & 0 \\ \ddot{y}_b & 0 \\ 0 & \ddot{\alpha}_b \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ g & 0 \\ 0 & 0 \end{bmatrix} \right] \begin{Bmatrix} M_b \\ \\ I_b \end{Bmatrix} = \begin{Bmatrix} F_{x_b} \\ F_{y_b} \\ \tau_b \end{Bmatrix} \tag{6.4}$$

Furthermore, as described in section 5.2.1, by substituting Eq. 6.2 into Eq. 6.4, the equation of motion for each body is obtained in terms of the actuators variables.

$$\left[\begin{bmatrix} {}^{t}\dot{\mathbf{J}}_b \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\end{Bmatrix} + {}^{t}\mathbf{J}_b \begin{Bmatrix}\ddot{\rho}_1\\ \ddot{\rho}_2\end{Bmatrix} & 0 \\ 0 & {}^{r}\dot{\mathbf{J}}_b \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\end{Bmatrix} + {}^{r}\mathbf{J}_b \begin{Bmatrix}\ddot{\rho}_1\\ \ddot{\rho}_2\end{Bmatrix} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ g & 0 \\ 0 & 0 \end{bmatrix}\right] \begin{Bmatrix} m_b \\ \\ I_b \end{Bmatrix} = \begin{Bmatrix} F_{x_b} \\ F_{y_b} \\ \tau_b \end{Bmatrix} \tag{6.5}$$

By pre-multiplying both sides of the equation by the Jacobian $\mathbf{J}_b$, the forces and torques are projected onto the joint-space, thus resulting Eq. 6.6 which depends only on the data obtained from the actuators.

$$\left[\mathbf{J}_b^T \mathbf{\Omega}_b + \mathbf{J}_b^T \mathbf{G}_b\right] \begin{Bmatrix} m_b \\ I_b \end{Bmatrix} = \mathbf{K_c}_b \begin{Bmatrix} m_b \\ I_b \end{Bmatrix} = \mathbf{J}_b^T \mathbf{\Gamma}_b = \begin{Bmatrix} F_{1_b} \\ F_{2_b} \end{Bmatrix} \tag{6.6}$$

Afterwards, adding the contribution of all solids, the rigid body model of the mechanism results as follows:

$$\mathbf{K}_c \mathbf{\Phi}_c = \mathbf{\Gamma}_c \tag{6.7}$$

Where,

$$\mathbf{K}_c = \left[\mathbf{K_{c_1}}, \mathbf{K_{c_2}}, \mathbf{K_{c_3}}, \mathbf{K_{c_4}}, \mathbf{K_{c_5}}, \mathbf{K_{c_6}}\right] \tag{6.8}$$

$$\mathbf{\Gamma}_c = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} \tag{6.9}$$

$$\mathbf{\Phi}_c = \{m_1, I_1, m_2, I_2, m_3, I_3, m_4, I_4, m_5, I_5, m_6, I_6\}^T \tag{6.10}$$

As some of the dynamic parameters in $\mathbf{\Phi}_c$ do not appear in the equations of motion, such as I_1, I_2 and I_3, they are not taken into account to build the final dynamic model. Finally, the linear system in Eq. 6.11 represents the

rigid body dynamics of the 2-PRR mechanism as a function of 9 parameters, where they have been sorted in masses first and inertias afterwards in $\mathbf{\Phi}_{rb}$.

$$\mathbf{K}_{rb}\left(\mathbf{q}_{\rho}, \dot{\mathbf{q}}_{\rho}, \ddot{\mathbf{q}}_{\rho}\right)_{2\times 9} \mathbf{\Phi}_{rb_{9\times 1}} = \mathbf{\Gamma}_{rb_{2\times 1}} \tag{6.11}$$

Where,

$$\mathbf{q}_{\rho} = \begin{Bmatrix} \rho_1 \\ \rho_2 \end{Bmatrix} \tag{6.12}$$

$$\mathbf{\Phi}_{rb} = [m_1, m_2, m_3, m_4, m_5, m_6, I_3, I_4, I_5]^T \tag{6.13}$$

6.1.2 Friction model

To take the friction of the mechanism into account, a viscous + Coulomb friction model is herein used. Hence, the friction of all the system is assumed to occur at the actuators. In Eq. 6.14 the friction force on both actuators is written as a linear system, where F_c represents the Coulomb friction and c the viscous friction.:

$$\mathbf{\Gamma}_f = \begin{bmatrix} sign(\dot{\rho}_1) & 0 & \dot{\rho}_1 & 0 \\ 0 & sign(\dot{\rho}_2) & 0 & \dot{\rho}_2 \end{bmatrix} \begin{Bmatrix} F_{c_1} \\ F_{c_2} \\ c_1 \\ c_2 \end{Bmatrix} = \mathbf{K}_f \cdot \mathbf{\Phi}_f \tag{6.14}$$

6.1.3 Identification Model

Subsequently, the friction dynamics are appended to the rigid body dynamics obtained in Eq. 6.11, thus resulting in the dynamic model for identification of Eq. 6.15 with 13 parameters to be identified.

$$\left[\mathbf{K}_{rb}\mathbf{K}_f\right] \begin{Bmatrix} \mathbf{\Phi}_{rb} \\ \mathbf{\Phi}_f \end{Bmatrix} = \mathbf{K}\left(\mathbf{q}_{\rho}, \dot{\mathbf{q}}_{\rho}, \ddot{\mathbf{q}}_{\rho}\right)_{2\times 13} \mathbf{\Phi}_{13\times 1} = \begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} \tag{6.15}$$

By performing a linear regression on Eq. 6.15 the dynamic paramerers in $\mathbf{\Phi}$ could be solved if the system was not under-determined. By using

several points of a trajectory the system can be expanded and thus the overdetermined linear system in Eq. 6.16 results where n_p is the number of points used.

$$\mathbf{W}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{(2\cdot n_p)\times 13} \mathbf{\Phi}_{13\times 1} = \mathbf{\Gamma}_{(2\cdot n_p)\times 1} \tag{6.16}$$

In general, the parameters in $\mathbf{\Phi}$ cannot be identified separately, on the contrary, they are identified as linear combinations. By following the procedure described in section 5.2.4, the system with $N_{base} \leqslant 13$ base parameters in Eq. 6.17 is obtained.

$$\mathbf{W}_{base}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{(2\cdot n_p)\times N_{base}} \mathbf{\Phi}_{base_{N_{base}\times 1}} = \mathbf{\Gamma}_{(2\cdot n_p)\times 1} \tag{6.17}$$

6.1.4 Identification trajectory

Regarding the identification trajectory, following the guidelines described in section 5.3 and in previous works [84, 99], it is parameterized by using periodic trajectories in the form of the Fourier series shown in equations 6.18-6.20.

$$\rho_j(t) = \rho_{j_0} + \sum_{i=1}^{n_h} \left[\frac{C_{ji}}{2\pi f\ i}\ sin(2\pi f\ i\ t) - \frac{D_{ji}}{2\pi f\ i}\ cos(2\pi f\ i\ t) \right] \tag{6.18}$$

$$\dot{\rho}_j(t) = \sum_{i=1}^{n_h} \left[C_{ji}\ cos(2\pi f\ i\ t) + D_{ji}\ sin(2\pi f\ i\ t) \right] \tag{6.19}$$

$$\ddot{\rho}_j(t) = \sum_{i=1}^{n_h} \left[-C_{ji}\ (2\pi f\ i)\ sin(2\pi f\ i\ t) + D_{ji}\ (2\pi f\ i)\ cos(2\pi f\ i\ t) \right] \tag{6.20}$$

Where ρ_{j_0} is the initial position of the j^{th} actuator, n_h is the number of harmonics, f is the Fourier series fundamental frequency and C_{ji} and D_{ji} are the sines and cosines amplitudes for each harmonic.

In order to calculate the optimum C_{ji}, D_{ji} and ρ_{j_0} parameters, the local optimization routine *fmincon* and the global optimization routine *Direct*

search method/patternsearch from the *Matlab/Optimization* toolbox were considered and further compared. These two methods were studied mainly because *fmincon* is a local optimization routine whereas the *Direct search method* falls into the global optimization routines category, so the purpose was to asses which one resulted in better results.

Since the actuators used are similar to the ones used for the MAST mechanism in section 4.2, a fundamental frequency of 0.4 Hz and 10 harmonics were used as the bandwidth of the actuators was found to be approximately 4 Hz. Also the initial values in Table 6.1 for the amplitudes and initial positions $\rho_1 = 0.2\,\text{m}$ and $\rho_2 = 0.25\,\text{m}$ were used for the optimization. Furthermore, considering the joint-space of the mechanism in Fig. 6.4, a 0.002 m and 0.0470 m lower position limits for actuators 1 and 2 respectively were used, while a 0.035 m superior limit was used in both cases. As for the velocity and acceleration limits, 0.57 m/s and $5\,\text{m/s}^2$ were used respectively for both actuators taken from the manufacturer datasheet. As a result, the trajectory in the joint-space shown in Fig. 6.4 resulted.

Table 6.1. Initial values of the amplitudes for the Fourier series

n_h	Act. 1		Act. 2	
	C_{1i} [m]	D_{1i} [m]	C_{2i} [m]	D_{2i} [m]
1	0.0066	0.0066	0.0060	0.0060
2	0.0033	0.0033	0.0030	0.0030
3	0.0022	0.0022	0.0020	0.0020
4	0.0016	0.0016	0.0015	0.0015
5	0.0013	0.0013	0.0012	0.0012
6	0.0011	0.0011	0.0010	0.0010
7	0.0009	0.0009	0.0009	0.0009
8	0.0008	0.0008	0.0008	0.0008
9	0.0007	0.0007	0.0007	0.0007
10	0.0007	0.0007	0.0006	0.0006

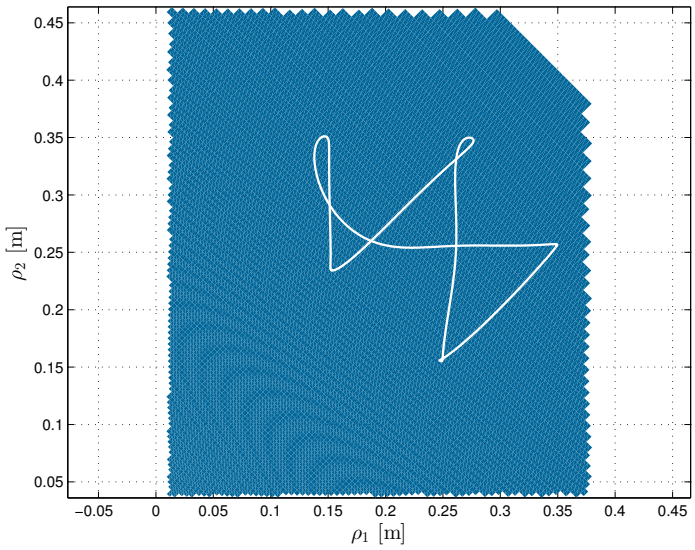


Figure 6.4. Optimized identification trajectory for the Scissor mechanism.

When the *fmincon* method was employed a condition number of 28.23 was obtained. Nevertheless, with the *Direct search method/patternsearch* a 20.78 of condition number resulted. Although in this case the difference is of only 7.45 points, the *Direct search method* was selected because provided a lower condition number.

6.1.5 Identification of the base parameters

The trajectory was executed using the control in the PXIe and the joints position, velocity were experimentally obtained using the encoder measurements from the motors with a sampling time of 2 ms. The acceleration was calculated by numerical differentiation of the velocity. Also, the current was measured to calculate the motors torque using the torque constant $K_t = 30.2\,\mathrm{mN\,m/A}$ from the datasheet. Furthermore, to eliminate noise and transient effects in the experimental signals, a third order low-pass filter with a 15 Hz cut-off frequency was used. The forces F_1 and F_2 were

calculated using Eq. 6.21, where I_c is the current of the motors, $i_r = 14$ is the gear ratio, and $p = 0.07\,\mathrm{m/rev}$ is the pitch of the linear guide. Additionally, the positions ρ_1 and ρ_2 were calculated using Eq. 6.22, where θ is the angle of the motors.

$$F_i = K_t \; I_{c_i} i_r \frac{2\pi}{p}\bigg|_{i=1,2} \tag{6.21}$$

$$\rho_i = \theta_i \frac{p}{2\pi i_r}\bigg|_{i=1,2} \tag{6.22}$$

In Fig 6.5 the commanded and experimental positions, velocity and acceleration signals are shown. It can be seen that the controller presents a good tracking of the commanded signals, and no major distortions are observed apart from some acceleration overshoots.

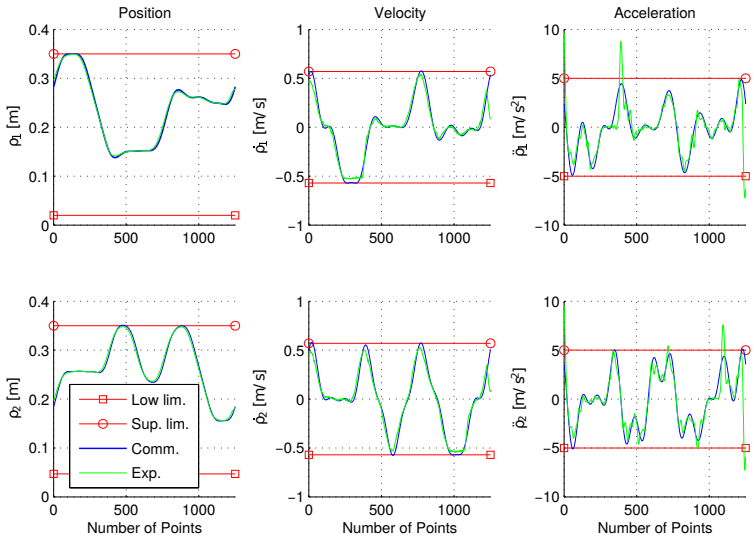


Figure 6.5. Comparison of the position velocity and acceleration signals of the identification trajectory signals.

After the post-processing and executing the identification algorithm in Matlab®, the base parameters in equations 6.23-6.30 and their values were obtained. It is observed in the first four base parameters that, despite inertial parameters are present, the main contribution is due to the Coulomb and viscous friction. Similarly, the base parameter in Eq. 6.28 is approximately $m_2 + 0.5m_5$.

$$\Phi_{base}(1) = 0.0075I_5 + 0.0075I_4 + 0.0075I_3 + F_{c_1} = 44.6742 \quad (6.23)$$

$$\Phi_{base}(2) = 0.008I_5 + 0.008I_4 + 0.008I_3 + F_{c_2} = 41.8721 \quad (6.24)$$

$$\Phi_{base}(3) = -0.0504I_5 - 0.0504I_4 - 0.0504I_3 + c_1 = 6.6274 \quad (6.25)$$

$$\Phi_{base}(4) = -0.0523I_5 - 0.0523I_4 - 0.0523I_3 + c_2 = 2.4461 \quad (6.26)$$

$$\Phi_{base}(5) = m_1 - 0.5m_5 + 4.6625I_5 + 4.6625I_4 + 4.6625I_3 = 19.5026 \quad (6.27)$$

$$\Phi_{base}(6) = m_2 + 0.5m_5 + 0.0092I_5 + 0.0092I_4 + 0.0092I_3 = 20.8867 \quad (6.28)$$

$$\Phi_{base}(7) = m_6 + 4.6417I_5 + 4.6417I_4 + 4.6417I_3 = 4.7265 \quad (6.29)$$

$$\Phi_{base}(8) = m_5 + m_4 + m_3 - 9.3088I_5 - 9.3088I_4 - 9.3088I_3 = 2.2236 \quad (6.30)$$

6.1.6 Identification of the individual parameters

Equations 6.23-6.30 are linear combinations of the individual dynamic parameters of the mechanism. To obtain the value for each of the 13 parameters, the linear combinations are *disassembled* considering the values of the trusted dynamic parameters. Since 8 linear combinations were found, 5 extra parameters are required. For the 2-$\underline{P}$RR mechanism, I_3, I_4, I_5, m_3, m_4, m_5 and m_6 are considered to be *well-known* beforehand as they have been manufactured and weighted in our dependencies. Following Eq. 5.43 up to 21 possible combinations exist to complete the linear system. However, only 15 of them can be used after removing the combinations which returned a rank deficient system. After solving all the combinations, the mean and the standard deviation for each parameter were obtained and are shown in Table 6.2.

Table 6.2. Parameters values obtained according to a priori known parameters set

Parameter	Mean	Standard deviation	Theoretic	Units
m_1	18.9629	0.7672	23.9744	kg
m_2	20.0594	0.6310	23.5028	kg
m_3	3.3610	1.4459	1.0267	kg
m_4	3.3610	1.4459	1.0267	kg
m_5	3.3610	1.4459	1.0267	kg
m_6	4.2582	0.0452	2.7750	kg
I_3	0.2651	0.1642	0.0366	kgm^2
I_4	0.2651	0.1642	0.0366	kgm^2
I_5	0.2651	0.1642	0.0366	kgm^2
F_{c_1}	44.6720	0.0012	-	N
F_{c_2}	41.8698	0.0013	-	N
c_1	6.6421	0.0082	-	N s/m
c_2	2.4614	0.0085	-	N s/m

It is observed that the identified parameters in the table present differences with respect to their theoretical values. The most significant differences are obtained for the masses and inertia of the bars. The masses of the bars present an error of 2.3343 kg, while the inertias present a deviation of $0.2285\,\mathrm{kgm}^2$. These differences are attributed to two main causes. On one hand, since the condition number is not 1, it is expected that the identified parameters values present errors caused by distortions in the measurements. On the other hand, as the mass of the bars are approximately 22 times smaller that the masses m_1 and m_2, it is possible that the masses and inertias of the bars do not contribute significantly to the dynamics of the mechanism, and therefore cannot be appropriately identified as it was mentioned by Pham and Gautier [101].

After updating the dynamic model with the mean parameters shown in Table 6.2,the identification trajectory was simulated and compared with the experimental torque. In Fig. 6.6, it is compared the experimental torque

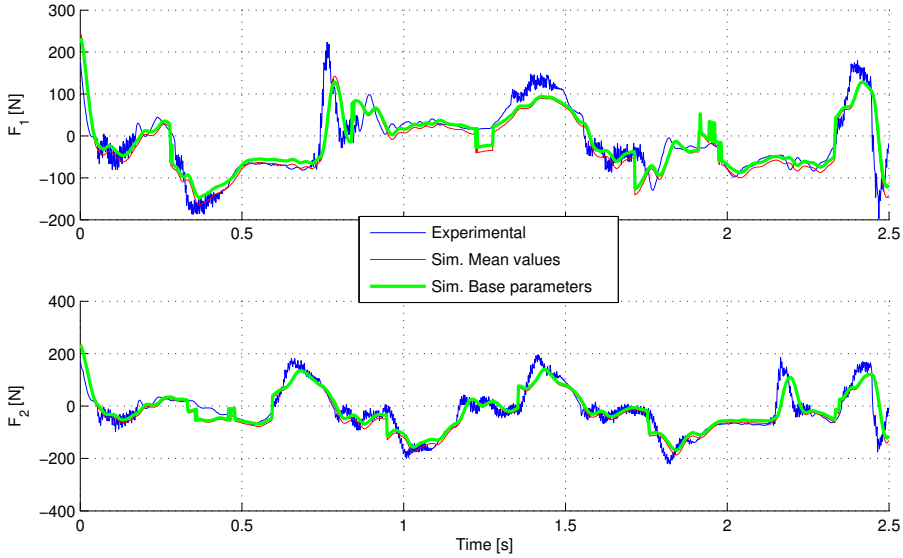


Figure 6.6. Measured forces vs. simulated forces with the base parameters and the mean values of the physical parameters.

with the *reduced* model with the base parameters, and the model with the mean values. It can be seen that model with the base parameters and the model with the mean values of the parameters are similar with a maximum deviation of 20 N between them. Furthermore, when the simulations are compared to the experimental signal, larger errors are observed with a maximum peak error of 81 N for a measured torque of 223.3 N.

6.1.7 Experimental validation

To validate the dynamic model with the parameters obtained it is necessary to use a trajectory different than the used for the identification. In this regard, a sinusoidal trajectory with a 3.5 Hz frequency, an amplitude of 11.45 mm and 7 cycles was programmed. Also, the control period was set to 5 ms. In Fig. 6.7, the tested trajectory is shown and compared with the commanded position signals. It can be seen a difference in amplitudes and

time delay between the commands and the experimental signals. These differences were expected as the motion frequency of 3.5 Hz is near the bandwidth of the actuators.

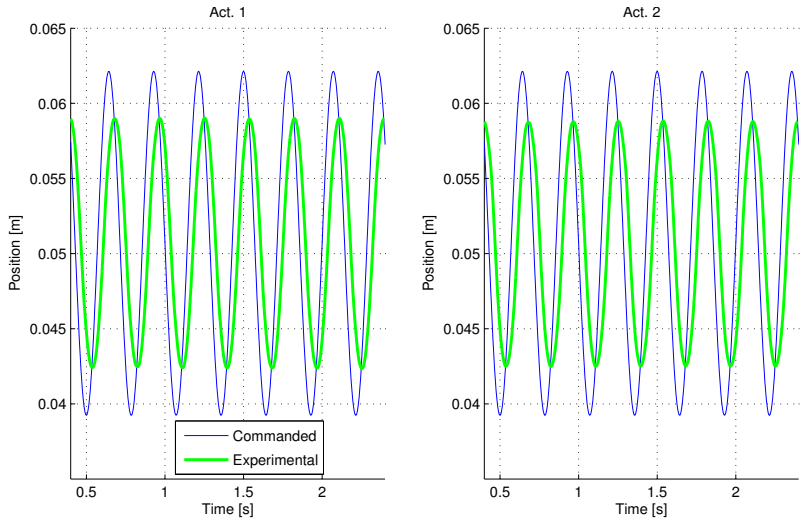


Figure 6.7. Commanded and measured validation trajectory for each joint.

In Fig. 6.8 a comparison of the torques in the motors has been carried out between the model with the identified parameters, the model with the base parameters and the experimental. It can be observed that the simulated torques are similar to the experimental ones. However, some deviations are still present specially for the second actuator. A maximum difference of approximately 92 N from an actual force of 190 N are appreciated for the second actuator between the simulations and the measurements. As for the first actuator, a maximum difference of approximately 75 N from a measured force of 156 N is observed between the measurements and the model with the mean values. These deviations are attributed to unmodeled dynamic phenomena, such as flexible behavior of the bodies, differences between the identified dynamic parameters and their real values.

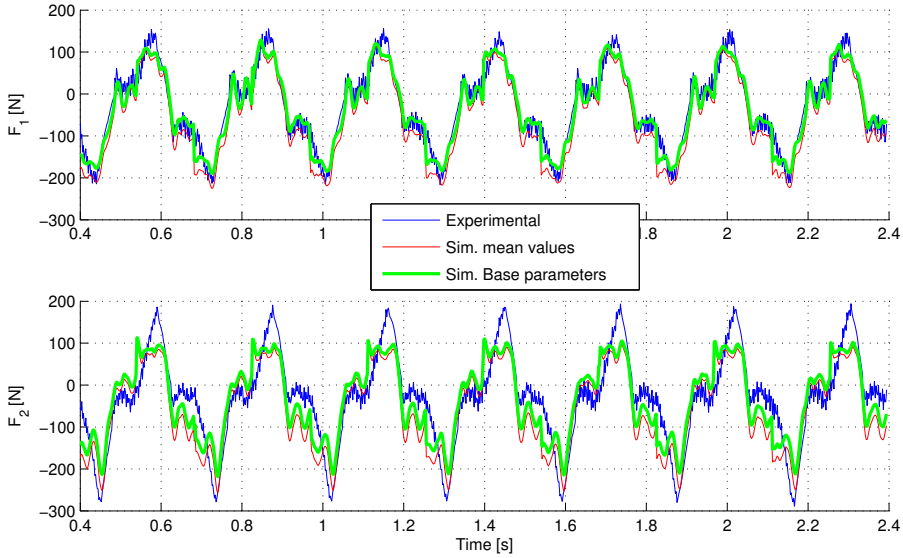


Figure 6.8. Measured and simulated torque comparison.

6.2 Case Study 2: 5R haptic mechanism

The *Reconfigurable Planar Linkage* mechanism, also called RePlaLink, is a 5R parallel robot with a third serial actuator mounted at the TCP. A representation of the mechanism is shown in Fig. 6.9. As the length of the lower bars of the mechanism are adjustable, the motors are placed in two separate horizontal planes on the *squared-shaped* frame, to avoid their collision when the mechanism is moved. Also, the horizontal distance between the main motors is adjustable, and the frame can be rotated to change the orientation of the working plane. The third actuator is used to change the orientation of a handle placed at the TCP of the mechanism, where a force and torque sensors are placed.

The purpose of the RePlaLink is to be used for simulating other mechanical devices or mechanism prototypes before manufacturing an actual

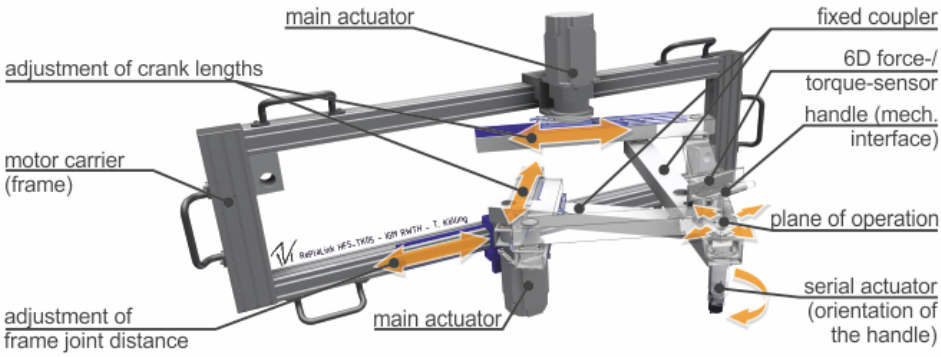


Figure 6.9. Representation of the RePlaLink mechanism and its components. (Courtesy of the IGM-RWTH Aachen).

device. In this way, the 5R mechanism is configured to simulate virtual prototypes taking the inputs from an operator. Therefore, it is operated as a haptic feedback system [9]. In Fig. 6.10 a use case is shown, where an operator is shown to moved a virtual prototype of a cabinet. Additionally, in Fig. 6.11 a hypothetical case study is shown, describing an operator interacting with a cabinet with 5 different designs, which the RePlaLink is intended to simulate.

The purpose of applying the identification procedure on the 5R mechanism is twofold. On the one hand, to evaluate further the identification procedure described with a mechanism different than the 2-PRR. On the other hand, to obtain a dynamic model considering friction that can be used for compensation by control.

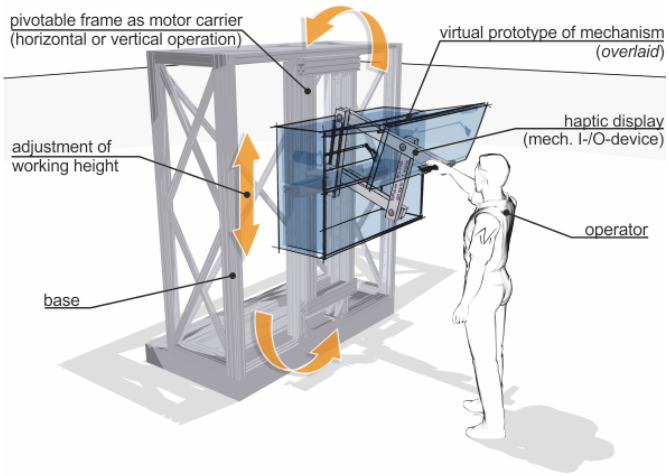


Figure 6.10. Representation of the RePlaLink in a general use case. (Courtesy of the IGM-RWTH Aachen).

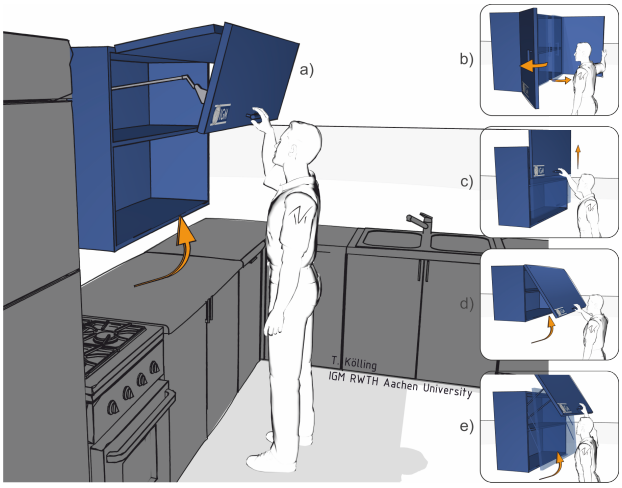


Figure 6.11. Representation of a hypothetical case study and four additional alternatives intended for simulation with the RePlaLink. (Courtesy of IGM-RWTH Aachen).

6.2.1 Rigid body model

The dynamic model for identification of the 5R haptic mechanism is again obtained following the procedure of chapter 5. In Fig. 6.12 a scheme of the mechanism is shown.

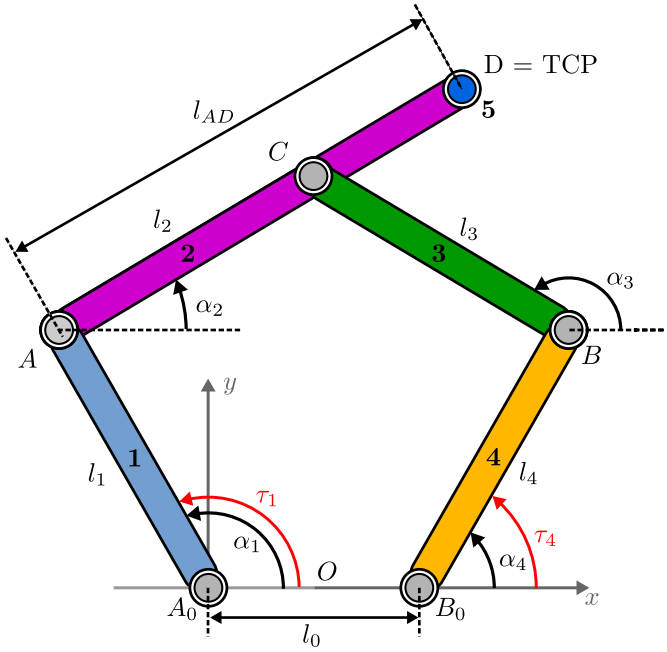


Figure 6.12. Scheme of the RePlalink.

To use the principle of energy equivalence, the mechanism was separated into the following 7 components with their respective dynamic parameters:

- Bar A_0-$A = m_1, I_1$
- Bar A-$D = m_2, I_2$
- Bar B-$C = m_3, I_3$

- Bar B_0-$B = m_4, I_4$
- Mass D= m_5
- Actuator at $A_0 = I_6$
- Actuator at $B_0 = I_7$

Furthermore, the mass of the handle was not taken into account as its weight was expected to be negligible compared to the rest of the elements. The third motor is considered as a concentrated mass located at point D, and its inertia was not considered for the identification procedure as it was not operated. Additionally, assuming a rigid body model of the actuators, the inertia of the main motors is considered at points A_0 and B_0 in Fig. 6.12. Solving the forward kinematics for each solid, the linear and angular velocities and accelerations for each solid are obtained as in Eq. 6.31 and Eq. 6.32.

$$\begin{Bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{\alpha}_b \end{Bmatrix} = \mathbf{J}_b \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_2 \end{Bmatrix} = \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{6.31}$$

$$\begin{Bmatrix} \ddot{x}_b \\ \ddot{y}_b \\ \ddot{\alpha}_b \end{Bmatrix} = \dot{\mathbf{J}}_b \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} + \mathbf{J}_b \begin{Bmatrix} \ddot{\alpha}_1 \\ \\ \ddot{\alpha}_4 \end{Bmatrix} = \begin{bmatrix} {}^t\dot{\mathbf{J}}_b \\ {}^r\dot{\mathbf{J}}_b \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} + \begin{bmatrix} {}^t\mathbf{J}_b \\ {}^r\mathbf{J}_b \end{bmatrix} \begin{Bmatrix} \ddot{\alpha}_1 \\ \\ \ddot{\alpha}_4 \end{Bmatrix} \tag{6.32}$$

As for the dynamics, considering the rigid body diagram of the components of Fig. 6.13, the equations of motion result as in Eq. 6.33. Substituting equations 6.31 and 6.32 into Eq. 6.33 where appropriate, the equations of motion are obtained in terms of the actuators variables α_1 and α_4 as in Eq. 6.34.

$$\left[\begin{bmatrix} \ddot{x}_b & 0 \\ \ddot{y}_b & 0 \\ 0 & \ddot{\alpha}_b \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ g & 0 \\ 0 & 0 \end{bmatrix} \right] \begin{Bmatrix} m_b \\ \\ I_b \end{Bmatrix} = \begin{Bmatrix} F_{x_b} \\ F_{y_b} \\ \tau_b \end{Bmatrix} \tag{6.33}$$

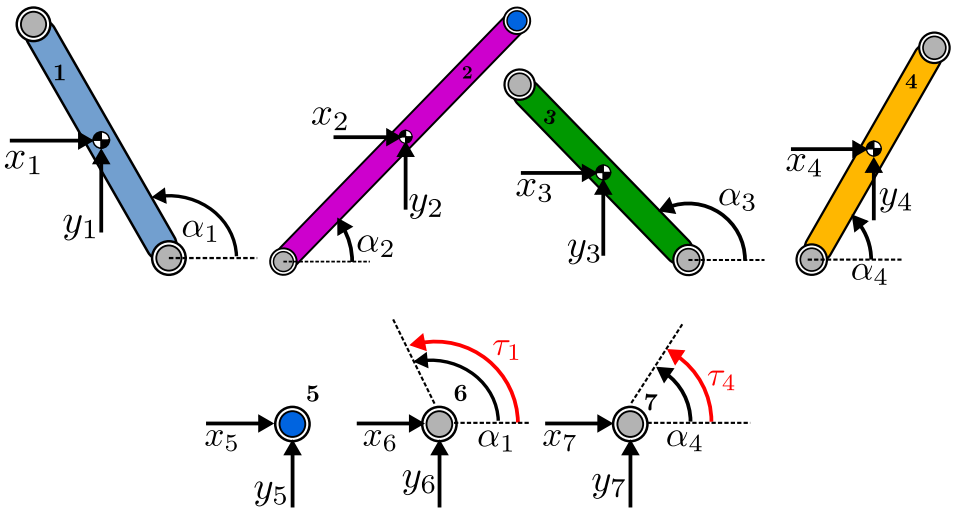


Figure 6.13. Rigid body diagrams of the components of the haptic mechanism.

$$\left[\mathbf{J}_b^T \mathbf{\Omega}_b + \mathbf{J}_b^T \mathbf{G}_b\right] \begin{Bmatrix} m_b \\ I_b \end{Bmatrix} = \mathbf{K}_b \mathbf{\Phi}_b = \mathbf{J}_b^T \mathbf{\Gamma}_b = \begin{Bmatrix} \tau_{1_b} \\ \tau_{4_b} \end{Bmatrix} \tag{6.34}$$

Where,

$$\mathbf{\Omega}_b = \begin{bmatrix} [{}^t\dot{\mathbf{J}}_b] \begin{Bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_4 \end{Bmatrix} + [{}^t\mathbf{J}_b] \begin{Bmatrix} \ddot{\alpha}_1 \\ \ddot{\alpha}_4 \end{Bmatrix} & 0 \\ & 0 \\ 0 & [{}^r\dot{\mathbf{J}}_b] \begin{Bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_4 \end{Bmatrix} + [{}^r\mathbf{J}_b] \begin{Bmatrix} \ddot{\alpha}_1 \\ \ddot{\alpha}_4 \end{Bmatrix} \end{bmatrix} \tag{6.35}$$

Since the mechanism was positioned horizontally, matrices $\mathbf{G}_b$ can be neglected as gravitational forces do not contribute to the dynamics of the system. Then, the rigid body model of the mechanism considering all the dynamic parameters results as follows:

$$\mathbf{K}_c \mathbf{\Phi}_c = \mathbf{\Gamma}_c \tag{6.36}$$

Where,

$$\mathbf{K}_c = \left[\mathbf{K_{c_1}}, \mathbf{K_{c_2}}, \mathbf{K_{c_3}}, \mathbf{K_{c_4}}, \mathbf{K_{c_5}}, \mathbf{K_{c_6}}, \mathbf{K_{c_7}}\right] \tag{6.37}$$

$$\mathbf{\Phi}_c = \{m_1, I_1, m_2, I_2, m_3, I_3, m_4, I_4, m_5, I_5, m_6, I_6, m_7, I_7\}^T \tag{6.38}$$

$$\mathbf{\Gamma}_c = \begin{Bmatrix} \tau_1 \\ \tau_4 \end{Bmatrix} \tag{6.39}$$

However, as m_6 and m_7 only rotate and I_5 does not exist as it is a concentrated mass, they do not contribute to the dynamics of the mechanism. Hence, a total of 11 rigid body parameters were considered to obtain the rigid body model of Eq. 6.40, where the terms have been grouped into masses and inertias.

$$\mathbf{K}_{rb}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{2\times 11} \mathbf{\Phi}_{rb_{11\times 1}} = \mathbf{\Gamma}_{rb_{2\times 1}} \tag{6.40}$$

Where,

$$\mathbf{q}_\rho = \begin{Bmatrix} \alpha_1 \\ \alpha_4 \end{Bmatrix} \tag{6.41}$$

$$\mathbf{\Phi}_{rb} = [m_1, m_2, m_3, m_4, m_5, I_1, I_2, I_3, I_4, I_6, I_7]^T \tag{6.42}$$

6.2.2 Friction model

Again, the friction of the complete system is also considered at the motors I_6 and I_7, as it is from where the measurements are taken. The Coulomb + viscous friction model of Eq. 5.17 was used resulting in:

$$\mathbf{F}_f = \begin{bmatrix} sign(\dot{\alpha}_1) & 0 & \dot{\alpha}_1 & 0 \\ 0 & sign(\dot{\alpha}_4) & 0 & \dot{\alpha}_4 \end{bmatrix} \begin{Bmatrix} F_{c_6} \\ F_{c_7} \\ c_6 \\ c_7 \end{Bmatrix} = \mathbf{K}_f \ \Phi_f \tag{6.43}$$

6.2.3 Identification model

The dynamic model at an instant considering the rigid body dynamics as well as the friction is written as follows:

$$\mathbf{K}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{[2\times 15]} \mathbf{\Phi}_{[15\times 1]} = \begin{Bmatrix} \tau_1 \\ \tau_4 \end{Bmatrix} \tag{6.44}$$

Where the total of 15 parameters to identify are:

$$\mathbf{\Phi} = \{m_1, m_2, m_3, m_4, m_5, I_1, I_2, I_3, I_4, I_6, I_7, F_{c_6}, F_{c_7}, c_6, c_7\}^T \tag{6.45}$$

When a trajectory is executed and n_p points are sampled, an overdetermined system as in Eq. 6.46 can be built based on Eq. 6.44.

$$\mathbf{W}\left(\mathbf{q}_\rho, \dot{\mathbf{q}}_\rho, \ddot{\mathbf{q}}_\rho\right)_{(2n_p)\times 15} \Phi_{15\times 1} = \mathbf{\Gamma}_{(2n_p)\times 1} \tag{6.46}$$

6.2.4 Identification trajectory

To obtain the base parameters, the procedure depicted in section 5.2.4 was followed, where Fourier series are used to parameterize the identification trajectory.

The local optimization routine *fmincon* and the direct search method *patternsearch* from the Matlab/Optimization toolbox were used to find the values of C_{ji}, D_{ji} and α_{j_0} that minimize the condition number of matrix $\mathbf{W}_{base}$. Additionally, 0.6981 rad and -0.3142 rad were considered as lower angular position limits for actuators α_1 and α_4 respectively. Also, 3.6652 rad and 2.4435 rad were used as the superior angular position limits for actuators α_1 and α_4 respectively. The trajectory was also constrained to be inside the joint-space of the mechanism using the *inpolygon* Matlab® function. Moreover, a velocity limit of 2 rad/s and an acceleration limit of 30 rad/s^2 were considered for both actuators.

When the function *fmincon* was used, the optimization failed to obtain a trajectory within the joint-space of the 5R haptic mechanism as in Fig. 6.14,

where the thin red line is the trajectory. A possible explanation is that the routine found a local minimum, which failed to respect the joint-space constraints.

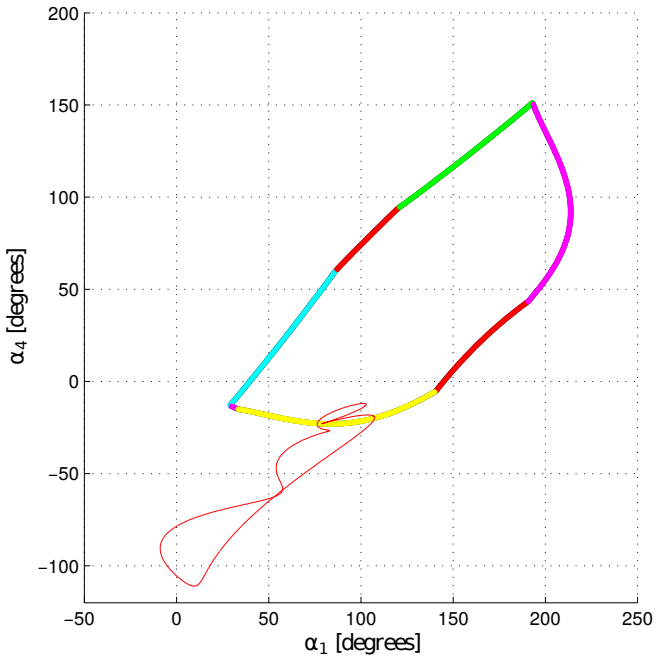


Figure 6.14. Failed optimized trajectory (thin red) relative to the joint-space of the RePlaLink.

In contrast, the global optimization direct search *patternsearch* function returned identification trajectories within the joint-space. Various iterations were carried to find the fundamental frequency and the number of harmonics for the Fourier series. From this tests it was found that a fundamental frequency of 0.45 Hz and 5 harmonics resulted in the lowest condition number of 62.21, with 2.0944 rad and 1.0472 rad as initial positions and the initial values for the amplitudes:

Table 6.3. Initial values of the amplitudes for the Fourier series

n_h	Act. 1		Act. 2	
	C_{1i} [m]	D_{1i} [m]	C_{2i} [m]	D_{2i} [m]
1	0.4533	0.3802	0.4152	0.4263
2	0.4334	0.3760	0.3869	0.4091
3	0.3349	0.3574	0.3470	0.1442
4	0.1488	0.2744	0.3085	0.1124
5	0.1128	0.1152	0.1391	0.0474

In this way, the trajectory for identification in the joint-space of Fig. 6.15 was obtained. It can be seen that the trajectory, thin red line, is within the joint-space.

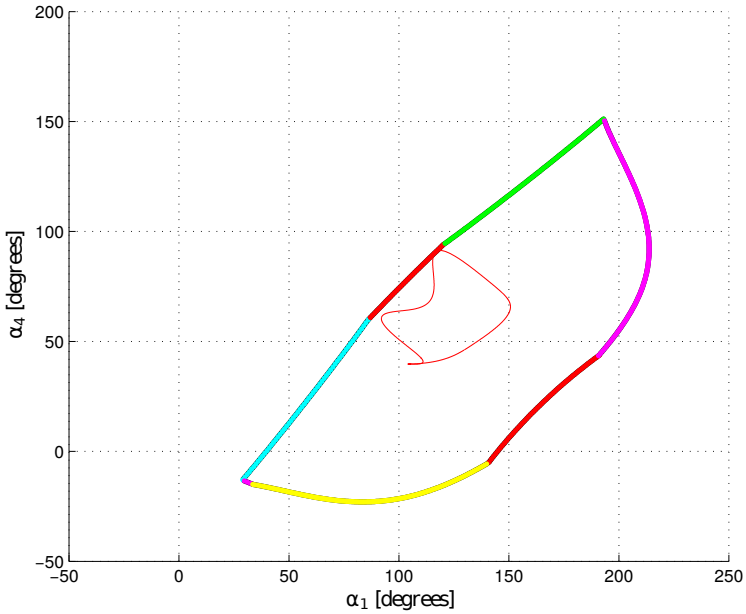


Figure 6.15. Optimized identification trajectory (thin red) relative to the joint-space of the RePlaLink.

Moreover, the position, velocity and acceleration of the actuated joints of the optimized trajectory are shown in Fig. 6.16, where it can be appreciated that the velocities and accelerations are within the established limits.

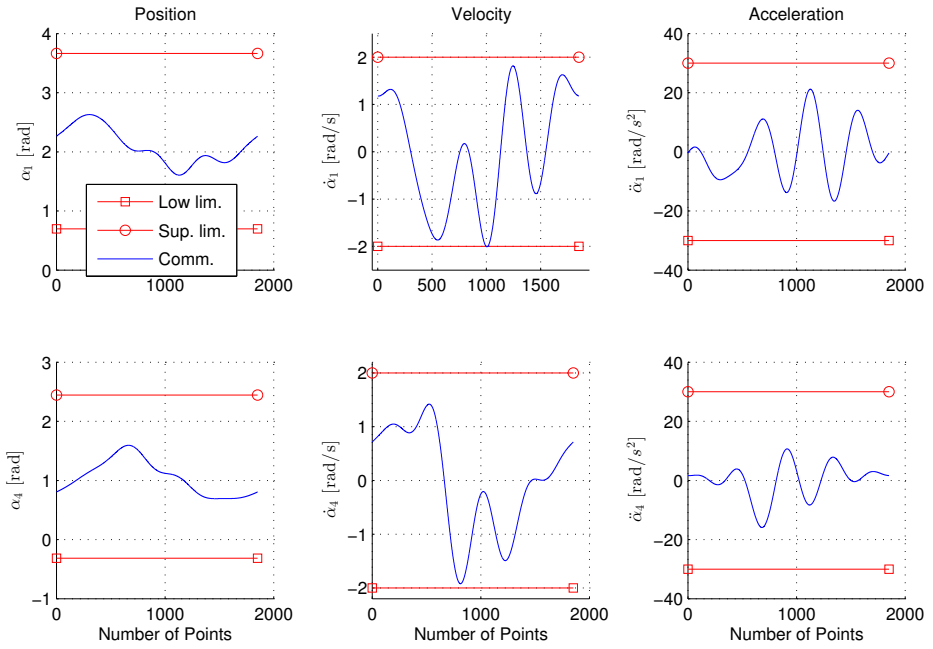


Figure 6.16. Position, velocity and acceleration of the optimized trajectory for identification.

6.2.5 Identification of the base parameters

The identification trajectory was then executed on the system. The B&R automation studio was used for the control with a period of 1.2 ms. The position and the velocity were measured using the encoders of the motors, and the acceleration was calculated from the velocity signal using numerical differentiation. As for the torque, it was obtained directly from the control

application, which is calculated from the current using the torque constant $K_t = 1.08\,\mathrm{N\,m/A}$ of the motors 8LSA45.E1045D200-10 and the gearbox 8GF60-064hh010klmm with gear ratio $i_r = 10$ by B&R. A third order Butterworth IIR low-pass filter with a 20 Hz cut-off frequency was used to filter the acceleration and torques.

After executing the identification trajectory in Matlab, a total of 9 base parameters will be identified, which are shown in equations 6.47 to 6.55. It can be noted from equations 6.47 to 6.50 that, although inertial parameters appear, they are negligible. Therefore, the contributions on these parameters are due to Coulomb and viscous frictions.

$$\begin{aligned}\Phi_{base}(1) = {} & F_{c_1} + 7.4\ 10^{-16}\ I_1 + \ 1.0\ 10^{-15}\ I_2 \\ & - 6.0\ 10^{-15}\ I_3 - 2.7\ 10^{-15}\ I_4 + 7.6\ 10^{-16}\ I_6 - 2.7\ 10^{-15}\ I_7 \end{aligned} \quad (6.47)$$

$$\begin{aligned}\Phi_{base}(2) = {} & F_{c_4} + 7.6\ 10^{-16}\, I_1 - 4.5\ 10^{-16}\, I_2 \\ & - 6.1\ 10^{-15}\, I_3 - 4.5\ 10^{-15}\, I_4 + 6.4\ 10^{-16}\, I_6 - 4.3\ 10^{-15}\, I_7 \end{aligned} \quad (6.48)$$

$$\begin{aligned}\Phi_{base}(3) = {} & c_1 - 2.4\ 10^{-16}\, I_1 + 8.8\ 10^{-16}\, I_2 - 2.8\ 10^{-15}\, I_3 \\ & - 1.7\ 10^{-15}\, I_4 - 1.6\ 10^{-16}\, I_6 - 1.6\ 10^{-15}\, I_7 \end{aligned} \quad (6.49)$$

$$\begin{aligned}\Phi_{base}(4) = {} & c_4 + 4.4\ 10^{-16}\, I_1 - 1.4\ 10^{-15}\, I_2 + 4.7\ 10^{-15}\, I_3 \\ & + 5.3\ 10^{-16}\, I_4 + 2.4\ 10^{-16}\, I_6 + 6.2\ 10^{-16}\, I_7 \end{aligned} \quad (6.50)$$

$$\begin{aligned}\Phi_{base}(5) = {} & 27.686\, I_1 + 13.533\, I_2 + 0.51015\, I_3 - 2.0\ 10^{-13}\, I_4 \\ & + 27.686\, I_6 - 2.1\ 10^{-13}\, I_7 + m_1 \end{aligned} \quad (6.51)$$

$$\begin{aligned}\Phi_{base}(6) = {} & 7.2328\, I_3 - 6.8439\, I_2 - 2.1\ 10^{-14}\, I_1 + 9.8\ 10^{-14}\, I_4 \\ & - 2.0\ 10^{-14}\, I_6 + 9.8\ 10^{-14}\, I_7 + m_2 \end{aligned} \quad (6.52)$$

$$\Phi_{base}(7) = 4.8\ 10^{-14}\, I_1 - 3.2\ 10^{-14}\, I_2 - 16.299\, I_3 - 1.4\ 10^{-13}\, I_4 + 4.7\ 10^{-14}\, I_6 - 1.4\ 10^{-13}\, I_7 + m_3 \quad (6.53)$$

$$\Phi_{base}(8) = 66.007\, I_3 - 2.2\ 10^{-13}\, I_2 - 4.4\ 10^{-13}\, I_1 + 35.225\, I_4 - 4.4\ 10^{-13}\, I_6 + 35.225\, I_7 + m_4 \quad (6.54)$$

$$\Phi_{base}(9) = 4.43\, I_2 - 4.2\ 10^{-15}\, I_1 - 0.27833\, I_3 - 2.3\ 10^{-14}\, I_4 - 4.6\ 10^{-15}\, I_6 - 2.4\ 10^{-14}\, I_7 + m_5 \quad (6.55)$$

After solving the system of equations, the values of the base parameters in equations 6.47-6.55 were identified to be the followings:

$$\Phi_{base}(1) = 0.9342 \quad (6.56)$$
$$\Phi_{base}(2) = 0.1672 \quad (6.57)$$
$$\Phi_{base}(3) = 2.6334 \quad (6.58)$$
$$\Phi_{base}(4) = 4.7132 \quad (6.59)$$
$$\Phi_{base}(5) = -3.2871 \quad (6.60)$$
$$\Phi_{base}(6) = 7.1563 \quad (6.61)$$
$$\Phi_{base}(7) = -18.0339 \quad (6.62)$$
$$\Phi_{base}(8) = 54.2796 \quad (6.63)$$
$$\Phi_{base}(9) = 6.3547 \quad (6.64)$$

6.2.6 Identification of the individual parameters

In the dynamic model of the $5R$ haptic mechanism, there are a total of 15 dynamic parameters. Among these, the 10 parameters in Eq. 6.65 were considered as the *trusted parameters* since they were measured or obtained from CAD.

$$\mathbf{\Phi}_t = \left\{ m_1, m_2, m_3, m_4, I_1, I_2, I_3, I_4, I_6, I_7 \right\}^T \quad (6.65)$$

Hence, up to 210 possible combinations were found using Eq. 6.66. Furthermore, a total of 64 combinations were used as the remaining resulted in a rank deficient matrix $\mathbf{C}$, so they were discarded.

$$N_{comb} = \frac{N_t!}{N_x!(N_t - N_x)!} = \frac{10!}{6!(4!)} = 210 \tag{6.66}$$

To estimate the parameters values, two approaches were followed in the case of the haptic mechanism, the mean of the value ranges obtained from the combinations, or an optimization with the *mean squared error* (MSE) between the experimental torques and the estimated torque as the objective function. In Table 6.4, each obtained parameter value is presented. The mean values, standard deviation and the result of the optimization are included and compared with the theoretical values.

Table 6.4. Parameters values obtained with the identification

Parameter	Mean	Std.	Opt.	Theoretical	Units
m_1	3.2150	0.0000	3.2150	3.2150	kg
m_2	15.7692	49.8521	7.4105	4.2540	kg
m_3	1.8900	0.0000	1.8900	1.8900	kg
m_4	47.9074	280.5484	2.9500	2.9500	kg
m_5	5.7803	2.2932	0.0000	0.4000	kg
I_1	0.1257	0.0000	0.1257	0.1257	kgm^2
I_2	0.4938	0.1899	0.8019	0.4062	kgm^2
I_3	0.6796	0.4937	0.0908	0.0783	kgm^2
I_4	1.3823	7.9645	0.1060	0.1060	kgm^2
I_6	0.0041	0.0000	0.0041	0.0041	kgm^2
I_7	1.2804	7.9645	0.0041	0.0041	kgm^2
F_{c_1}	0.9342	0.0000	0.9342	-	N m
F_{c_4}	0.1672	0.0000	0.1672	-	N m
c_1	2.6334	0.0000	2.6334	-	N m s/rad
c_4	4.7132	0.0000	4.7132	-	N m s/rad

It can be seen that, for some parameters, their mean values present large standard deviation, such as m_2 and m_4. The mean of mass m_2 was found to be approximately twice the theoretical value, while the mean of m_4 is approximately 16 larger than its theoretical value. It can also be appreciated how the results are improved with the optimization. For example, the inertia I_7 obtained with the optimization equals its theoretical value, whereas its mean is approximately 300 times larger. However, as for the inertia I_2, the optimization returned twice the theoretical values, which is worst than its mean value.

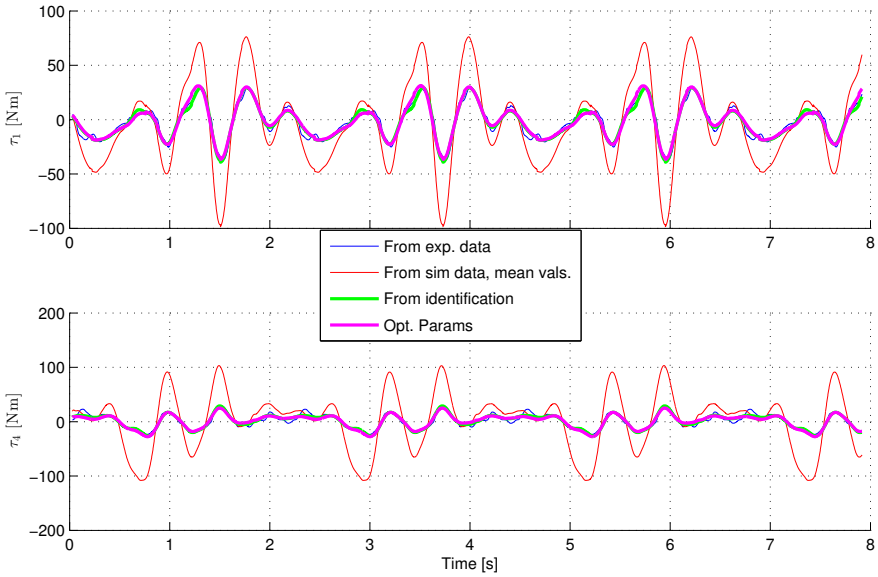


Figure 6.17. Experimental and identified model with optimization torque signals for both actuators.

In Fig. 6.17, the measured and simulated torques at the output of the gearbox are shown. It can be appreciated that when the mean values were used in the dynamic model, it was able to follow the *shape* of the experi-

mental data although substantial deviations in amplitude of about 63.5 N m from a measured torque of 34.8 N m are present. Nevertheless, with the optimized parameters, the updated model was able to closely follow the experimental torque, as well as the identified model with base parameters.

Despite having found that set of parameters values that appear to model the system, a experimental validation has to be carried out in order to asses if, with the obtained parameters values, it is possible to predict the torque when a different trajectory is executed.

6.2.7 Experimental validation

A trajectory simulating a *four-bar mechanism* was programmed with a 1.2 ms control period. In Fig. 6.18 the commanded and executed signal are presented for both actuators. It can be seen that, the experimental signal presents a good tracking with respect the commanded signals, and no significant distortions or time offsets are present.

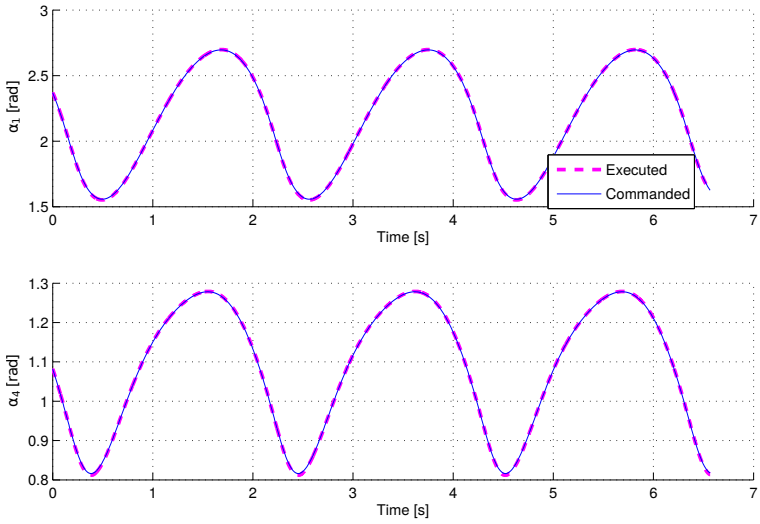


Figure 6.18. Commanded validation trajectory for both actuators.

Also, the acceleration signal was calculated from the experimental velocity signal with a numerical derivation. Afterwards, the acceleration signals and torque were filtered with a low pass filter as carried out previously. Finally, the position, velocity and acceleration were used to construct matrix $\mathbf{K}$ and the identified parameters were used as in Eq. 6.44 to calculate the torque in the motors. In Fig. 6.19, both, the experimental and simulated torques at the output of the gearbox are shown for both actuators.

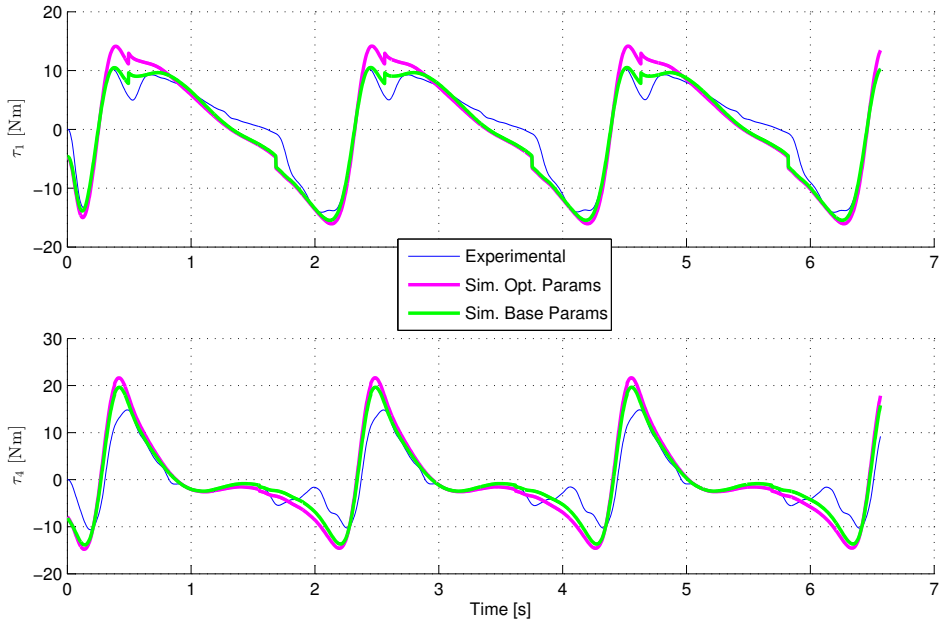


Figure 6.19. Experimental and simulated torque for the actuators for the validation trajectory.

It is observed that the torque calculated using the parameters obtained with the optimization, *Sim. Opt. Params.*, is very similar to the one registered in the experimental signal, although differences are still present. A maximum error of approximately 7 Nm was observed for the first actuator

compared to a measured torque of 4.9 Nm, whereas an error of 8.75 Nm is appreciated for the second compared to a measured torque of 12.8 Nm. As for the simulation with the base parameters, *Sim. Base Params.*, it is observed that less errors are obtained. For the first actuator, a maximum error of 4.2 Nm compared to a measured torque of 4.9 Nm was obtained, while a maximum error of 6.7 Nm compared to a measured torque of 12.8 Nm is observed for the second actuator.

6.3 Conclusions

In this chapter, a 2-PRR and a 5R haptic mechanisms were used as case studies of the identification of rigid body and friction parameters. The rigid body model of both mechanisms was obtained using the principle of energy equivalence as described in section 5.2.1. On the other hand, the total friction of the mechanisms was considered at the actuators, for which a simple friction model with Coulomb and viscous effects was used.

The resulting rigid body and friction models were assembled together to obtain a model linear in the unknown dynamic parameters. In that way, a linear system can be built that can be used for the identification of the unknown dynamic parameters, such as masses, inertias, and the friction model parameters.

Moreover, it was mentioned that, as the proper identification of the parameters depends on the trajectory used, then a trajectory optimization was performed in order to minimize the condition number of matrix $\mathbf{W}_{base}$. To that end, the joint-space trajectory was parameterized as Fourier series, for which the optimum amplitudes and initial positions of the actuators should be found. Two optimization methods were employed in this regard and their results were further compared, the *fmincon* local optimization, and the *patternsearch* global optimization from *Matlab*. In the case of the 2PRR mechanism, with the *fmincon* function resulted a condition number of 28.23, while a condition number of 20.78 was obtained with the *patternsearch* function. The difference obtained between both meth-

ods was not significant. However, the *patternsearch* method was used as it resulted in a lower condition number. As for the 5R haptic mechanism, the *fmincon* failed at finding a suitable trajectory for identification within the joint-space limits of the mechanism. It is believed that the constrained joint-space of the mechanism results in a difficulty for the optimization algorithm to find a minimum value in the direction of search. In contrast, with the *patternsearch* algorithm, an identification trajectory was found within the joint-space limits of the mechanism with a condition number of 62.2.

Then, the trajectories were executed on the mechanisms and the identification procedure was used to obtain the base parameters, 8 parameters for the 2PRR mechanism, and 9 in the case of the 5R haptic mechanism. Also, in sections 6.1.6 and 6.2.6 a procedure was carried out to find the values of the unknown parameters based on the *trusted parameters* of the system.

In the case of the 2-PRR, it was found that the most significant errors were obtained for the masses and inertias of the bars with approximately 220 % difference, while the equivalent masses of the actuators were found with a difference of 20 % with respect to their theoretical value.

As for the haptic mechanism case, the differences between the mean and the theoretical values for some of the parameters were found even larger than in the case of the 2-PRR mechanism. As for example, the mass m_4 has a deviation of approximately 1500 %. Additionally, large standard deviations were obtained for some parameters of the haptic mechanism. This indicates that the parameters values vary significantly with every feasible combination.

Hence, it could be assumed that a more accurate value may be found within the minimum and maximum values obtained with the combinations, for which an optimization procedure was used to minimize the mean squared error between the simulated and experimental torques. After it was used,

the optimization resulted in more precise estimations of the parameters values with respect to their theoretical values.

Nevertheless, important differences were still appreciated in the identified parameters for both mechanisms. A source of errors is assumed to be from the condition number of matrix $\mathbf{W}_{base}$ for which the base parameters are calculated. However, an optimization was already used to minimize this condition number as it was mentioned before. Then, it is possible that the coupling of the kinematic chains and the available joint-space is limiting the excitation of the unknown dynamic parameters, thus making more difficult the identification of the parameters. Therefore, identification procedures and techniques for *lower mobility mechanisms* should be further investigated but are beyond the scope of this work. Another possibility is that the values obtained correspond to a local minimum due to the optimization algorithm used. Hence, it is also possible to improve the parameters values results if global optimization algorithms are used instead, but this was not addressed in this work.

After the identification of the parameters, a validation step was done in both case studies to evaluate how the dynamic models with the identified parameters deviate from the dynamics of the actual systems. In this regard, a different trajectory was executed, and the resulting torques from the simulation and the experiments were compared. It was found that for the 2-$\underline{P}$RR mechanism, a maximum difference of 40 N can be appreciated. On the other hand, for the haptic mechanism a maximum error of 9 N m was observed. A cause for these differences are due to errors in the identified parameters, which has been already discussed. Another source of errors may be due to unmodeled dynamics or simplifications in the models as in the case of the haptic mechanisms, where the third actuator was not considered. However, despite the differences observed, the models can be used as a first approximation for simulation of the dynamic behavior of the mechanisms.

Part IV

Conclusions and Future Research

Chapter 7

Contributions, Conclusions and Future Research

Troughtout this thesis, a method for the mechatronic analysis of parallel kinematic mechanisms has been developed. First, in chapter 2, a review of the state of the art was conducted over several topics regarding mechatronics, motion control, parallel kinematics machines, dynamic and mechatronic modeling, friction and their parameters identification.

In chapter 3, it is proposed a method to develop a mechatronic model to be used for the mechatronic analysis of parallel kinematic mechanisms, such as pick & place mechanisms or Multi-Axial Shaking Tables (MAST). Furthermore, in chapter 4 the proposed method was applied to two case studies, a 5R planar parallel manipulator and a 2PRU-1PRS spatial MAST mechanism.

The experimental identification of dynamic parameters of planar mechanisms was addressed in chapter 5. In this regard, two additional case studies were used in chapter 6 for the experimental validation of the identification procedure, a 2-$\underline{P}$RR and the RePlaLink mechanism.

7.1 Contributions and conclusions

In view of the state of the art and after applying the proposed methodologies for the mechatronic analysis of parallel kinematics manipulators, the contributions and conclusions of this thesis are summarized as follows:

- It is proposed a methodology to obtain a mechatronic model of parallel kinematic mechanisms considering the actuators, the control and the manipulator. A remarkable contribution of the approach taken is that the manipulator is decoupled from the actuation system and modeled as a disturbance to the actuators. In this way, alternative designs of the mechanism, actuator and control, can be easily evaluated.

- With the mechatronic models obtained using the proposed methodology, the performance of parallel kinematics mechanism can be assessed in the time and frequency domains, evaluating the motor torque, position and velocity, as well as the bandwidth of the manipulator.

- The bandwidth of the manipulator can be used to determine if its design is appropriate for a specific application, such as Multi-Axial Shaking Tables. This characteristic can be easily found for an actuator since nonlinearities are not present, as its dynamic parameters remain constant. However, nonlinearities exist in parallel kinematics manipulators due to their kinematics and workspace position dependent inertial, centrifugal, Coriolis and gravitational terms. Therefore, in this work an estimation of the bandwidth of parallel manipulators is made by limiting the amplitudes of the commanded signals, thus reducing the effects of the nonlinearities of the kinematics and dynamics. Furthermore, to obtain the bandwidth of the parallel manipulator, sinusoidal motions in the workspace were commanded, and the time delay and amplitude of the response of the actuators were measured and compared with the commands.

- The resulting mechatronic model using the approach of this work, can also be used to evaluate the effect of a parallel mechanism on the actuation system, thus allowing to obtain important information for the proper selection of the actuators and its components, such as motor, gearbox and linear guides.

- A modeling procedure of electromechanical actuators was described, for which the electrical dynamics were considered and lumped parameters models of 1, 2 or higher degrees of freedom were used for the mechanical dynamics. In that way, inertias, flexibilities, stiffness and damping can be also taken into account.

- It was shown how the most common control alternatives of joint-space, workspace and computed torque control can be integrated into the mechatronic model, thus allowing to evaluate several control strategies that can assist in determining the most suitable control for a specific application.

- It was described how to apply the principle of energy equivalence to obtain expressions of the dynamics of parallel mechanisms. Components with planar motion and the translation of the end-effector of a mechanism, are analyzed using Lagrange's equations in standard form. On the other hand, the spatial rotation dynamics of components are analyzed using quasi-coordinates and Boltzmann-Hamel equations, for which a particularization was given. With this approach, the expression of the dynamics are obtained in a systematic manner, which is less error prone rather than trying to obtain the dynamic model for a specific mechanism.

- It was found that by separating the motion of the center of mass from the spatial rotation motion of a component, its dynamics can be analyzed more effectively. On one hand, complex trigonometric expressions appear when Lagrange's equations are applied to analyze the spatial rotation motion. On the other hand, if Boltzmann-Hamel

equations are applied to analyze the motion of the center of mass, also unnecessary complex expressions appear.

- Two case studies were used to validate the proposed mechatronic modeling procedure, a 5R planar parallel mechanism and a 2PRU-1PRS MAST mechanism, for which a comparison between the simulation of their mechatronic models and experimental signals was carried out. It was shown for both cases that the mechatronic models were capable of simulating the performance of the actual mechanisms. Additionally, three main sources of deviations were identified, the controllers, modeling errors and difference in the dynamic parameters values with respect to the actual ones.

- An identification procedure of the dynamic parameters of planar parallel mechanisms was proposed to adjust the parameters used in the mechatronic models. To that end and as a contribution, the principle of energy equivalence was again applied to obtain a dynamic model for identification of planar mechanisms, considering the rigid body and Coulomb + viscous friction model parameters. To the best of the author knowledge, the approach taken has not been addressed before. Additionally, the procedure to obtain the dynamic model for identification was described, and also the procedure to obtain the base parameters of a rank deficient linear system dynamic model was given. Furthermore, a method to identify the dynamic parameters separately was described. This method uses the a priori known values of some dynamic parameters to solve for the unknown ones in the base parameters. Finally, the identification procedure was applied on two additional case studies.

- To obtain the identification trajectory for the mechanisms, two optimizations methods were applied, the local optimization method *fmincon* and the global optimization method *patternsearch*. The resulting condition number of matrix $\mathbf{W}_{base}$ was used as a criterion to compare these two methods. No major differences resulted when the *fmincon* and the *patternsearch* were applied for the 2-PRR mechanism. How-

ever, it was found that the local optimization function *fmincon* failed at finding an optimal trajectory within the joint-space limits. In contrast, an identification trajectory was found with the *patternsearch* method. It is believed that joint-space limits may be limiting the excitation of the dynamic parameters, and that the *fmincon* function is falling into a local minimum.

- The values of the dynamic parameters of the 2-PRR and the RePlaLink mechanisms were estimated. Moreover, the adjusted dynamic model were simulated and compared to the experimental signals obtained using a validation trajectory. On the one hand, the dynamic model of the 2-PRR mechanism with the mean value of the dynamic parameters performed similar to the experimental signals. On the other hand, it was shown for the RePlaLink that the mean value of some parameters deviated considerably and could not be use in the dynamic model of the mechanism. As a solution, an optimization was applied to find better estimates of the dynamic parameters than the mean values. As a result, it was shown for the RePlaLink that the model with the parameters obtained using the optimization performed similarly to the experimental signals.

7.2 Future research

Based on the results obtained and the work in this thesis, several areas for improvement and future research are herein commented:

- In this thesis, simple single-axis controllers have been considered as the control for the prototypes here studied. It is believed that the overall system performance should be improved by using more advanced control alternatives, specially model-based controls like the computed torque control or adaptive controllers. Furthermore, it can be noted that the dynamics obtained using the principle of energy equivalence can result in good candidates to be implemented for dy-

namic compensation in control. These subjects should be investigated in future research.

- The mechatronic model here obtained can be used to evaluate alternative configurations of actuators components. It would result convenient to introduce the mechatronic model in an automatic procedure to choose the optimal configuration of the actuators for a given parallel kinematic mechanisms with specific performance characteristics in terms of bandwidth and precision.

- A simple friction model has been here considered. However, more complex models, such as the *Generalized Maxwell-Slip* friction model, can be used to further evaluate if friction can be modeled in detail with improvements in the performance of mechatronic models.

- In this thesis, a modal test has been used to obtain the parameters of two degrees of freedom model of the actuators. However, it should be investigated how to identify such parameters using the actuators motions and signals.

Part V

Appendices

Appendix A

Real-Time Position Controller Development

A custom position controller software has been developed in the framework of this thesis. This development has been carried out to provide a flexible and scalable controller required for the experimentation on this thesis which also can be used for future developments and testing. This controller is based on existing hardware platforms for scientific and engineering applications by National Instruments® with real-time capabilities.

A.1 Controller purpose and requirements

In this thesis, the controller named "Lynx" was designed and programmed to offer the following functions.

- A user-friendly interface to process user inputs.
- Loading of a programmed trajectory by the user.

- Defining homing settings.
- Controlled execution of a *homing trajectory*.
- Controlled execution of a programmed trajectory.
- Data acquisition of relevant information of the motors for later analysis.

A.2 Controller design

A client-server architecture was chosen to provide a clear separation between priority software processes, which must execute in a deterministic manner, as a difference to those processes not requiring such constraints. Fig. A.1 represents the architecture of the controller system.

As it can be appreciated, the controller has two layers. On one hand, the "high level controller" is in charge of the user-interface in the PC, processing the user inputs accordingly and presenting the visualization of important information. Then, the PC communicates with the PXI which is in charge of the deterministic tasks of the system such as position control and data acquisition of the actual position, velocity and current from the motors. The communication between these two is done through TCP/IP protocol.

On the other hand, the "low-level controller" represented by the EPOS, is in charge of directly controlling the motors by regulating their velocities to the velocity commands sent by the PXI through the CANopen protocol. Also the EPOS are in charge of regulating the current supplied to the motors at a fixed DC voltage with the current control loop.

The client-server architecture consists of two software distributed applications, which communicate to each other using the TCP/IP mentioned previously. The *client* requests specific *services* or functions to the *server*,

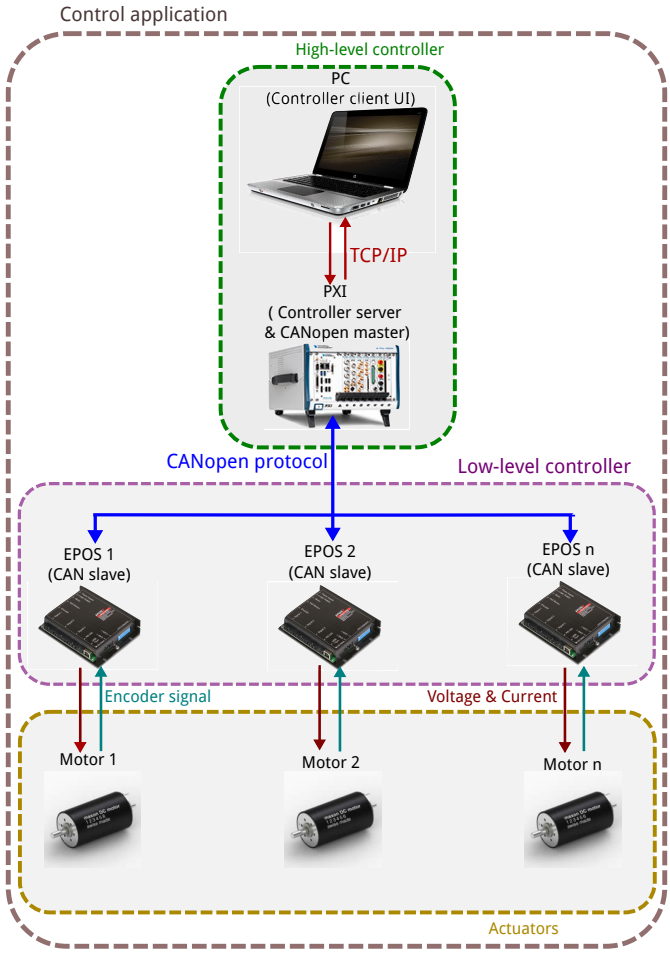


Figure A.1. Controller architecture.

such as moving a motor, configuring the controller parameters or performing a *homing* task. On the other hand, the server processes the client's requests and performs its functions accordingly, while communicating to the client important information, such as the *state* of the motors. Moreover, the implementation of either the client as well as the server software,

is done in this work following a state machine model [102].

A.2.1 State machine architecture

A software application can be represented by a machine metaphor. In this way, it can be thought that a particular situation and conditions define a *machine state*. Moreover, a machine can have a set of finite states which describe different situations and conditions. It is said to be *finite* because the range of states in which the machine can be in is limited and clearly differentiated. As an example, let us consider the simple electrical circuit in Fig. A.2 whose purpose is to light a bulb. The whole circuit is regarded as the machine and it can be seen that two states are possible, *on* and *off*. The transition between the two states is triggered by the input to the switch.

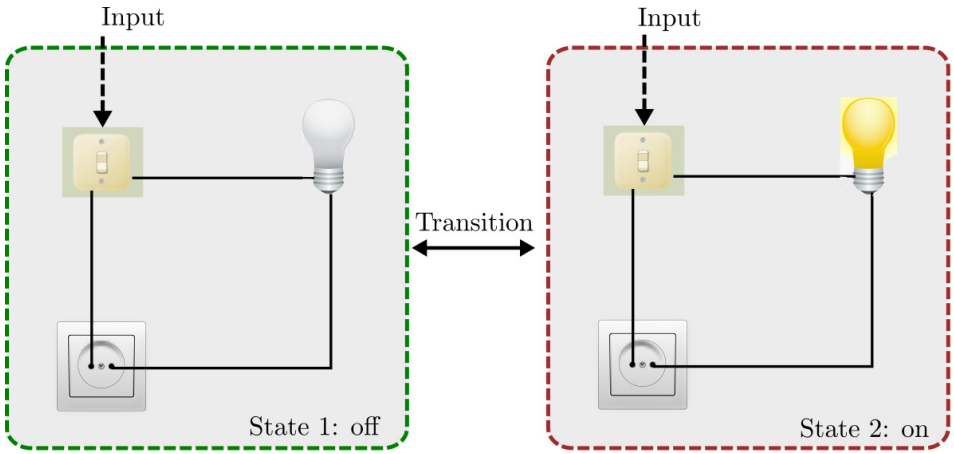


Figure A.2. Light bulb circuit representation of a two state machine.

In the control application, several state machines have been employed to program specific functionalities, such as *initialization*, *operation* and *shutdown*. In Fig. A.3 a Labview code snippet with a typical state machine programming is shown.

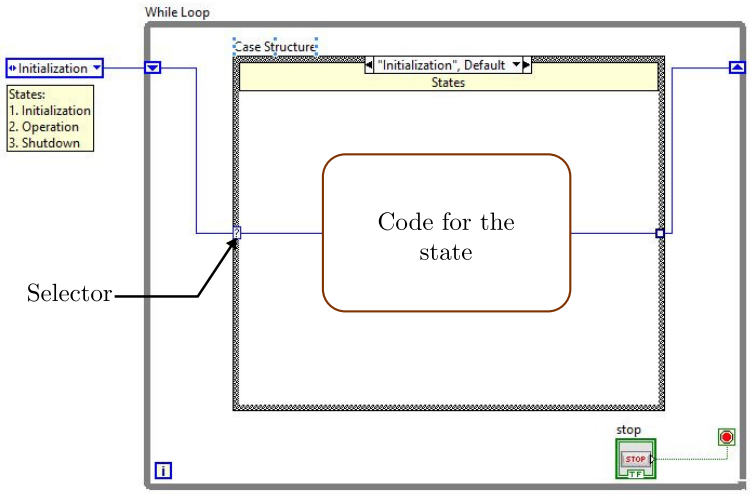


Figure A.3. State machine arquitectur code in Labview.

Additionally, a specific state in the main program may have internal states which are used to activate the *control*, *homing*, *configuration* and *jog tasks* or subprograms. These tasks handle more specific functionalities in the application, such as moving the motors to obtain a reference position with the homing task, or follow a programmed trajectory with the control task. As a result, the application is programmed in a modular way which can be easily extended.

Furthermore, each task follows a master-slave approach. In this regard, a *master* loops are used to handle non-important operations specific to the tasks, like handling requests for tasks, and supplied information to the processor loop. *Slave* loops are used to process the information supplied by the master in a previously defined manner. In Fig. A.4 a master-slave code snippet is shown, where the master loop contains an *event handler* to manage requests for the task, a *communication object* to pass commands to the slave loop, which performs the actual function of the task, such as moving a motor. In this work, the master loop of a task is used to communicate the

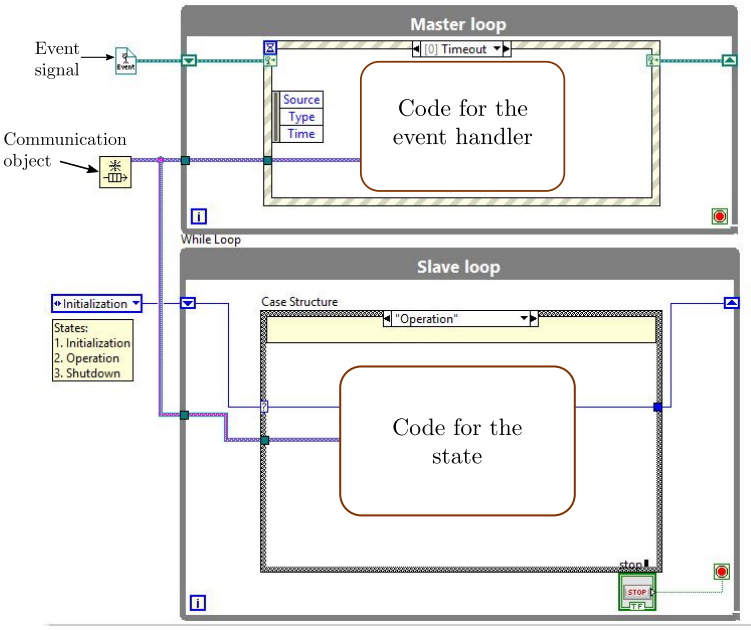


Figure A.4. Master-slave arquitecture in Labview.

activation or deactivation of the task to the main program, and the slave loops is where the task functions are programmed and processed.

A.3 User-interface controller client

The user must have a way to interact with the software to control the actions of the system. A common approach is to have a command line interface. Such interfaces usually expose many functionalities to the user. However, often times they are limited to advanced users as the command line interface lack of visualization features, which is difficult for regular users to interact with. In this thesis, the visual interface in Fig. A.5 was designed to provide a straight-forward user experience. It is divided into two main panels, the *input* and *status* panels.

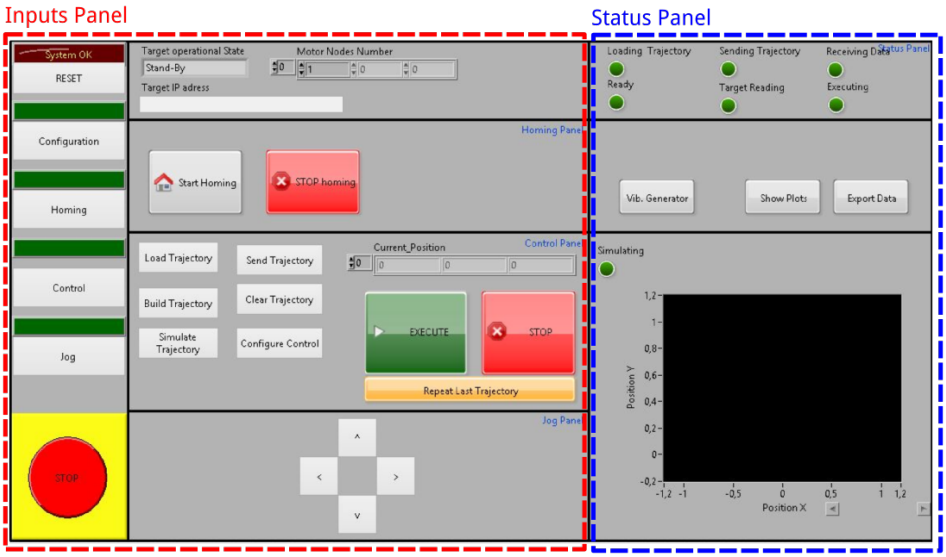


Figure A.5. Screenshot of the controller user interface.

A.3.1 The inputs panel

The inputs panel in Fig. A.6 is where user commands related to the configuring and execution of a given task is carried out. It is divided into the *tasks panel*, where the configuration, homing, control and jog tasks are selected depending of the required function, and the *commands panels* where the specific commands of a given tasks are chosen, such as start or stop the homing, and load and execute a trajectory.

Only one task is allowed to operate at a time. Hence the commands of a given task are processed if only the respective task is in an operational state, indicated by a light green *led* above the task button. The motivation behind this design is to avoid inappropriate commands for a specific situation to be sent to the PXI, which may result in unexpected behavior and possible dangerous motions.

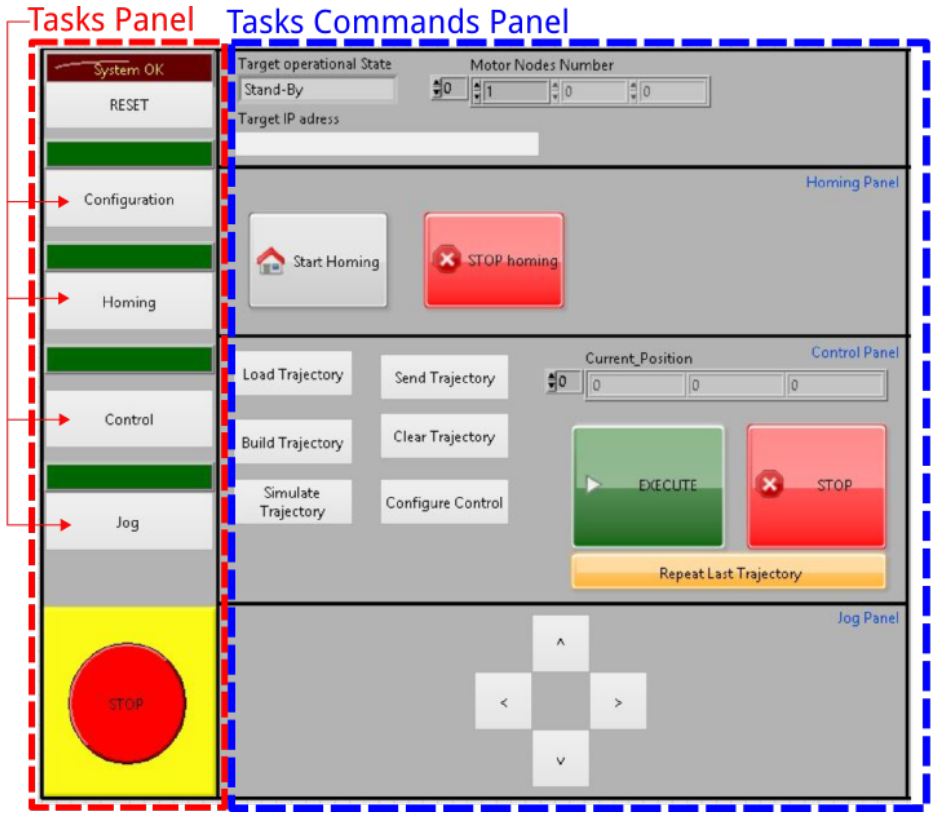


Figure A.6. Screenshot of the inputs panel.

A.3.2 The status panel

The status panel shown in Fig. A.7 is where the information concerning the state and data of the system is visualized. It is divided into three subpanels.

The indicators panel present critical information of the system in a boolean state. The data management is where the data acquired can be visualized. Lastly, the mechanism visualization panel is where a representation of the mechanism is shown. A brief description for each indicator

meaning and control function in this panel is given:

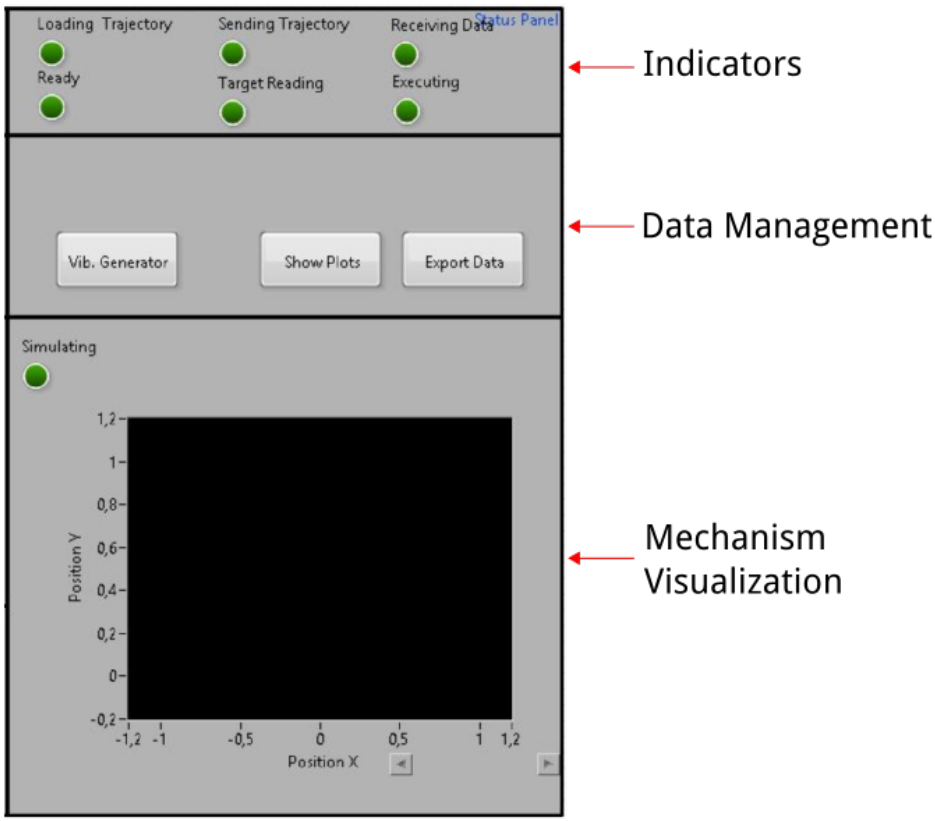


Figure A.7. Screenshot of the status panel.

- **Loading trajectory:** This led turns light green if a file with a programmed trajectory is being loaded into the controller. This is only possible when the *control task* is active.
- **Ready:** This led turns light green after the trajectory has been loaded completely.

- **Sending trajectory:** After a trajectory has been loaded, it must be downloaded into the PXI where the actual position control takes place. By pressing the "Send Trajectory" in the tasks command panel, the trajectory downloading process starts and this indicator turns on. When the client-PC pushes the last point of the trajectory in the TCP/IP data stream, hence, successful trajectory downloading, this indicators switches back off.

- **Target reading:** The trajectory data sending involves two separate asynchronous processes in the software. The first process takes place in the PC and its state is described by the "Sending Trajectory" led. The second process takes place in the PXI and it is in charge of receiving the trajectory points. Hence, this LED turns on to indicate that the target is receiving a trajectory. Upon successfully receiving all points, this LED switches back off.

- **Receiving data:** This LED indicates that motor information, such as position, velocity and current, is being received by the PC.

- **Executing:** This LED indicates that a trajectory is being executed by the position control and that the motors have power. The data management section is where additional functionalities are added to the software in the form of plug-ins. Such plug-ins are used for performing data operations such as generating, visualizing and/or exporting. The "Vib. Generator" is used for generating a trajectory for vibrations tests. The "Show Plots" is used to plot the position, velocity and current data received from the motors. Also, the "Export data" is used to export the received data from the motors to a text file for later analysis and/or processing. Lastly, the *Mechanism Visualization* is where the mechanism is plotted simultaneously as the actual prototype is moving.

A.3.3 States of the user interface application

The functioning of the user interface follows a three step process execution as shown in Fig. A.8a. In this way, upon receiving a start signal (running the application), the initialization steps is executed, then the execution enters into operation state where the main functionalities are handled, and finally the shutdown state is where a controlled exit is performed, in such a way as to set variables to a default value, turn off the EPOS and clean memory.

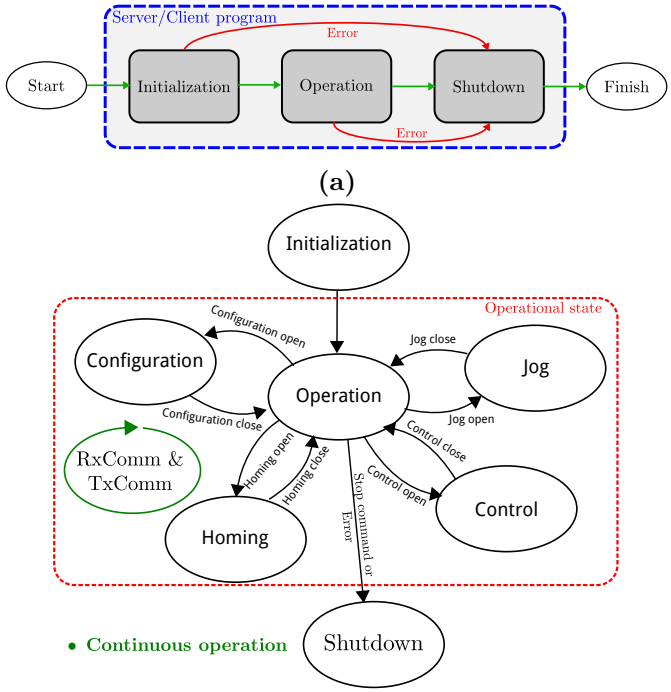


Figure A.8. (a) Operating states of the main program for the embedded controller and client application, and (b) state machine model of the user interface and the embedded controller.

The application can be modeled by the state machine shown in Fig. A.8b. There can be seen the following states: *initialization*, *operation*, *configuration*, *homing*, *jog*, *control* and *stop*. It is worth mentioning that the configuration, homing, jog and control states correspond to tasks with specific functions within the operation state.

A.3.3.1 Initialization

In the initialization state the global variables of the applications are set to their default value, such as the IP address, status panel indicators variables and the stop global variable. Additionally, communication objects, such as queues, and first-in-first-out memory buffers, FIFO's.

A.3.3.2 Operation

The operation state is used to receive and process any interface events such as indicators and buttons pressed. Depending on the user input, specific tasks, control, can be activated and task-specific commands associated to each button in the user interface are sent to their respective handlers.

A.3.4 User interface tasks

Four tasks are used to handle the main functions of the system. These functions are Configuration, Homing, Control and Jog. Furthermore, the transmit TxComm and receive RxComm communication tasks are used to handle the client-server communications. On one hand, TxComm is used to transmit data to the server, and RxComm, is used to receive the data from the server and route it accordingly.

A.4 Controller server

The PXI runs the hard real-time operating system VXworks provided by National Instruments [103]. It's purpose is to provide a deterministic operating system that guarantees that critical tasks, such as the control task

of the motors, are executed when required and within time tolerances in the microseconds range. Otherwise, accurate motions would not be guaranteed or acquisition data may be lost. Deterministic process execution implies that other unnecessary system processes are avoided, which otherwise would consume CPU time or memory. For this reason, to avoid the management of a graphical interface, which would consume system resources, the server application does not have an user interface. This functionality in handled by the client-PC.

A.4.1 The main program

The *main program* is where the functions and execution steps of the server are programmed. It is coded in a modular way for specific functionalities similarly to the user interface. Each task correspond to one task or functionality of the controller, such as the *control task*. In this way, specific functions of the controller are separated and are only executed when they are called by the main program upon receiving a command request from the client-PC, otherwise the task remain in an idle state.

Moreover, the main program is in charge of general purpose tasks such as initializations, pre-configuring, commands processing, safety operations and shutdown. In this regard, the main program is designed with the three states of execution as the user interface and shown in Fig. A.8a. Also, the server application follows the same state model diagram in Fig. A.8b. The difference between the user interface and the control server is the specific functions of the components.

A.4.1.1 Initialization

The initialization is where the pre-configuring of the server program is done and is executed once the PXI is turned on or the server program is loaded onto the PXI. In this step, the following actions are carried out:

1. *Global variables* and *functional global variables* are created and ini-

tialized to a pre-defined value.

2. Communication with the client-PC is established using the TCP/IP network interface.

3. Creation of the *FIFO's* used for the state manager and for the trajectory.

4. Creation of the events for the *Task Manager*, the *Acknowledge Opened Event* and *Acknowledge Closed Event*, which signal when a task has been activated or deactivated respectively.

5. Error check during the initialization step.

6. Close the initialization step and switch to the operation state if no errors have occurred, otherwise shutdown the server program.

7. Initialization of the CANopen interface and CANopen SDO's and PDO's objects, which are used to send data packets to the EPOS.

8. Configuration of critical digital inputs for the low level controllers such as *Quick Stop* and *Negative* and *Positive* limit switches.

A.4.1.2 Operation

The main program enters into the *Operation* state after successfully completing the *Initialization* state, otherwise it is ignored and the systems skips to *Terminate* state.

In Operation state is where the commands sent from the client-PC are processed and handled accordingly to activate or switch tasks. Also, any task-specific command is here routed to its respective task. The inner workings of the Operation state also follows the *State Machine model*.

A.4.1.3 Terminate

When the control software is turned off or a critical error occurs, the server program enters into *Terminate* state to perform a *clean* stop of all the processes or tasks in execution. In this way, tasks are safely stopped and any process is appropriately closed. Also, any memory used by these tasks and processes is freed. The following steps are taken in this state.

1. Set global variables to default values.
2. Clear any remaining data in the FIFO's and close their references.
3. Close the reference to the software *Events*.
4. Close the Client-server communication objects.
5. Shutdown the CANopen slaves and close any reference to CANopen communication objects.

A.5 Communication protocols and objects

A communication protocol has been developed to pass information between different parts of the software. The types of communications used in Lynx are the following.

1. Communication type 0 (CommType0): Used to send *open* or *close* commands to the tasks in the server, to whether activate or deactivate a task. If a specific tasks receives the command *open*, it is activated and the others are locked until the current one is deactivated once a *close* command is received. Additionally, a *special* command to signal the *shutdown* of the server is sent this way.
2. Communication type 1 (CommType1): reserved for future functionalities.

3. Communication type 2 (CommType2): This communication type is used to exchange trajectory data from the PC to the server or measurement data from the server to the PC.
4. Communication type 3 (CommType3): reserved for future functionalities.
5. Communication type 4 (CommType4): used to send task-specific commands to the server. In this way, the motion execution is triggered as in the control, jog and homing tasks.

In Fig. A.9 the communications in the control are depicted. The communication between the PC and the server are handled by the *TxComm* and *RxComm* tasks in both applications. The TxComm is in charge of transmitting the data to the RxComm object in the other end through an ethernet TCP/IP connection. Furthermore, the RxComm object is in charge of routing the communication commands and data to the appropriate handler, such as the tasks or the state manager.

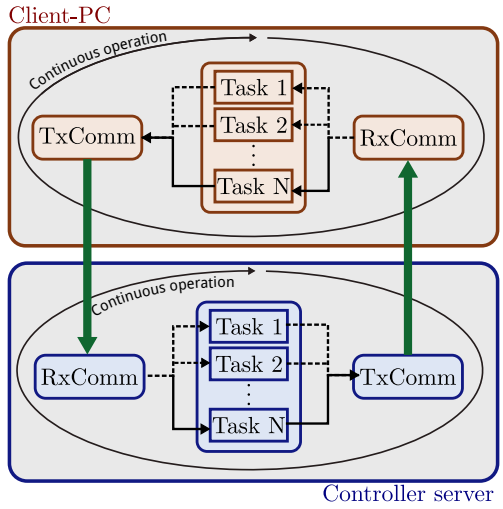


Figure A.9. Scheme of the communications between components.

Appendix B

Jacobian Matrices

B.1 Jacobian matrices for the 5R mechanisms

The following equations can be obtained from the kinematic analysis of point P:

$$\mathbf{J}_x \begin{Bmatrix} \dot{x}_p \\ \\ \dot{y}_p \end{Bmatrix} = \mathbf{J}_q \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.1}$$

Where,

$$\mathbf{J}_x = \begin{bmatrix} l_2 cos(\alpha_2) & l_2 sin(\alpha_2) \\ l_2 cos(\alpha_3) & l_2 sin(\alpha_3) \end{bmatrix} \tag{B.2}$$

$$\mathbf{J}_q = \begin{bmatrix} \mathbf{J}_{q11} & 0 \\ 0 & \mathbf{J}_{q22} \end{bmatrix} \tag{B.3}$$

$$\mathbf{J}_{q11} = l_1 l_2 cos(\alpha_1) sin(\alpha_2) - l_1 l_2 sin(\alpha_1) cos(\alpha_2) \tag{B.4}$$

$$\mathbf{J}_{q22} = l_3 l_4 sin(\alpha_3) cos(\alpha_4) - l_3 l_4 cos(\alpha_3) sin(\alpha_4) \tag{B.5}$$

Hence,

$$\begin{Bmatrix} \dot{x}_p \\ \\ \dot{y}_p \end{Bmatrix} = \mathbf{J}_x^{-1}\mathbf{J}_q \begin{Bmatrix} \alpha_1 \\ \\ \alpha_4 \end{Bmatrix} \tag{B.6}$$

Similarly, the relation between the velocity of point p and the angular velocity of the passive angles $\alpha_2,\ \alpha_3$ is written as follows:

$$\mathbf{J}_{x_{pas}} \begin{Bmatrix} \dot{x}_p \\ \\ \dot{y}_p \end{Bmatrix} = \mathbf{J}_{q_{pas}} \begin{Bmatrix} \dot{\alpha}_2 \\ \\ \dot{\alpha}_3 \end{Bmatrix} \tag{B.7}$$

Where,

$$\mathbf{J}_{x_{pas}} = \begin{bmatrix} l_1 cos(\alpha_1) & l_1 sin(\alpha_1) \\ l_4 cos(\alpha_4) & l_4 sin(\alpha_4) \end{bmatrix} \tag{B.8}$$

$$\mathbf{J}_{q_{pas}} = \begin{bmatrix} \mathbf{J}_{q_{pas_{11}}} & 0 \\ 0 & \mathbf{J}_{q_{pas_{22}}} \end{bmatrix} \tag{B.9}$$

$$\mathbf{J}_{q_{pas_{11}}} = l_1 l_2 sin(\alpha_1) cos(\alpha_2) - l_1 l_2 cos(\alpha_1) sin(\alpha_2) \tag{B.10}$$

$$\mathbf{J}_{q_{pas_{22}}} = l_3 l_4 cos(\alpha_3) sin(\alpha_4) - l_3 l_4 sin(\alpha_3) cos(\alpha_4) \tag{B.11}$$

Hence,

$$\begin{Bmatrix} \dot{\alpha}_2 \\ \\ \dot{\alpha}_3 \end{Bmatrix} = \mathbf{J}_{q_{pas}}^{-1} \mathbf{J}_{x_{pas}} \begin{Bmatrix} \dot{x}_p \\ \\ \dot{y}_p \end{Bmatrix} \tag{B.12}$$

Substituting Eq. B.6 into Eq. B.12, the expression of the angular velocity of the passive angles with respect to the active angles $\alpha_1,\ \alpha_4$ results as follows:

$$\begin{Bmatrix} \dot{\alpha}_2 \\ \\ \dot{\alpha}_3 \end{Bmatrix} = \mathbf{J}_{q_{pas}}^{-1} \mathbf{J}_{x_{pas}} \mathbf{J}_x^{-1} \mathbf{J}_q \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} = \mathbf{J}_{pas} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.13}$$

B.1.0.4 Solid 1

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\alpha}_1 \end{Bmatrix} = \mathbf{J}_1 \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} = \begin{bmatrix} -\frac{l_1}{2} sin(\alpha_1) & 0 \\ \frac{l_1}{2} cos(\alpha_1) & 0 \\ 1 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.14}$$

B.1.0.5 Solid 2

$$\begin{Bmatrix} \dot{x}_2 \\ \dot{y}_2 \\ \dot{\alpha}_2 \end{Bmatrix} = \mathbf{J}_2 \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix}$$

$$= \begin{bmatrix} -l_1 sin(\alpha_1) - \frac{l_2}{2} sin(\alpha_2)\mathbf{J}_{pas_{11}} & -\frac{l_2}{2} sin(\alpha_2)\mathbf{J}_{pas_{12}} \\ l_1 cos(\alpha_1) + \frac{l_2}{2} cos(\alpha_2)\mathbf{J}_{pass_{11}} & \frac{l_2}{2} cos(\alpha_2)\mathbf{J}_{pass_{12}} \\ \mathbf{J}_{pas_{11}} & \mathbf{J}_{pas_{12}} \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.15}$$

B.1.0.6 Solid 3

$$\begin{Bmatrix} \dot{x}_3 \\ \dot{y}_3 \\ \dot{\alpha}_3 \end{Bmatrix} = \mathbf{J}_3 \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix}$$

$$= \begin{bmatrix} -\frac{l_3}{2} sin(\alpha_3)\mathbf{J}_{pas_{21}} & -l_4 sin(\alpha_4) - \frac{l_3}{2} sin(\alpha_3)\mathbf{J}_{pas_{22}} \\ \frac{l_3}{2} cos(\alpha_3)\mathbf{J}_{pass_{21}} & l_4 cos(\alpha_4) + \frac{l_3}{2} cos(\alpha_3)\mathbf{J}_{pass_{22}} \\ \mathbf{J}_{pas_{21}} & \mathbf{J}_{pas_{22}} \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.16}$$

B.1.0.7 Solid 4

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\alpha}_1 \end{Bmatrix} = \mathbf{J}_1 \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} = \begin{bmatrix} 0 & -\frac{l_4}{2} sin(\alpha_4) \\ 0 & \frac{l_4}{2} cos(\alpha_4) \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \dot{\alpha}_1 \\ \\ \dot{\alpha}_4 \end{Bmatrix} \tag{B.17}$$

To obtain the passive angles $\alpha_3,\ \alpha_4$, two circles centered at point A and B respectively, are used to obtain the two possible solutions of y_p, which

are the intersection points of these two circles. To that end, the equations of these two circles are written as follows:

$$(x_p - x_A)^2 + (y_p - y_A)^2 = l_2^2 \tag{B.18}$$

$$(x_p - x_B)^2 + (y_p - y_B)^2 = l_3^2 \tag{B.19}$$

Expanding both equations and subtracting B.18 by B.19, the expression of x_p results as in Eq. B.20.

$$x_p = \frac{l_2^2 - l_3^2 - 2y_p(y_B - y_A) - y_A^2 + y_B^2 - x_A^2 + x_B^2}{2(x_B - x_A)} \tag{B.20}$$

Eq. B.20 can be substituted into Eq. B.18 to find y_p, hence obtaining the the following expression:

$$\phi y_p^2 + \beta y_p + \gamma = 0 \tag{B.21}$$

Where,

$$\phi = 1 - \left(\frac{y_A - y_B}{x_B - x_A}\right)^2 \tag{B.22}$$

$$\beta = \frac{(y_A - y_B)}{(x_B - x_A)^2} M - 2x_A \frac{(y_A - y_B)}{(x_B - x_A)} - 2y_A \tag{B.23}$$

$$\gamma = \frac{M^2}{4(x_B - x_A)^2} - \frac{x_A}{(x_B - x_A)} M + y_A^2 + x_A^2 - l_2^2 \tag{B.24}$$

The passive angles are then obtained by using the dot product between $\overleftrightarrow{AP}$ and the unit vector $\hat{i}$ in the x direction for α_2 as in Eq. B.25, and between $\overleftrightarrow{BP}$ and $\hat{i}$ for α_3 as in Eq. B.26. Also, y_p is defined to be the solution when Eq. B.27 is fulfilled.

$$\alpha_2 = arcos(\frac{\overrightarrow{AP} \cdot \hat{i}}{l_2}) \tag{B.25}$$

$$\alpha_3 = arcos(\frac{\overrightarrow{BP} \cdot \hat{i}}{l_3}) \tag{B.26}$$

$$sign(\overrightarrow{BP} \times \overrightarrow{AP}) sign(y_p) = 1 \tag{B.27}$$

B.2 Jacobian matrices of the 2PRU-1PRS

From the kinematic analysis of the manipulator [104] the following linear system results.

$$\mathbf{J}_x \begin{Bmatrix} \dot{z}_p \\ \dot{\psi} \\ \dot{\theta} \end{Bmatrix} = \mathbf{J}_q \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} \tag{B.28}$$

Where,

$$J_{x_{11}} = Lsin(\gamma_1) \tag{B.29}$$
$$J_{x_{12}} = Lrsin(\theta)cos(\psi)cos(\gamma_1) \tag{B.30}$$
$$J_{x_{13}} = Lrcos(\theta)sin(\psi)cos(\gamma_1) + Lrsin(\theta - \gamma_1) \tag{B.31}$$
$$J_{x_{21}} = Lsin(\gamma_2) \tag{B.32}$$
$$J_{x_{22}} = Lr(sin(\psi)cos(\gamma_2) + cos(\theta)cos(\psi)sin(\gamma_2)) \tag{B.33}$$
$$J_{x_{23}} = Lrsin(\theta)sin(\psi)sin(\gamma_1) \tag{B.34}$$
$$J_{x_{31}} = Lsin(\gamma_3) \tag{B.35}$$
$$J_{x_{32}} = -Lrsin(\theta)cos(\psi)cos(\gamma_3) \tag{B.36}$$
$$J_{x_{32}} = -Lrcos(\theta)sin(\psi)cos(\gamma_3) + Lrsin(\theta + \gamma_3) \tag{B.37}$$

And,

$$\mathbf{J}_q = \begin{bmatrix} Lsin(\gamma_1) & 0 & 0 \\ 0 & Lsin(\gamma_2) & 0 \\ 0 & 0 & Lsin(\gamma_3) \end{bmatrix} \tag{B.38}$$

Also, the TCP (point p) of the manipulator can only move in the x and z directions. Hence, the following relations can be found.

From bar 1:

$$\dot{x}_p = L\dot{\gamma_1}sin(\gamma_1) + r\dot{\theta}sin(\theta) \tag{B.39}$$
$$\dot{z}_p = \dot{\rho_1} + L\dot{\gamma_1}cos(\gamma_1) + r\dot{\theta}cos(\theta) \tag{B.40}$$

From bar 2:

$$-r\dot{\psi}sin(\psi) = L\dot{\gamma_2}sin(\gamma_2) \tag{B.41}$$
$$\dot{z}_p = \dot{\rho_2} + L\dot{\gamma_2}cos(\gamma_2) + r\dot{\theta}sin(\theta)sin(\psi) - r\dot{\theta}cos(\theta)cos(\psi) \tag{B.42}$$

From bar 3:

$$\dot{x}_p = -L\dot{\gamma_3}sin(\gamma_3) - r\dot{\theta}sin(\theta) \tag{B.43}$$
$$\dot{z}_p = \dot{\rho_3} + L\dot{\gamma_3}cos(\gamma_3) - r\dot{\theta}cos(\theta) \tag{B.44}$$

Moreover, the velocities of the center of mass for each bar can be calculated from the velocities of point p. In the following, the expressions for the bar 1 will be obtained.

$$\dot{x}_1 = \frac{1}{2}\left(\dot{x}_p - r\dot{\theta}sin(\theta)\right) \tag{B.45}$$
$$\dot{z}_1 = \frac{1}{2}\dot{\rho}_1 + \frac{1}{2}\left(\dot{z}_p - r\dot{\theta}cos(\theta)\right) \tag{B.46}$$
$$\dot{\gamma_1} = \frac{1}{Lsin(\gamma_1)}\left(\dot{x}_p - r\dot{\theta}sin(\theta)\right) \tag{B.47}$$
$$\tag{B.48}$$

Substituting the $\dot{x}_p$ and $\dot{z}_p$ by expressions B.39 and B.40 respectively, the following expression results.

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{z}_1 \\ \dot{\gamma}_1 \end{Bmatrix} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}\left(rsin(\theta)+rcos(\theta)sin(\psi)\right) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \begin{Bmatrix} \dot{z}_p \\ \dot{\psi} \\ \dot{\theta} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \frac{\dot{\rho}_1}{2} \\ 0 \end{Bmatrix} \tag{B.49}$$

The above expression can be rewritten as a function of the inputs as follows.

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{z}_1 \\ \dot{\gamma}_1 \end{Bmatrix} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}\left(rsin(\theta)+rcos(\theta)sin(\psi)\right) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} + \begin{Bmatrix} 0 \\ \frac{\dot{\rho}_1}{2} \\ 0 \end{Bmatrix} \tag{B.50}$$

Finally, the Jacobian matrix for the bar 1 results in:

$$\mathbf{J}_{b_1} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}\left(rsin(\theta)+rcos(\theta)sin(\psi)\right) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q + \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{B.51}$$

With the linear system being:

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{z}_1 \\ \dot{\gamma}_1 \end{Bmatrix} = \mathbf{J}_{b_1} \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} \tag{B.52}$$

By following a similar procedure for bars 1 and 2, the linear systems and their respective Jacobians are obtained.

$$\begin{Bmatrix} \dot{y}_2 \\ \dot{z}_2 \\ \dot{\gamma}_2 \end{Bmatrix} = \mathbf{J}_{b_2} \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} \tag{B.53}$$

$$\begin{Bmatrix} \dot{x}_3 \\ \dot{z}_3 \\ \dot{\gamma}_3 \end{Bmatrix} = \mathbf{J}_{b_3} \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} \tag{B.54}$$

With,

$$\mathbf{J}_{b_2} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\psi) & 0 \\ 0 & -\frac{cos(\gamma_2)}{2sin(\gamma_2)}rsin(\psi) & 0 \\ 0 & -\frac{rsin(\psi)}{Lsin(\gamma_2)} & 0 \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{B.55}$$

$$\mathbf{J}_{b_3} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}\left(rsin(\theta) - rcos(\theta)sin(\psi)\right) \\ \frac{1}{2} & 0 & \frac{1}{2}rcos(\theta) \\ 0 & \frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_3)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_3)} \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix} \tag{B.56}$$

As for the rotation Jacobian of the end effector, the following linear system relates the angular velocities of point p with the angular velocities $\dot{\psi}$ and $\dot{\theta}$.

$$\begin{Bmatrix} w_x \\ w_y \\ w_z \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \begin{Bmatrix} \dot{\psi} \\ \dot{\theta} \end{Bmatrix} \tag{B.57}$$

Moreover, the equation can be expressed as a function of the inputs as follows.

$$\begin{Bmatrix} w_x \\ w_y \\ w_z \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q|_R \begin{Bmatrix} \dot{\rho}_1 \\ \dot{\rho}_2 \\ \dot{\rho}_3 \end{Bmatrix} \tag{B.58}$$

Where $\mathbf{J}_x^{-1}\mathbf{J}_q|_R$ is the submatrix of $\mathbf{J}_x^{-1}\mathbf{J}_q$ from row 2 and 3, which is the matrix that relates the rotation motion of point p with the inputs. Hence, the Jacobian matrix of the end effector results in.

$$\mathbf{J}_R = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \mathbf{J}_x^{-1}\mathbf{J}_q|_R \tag{B.59}$$

B.3 Jacobian matrices of the 2-PRR

B.3.0.8 Solid 1

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\alpha}_1 \end{Bmatrix} = \mathbf{J}_1 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.60}$$

B.3.0.9 Solid M_2

$$\begin{Bmatrix} \dot{x}_2 \\ \dot{y}_2 \\ \dot{\alpha}_2 \end{Bmatrix} = \mathbf{J}_2 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.61}$$

B.3.0.10 Solid M_3

$$\begin{Bmatrix} \dot{x}_3 \\ \dot{y}_3 \\ \dot{\alpha}_3 \end{Bmatrix} = \mathbf{J}_3 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 3/4 & -3/4 \\ -A/2 & -A/2 \\ B & B \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.62}$$

Where,

$$A = \frac{1}{2\tan(\alpha)} \tag{B.63}$$

$$B = \frac{1}{2\sin(\alpha)\ L} \tag{B.64}$$

B.3.0.11 Solid 4

The body 4 moves exactly as body 3, thus, its kinematic is the same as body 3.

$$\begin{Bmatrix} \dot{x}_4 \\ \dot{y}_4 \\ \dot{\alpha}_4 \end{Bmatrix} = \mathbf{J}_4 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 3/4 & -3/4 \\ -A/2 & -A/2 \\ B & B \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.65}$$

B.3.0.12 Solid 5

$$\begin{Bmatrix} \dot{x}_5 \\ \dot{y}_5 \\ \dot{\alpha}_5 \end{Bmatrix} = \mathbf{J}_5 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 1/4 & -3/4 \\ -A/2 & -A/2 \\ -B & -B \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.66}$$

B.3.0.13 Solid M_6

$$\begin{Bmatrix} \dot{x}_6 \\ \dot{y}_6 \\ \dot{\alpha}_6 \end{Bmatrix} = \mathbf{J}_6 \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} = \begin{bmatrix} 1/2 & -1/2 \\ -A & -A \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \dot{\rho}_1 \\ \\ \dot{\rho}_2 \end{Bmatrix} \tag{B.67}$$

Appendix C

Published JCR articles

Mechatronics 000 (2016) 1–15

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics

Constantino Roldán-Paraponiaris*, Francisco J. Campa, Oscar Altuzarra

Department of Mechanical Engineering, University of the Basque Country UPV-EHU, C Alameda Urquijo S/N, Bilbao 48013, Spain

ARTICLE INFO

Article history:
Received 26 July 2016
Revised 7 October 2016
Accepted 22 October 2016
Available online xxx

Keywords:
Dynamic modeling
Mechatronic model
Parallel mechanisms
Multi-axial simulation table

ABSTRACT

In this work a mechatronic model was developed for a parallel Multi-Axial Simulation Table (MAST) mechanism. The dynamics of the mechanism was obtained using the principle of energy equivalence and Boltzmann–Hamel equations. In this way, the procedure to obtain the explicit dynamic equations is simplified and has the advantage of being systematic. Also, the actuators and the control were modeled and integrated to simulate and study the system's positioning and torque.

A remarkable contribution of this work is that the mechatronic model developed considers the mechanism as a disturbance to the actuators in a decoupled manner, allowing to easily evaluate alternative designs of whether the actuators, the mechanism or both. Additionally, the procedure taken has been validated with experimental data from an actual MAST prototype.

1. Introduction

In the last twenty years, parallel kinematics machines have been increasingly used in several fields due to their high performance. For instance, hexapods and tripods are being used for scientific instrumentation due to their high precision positioning in several degrees of freedom [1]. Some solutions present a high stiffness-mass ratio and acceleration which makes them suitable for light machining tasks [2] in an industrial environment. Also, their capability of generating high accelerations makes them interesting for pick & place [3] or for generating harmonically complex motions with great bandwidth, as is the case of excitation tables [4].

Nevertheless, despite their increasing use, they are still complex machines to design, due to their kinematics, dynamics and control. That is why a mechatronic approach with model based design becomes essential for the conception of these machines. For that purpose, complex and detailed simulation tools capable of modeling the manipulator, drives and control dynamics in a cost efficient manner are required.

In that sense, a lot of effort has been putted into modeling the kinematics and dynamics for serial and parallel manipulators. Žlajpah [5] presents an overview of several computational tools (e.g. Matlab/Simulink, Dymola/Modelica) commonly used for simulations purposes focused on robotic systems. As for the dynamics, Lagrange and recursive Newton–Euler methods and the principle of virtual work are commonly employed to obtain the dynamic expressions for serial robotic mechanisms [6]. However, their applicability on parallel manipulators becomes a difficult task because of the kinematic constraints of the closed loops.

Previous works make use of multi-body models of the mechanism [7] or the forward dynamic problem in order to build a mechatronic model. In the first case, it can be justified when high loads are applied to the manipulator, because multibody models are capable of considering the flexible behavior of the machine components. Nevertheless, in that case, expensive software packages may be a limiting factor. On the other hand, with both approaches the contribution of the actuators to the global dynamics is often overlooked, modeling them as a simple inertia and thus assuming a rigid body behavior.

What is more, those formulations can't be used to introduce them in the control algorithm to perform a Compute torque control or a Feed-forward torque control. For that task, the inverse dynamic problem (IDP) has been traditionally used, as it provides the needed torques or forces in the actuators to perform the commanded motion. There are several works where the authors have used this approach to implement control schemes such as model-based control [8,9]. Codourey [10], developed a model-based control using the IDP to implement a feedforward control for a Delta robot. Similarly, Yang et al. [11] developed a computed force and

* Corresponding author.
E-mail addresses: constantino.roldan@ehu.eus (C. Roldán-Paraponiaris), fran.campa@ehu.eus (F.J. Campa), oscar.altuzarra@ehu.eus (O. Altuzarra).

http://dx.doi.org/10.1016/j.mechatronics.2016.10.017
0957-4158/

velocity control for a 6-DOF parallel mechanism also using the inverse dynamics.

Regarding the method to solve the IDP of parallel mechanisms, several formulations have been proposed in the past, as the Newton–Euler [12,13], the principle of virtual work [14] or Lagrangian methods [15]. Likewise, Lagrangian analysis is frequently employed for open-chains mechanisms. It's use in parallel mechanism yields in very large and often complex set of equations because of the kinematic constraints due to the closed loops of such systems. Also, an interesting approach with Newton–Euler has been taken in [13], where intermediate variables from the joint-space and matrix algebraic manipulation tools are used to obtain explicit dynamic models for a Gough–Stewart platform . In general, their applicability on parallel mechanisms is difficult due to the kinematic constraints caused by the closed loops [16]. As an alternative, with other methods of analytical mechanics (i.e. Boltzmann–Hamel equations, quasi-velocities and principle of energy equivalence [17]), the difficulty in finding a dynamic model suitable for computer simulations is greatly reduced.

Moreover, the common approach is to focus on the manipulator and then include the actuators. However, in several applications, especially when the payload and the manipulator are relatively light, the control cycle time or even the actuators dynamics can be more restrictive due to their finite stiffness, which limits the bandwidth and thus the dynamic performance of the machine in terms of speed, acceleration and trajectory tracking [18–20].

In the present work, a procedure for the mechatronic modeling of parallel kinematics machines is proposed, taking into account the rigid body dynamics of the manipulator, the compliant dynamics of the actuators and the cycle time of the control loops. The method is based on decoupling the dynamics of the actuators from the manipulator, in such a way that forces needed to move the manipulator are considered as a disturbance from the point of view of the actuators. This scope allows modeling the manipulator dynamics using the inverse dynamic problem, relating the motion of the actuators with the forces that generate the motion of the manipulator. To do so, although any method can be used, here it is proposed to use the Principle of energy equivalence and the Boltzmann–Hamel equations to compute the IDP. The actuators modeling and their transmission chain can be performed with great detail using a model of several degrees of freedom affected by the disturbance forces from the mechanism and the friction. Finally, the cycle time of the position, velocity and current control loops is taken into account. The whole model has been programmed in Matlab Simulink.

There are several advantages for this procedure. First, it is easy to evaluate alternative designs. Given the fact that models of control, actuators and manipulator are decoupled and represented by blocks, it is possible to replace them with new blocks representing alternative configurations. This reduces the time and effort required in the design and simulation stage for a given application yet being reliable. Second, it is possible to better evaluate the interaction between control, actuators and mechanism. For example, the bandwidth of the actuators alone vs. the bandwidth of the whole manipulator can be analyzed. Also, it is possible to isolate and evaluate the influence of the dynamic parameters of the manipulator or the actuator transmission chain on the tracking error. Simulation of the cycle time and its effect on the trajectory tracking and the driving forces allows also a better definition of the control specifications of the final prototype. Third, the use of the inverse dynamic problem results in an explicit set of equations that allows a fast computation comparing with multibody techniques and can be used also to improve the control algorithm if needed. Fourth, the use of the Principle of energy equivalence and the Boltzmann–Hamel equations allows for a more systematic and error free computation of the IDP for parallel mechanisms.

This article is organized as follows. First, in Section 2, the proposed procedure for mechatronic modeling of parallel kinematic machines will be detailed. Second, in Section 3, a case study based on a 3PRS mechanism will be given where the aforementioned procedures are employed. Third, the results of an experimental validation will be commented in Section 4. Finally, the main conclusions are presented.

2. Mechatronic modeling for parallel kinematic mechanisms

The method here proposed for the mechatronic modeling of parallel manipulators considers the actuators and the manipulator as two independent subsystems whose interaction is due to the Newton's third law. That is, from the actuators viewpoint, the mechanism generates some forces that work as a disturbance against their motion but, at the same time, those forces are the input that provides the manipulator's motion. That interaction is represented by F_i forces in Fig. 1 in a generic parallel manipulator. The result is that the actuators, which often times limit the overall system's performance, can be modeled in a more detailed fashion. On the other hand, to include the influence of the mechanism in the mechatronic model, the inverse dynamic problem (IDP) is solved, with the advantage that those equations can also be used in the control algorithm. Also, the control algorithm as in Fig. 2 and the cycle time of the closed loops has been considered due to their impact on the trajectory tracking, bandwidth and disturbance rejection. To the best of the authors knowledge, this approach that considers the mechanism as a disturbance for the actuators allows a deeper analysis of the interaction between control, actuators and mechanism and has not been addressed before.

2.1. Mechatronic model of the manipulator

In Fig. 1a mechatronic model of a manipulator is shown. It is assumed a joint space position control, where the control reacts to the position error measured in the actuated joints $\boldsymbol{\rho}$. This decision was taken since it is widely found in general industrial applications. However, more complex control algorithm can also be employed as will be shown in Section 2.3. In this way, the end platform position commands $\mathbf{x_0}$ are converted to the joint space through the inverse kinematic problem. Those $\mathbf{q}_{\rho_0}$ commands are then introduced into the mechatronic model of the actuators. As a result, the actuators reach a position $\mathbf{q}_{\rho}$ and the end platform location $\mathbf{x}$ is calculated with the direct kinematic problem. A rigid body behavior is here considered for the mechanism. To model the influence of the mechanism dynamics on the global behavior, the forces $\mathbf{t_p}$ generated to perform the motion are calculated with the IDP once actuators and platform motion are known. Those forces are then introduced as a disturbance into the actuators mechatronic model, in which a cascaded control in position, velocity and current is assumed. There, the mechanism forces are converted into a torque disturbance on the motor, thus acting as an opposition to the actuator.

2.2. Mechatronic model of the actuators

Regarding the actuator control model, a proportional position control with gain k_v is considered. If two encoders are used, a rotary one for the motor and a linear one for the actuator table, it is possible to control directly the linear position $\boldsymbol{\rho}$. The velocity and current control loops in Fig. 2 are based on a PI control, where k_p, k_i, k_{pc} and k_{ic} are the proportional and integral gains, respectively for each loop. Furthermore, their cycle times can be taken into account, sampling the signals and using the z-domain for the transfer functions of the system. Also, the response of the electrical part is modeled by taking into account the resistance R and inductance L

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

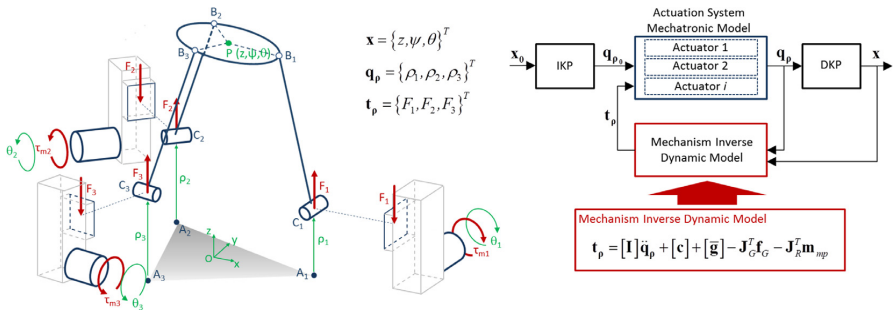


Fig. 1. Mechatronic model of a parallel kinematics machine with joint sace position control: manipulator position in workspace ($\mathbf{x}$), manipulator position in joint space ($\boldsymbol{\rho}$), actuators position at the motor encoders ($\boldsymbol{\theta_i}$), reactions between the manipulator and actuators ($\boldsymbol{\tau}$), motor torques ($\boldsymbol{\tau_{mi}}$).

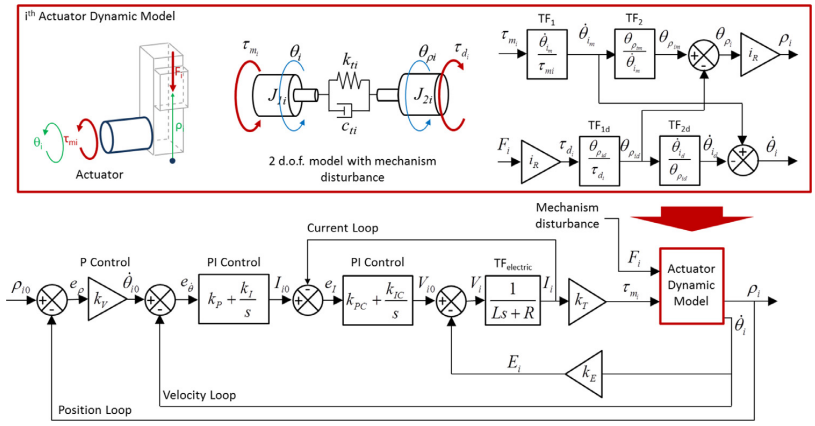


Fig. 2. ith actuator mechatronic model.

of the circuit in $TF_{electric}$, as well as the torque constant k_T and the counter electromotive constant k_E.

With respect to the mechanical behavior of the actuators, the two degrees of freedom model as shown in Fig. 2 has been used, where the motor torque τ_m and the disturbance force F are the inputs. The latter is converted to a torque disturbance applied to the motor by means of Eq. (1), where p is the pitch of the linear guide and i_r is the gear ratio of the gearbox. Likewise, the outputs are the linear position of the guide ρ, whose angular equivalent is θ_ρ, and the motor's angular velocity $\dot{\theta}$. Furthermore, the actual angular position of the guide is obtained from the ideal position θ_{ρ_m} and the position variation due to the disturbance θ_{ρ_d}. Similarly, the actual velocity is obtained from the motor's ideal velocity $\dot{\theta}_m$ and the velocity variation $\dot{\theta}_d$ also caused by the disturbance.

The aforementioned magnitudes are related by four transfer functions (TF). The first, TF_1 in Eq. (2), relates the angular position of the motor without disturbance θ_m and the input torque τ_m. The actuator's flexible dynamics are represented by the damping C_t and stiffness K_t. Moreover, the inertia of the driving part is represented by J_1 whereas J_2 represents the inertia of the transmission. The second transfer function, TF_2 also in Eq. (2), relates the equivalent ideal angular position of the linear guide θ_{ρ_m} with the motor's position θ_m. The remaining transfer functions are employed to model the disturbances. On one hand, TF_{1d} in Eq. (3) relates the disturbance torque τ_d with the angular equivalent of the position disturbance in the linear guide θ_{ρ_d}. On the other hand, TF_{2d} (see Eq. (3)) relates θ_{ρ_d} with the position variation at the motor's shaft due to the disturbance θ_d. It should be noted that these transfer functions must be derived to match the magnitudes when appropriate. Finally, depending on the transmission complexity and stiffness, they can be obtained from a one, two, or N degrees of freedom model. The one DoF model is suitable for actuators that could be considered as stiff or that operate in a relatively low frequency range. Otherwise, a N DoF model is required if the damping and flexibility of its components is not negligible.

$$i_R = \frac{p}{2\pi i_r} \tag{1}$$

$$TF_1 = \frac{\theta_m}{\tau_m} = \frac{J_2 s^2 + c_t s + k_t}{s^2\left(J_1 J_2 s^2 + (J_1 + J_2) c_t s + (J_1 + J_2) k_t\right)}$$
$$TF_2 = \frac{\theta_{\rho_m}}{\theta_m} = \frac{c_t s + k_t}{\left(J_2 s^2 + c_t s + k_t\right)} \tag{2}$$

$$TF_{1d} = \frac{\theta_{\rho_d}}{\tau_d} = \frac{J_1 s^2 + c_t s + k_t}{s^2\left(J_1 J_2 s^2 + (J_1 + J_2) c_t s + (J_1 + J_2) k_t\right)}$$
$$TF_{2d} = \frac{\theta_d}{\theta_{\rho_d}} = \frac{c_t s + k_t}{\left(J_1 s^2 + c_t s + k_t\right)} \tag{3}$$

2.3. Control alternatives

Despite the approach here proposed has been applied to a joint space control, it is also possible to consider other control strategies. The main problem of the joint space control is that it reacts against the position error measured in the actuated joints of the

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

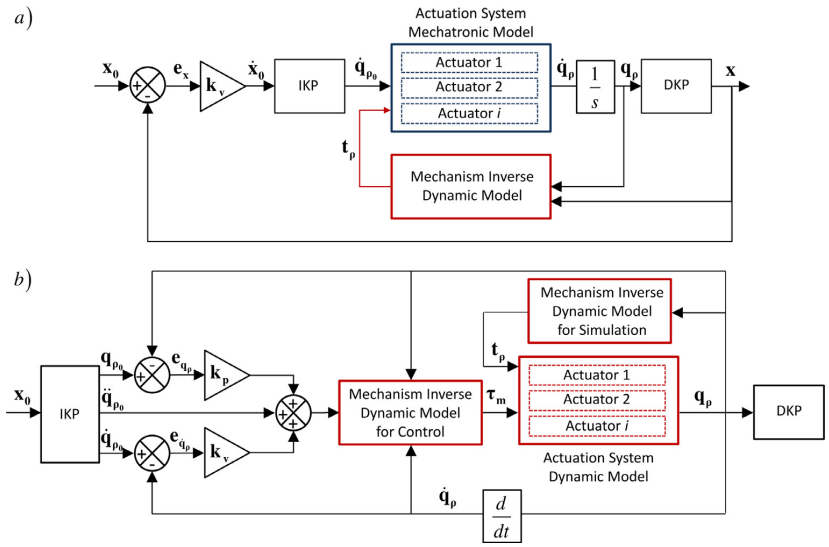


Fig. 3. Mechatronic model for: a) Workspace position control. b) Computed torque control.

manipulator, so every error in the kinematic chain of the mechanism is not considered. Nevertheless, it is a convenient alternative due to the generally difficult access to a direct measurement of the end platform in parallel kinematic machines.

As opposed to joint-space control schemes, workspace control reacts to the position error $\mathbf{e}_x$ measured directly in the end platform. This alternative can be modeled as it is shown in Fig. 3a and compared to the joint space alternative. In this case, the position control is performed in the manipulator level and not in the actuator model. As such and upon receiving a workspace position $\mathbf{X}_0$, the error is calculated with the actual position $\mathbf{X}$ which is then passed through the gain $\mathbf{k}_v$. A velocity command $\dot{\mathbf{X}}_0$ results and is converted to the actuated joints coordinates $\dot{\mathbf{q}}_{\rho_0}$ by using the inverse kinematic problem (IKP). At the same time, the disturbance $\mathbf{t}_\rho$ is calculated and fed into the actuators models along with the joint coordinates. Finally the actual joints positions are obtained and passed through the direct kinematic problem (DKP) to obtain the actual workspace positions $\mathbf{X}$.

Other control algorithms, such as estimators-based controllers or *computed torque control* (CTC) can also be employed [21] and [22]. As an example, a simple scheme for the CTC case is shown in Fig. 3b. First, the command $\mathbf{X}_0$ is converted to position ($\mathbf{q}_{\rho_0}$), velocity ($\dot{\mathbf{q}}_{\rho_0}$) and acceleration ($\ddot{\mathbf{q}}_{\rho_0}$) at the joints with the IKP. Then, the position and velocity errors ($\mathbf{e}_{q_\rho}$, $\mathbf{e}_{\dot{q}_\rho}$) are computed and passed through the position and velocity gains ($\mathbf{k_p}$, $\mathbf{k_v}$) respectively and added to $\ddot{\mathbf{q}}_{\rho_0}$. Afterwards, the acceleration signal is used in the inverse dynamic model (IDP) of the mechanism to obtain the required torques τ_m for the actuators. Also, the disturbance $\mathbf{t}_\rho$ are calculated and introduced to the actuators's models where the actual positions, velocities and accelerations in $\mathbf{q}_\rho$ are obtained. Finally, these are then used in the DKP to calculate the workspace position of the mechanism.

2.4. Inverse dynamic problem modeling

For simulation and control purposes, differential equations in terms of the actuation geometrical kinematic variables, mass properties and applied forces by the actuators are required. This form of dynamic problem results in inertial $\mathbf{I}_G$, gravitational $\tilde{\mathbf{g}}_G$ and Coriolis $\mathbf{c}_G$ terms obtained separately.

With the input variables of the mechanism defined as $\mathbf{q}_\rho$, each link can be expressed with the Lagrange function $\mathcal{L}_{mech}$ (Eq. (4)) in terms of these input variables to apply the Euler operator $\mathcal{E}_{\mathbf{q}_\rho}$ [17].

$$\mathcal{E}_{\mathbf{q}_\rho}(\mathcal{L}_{mech}) = \frac{d}{dt}\frac{\partial \mathcal{L}_{mech}}{\partial \dot{\mathbf{q}}_\rho} - \frac{\partial \mathcal{L}_{mech}}{\partial \mathbf{q}_\rho} = \mathbf{f}_\rho = \mathbf{t}_\rho + \mathbf{Q}_\rho \tag{4}$$

In this equation, $\mathbf{f}_\rho$ is the generalized forces vector which include input torques $\mathbf{t}_\rho$ in terms of the input variables $\mathbf{q}_\rho$, and $\mathbf{Q}_\rho$ represents the generalized output forces. This equation will be used to simulate the torque depending on the kinematic variables of the actuators. In the case of parallel mechanism, obtaining such equation is a difficult task due to the constraints of closed-chains kinematics [15]. By using dependent coordinates this problem can be simplified. Basically, the mechanism can be separated and each of its subsystem can be analyzed independently, as long as the motion conditions are the same as the assembled system. This approach has been followed before in [23] and [24] using Lagrange multipliers $\lambda's$ which have also to be calculated. An alternative is to employ the principle of energy equivalence [15].

Essentially, it states that the sum of each subsystem's energy is the same as the assembled system energy. This avoids the additional calculus of the Lagrange multipliers. However, in spatial rotation such as in the case of the end effector, the kinetic energy results in multiplication of angular derivatives terms by trigonometric functions, which yields in a complex expression of $\mathcal{L}_{mech}$. To avoid this, quasi-velocities can be used as generalized coordinates requiring to employ Boltzmann–Hamel equations instead of the Euler operator.

2.4.1. Principle of energy equivalence applied to planar motion

The mechanism is separated into N free-body subsystems each one represented by its generalized coordinates q_{bi} and grouped into the set $\mathbf{q}_b$. To maintain the energy equivalence with the original system, all subsystems have to move as if they were assembled. This condition implies that $\mathbf{q}_b$ is a function of the generalized coordinates of the assembled mechanism $\mathbf{q}_\rho$. Thus, the virtual displacements $\delta\mathbf{q}_b$ for all subsystems are related with the Jacobians

to the virtual displacement $\delta\mathbf{q}_\rho$ of the generalized coordinates as shown in Eq. (5).

$$\delta\mathbf{q}_b = \frac{\partial \mathbf{q}_b}{\partial \mathbf{q}_\rho}\,\delta\mathbf{q}_\rho = \mathbf{J}\,\delta\mathbf{q}_\rho; \qquad \delta\mathbf{q}_{b_i} = \mathbf{J}_{b_i}\,\delta\mathbf{q}_\rho \qquad i = 1\ldots N \tag{5}$$

In this equation, $\mathbf{J}$ and $\mathbf{J}_{b_i}$ are the Jacobians for the assembled and for each subsystem respectively, which were obtained from the kinematic problem. They relate the generalized coordinates of each subsystem $\mathbf{q}_{b_i}$ with the ones of the assembled mechanism $\mathbf{q}_\rho$. Also, the energy equivalence implies that, the virtual work of the assembly δW_ρ is equal to the sum of the virtual work of all subsystems $\sum_{i=1}^{N}\delta W_{b_i}$. This relation is presented in detail in Eq. (6).

$$\delta W_\rho = \delta\mathbf{q}_\rho^T\mathbf{f}_\rho = \delta W_b = \sum_{i=1}^{N}\delta W_{b_i} = \sum_{i=1}^{N}\delta\mathbf{q}_{b_i}^T\mathbf{f}_{b_i} \tag{6}$$

Furthermore, by substituting Eq. (6) into Eq. (5), the forces due to the manipulator's bars $\mathbf{f}_\rho$ are obtained.

$$\mathbf{f}_\rho = \mathbf{J}^T\mathbf{f}_b = \sum_{i=1}^{N}\mathbf{J}_{b_i}^T\mathbf{f}_{b_i} \tag{7}$$

Finally, when each subsystem is disassembled, joint forces between solids arise which also produce virtual work. However, when the contributions $\mathbf{J}_{b_i}^T\mathbf{f}_{b_i}$ of each subsystem are summed up like in Eq. (7), that virtual work gets canceled by the corresponding reaction of the adjacent subsystem, thus it is unnecessary to take them into account.

2.4.2. Boltzmann–Hamel equations applied for spatial rotation

The end-effector of the 3PRS mechanism herein considered is capable of constrained translation and rotation motions. Decoupling both motions, the dynamics can be studied more effectively. Regarding the contribution of the platform translation to the global dynamics, by considering the coordinates of the center of mass as generalized coordinates $\mathbf{q}_G = [x_G,\ y_G,\ z_G]^T$, an applied external force at that point $\mathbf{f}_G$, and the kinematic relationship with the input variables through the corresponding Jacobian $\mathbf{J}_G$, Eqs. (8) and (9) are obtained, where $\mathbf{M}_G$ is the mass matrix and $\mathbf{I}_G$, $\bar{\mathbf{g}}_G$ and $\mathbf{c}_G$ are recalled to be the inertial, gravitational and Coriolis terms matrices.

$$\mathbf{J}_G^T\mathbf{f}_G = \mathbf{J}_G^T\mathbf{M}_G\mathbf{J}_G\ddot{\mathbf{q}}_\rho + \mathbf{J}_G^T\mathbf{M}_G\dot{\mathbf{J}}_G\dot{\mathbf{q}}_\rho + \mathbf{J}_G^T\mathbf{g}_G \tag{8}$$

$$\mathbf{J}_G^T\mathbf{f}_G = \mathbf{I}_G\ddot{\mathbf{q}}_\rho + \mathbf{c}_G + \bar{\mathbf{g}}_G \tag{9}$$

However, when trying to obtain an expression of the rotation kinetic energy, complex terms of trigonometric functions multiplying angular derivatives arises as a result of the spatial rotation of the element. This yield in a effort when trying to obtain the expressions of the dynamics. By using quasi-velocities as the generalized coordinates, as opposed to position coordinates, the problem can be simplified. Nevertheless, this would imply that Boltzmann–Hamel equations have to be used instead of Euler's operator.

In this work, Boltzmann–Hamel equations are applied to obtain the pure rotation dynamics of the final platform. For this, the angular velocities $\mathbf{w}$ with respect to a reference system attached to the mobile platform, and the Euler's angle $\mathbf{q}_e$, are expressed as in Eqs. (10) and (11).

$$\mathbf{w} = [\omega_x, \omega_y, \omega_z]^T \tag{10}$$

$$\mathbf{q}_e = [\theta_e, \psi_e, \phi_e]^T \tag{11}$$

Furthermore, the relationship between $\mathbf{w}$ and $\dot{\mathbf{q}}_e$ can be written as in Eq. (12), where $\mathbf{D}_{\boldsymbol{\phi}}{}^T$ is the projection matrix between both vectors. It is important to stress that matrix $\mathbf{D}_{\boldsymbol{\phi}}{}^T$ may be rank deficient depending on the orientation of the rigid body relative to the defined coordinate system. Such a problem is commonly found with every parameterization of the orientation and is specially present in the case of large rotation angles ($\geq$180). Nevertheless, rank deficiency can be avoided by properly choosing an orientation of the coordinate system that ensures that the singularities are beyond the actual orientation workspace. This guarantees a full rank matrix and hence good numerical results. Another approach is to employ *quaternions* to express the rotation of the platform. Yet, this would result in unnecessary more complex dynamic equations for the mechanism herein studied since the maximum permitted angle of motion is defined to be 20°.

$$\mathbf{w}(\dot{\mathbf{q}}_e, \mathbf{q}_e) = \mathbf{D}_\phi{}^T\dot{\mathbf{q}}_e = \begin{bmatrix} \cos\phi_e & \sin\theta_e\sin\phi_e & 0 \\ -\sin\phi_e & \sin\theta_e\cos\phi_e & 0 \\ 0 & \cos\theta_e & 1 \end{bmatrix}\begin{Bmatrix} \dot{\theta}_e \\ \dot{\psi}_e \\ \dot{\phi}_e \end{Bmatrix} \tag{12}$$

As a result of applying the Boltzmann–Hamel equations as described in [15,16], the moment due to the rotation of the platform $\mathbf{m}_{mp}$ is obtained as in Eq. (13). In this equation, $\mathbf{I}_{mp}$ is the inertia matrix with respect of a coordinate system fixed to the platform.

$$\mathbf{m}_{mp} = \mathbf{I}_{mp}\dot{\mathbf{w}} + \mathbf{w}\times(\mathbf{I}_{mp}\mathbf{w}) \tag{13}$$

Furthermore, on account of considering the mechanism as a disturbance to the actuators, this moment has to be expressed with respect to the input variables. Hence, the Jacobian matrix $\mathbf{J_R}$, which relates the inputs in $\mathbf{q}_\rho$ and the velocities $\mathbf{w}$, is used to *project* the rotation dynamics onto the actuation system. Therefore, Eq. (13) is operated and rewritten as follows.

$$\mathbf{J}_R^T\mathbf{m}_{mp} = \mathbf{J_R}^T\left[\mathbf{D}_\phi\mathbf{I}_{mp}\dot{\mathbf{w}} + 2\dot{\mathbf{D}}_\phi\mathbf{I}_{mp}\mathbf{w} - \mathbf{A}^T\mathbf{I}_{mp}\mathbf{w}\right] \tag{14}$$

Where

$$\mathbf{A} = \left(\frac{\partial\dot{\mathbf{w}}}{\partial\dot{\mathbf{q}}_e}\right) \tag{15}$$

The time derivatives of $\mathbf{w}$ and $\dot{\mathbf{q}}_e$ are defined as

$$\dot{\mathbf{w}} = \dot{\mathbf{D}}_\phi^T\dot{\mathbf{q}}_e + \mathbf{D}_\phi^T\ddot{\mathbf{q}}_e \tag{16}$$

$$\ddot{\mathbf{q}}_e = \mathbf{J}\ddot{\mathbf{q}}_\rho + \dot{\mathbf{J}}\dot{\mathbf{q}}_\rho \tag{17}$$

Finally, by substituting when appropriate into Eq. (14), the expression representing the rotation dynamics is found separated by the inertial term $\bar{\mathbf{I}}_{mp}$ and the quadratic velocity term $\mathbf{c}_{mp}$.

$$\mathbf{J}_R^T\mathbf{m}_{mp} = \mathbf{J}_R^T\mathbf{I}_{mp}\mathbf{J}_R\ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} = \bar{\mathbf{I}}_{mp}\ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} \tag{18}$$

Where,

$$\mathbf{c}_{mp} = \left[\mathbf{J}^T\left[\mathbf{D}_\phi\mathbf{I}_{mp}\mathbf{D}_\phi{}^T\right]\dot{\mathbf{J}} + \mathbf{J}^T\left[\mathbf{D}_\phi\mathbf{I}_{mp}\dot{\mathbf{D}}_\phi^T + 2\dot{\mathbf{D}}_\phi\mathbf{I}_{mp}\mathbf{D}_\phi{}^T - \mathbf{A}^T\mathbf{I}_{mp}\mathbf{D}_\phi{}^T\right]\mathbf{J}\right]\dot{\mathbf{q}}_\rho \tag{19}$$

It is important to note that Eqs. (18) and (19) do not lack of generality and thus can be used for any manipulator, provided that $\mathbf{I}_{mp}$ and $\mathbf{J_R}$ are appropriately modified. As a contribution, the proposed procedure has the advantage of being systematic, which is less error prone than when trying to obtain particular dynamic equations for a given mechanism.

3. Case study

The case study is based on an actual 3PRS Multi-Axial Simulation Table (MAST) parallel mechanism. A picture of the prototype used is shown in Fig. 4. Moreover, the important geometric variables are also shown in Fig. 4. The dimensions of the actual mechanism are $r = 0.35$ m, $L = 0.26$ m and $H = 0.4$ m. The mass of the legs is 0.07 kg and the mass of the platform is 2.06 kg.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

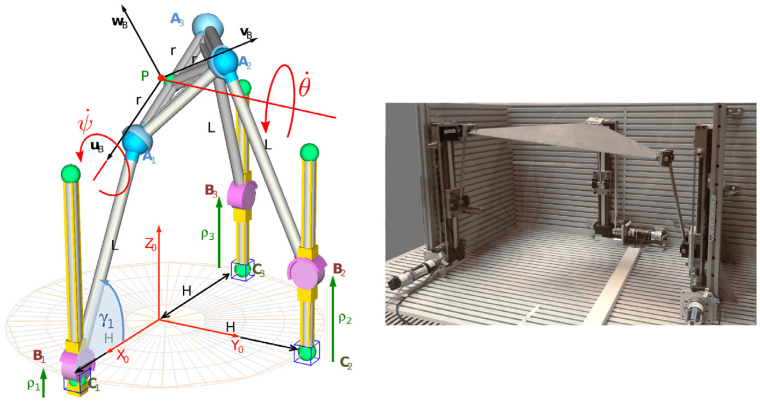


Fig. 4. 3PRS prototype.

Fig. 5. Picture of the actuator.

3.1. Actuators modeling

The prismatic joints used to drive the mechanism are formed by a linear belt guide Igus® ZLW-1040-02-S-100-L/R-300 with a 300 mm stroke [25], actuated by a Maxon® RE-40 DC motor with a GP42C gearbox with a 15:1 gear ratio [26]. The linear guide is connected to the gearbox with a flexible coupling. A picture of the actuation system is shown in Fig. 5.

As mentioned previously, the dynamics of the actuators is represented by a two degrees of freedom model as in Fig. 2. In Eq. (20), J_1 is the sum of the inertias of all the components up to the input shaft of the linear guide. As for the inertia J_2, it is calculated from the mass of the load M_{load} which is formed by the mass of the drive carriage and the inertia of the linear guide J_{guide} obtained from the manufacturer. Furthermore the mass M_{load} is translated into a rotational inertia as in Eq. (21). Finally, J_t is the sum of inertia J_1 and J_2.

$$J_1 = J_{motor} + J_{gearbox} + J_{coupling} \tag{20}$$

$$J_2 = J_{guide}\left(\frac{1}{i_r}\right)^2 + M_{load}\left(\frac{p}{2\cdot\pi\cdot i_r}\right)^2 \tag{21}$$

Where,

$$M_{load} = M_{drive} = 2.27[kg] \tag{22}$$

Also, the linear guides have Coulomb friction τ_f which plays an important role specially at low velocities. This parameter is calculated with equation Eq. (23), obtained from the manufacturer's catalogue [25].

$$\tau_f = 0.2 + \left(4.07\times 10^{-2}\right)M_{load} \tag{23}$$

It is worth mentioning that K_t and C_t in Eqs. (2) and (3) are unknown parameters. Hence, they must be experimentally identified or estimated. To that end, a modal analysis was carried out to obtain the unknown parameters value from the frequency response functions. An accelerometer was placed at the drive carriage and an impact test was performed with a PCB-086-C03 modal hammer. The input signals from the hammer and the vibrations registered by the accelerometer were passed through the OROS-OR35 signal analyzer to obtain the parameters values. Additionally, J_{motor}, $J_{gearbox}$, J_{guide} and $J_{coupling}$ were taken from the manufacturers datasheet [26].

3.2. Control and modeling

The control employed in this study is based in a monoarticular local control, where each actuator's position is independently controlled. In this way, the control is performed in the joint space. Moreover, the demanded pose of the mechanism is first passed through the inverse kinematic problem where the joint position are obtained. These are then fed into the joint controller, each one consisting of a cascaded position, velocity and current control. In Fig. 6 a scheme with the actual control implementation is shown.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

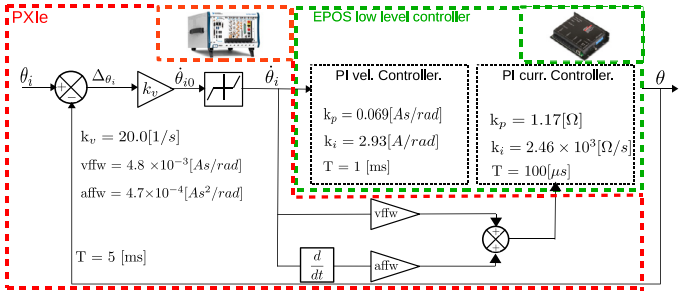


Fig. 6. Actual controller.

3.3. Mechanism dynamics

In this section the dynamics of the MAST prototype are obtained. As for the mechanism bars and the translation dynamics of the platform, the principle of energy equivalence depicted in Section 2.4.1 will be followed. On the other hand, the rotation dynamics of the platform will be analyzed using the Boltzmann–Hamel equations as explained in Section 2.4.2.

3.3.1. Bars modeling

To model the dynamics of the mechanism's bars, the Lagrangian equation for each one is obtained and then the Euler's operator is applied as in Eq. (4). In this way, Eq. (24) results, where m_{b_i} and I_{b_i} are the mass and inertia of the ith bar respectively, and g is the gravitational acceleration.

$$\mathbf{f}_{b_i} = \begin{Bmatrix} 0 \\ F_{b_i} \\ 0 \end{Bmatrix} = \mathcal{E}_{\mathbf{q}_{b_i}}\left(\mathcal{L}_{b_i}\right) = \begin{bmatrix} m_{b_i} & 0 & 0 \\ 0 & m_{b_i} & 0 \\ 0 & 0 & I_{b_i} \end{bmatrix} \ddot{\mathbf{q}}_{b_i} + \begin{Bmatrix} 0 \\ m_{b_i} g \\ 0 \end{Bmatrix} = \mathbf{M}_{b_i} \ddot{\mathbf{q}}_{b_i} + \mathbf{g}_{b_i} \tag{24}$$

Moreover, by using the kinematic relationship $\dot{\mathbf{q}}_{b_i} = \mathbf{J}_{b_i} \dot{\mathbf{q}}_\rho$ Eq. (25) is obtained. $\mathbf{J}_{b_i}$ is the Jacobian of each bar which relates the velocity of the center of gravity with the input variables.

$$\mathbf{f}_{b_i} = \mathbf{M}_{b_i} \mathbf{J}_{b_i} \ddot{\mathbf{q}}_\rho + \mathbf{M}_{b_i} \dot{\mathbf{J}}_{b_i} \dot{\mathbf{q}}_\rho + \mathbf{g}_{b_i} \tag{25}$$

The expression of the $\mathbf{J}_{b_i}$ is shown in the appendix. What is more, the contribution of each leg i to the overall mechanism's dynamics $\mathbf{f}_{\rho_{b_i}}$ is found by employing Eq. (7), thus obtaining the expression in Eq. (26).

$$\mathbf{f}_{\rho_{b_i}} = \mathbf{J}_{b_i}^T \mathbf{f}_{b_i} = \left[\mathbf{J}_{b_i}^T \mathbf{M}_{b_i} \mathbf{J}_{b_i}\right] \ddot{\mathbf{q}}_\rho + \left[\mathbf{J}_{b_i}^T \mathbf{M}_{b_i} \dot{\mathbf{J}}_{b_i} \dot{\mathbf{q}}_\rho\right] + \left[\mathbf{J}_{b_i}^T \mathbf{g}_{b_i}\right] \tag{26}$$

Furthermore, Eq. (26) can be simplified by operating the matrices and renaming them afterwards. In this way, Eq. (27) results.

$$\mathbf{f}_{\rho_{b_i}} = \mathbf{I}_{b_i} \ddot{\mathbf{q}}_\rho + \mathbf{c}_{b_i} + \bar{\mathbf{g}}_{b_i} \tag{27}$$

Where $\mathbf{I}_{b_i}$ and $\bar{\mathbf{g}}_{b_i}$ are the inertia matrix and gravitational vector respectively. These are only functions of the position of the mechanism. On the other hand, $\mathbf{c}_{b_i}$ is the matrix containing velocity quadratic terms.

3.3.2. Modeling of the end-effector

As mentioned before, the MAST mechanism is capable of constrained translation and rotation motions. The translation dynamics has been analyzed as described in Section 2.4.2 and following Eqs. (8) and (9). On the other hand, the rotation dynamics is analyzed by considering a virtual inertial body and applying Boltzmann–Hamel equations. A moving coordinate system is placed at the point P as shown in Fig. 4. Furthermore, the angular velocity $\mathbf{w}$ in the moving frame is used as *quasi-velocities* for the application of the Boltzmann–Hamel equations. Since only the rotation motion of the platform is being analyzed, the translation and potential energies are neglected. Hence, the Lagrangian function of the moving platform results in:

$$\mathcal{L}_{mp} = \bar{\mathbf{T}} = \frac{1}{2} \mathbf{w}^T \mathbf{I}_{mp} \mathbf{w} \tag{28}$$

Where $\mathbf{w}$ is the angular velocity,

$$\mathbf{w} = [\omega_x, \omega_y, \omega_x]^T \tag{29}$$

and $\mathbf{I}_{mp}$ is the inertia tensor of the platform in the local reference system. Also, the assembled condition of the platform with the rest of the mechanism is provided by the loop-closure and constrained equations. These relate the angular velocity $\mathbf{w}$ with the input variables velocities in $\dot{\mathbf{q}}_\rho$ as follows.

$$\mathbf{w} = \mathbf{J}_R \dot{\mathbf{q}}_\rho \tag{30}$$

Where the expression of $\mathbf{J}_R$ is shown in the appendix. Moreover, by substituting Eq. (12) into Eq. (30), the relationship between the Euler angles and the inputs can be found.

$$\dot{\mathbf{q}}_e = \left(\mathbf{D}_\phi{}^T\right)^{-1} \mathbf{J}_R \dot{\mathbf{q}}_\rho = \mathbf{J} \dot{\mathbf{q}}_\rho \tag{31}$$

By following the procedure depicted in Section 2.4.2, and recalling Eqs. (18) and (19), the expression of the rotation dynamics is obtained.

$$\mathbf{J}_R^T \mathbf{m}_{mp} = \mathbf{J}_R^T \mathbf{I}_{mp} \mathbf{J}_R \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp} = \bar{\mathbf{I}}_{mp} \ddot{\mathbf{q}}_\rho + \mathbf{c}_{mp}$$

Where,

$$\mathbf{c}_{mp} = \left[\mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T\right] \mathbf{J} + \mathbf{J}^T \left[\mathbf{D}_\phi \mathbf{I}_{mp} \dot{\mathbf{D}}_\phi^T + 2 \dot{\mathbf{D}}_\phi \mathbf{I}_{mp} \mathbf{D}_\phi{}^T - \mathbf{A}^T \mathbf{I}_{mp} \mathbf{D}_\phi{}^T\right] \mathbf{J}\right] \dot{\mathbf{q}}_\rho$$

3.3.3. Global dynamics

Finally, the explicit dynamic equations of the 3PRS is obtained by adding the dynamic contribution of the legs (Eq. (27)), and the translational and rotational dynamic contributions of the platform (Eqs. (9) and (18) respectively). In this way, Eq. (32) is obtained where $\mathbf{I}_{Ls}$, $\mathbf{c}_{Ls}$ and $\mathbf{g}_{Ls}$ are the inertial, Coriolis and gravitational terms of the bars. Similarly, it is recalled that $\mathbf{I}_G$ and $\mathbf{c}_G$ represent the inertial and Coriolis terms for the translational dynamics whereas $\bar{\mathbf{I}}_{mp}$ and $\mathbf{c}_{mp}$ represent the dynamics due to the rotation.

$$\mathbf{t}_\rho + \mathbf{J}_G^T \mathbf{f}_G + \mathbf{J}_R^T \mathbf{m}_{mp} = \left[\mathbf{I}_{Ls} + \mathbf{I}_G + \bar{\mathbf{I}}_{mp}\right] \ddot{\mathbf{q}}_\rho + \left[\mathbf{c}_{Ls} + \mathbf{c}_G + \mathbf{c}_{mp}\right] + \left[\bar{\mathbf{g}}_{Ls} + \bar{\mathbf{g}}_G\right] \tag{32}$$

It can be seen that with the approach followed, the Jacobians $\mathbf{J}_{bi}$, $\mathbf{J}_G$ and $\mathbf{J}_R$ are obtained from the kinematic analysis in a straightforward way after applying the loop-closure and constrained equations. Furthermore, matrices $\mathbf{D}_\phi{}^T$ and $\mathbf{A}$ can be defined for an Euler convention, hence, not depending on any specific mechanism.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

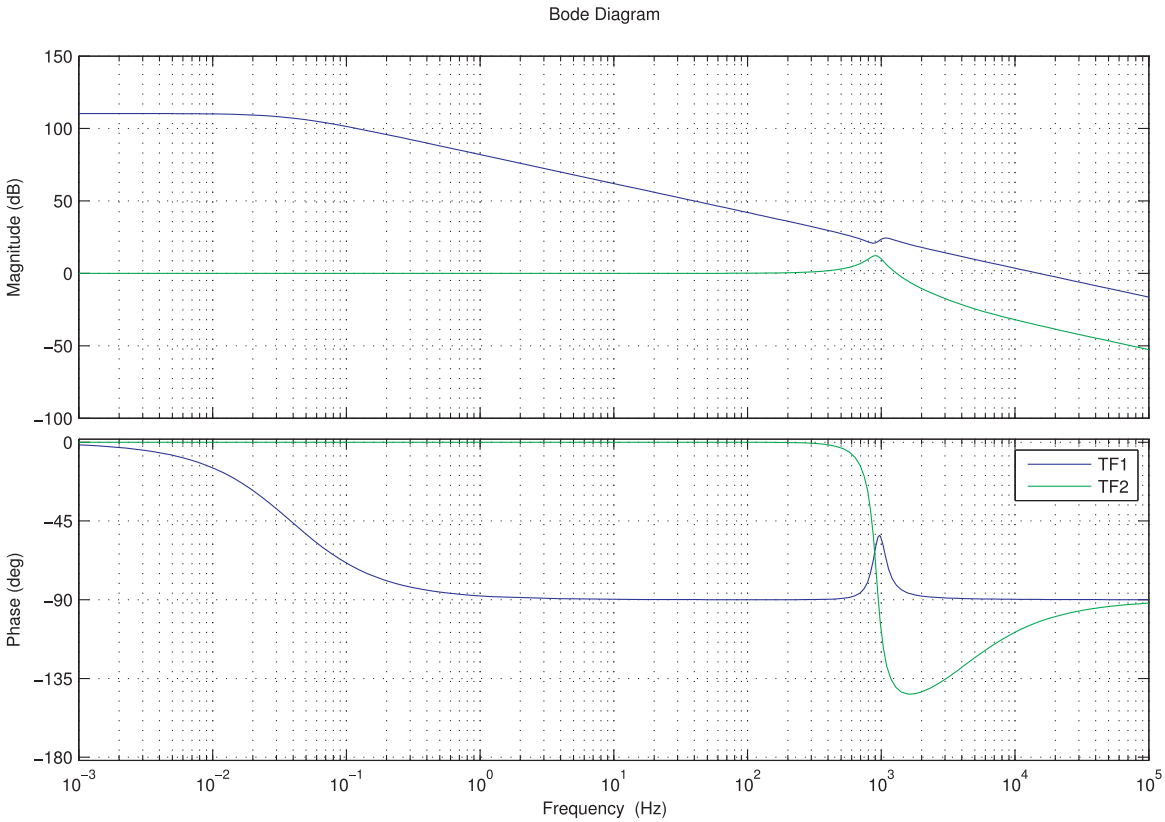


Fig. 7. Bode plot of the two transfer functions of the actuators.

4. Experimental validation

To validate the model of the system, the frequency response was analyzed and the bandwidth of the simulation model was compared with the experimental results obtained from the prototype. Also, the modeled and measured torque outputs were compared. As a result, the interaction of the mechanism's and actuators's models can be easily studied and compared with experimental data.

4.1. Frequency domain testing

In Fig. 7 a bode diagram is shown with the magnitude and phase response of the transfer functions in Eqs. (2) and (3) using the values of Table 1. It can be seen that the resonant frequency of the TF_2 is located at 900 [Hz] approximately, which is significantly higher that the intended operating frequency range of the system.

Also, for a MAST mechanism it is important to study the bandwidth of the system to know up to which frequency could be simulated. For this reason, pure sinusoidal motions in z, ψ and θ were simulated in the 0.5–20 Hz frequency range using the mechatronic model. For the experimental tests, the same motions have been commanded with frequencies set to 0.4, 1, 5 10 and 20 Hz. A comparison between the simulation and the experimental tests is shown in Figs. 8 and 9. Moreover, to study the interaction between the actuation system and the mechanism, the bandwidth with and without the mechanism is compared in Fig. 10

Table 1
Parameter values of the actuator model.

Parameter	Value	Units
f_n [a]	986.5	$[Hz]$
K_t [a]	67.702	$\left[\frac{Nm}{rad}\right]$
C_t [a]	3.00×10^{-08}	$\left[\frac{Nms}{rad}\right]$
M [a]	0.076	$[kg]$
J_{motor} [b]	1.42×10^{-5}	$[kgm^2]$
$J_{gearbox}$ [b]	1.4×10^{-6}	$[kgm^2]$
$J_{coupling}$ [b]	4.68×10^{-6}	$[kgm^2]$
J_{guide} [b]	4.524×10^{-4}	$[kgm^2]$
J_{brake} [b]	1.00×10^{-6}	$[kgm^2]$

[a] Experimentally identified.
[b] From manufacturer.

From Figs. 8 and 9 it can be seen that the simulations are in close correspondence with the experimental results obtained for each motion. Furthermore, when the two motions are compared, no significant differences are present in the system's performance despite the fact that both motions are different. This leads to the idea that the mechanism does not exerts a significant influence on the system's performance. This can be further analyzed with the z motion in Fig. 10.

As mentioned before, in Fig. 10 the response magnitude and phase for the z motion is shown. What is more, the same motion is compared with the MAST attached and with only the actuators. In this way, it can be easily analyzed the effect of the mechanism in the system's response. It is appreciated that up to 2 Hz

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

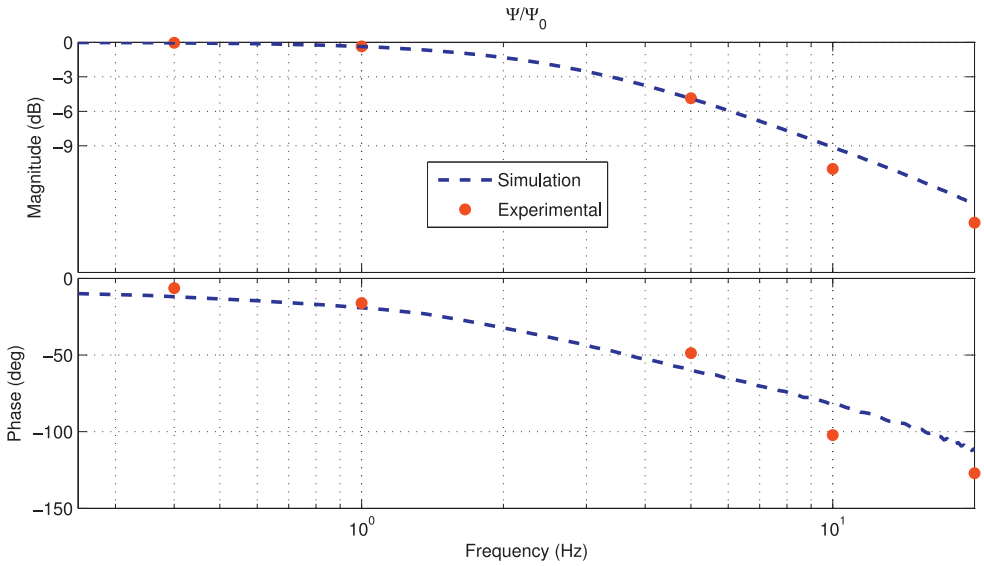


Fig. 8. Manipulator closed position loop transfer function for the ψ motion.

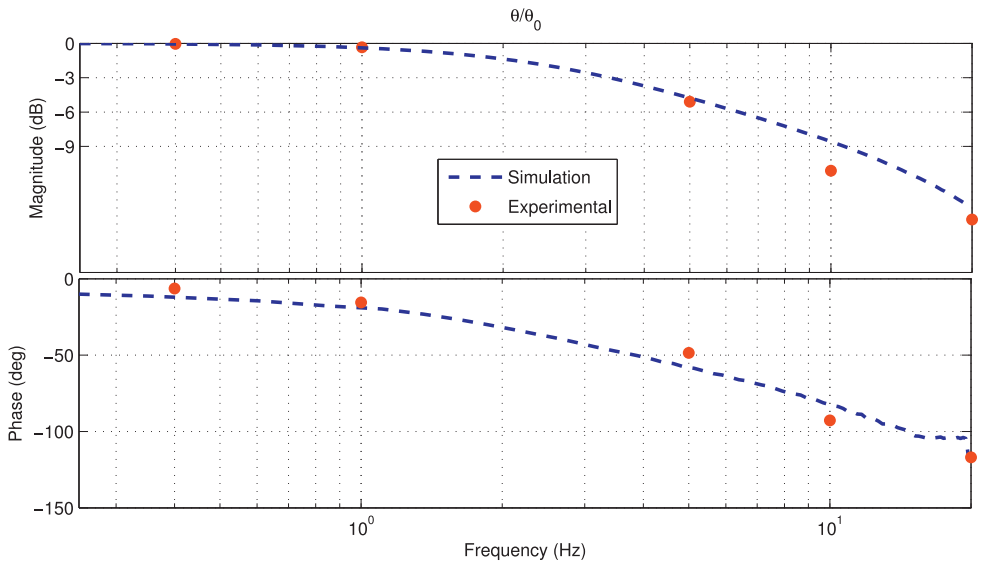


Fig. 9. Manipulator closed position loop transfer function for the θ motion.

both systems behave similarly. However at 2 Hz the magnitude of their response start diverging. What is more, it has been found that the actuators without the MAST have a bandwidth about 4 Hz, whereas with the mechanism the bandwidth is reduced to 3.5 Hz. Thus, it can be noted that the actuators are the most limiting components and that the mechanism itself has only a 0.5 Hz effect on the system's performance.

4.2. Time domain tests

Finally, in Figs. 11–13 the position and the torques are shown for the three pure motions executed at frequencies 0.4, 5 and 10 Hz. Two set of model parameters were used to simulated the model and compare the results with the measurements. The simulation with the parameter set *Sim. Manufac.* was constructed with the data as obtained from the manufacturers, whereas the simulation *Sim. Ident.* was constructed by modifying the inertia $J1$ to $J1/1.56$ and the Coulomb friction modified to 7.18×10^{-2} Nm as it will be explained in the next paragraph.

From the figures, the simulated positions present and maximum deviation of approximately 11% in magnitude and 13% in phase respect to the experimental signals for the frequencies shown and motions executed. Also, it can be observed that as the frequencies of the motions are increased, the magnitude of the

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

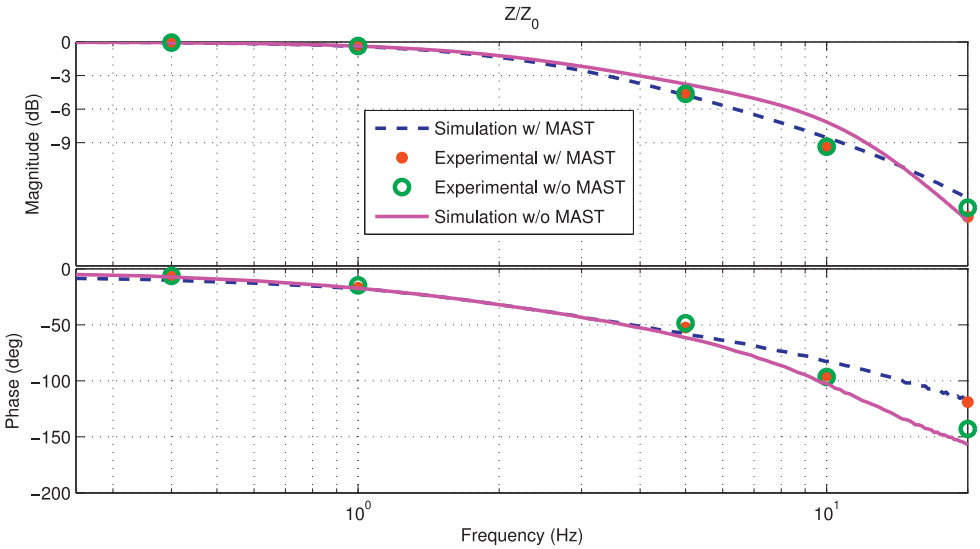


Fig. 10. Manipulator closed position loop transfer function for the actuator model with and without the MAST attached.

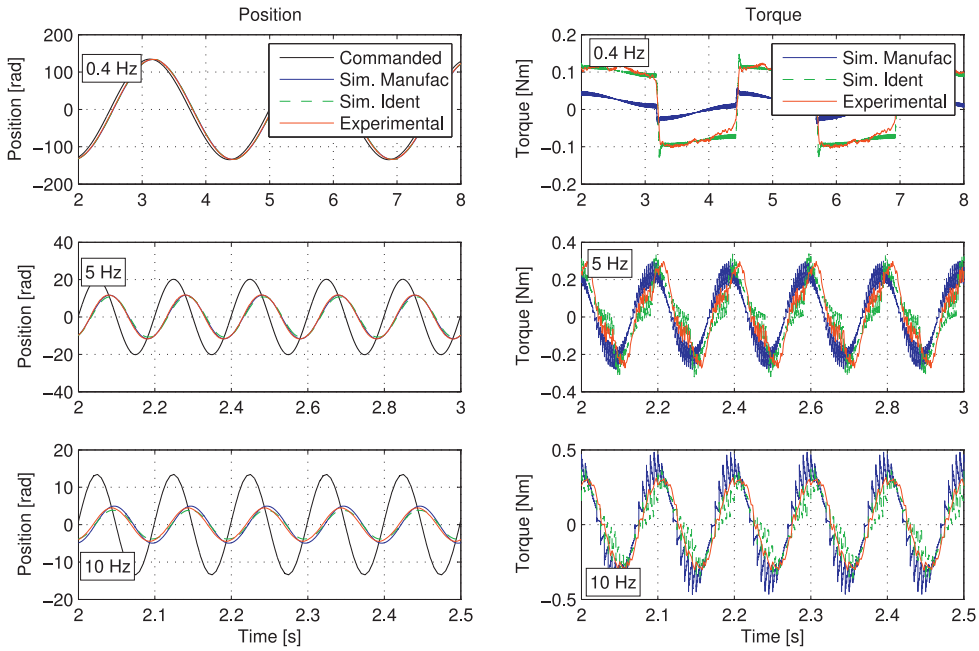


Fig. 11. Position and torque signals comparison for the z motion.

response is reduced in accordance with the Bodes shown previously. Indeed, at 0.4 Hz the magnitude of the response was found to be approximately 99.4%, whereas at 5 Hz and 10 Hz the obtained magnitudes were 65.4% and 43.2% respectively. Likewise, the phase lag between the commands and the actual signals are appropriately estimated by the model when compared with the experimental data. It can be seen that at 0.4 Hz, 5 Hz and 10 Hz, the phase lags were found to be -6.3°, -62.5° and -103.7° respectively.

Nevertheless and to validate the model, the position response must not be used alone as the controller's action may be masquerading dynamic deviations with the appearance of good positioning results. This effect can be clearly seen by comparing the position and the torque of the *Sim. Manufac.* model at 0.4 Hz. It is observed that, despite an apparent good positioning with respect to the experimental data, the simulated torque differs from the experimental torque obtained by approximately 70%. This indicates that the dynamics are not correctly represented by this model.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

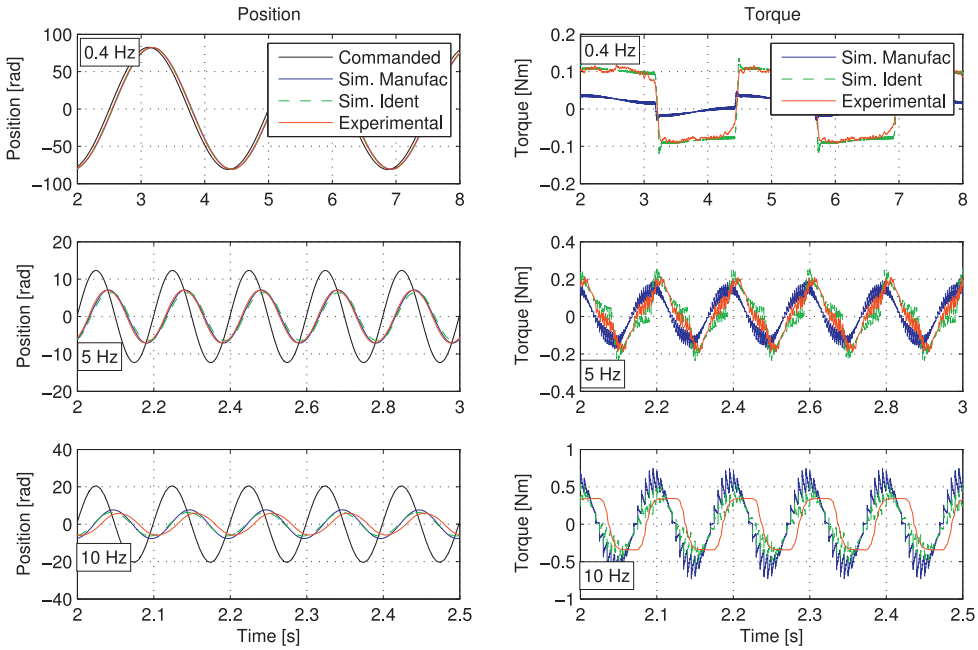


Fig. 12. Position and torque signals comparison for the ψ motion.

Moreover, the torques obtained from the simulation with the two parameters set are different. When the data as given by the manufacturer was introduced, a difference in the torque of approximately 0.10[Nm] at 0.4 Hz can be appreciated as it was pointed out before. This implies that the Coulomb friction force in the actual system is higher. Also, at a frequency of 10 Hz, the model estimated 0.1[Nm] higher torque than the experimental signals. Since the acceleration forces increase with the frequencies, it follows that the actual inertia should be less than the initially estimated with the manufacturer data. For this reason, the Coulomb friction and the inertia J_1 were adjusted as commented before. In this way, the simulation provided better results than when the parameters set with the original manufacturer data was used.

On top of that, it is worth observing that even when the inputs are pure sinusoids, the resulted torques are not sinusoids as well. This effect is caused by the non-linearities of the dynamics and the friction present in the actuation system. The latter can be easily identified at lower frequencies, specially at 0.4 Hz where an almost square signal is appreciated. Finally, it can also be seen that at 10 Hz the experimental torque are *trimmed* by the action of a low-pass filter attributed to the low level controller.

5. Conclusions

In the present work a mechatronic model has been developed for a 3PRS parallel manipulator. A key contribution of this work is that the mechanism dynamics is integrated as a disturbance applied to the actuators. This results in a simple yet reliable model that integrates the actuators dynamics with the manipulator model in a decoupled manner. In this way, different configurations or design of individual components can be easily modified and then integrated to evaluate the overall system's performance without compromising reliability of the results.

Furthermore, the manipulator's dynamics was described using Boltzmann–Hamel equations and the principle of energy equivalence as shown is Section 3.3. It was found that, since the translation and rotation motions are decoupled, the main advantage of Boltzmann–Hamel equations becomes apparent when applied to analyze the rotation dynamics. Furthermore, the approach taken simplifies obtaining explicit dynamic equations, where complex and nonlinear expressions are avoided by making use of the kinematic Jacobians. As a contribution, the proposed procedure has the advantage of being systematic, which is less error prone than when trying to obtain particular dynamic equations for a given mechanism.

Also, the actuators and the control scheme have been modeled. They were further integrated into the mechatronic model and its performance was compared with an actual prototype of a MAST. It was observed that the mechatronic model resulted in an useful simulation tool to assess the performance of the system in terms of bandwidth.

Moreover, when the simulations and experimental signals where compared, it was observed a reasonably agreement. The motor positions of both signals present the same amplitude and the phase lag shows a slight difference attributed to the friction and other phenomena yet to be explored. Also, it was observed that with the parameters values obtained from the manufacturer, the model provided fairly good results. However, with the Coulomb friction and inertia adjusted, the model provided more accurate results. This shows that an identification step should be taken in order to take into account differences between the theoretic and actual values for the parameters.

Acknowledgment

The authors of this paper wish to acknowledge the funding received from the Spanish Government via the Ministerio de Economía y Competitividad (BES-2012-053723 under Project DPI2011-22955 and DPI2015-64450-R), the ERDF of the European Union, the Government of the Basque Country (SAIOTEK 2013

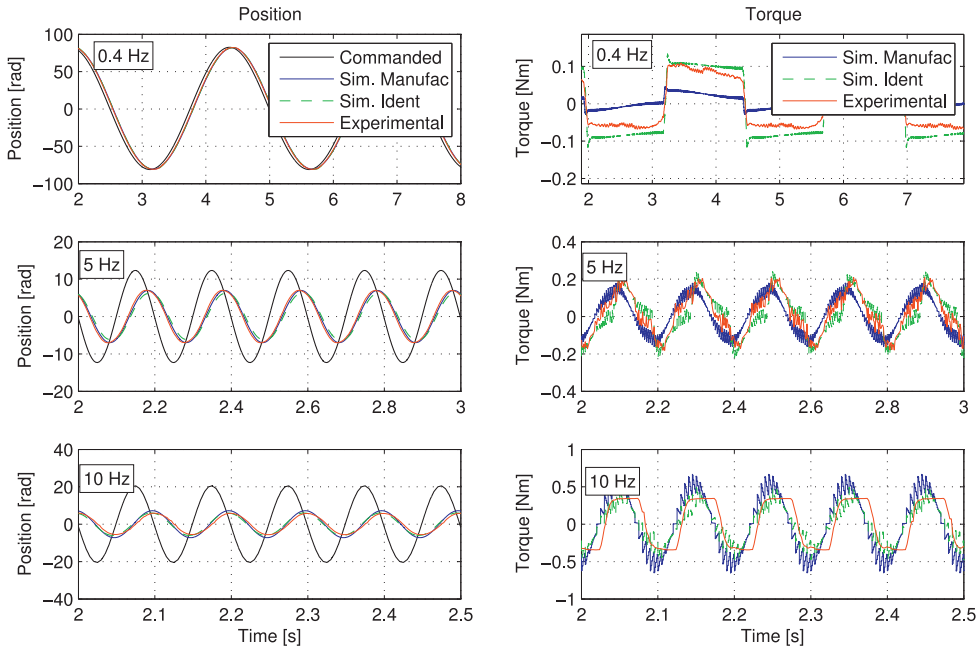


Fig. 13. Position and torque signals comparison for the θ motion.

SAI13/245), and the financial support from the University of the Basque Country(UPV/EHU) under the program UFI 11/29.

Appendix A. Jacobian matrices

From the kinematic analysis of the manipulator [27] the following linear system results.

$$[J_x]\begin{Bmatrix}\dot{z}_p\\ \dot{\psi}\\ \dot{\theta}\end{Bmatrix} = [J_q]\begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} \tag{A.1}$$

Where,

$$J_{x_{11}} = Lsin(\gamma_1) \tag{A.2}$$

$$J_{x_{12}} = Lrsin(\theta)cos(\psi)cos(\gamma_1) \tag{A.3}$$

$$J_{x_{13}} = Lrcos(\theta)sin(\psi)cos(\gamma_1) + Lrsin(\theta - \gamma_1) \tag{A.4}$$

$$J_{x_{21}} = Lsin(\gamma_2) \tag{A.5}$$

$$J_{x_{22}} = Lr(sin(\psi)cos(\gamma_2) + cos(\theta)cos(\psi)sin(\gamma_2)) \tag{A.6}$$

$$J_{x_{23}} = Lrsin(\theta)sin(\psi)sin(\gamma_1) \tag{A.7}$$

$$J_{x_{31}} = Lsin(\gamma_3) \tag{A.8}$$

$$J_{x_{32}} = -Lrsin(\theta)cos(\psi)cos(\gamma_3) \tag{A.9}$$

$$J_{x_{32}} = -Lrcos(\theta)sin(\psi)cos(\gamma_3) + Lrsin(\theta + \gamma_3) \tag{A.10}$$

And,

$$J_q = \begin{bmatrix} Lsin(\gamma_1) & 0 & 0\\ 0 & Lsin(\gamma_2) & 0\\ 0 & 0 & Lsin(\gamma_3)\end{bmatrix} \tag{A.11}$$

Also, the *TCP* (point p) of the manipulator can only move in the x and z directions. Hence, the following relations can be found.

From bar 1:

$$\dot{x}_p = L\dot{\gamma}_1 sin(\gamma_1) + r\dot{\theta}sin(\theta) \tag{A.12}$$

$$\dot{z}_p = \dot{\rho}_1 + L\dot{\gamma}_1 cos(\gamma_1) + r\dot{\theta}cos(\theta) \tag{A.13}$$

From bar 2:

$$-r\dot{\psi}sin(\psi) = L\dot{\gamma}_2 sin(\gamma_2) \tag{A.14}$$

$$\dot{z}_p = \dot{\rho}_2 + L\dot{\gamma}_2 cos(\gamma_2) + r\dot{\theta}sin(\theta)sin(\psi) - r\dot{\theta}cos(\theta)cos(\psi) \tag{A.15}$$

From bar 3:

$$\dot{x}_p = -L\dot{\gamma}_3 sin(\gamma_3) - r\dot{\theta}sin(\theta) \tag{A.16}$$

$$\dot{z}_p = \dot{\rho}_3 + L\dot{\gamma}_3 cos(\gamma_3) - r\dot{\theta}cos(\theta) \tag{A.17}$$

Moreover, the velocities of the center of mass for each bar can be calculated from the velocities of point p. In the following, the expressions for the bar 1 will be obtained.

$$\dot{x}_1 = \frac{1}{2}\left(\dot{x}_p - r\dot{\theta}sin(\theta)\right) \tag{A.18}$$

$$\dot{z}_1 = \frac{1}{2}\dot{\rho}_1 + \frac{1}{2}\left(\dot{z}_p - r\dot{\theta}cos(\theta)\right) \tag{A.19}$$

$$\dot{\gamma}_1 = \frac{1}{Lsin(\gamma_1)}\left(\dot{x}_p - r\dot{\theta}sin(\theta)\right) \tag{A.20}$$

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

Substituting the $\dot{x}_p$ and $\dot{z}_p$ by expressions A.12 and (A.13) respectively, the following expression results.

$$\begin{Bmatrix}\dot{x}_1\\ \dot{z}_1\\ \dot{\gamma}_1\end{Bmatrix} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}(rsin(\theta)+rcos(\theta)sin(\psi)) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \times \begin{Bmatrix}\dot{z}_p\\ \dot{\psi}\\ \dot{\theta}\end{Bmatrix} + \begin{Bmatrix}0\\ \frac{\dot{\rho}_1}{2}\\ 0\end{Bmatrix} \tag{A.21}$$

The above expression can be rewritten as a function of the inputs as follows.

$$\begin{Bmatrix}\dot{x}_1\\ \dot{z}_1\\ \dot{\gamma}_1\end{Bmatrix} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}(rsin(\theta)+rcos(\theta)sin(\psi)) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \times \left[J_x^{-1}J_q\right] \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} + \begin{Bmatrix}0\\ \frac{\dot{\rho}_1}{2}\\ 0\end{Bmatrix} \tag{A.22}$$

Finally, the Jacobian matrix for the bar 1 results in:

$$J_{b_1} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}(rsin(\theta)+rcos(\theta)sin(\psi)) \\ \frac{1}{2} & 0 & -\frac{1}{2}rcos(\theta) \\ 0 & -\frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_1)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_1)} \end{bmatrix} \times \left[J_x^{-1}J_q\right] + \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A.23}$$

With the linear system being

$$\begin{Bmatrix}\dot{x}_1\\ \dot{z}_1\\ \dot{\gamma}_1\end{Bmatrix} = \left[J_{b_1}\right] \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} \tag{A.24}$$

By following a similar procedure for bars 1 and 2, the linear systems and their respective Jacobians are obtained.

$$\begin{Bmatrix}\dot{y}_2\\ \dot{z}_2\\ \dot{\gamma}_2\end{Bmatrix} = \left[J_{b_2}\right] \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} \tag{A.25}$$

$$\begin{Bmatrix}\dot{x}_3\\ \dot{z}_3\\ \dot{\gamma}_3\end{Bmatrix} = \left[J_{b_3}\right] \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} \tag{A.26}$$

With,

$$J_{b_2} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\psi) & 0 \\ 0 & -\frac{cos(\gamma_2)}{2sin(\gamma_2)}rsin(\psi) & 0 \\ 0 & -\frac{rsin(\psi)}{Lsin(\gamma_2)} & 0 \end{bmatrix} \left[J_x^{-1}J_q\right] + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{A.27}$$

$$J_{b_3} = \begin{bmatrix} 0 & -\frac{1}{2}rsin(\theta)cos(\psi) & -\frac{1}{2}(rsin(\theta)-rcos(\theta)sin(\psi)) \\ \frac{1}{2} & 0 & \frac{1}{2}rcos(\theta) \\ 0 & \frac{rsin(\theta)cos(\psi)}{Lsin(\gamma_3)} & -\frac{rsin(\theta)+rcos(\theta)sin(\psi)}{Lsin(\gamma_3)} \end{bmatrix} \times \begin{bmatrix} \\ J_x^{-1}J_q \\ \\ \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix} \tag{A.28}$$

As for the rotation Jacobian of the end effector, the following linear system relates the angular velocities of point p with the angular velocities $\dot{\psi}$ and $\dot{\theta}$.

$$\begin{Bmatrix}w_x\\ w_y\\ w_z\end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \begin{Bmatrix}\dot{\psi}\\ \dot{\theta}\end{Bmatrix} \tag{A.29}$$

Moreover, the equation can be expressed as a function of the inputs as follows.

$$\begin{Bmatrix}w_x\\ w_y\\ w_z\end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \left[J_x^{-1}J_q|_R\right] \begin{Bmatrix}\dot{\rho}_1\\ \dot{\rho}_2\\ \dot{\rho}_3\end{Bmatrix} \tag{A.30}$$

Where $J_x^{-1}J_q|_R$ is the submatrix of $J_x^{-1}J_q$ from row 2 and 3, which is the matrix that relates the rotation motion of point p with the inputs. Hence, the Jacobian matrix of the end effector results in.

$$J_R = \begin{bmatrix} 1 & 0 \\ 0 & cos(\psi) \\ 0 & -sin(\psi) \end{bmatrix} \left[J_x^{-1}J_q|_R\right] \tag{A.31}$$

References

[1] Physik instrumente (pi) gmbh & co. kg. 2015. Last visited on October 14th; http://www.physikinstrumente.com/products/parallel-kinematic-hexapods.html.
[2] Pkmtricept sl. 2015. Last visited on October 15th; http://www.pkmtricept.com/.
[3] Adept technology, inc.2015. Last visited on October 14th; http://www.adept.com/products/robots.
[4] Moog inc. 2015. Last visited on October 15th; http://www.moog.com/products/hexapods-positioning-systems/.
[5] Žlajpah L. Simulation in robotics. Math Comput Simul 2008;79(4):879–97. 5th Vienna International Conference on Mathematical Modelling/Workshop on Scientific Computing in Electronic Engineering of the 2006 International Conference on Computational Science/Structural Dynamical Systems: Computational Aspects; http://www.sciencedirect.com/science/article/pii/S0378475408001183.
[6] Siciliano B, Sciavicco L, Villani L, Oriolo G. Robotic: modelling, planning and control. Advanced Ttxtbooks in control and signal processing. Springer-Verlag London Limited; 2010.
[7] Wang J, Gosselin C, Cheng L. Modeling and simulation of robotic systems with closed kinematic chains using the virtual spring approach. Multibody Syst Dyn 2002;7(2):145–70.
[8] Massimo Callegari MP, Palpacelli M-C. Dynamics modelling and control of the 3-rcc translational platform. Mechatronics 2006(16):589–605.
[9] Lou Y, Li Z, Zhong Y, Li J, Li Z. Dynamics and contouring control of a 3-dof parallel kinematics machine. Mechatronics 2011;21(1):215–26.
[10] Codourey A. Dynamic modeling of parallel robots for computed-torque control implementation. Int J Rob Res 1998;17(12):1325–36.
[11] Yang C, Huang Q, Han J. Computed force and velocity control for spatial multi-dof electro-hydraulic parallel manipulator. Mechatronics 2012;22(6):715–22. Special Issue on Intelligent Mechatronics (LSMS2010 & ICSEE2010)
[12] Ebert-Uphoff I, Gosselin C. Dynamic modeling of a class of spatial statically-balanced parallel platform mechanisms. In: Robotics and automation, 1999. proceedings. 1999 IEEE international conference on, vol. 2; 1999. p. 881–8.
[13] Oftadeh R, Aref M, Taghirad H. Explicit dynamics formulation of stewart-gough platform: A newton-euler approach. In: Intelligent robots and systems (IROS), 2010 IEEE/IRSJ international conference on; 2010. p. 2772–7.
[14] Callegari M, Palpacelli M-C, Principi M. Dynamics modelling and control of the 3-rcc translational platform. Mechatronics 2006;16(10):589–605. http://www.sciencedirect.com/science/article/pii/S0957415806000754.
[15] Abdellatif H, Heimann B. Computational efficient inverse dynamics of 6-dof fully parallel manipulators by using the lagrangian formalism. Mech Mach Theory 2009;44(1):192–207. http://www.sciencedirect.com/science/article/pii/S0094114X08000311.
[16] Ebert-Uphoff K, Kozak I. Review of the role of quasi-coordinates for the kinematic and dynamic modeling of parallel manipulators. In: Proceedings of the workshop on fundamental issues and future research directions for parallel mechanisms and manipulators; 2002. p. 328–38. Quebec.
[17] Lurie A. Analytical mechanics. Foundations of engineering mechanics. Springer Berlin Heidelberg; 2002. ISBN 978-3-642-53650-2.
[18] Magnani G, Rocco P. Mechatronic analysis of a complex transmission chain for performance optimization in a machine tool. Mechatronics 2010;20(1):85–101. Special Issue on Servo Control for Data Storage and Precision Systems, from 17th {IFAC} World Congress 2008.
[19] Ellis G. Control system design guide. Burlington: Academic Press; third edition; 2004. ISBN 978-0-12-237461-6. http://www.sciencedirect.com/science/article/pii/B9780123746160500126.
[20] Groß H, Hamann J, Wiegärtner G. Electrical feed drives in automation: basics, computation, dimensioning. Publicis MCD corporate ublishing, Erlangen and Munich; 2001.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

[21] Paccot F, Andreff N, Martinet P. A review on the dynamic control of parallel kinematic machines: theory and experiments. Int J Rob Res 2009;28(3):395–416. http://ijr.sagepub.com/content/28/3/395.full.pdf+html; http://ijr.sagepub.com/content/28/3/395.abstract.
[22] Flottmeier S, Trchtler A. 2-DOF state control scheme for the motion control of a parallel kinematic machine. In: 2013 Conference on control and fault-tolerant systems (SysTol); 2013. p. 744–9.
[23] Pang H, Shahinpoor M. Inverse dynamics of a parallel manipulator. J Robot Syst 1994;11(8):693–702. http://dx.doi.org/10.1002/rob.4620110803.
[24] Tsai M-S, Yuan W-H. Inverse dynamics analysis for a 3-prs parallel mechanism based on a special decomposition of the reaction forces. Mech Mach Theory 2010;45(11):1491–508. http://www.sciencedirect.com/science/article/pii/S0094114X10001229.
[25] Igus-GmbH. Drylin zlw -1040 toothed belt axis online catalog. 2016. http://www.igus.eu/wpck/4773/zlw_1040.
[26] Maxon. Maxon motor online catalog. 2016. http://www.maxonmotor.com/maxon/view/catalog/.
[27] Carretero J, Podhorodeski R, Nahon M, Gosselin C. Kinematic analysis and optimization of a new three degree-of-freedom spatial parallel manipulator. ASME J Mech Des 1999;122(1):17–24.

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

Constantino Roldán Paraponiaris obtained his undergraduate in Mechanical Engineering. Furthermore, he received his MSc. in Mechanical Engineering Design and Manufacturing from the University of the Basque Country (UPV-EHU, Spain) in 2011. He is currently a Graduate Student and Assistant Researcher under the supervision of Fran Campa and Oscar Altuzarra in the Department of Mechanical Engineering at the UPV-EHU. He works in the fields of Mechanical Systems and Mechatronics, focusing on: System Identification and Modeling, Real-Time Control programming, and Control Design and Integration on Mechanical Systems.

Francisco J. Campa received his MSc Mechanical Engineering degree from the Faculty of Engineering of Bilbao, Universidad del PaÍs Vasco (UPV/EHU), Spain, in 2003 and the Ph.D. degree in Mechanical Engineering from the same University in 2010. Since 2008, he has been a lecturer in the Department of Mechanical Engineering at the Faculty of Engineering of Bilbao, Universidad del PaÍs Vasco (UPV/EHU). His research interests are Dynamics of Mechanical Systems and Manufacturing Processes, Machine-Tools and Mechatronics.

Oscar Altuzarra received his MSc Mechanical Engineering degree from the Faculty of Engineering of Bilbao, Universidad del País Vasco (UPV/EHU), Spain, in 1995 and the Ph.D. degree in Mechanical Engineering from the same University in 1999. He also attended a course in Mechanical Engineering at Coventry University, Coventry, UK in 1992. Since 1996, he has been in the Department of Mechanical Engineering at the Faculty of Engineering of Bilbao, Universidad del País Vasco (UPV/EHU). In 2010 he obtained the position of Full Professor in Mechanical Engineering. His educational duties are in the subjects related to Fundamental Mechanics and Theoretical Kinematics. His research interests are theoretical kinematics, mechanisms, design of parallel kinematic machines, robotics, and computational analysis of the kinematics and dynamics of complex mechanical problems. He is ASME and IEEE member, a member of the IFToMM Technical Committee for Robotics, and Associate Editor of the Journal of Mechanical Design since 2012. He has been the coordinator of the Spanish Research Network on Mechanical Engineering in 2011, an initiative of the Spanish Association of Mechanical Engineers (AEIM). He is also serving as assistant coordinator evaluating research projects in the field of mechanisms and machines for the National Agency for Research Evaluation (ANEP), that asses research projects for public funding in Spain.
His Research Lines:
- Research on Computational Kinematics
- Research on Lower Mobility Parallel Manipulators with Mixed Freedoms
- Research on Multi Axial Simulation Tables

Please cite this article as: C. Roldán-Paraponiaris et al., Mechatronic modeling of a parallel kinematics multi-axial simulation table based on decoupling the actuators and manipulator dynamics, Mechatronics (2016), http://dx.doi.org/10.1016/j.mechatronics.2016.10.017

Mechatronics 39 (2016) 77–88

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Experimental validation of the kinematic design of 3-PRS compliant parallel mechanisms

A. Ruiz, F. J. Campa*, C. Roldán-Paraponiaris, O. Altuzarra, C. Pinto

Department of Mechanical Engineering, University of the Basque Country UPV/EHU, Escuela Superior de Ingeniería de Bilbao, Alameda Urquijo s/n, 48013, Bilbao, Spain

ARTICLE INFO

Article history:
Received 11 December 2015
Revised 18 July 2016
Accepted 15 August 2016
Available online 24 August 2016

Keywords:
Parallel kinematics
Compliant mechanisms
Flexure joints design

ABSTRACT

In this paper, a procedure for the kinematic design of a 3-PRS compliant parallel manipulator of 3 degree of freedom is proposed. First, under the assumption of small displacements, the solid body kinematics of the 3-PRS has been studied, performing a comprehensive analysis of the inverse and forward kinematic problem, and calculating the rotations that the revolute and spherical flexure joints must perform. Then, after defining some design requirements and therefore the necessary displacements to fulfill, a design process based on the finite element calculations has been stablished, giving the necessary guidelines to reach the optimal solution on a 3-PRS compliant mechanism. Also, a prototype has been tested, using a coordinate measuring machine to verify its dimensions and the resulting displacements in the end effector and the actuated joints. Finally, those measurements have been compared with the FEM and the rigid body kinematics predictions, contrasting the validity of those two modelling approaches for the kinematic design of compliant mechanisms.

1. Introduction

At present, the requirements of precision and surface finish in different areas as scientific, medical, metrological and communication fields have led to the necessity of creating new mechanisms capable of achieving accuracy constraints on the order of microns or nanometers. A well-known solution for many of these high-tech applications are the compliant mechanisms [1], which allow generating motions with great repeatability, precision, speed and a high bandwidth, through the controlled deformation in several points of their monolithic structure. Those flexible points are flexure joints, that substitute the traditional joints used in typical mechanisms overcoming the problems they usually suffer, as friction, clearances, hysteresis and wear. Nevertheless, the main drawback of these mechanisms is a more complex design, because it is necessary to control the elastic deformation of the joints, and the appearance of parasitic motions together with the desired displacements. For that reason, it is important to develop approaches to study, design and validate these compliant machines, which is one of the objectives of the present work.

Here, the design of a 3-PRS compliant parallel mechanism [2] is proposed. It is a fusion of a parallel mechanism [3–6] and a compliant mechanism, so the advantages of both devices can be obtained: higher mechanical stiffness, higher loading capacity, and higher positioning accuracy from the parallel mechanism and the zero backlash, no friction, no need for lubrication from the flexure devices. The axisymmetric arrangement of a 3-PRS causes that it is a common solution among the compliant parallel mechanisms for precision operations.

As a function of the kinematics of the device and the geometry of the joints, it is possible to create a large number of combinations. An example of a 1 degree of freedom (dof) compliant device is presented by Kim et al. in [7], that consists of a single-axis flexure-based nano-positioning stage with a range of motion up to a millimeter and a compact stage. A reduction of the parasitic movements and an increase of the accuracy are achieved by means of their process design, providing an optimization of the structure. Others examples of 1 dof compliant devices can be observed in [8–11]. In the same way, it is possible to develop mechanisms with 2 dof, as for example the stage presented by Wang et al. in [12], where the design of a planar motion stage based on flexure elements allows large ranges of motion along x and y direction without causing over-constraint or significant errors. Also, the authors provide some useful steps for designing and analyze by FEM compliant devices. Another case of 2 dof compliant mechanism is presented in [13], where the design of a flexure-based XY stage tries to obtain a relatively large range and high scanning speed device. Its experimental setup and characterization is shown

* Corresponding author: Fax: +34 946014215.
E-mail address: fran.campa@ehu.eus (F. J. Campa).

http://dx.doi.org/10.1016/j.mechatronics.2016.08.006
0957-4158/

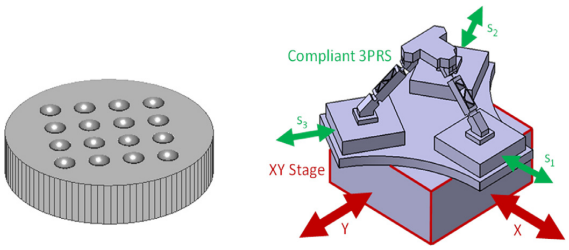


Fig. 1. Workpiece and schematic of the hybrid manipulator.

based on the FEM analysis performed. Also, large number of 3 dof mechanisms have been developed in the current literature. For example, an ultra-precision XYθz flexure stage with nanometer accuracy is presented by Kim et al. in [14]. Other sample of this type of devices is shown in [15], where a high-performance three-axis serial-kinematic nano-positioning stage for high-bandwidth applications is presented by Kenton and Leang. Moreover, the necessary steps to achieve a good design, characterization and control are presented. Another 3 dof mechanism is presented in [16], where a XYZ compliant parallel manipulator composed by identical spatial double four-beam modules, is presented. Additional examples of this type of devices are presented in [2,17]. To conclude, some examples of 6 dof mechanisms are indicated. A micro-scale manipulator based on a six-DOF compliant parallel mechanism is presented by Liang et al. [18]. This design is featured by piezo-driven actuators and integrated force sensor capable of delivering six-DOF motions with high precision and providing real-time force information for feedback control. On the other hand, a low stiffness 6 degrees of freedom compliant precision stage is given in [19]. In it, Dunning et al detail the dimensional optimization of the design and the process of experimental validation of the prototype. Further instances of 6 dof devices are shown in [20,21].

The present work will be focused on the kinematics study, design process, construction and experimental validation of the kinematics of a prototype of the 3-PRS compliant mechanism. Firstly, the proposed design approach is explained and developed in Sections 2–5, where the rigid body kinematics of the 3-PRS will be developed, the kinematic requirements for the actuators and joints will be calculated and the design process of the flexible joints by FEM will be explained. Finally, the experimental procedure to verify the kinematics of a developed prototype will be shown. To end up, a discussion about the results reached by the ideal kinematics of a 3-PRS, the FEM kinematics and the prototype will be made.

2. Procedure for the kinematic design

The background of the present work is the development of a hybrid milling machine of 5 degrees of freedom for milling moulds for microlenses, where the spindle is fixed to a portal and the workpiece is manipulated by a XY stage under a 3-PRS compliant parallel mechanism with direct measurement of the platform position. The milling of microlenses consists on milling a matrix of N × N concave aspherical cavities on one face of a cylindrical workpiece of diameters ranging from 10 to 20 mm. The lenses have diameters from 0.5 mm to 2 mm and a sagittal depth less than 1 mm. Hence, in the hybrid manipulator, the XY stage carries out the long travel range in X and Y while the parallel kinematics stage has to provide mainly the Z motion and, if possible, two rotations around X and Y to improve the tool orientation, see Fig. 1. The XY stage also compensates the parasitic motions in X and Y of the parallel kinematics stage.

To obtain the kinematic requirements in terms of displacements needed, a standard milling process has been designed, with a matrix of 4 × 4 cavities in a cylindrical mould of 18 mm. The cavities are spherical with a diameter of 2 mm, and are machined in down-milling with a spiral down strategy in counter-clock direction. The offset between the part and the tool tip for the motions between cavities is of 1 mm. As a conclusion, the needed range in X and Y is ±10 mm and in Z ±2 mm, with the objective of maximize the two rotations as much as possible. As there exists commercial solutions for the XY stage, the final purpose of this work is the design of the compliant parallel stage. A compliant solution has been considered as the structural requirements are not very demanding, given the fact that the estimated cutting forces of the micromilling process are below 1 N, and the travel range can be reached with enough precision. The selection of a 3-PRS is due to its stiffness and thermal properties due to its axisymmetric configuration.

Hence, the aim of this study is to achieve the kinematic design of a 3-PRS compliant parallel manipulator capable of performing displacements in Z direction as well as two rotations around X and Y axes. To reach this objective, the following procedure has been developed. Firstly, the rigid body kinematics of the 3-PRS is solved, assuming that under small displacements, there will not appear parasitic translations in the compliant joints that change the rotation center location, hence, the kinematics of the rigid body mechanism will be similar to the compliant one. This assumption is a common practice designing compliant mechanisms [7,17,18]. The rigid body kinematics provides the necessary displacements in the actuators as well as the displacements that the flexure joints must fulfill. Once these requirements are stablished, it is possible to design the flexure joints by means of an iterative process based on FEM calculations, where the stress in the joints becomes a crucial design parameter. First, the FEM study is applied to an isolated flexure joint, to analyze if provides the demanded displacements, and the behavior of those joints in the whole manipulator are again analyzed by FEM, to reach a homogenous distribution of the stress in all the joints. In the following sections, each of these points will be widely explained, providing the achieved relations, the developments for the design and the validation, and the obtained results, both theoretical as experimental ones.

3. Rigid body kinematics of the 3- PRS

The rigid body kinematics has been solved following the work from [22]. The notation P, R, S denotes prismatic, revolute and spherical joint, respectively. Each limb connects the base to the moving platform by a P joint, an R joint and an S joint in sequence, where the P joint is actuated by a linear actuator. Thus, the moving platform is attached to the base by three identical PRS linkages, $\mathbf{C_iB_i}$, whose length is L, see Fig. 2. The points B_i are located in the moving platform in a circumference of radius b, whose center is defined as point P. The angles between the legs and the horizontal plane are α_i. which is equal to 45° in the default position. Three prismatic actuators at 120° are used; being the joint space coordinates s_i. The location of the actuators in the zero position is defined by the points A_i, placed on a circle of radius a, whose center is defined by the point O.

Two reference systems are defined to solve the kinematics. A fixed frame {x,y,z} is located at point O, whose X axis is coincident with the slider direction $\mathbf{OA_1}$ and its Z axis is placed vertically. In the same way, a moving frame {u, v, w} is laid at point P, whose U axis is coincident with the direction $\mathbf{PB_1}$ and its W axis is perpendicular to the moving platform.

In the following subsections, the motion capabilities and the main parameters of the mechanism will be displayed for a better understanding.

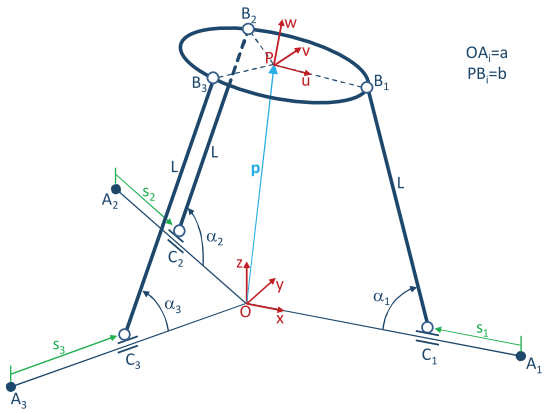


Fig. 2. Schematic of the 3-PRS manipulator analyzed.

3.1. Parasitic movements

The 3-PRS mechanism belongs to the zero torsion parallel mechanisms group [23], and although it is commonly considered as a 3 degrees of freedom manipulator, that is, translation in Z and rotations around X and Y, the truth is that once a set of input displacements is introduced, in the platform there appear three small coupled motions, two translations in X and Y and a rotation around Z. As they are not the desired degrees of freedom, they are usually called parasitic motions. In this section, these parasitic movements will be calculated as they will be compensated by the XY stage below the parallel manipulator.

To begin with, the position of the center of the moving platform is described by the coordinates p_x, p_y, and p_z, and its orientation is given by three angles Ψ, θ and ϕ, which are defined as rotations around the X, Y and Z axes of the fixed frame. Therefore, the position of the center of the moving platform can be expressed as:

$$\mathbf{OP} = \begin{Bmatrix} p_x & p_y & p_z \end{Bmatrix}^T \tag{1}$$

On the other hand, the zero position of the actuators in the fixed frame {x,y,z} is defined by:

$$\begin{aligned} \mathbf{OA_1} &= \begin{Bmatrix} a & 0 & 0 \end{Bmatrix}^T \\ \mathbf{OA_2} &= \begin{Bmatrix} -a/2 & a\sqrt{3}/2 & 0 \end{Bmatrix}^T \\ \mathbf{OA_3} &= \begin{Bmatrix} -a/2 & -a\sqrt{3}/2 & 0 \end{Bmatrix}^T \end{aligned} \tag{2}$$

Also, it is possible to define the position of the spherical joints in the moving frame {u,v,w} by means of the following relations; where the sub-index 'm' denotes that the vectors are expressed in the moving frame.

$$\begin{aligned} \mathbf{PB_{1m}} &= \begin{Bmatrix} b & 0 & 0 \end{Bmatrix}^T \\ \mathbf{PB_{2m}} &= \begin{Bmatrix} -b/2 & b\sqrt{3}/2 & 0 \end{Bmatrix}^T \\ \mathbf{PB_{3m}} &= \begin{Bmatrix} -b/2 & -b\sqrt{3}/2 & 0 \end{Bmatrix}^T \end{aligned} \tag{3}$$

To transform any vector from the moving frame to the fixed frame, it is necessary to know the rotation matrix **R** that relates both systems, which is expressed as follows:

$$\begin{aligned} \mathbf{R} = \mathbf{R_y}(\theta)\mathbf{R_x}(\psi)\mathbf{R_z}(\phi) &= \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \\ &= \begin{bmatrix} c\theta c\phi + s\psi s\theta s\phi & -c\theta s\phi + s\psi s\theta c\phi & c\psi s\theta \\ c\psi s\phi & c\psi c\phi & -s\psi \\ -s\theta c\phi + s\psi c\theta s\phi & s\theta s\phi + s\psi c\theta c\phi & c\psi c\theta \end{bmatrix} \end{aligned} \tag{4}$$

where c stands for to cosine and s refers to the sine. Also, u, v and w are defined as three unit vectors along the U, V and W axes of the moving reference system *P*.

The position of each spherical joint regarding to the fixed reference system can be expressed as:

$$\mathbf{OB_i} = \mathbf{OP} + \mathbf{R} \cdot \mathbf{PB_{im}} \quad i = 1, 2, 3 \tag{5}$$

Developing $\mathbf{OB_i}$ for each limb, the following three equations are obtained:

$$\begin{aligned} \mathbf{OB_1} &= \begin{Bmatrix} p_x + u_x b & p_y + u_y b & p_z + u_z b \end{Bmatrix}^T \\ \mathbf{OB_2} &= \begin{Bmatrix} p_x - \frac{u_x b}{2} + \frac{\sqrt{3}v_x b}{2} & p_y - \frac{u_y b}{2} + \frac{\sqrt{3}v_y b}{2} & p_z - \frac{u_z b}{2} + \frac{\sqrt{3}v_z b}{2} \end{Bmatrix}^T \\ \mathbf{OB_3} &= \begin{Bmatrix} p_x - \frac{u_x b}{2} - \frac{\sqrt{3}v_x b}{2} & p_y - \frac{u_y b}{2} - \frac{\sqrt{3}v_y b}{2} & p_z - \frac{u_z b}{2} - \frac{\sqrt{3}v_z b}{2} \end{Bmatrix}^T \end{aligned} \tag{6}$$

The revolute hinges set that the spherical joints can only move in the fixed planes defined by the linear actuators $\mathbf{OA_i}$ and the legs of the manipulator $\mathbf{C_iB_i}$. Therefore, the following three mechanical constraints are imposed to the mechanism:

$$OB_{1y} = 0 \tag{7}$$

$$OB_{2y} = -\sqrt{3}OB_{2x} \tag{8}$$

$$OB_{3y} = \sqrt{3}OB_{3x} \tag{9}$$

Substituting the expressions of $\mathbf{OB_i}$ from the Eq. (6) into the Eqs. (7)–(9), and developing these expressions, the following equivalences are reached:

$$p_y = -u_y b \tag{10}$$

$$p_x = \frac{b}{2}(u_x - v_y) \tag{11}$$

$$u_y = v_x \tag{12}$$

Substituting the expressions of u_i and v_i from the rotation matrix **R**, in Eq. (4), into Eqs. (10) and (11), the 3-PRS dependent variables or parasitic motions are obtained:

$$\phi = \operatorname{atan}\left(\frac{\sin\psi\sin\theta}{\cos\psi + \cos\theta}\right) \tag{13}$$

$$p_x = \frac{b}{2}(\cos\theta\cos\phi + \sin\psi\sin\theta\sin\phi - \cos\psi\cos\phi) \tag{14}$$

$$p_y = -b\cos\psi\sin\phi \tag{15}$$

From Eq. (13) it can be stated that for small rotations around Z and Y, the parasitic rotation ϕ will be negligible.

3.2. Inverse kinematics modeling

The aim of the inverse kinematic problem is to find the values that should take the active joint coordinates of the mechanism s_i, to reach a given position and orientation of the platform. Firstly, referring to Fig. 2, the following relationships can be obtained.

$$\mathbf{A_iB_i} = \mathbf{OB_i} - \mathbf{OA_i} \quad i = 1, 2, 3 \tag{16}$$

Where $\mathbf{OB_i}$ and $\mathbf{OA_i}$ are expressed in Eq. (6) and (2). Also, the expression for $\mathbf{C_iB_i}$ can be written as:

$$\mathbf{C_iB_i} = L \cdot \mathbf{l_{i0}} = \mathbf{A_iB_i} - \mathbf{A_iC_i} = \mathbf{A_iB_i} - s_i \cdot \mathbf{s_{i0}} \quad i = 1, 2, 3 \tag{17}$$

Where L is the length of the limbs, $\mathbf{l_{i0}}$ represents the unit vector along the direction of each leg and the unit vectors $\mathbf{s_{i0}}$ define the positive direction of motion of the actuators and are written as:

$$\mathbf{s_{i0}} = -\frac{\mathbf{OA_i}}{OA_i} \quad i = 1, 2, 3 \tag{18}$$

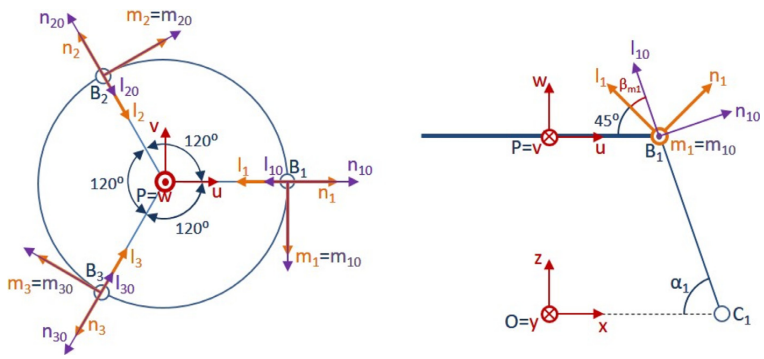


Fig. 3. Reference systems for the spherical joints rotation study.

Rearranging and squaring the components of the Eq. (17), a quadratic equation is found.

$$s_i^2 - 2s_i \cdot \mathbf{s_{i0}} \cdot \mathbf{A_iB_i} + \mathbf{A_iB_i} \cdot \mathbf{A_iB_i} - L^2 = 0 \qquad i = 1, 2, 3 \tag{19}$$

Solving Eq. (19) allows us to obtain the solutions for the inverse kinematic problem.

$$s_i = (\mathbf{s_{i0}} \cdot \mathbf{A_iB_i}) \pm \sqrt{(\mathbf{s_{i0}} \cdot \mathbf{A_iB_i})^2 - \mathbf{A_iB_i} \cdot \mathbf{A_iB_i} + L^2} \qquad i = 1, 2, 3 \tag{20}$$

3.3. Passive coordinates

In this subsection, the expressions for the angles of the both joints will be developed to calculate the rotations that are produced in the hinges while the manipulator is performing a programmed operation.

3.3.1. Rotation at revolute joints

The rotation in the revolute joints as consequence of a displacement of the manipulator can be obtained solving the following development. Referring to Fig. 2:

$$\mathbf{l_{i0}} = \frac{\mathbf{C_iB_i}}{L} \qquad i = 1, 2, 3 \tag{21}$$

Knowing this information, the angles in the revolute joints can be calculated using the dot product of $\mathbf{s_{i0}}$ and $\mathbf{l_{i0}}$, which have been shown in Eqs. (17) and (18).

$$\alpha_i = \operatorname{acos}\left(\frac{\mathbf{l_{i0}} \cdot \mathbf{s_{i0}}}{l_{i0} \cdot s_{i0}}\right) \qquad i = 1, 2, 3 \tag{22}$$

3.3.2. Rotation at spherical joints

To study the rotation at the spherical joints, two sets of moving frames have been used. The first set, $\mathbf{F_i} = \{m_i\ n_i\ l_i\}$, will be fixed to the moving platform, and the other, $\mathbf{F_{io}} = \{m_{i0}\ n_{i0}\ l_{i0}\}$, will move together with the leg. Both have the origin located at the spherical joints B_i and overlap in the default position, with $\mathbf{m_{i0}}$ and $\mathbf{m_i}$ parallel to each revolute joint axis and $\mathbf{l_{i0}}$ and $\mathbf{l_i}$ aligned with each limb. A scheme can be seen in Fig. 3 for a pure vertical translation of the moving platform. Rotation angles around the m- and n-axes, β_{mi} and β_{ni}, will make reference to the deflection of the joint and rotation around the l-axis, β_{li}, will be the torsional deformation.

Let's consider here the following four rotation matrices: $\mathbf{R_{im}}$ to relate $\mathbf{F_i}$ with {u v w}, $\mathbf{R}$ mentioned in Eq. (4) to relate {u v w} with {x y z}, $\mathbf{R_{iSjoint}}$ reflects the spherical joints rotation and relates $\mathbf{F_i}$ with $\mathbf{F_{i0}}$, and $\mathbf{R_{iRjoint}}$ reflects the revolute joints rotation and relates $\mathbf{F_{i0}}$ with {x y z}. They meet the following relation:

$$\mathbf{R_{iRjoint}} \cdot \mathbf{R_{iSjoint}} = \mathbf{R} \cdot \mathbf{R_{im}} \qquad i = 1, 2, 3 \tag{23}$$

Hence, to obtain the spherical joint rotation, the angles between frames $\mathbf{F_i}$ and $\mathbf{F_{i0}}$ must be calculated, which means calculating $\mathbf{R_{iSjoint}}$ as a function of the other matrices:

$$\mathbf{R_{iSjoint}} = \mathbf{R}^T_{\mathbf{iRjoint}} \cdot \mathbf{R} \cdot \mathbf{R_{im}} \qquad i = 1, 2, 3 \tag{24}$$

To do that, first, vectors $\mathbf{m_{i0}}$ are developed as the characteristic vectors of the three planes at 120º that constrain the motion of each leg to each plane.

$$\begin{aligned} \mathbf{m_{10}} &= \{0 \quad -1 \quad 0\}^T \\ \mathbf{m_{20}} &= \left\{\tfrac{\sqrt{3}}{2} \quad \tfrac{1}{2} \quad 0\right\}^T \\ \mathbf{m_{30}} &= \left\{\tfrac{-\sqrt{3}}{2} \quad \tfrac{1}{2} \quad 0\right\}^T \end{aligned} \tag{25}$$

Unit vectors $\mathbf{l_{io}}$ are calculated as shown in Eq. (21). Finally, the unit vectors $\mathbf{n_{i0}}$ can be obtained by means of the cross product of $\mathbf{l_{io}}$ and $\mathbf{m_{i0}}$.

$$\mathbf{n_{i0}} = \mathbf{l_{i0}} \wedge \mathbf{m_{i0}} \qquad i = 1, 2, 3 \tag{26}$$

As a result, the rotation matrices $\mathbf{R_{iRjoint}}$ that relate the frames of the legs $\mathbf{F_{io}}$ with the global reference system {x, y, z} are:

$$\mathbf{R_{iRjoint}} = \begin{bmatrix} \mathbf{m_{i0}} & \mathbf{n_{i0}} & \mathbf{l_{i0}} \end{bmatrix} \qquad i = 1, 2, 3 \tag{27}$$

On the other hand, the rotation matrices $\mathbf{R_{im}}$, that relate $\mathbf{F_i}$ and {u, v, w} systems, are calculated. The unit vectors for each system $\mathbf{F_i}$ can be defined by the following expressions in the moving frame:

$$\begin{aligned} \mathbf{m_{1m}} &= \{0 \quad -1 \quad 0\}^T \\ \mathbf{n_{1m}} &= \{\sin\alpha_0 \quad 0 \quad \cos\alpha_0\}^T \\ \mathbf{l_{1m}} &= \{-\cos\alpha_0 \quad 0 \quad \sin\alpha_0\}^T \end{aligned} \tag{28}$$

$$\begin{aligned} \mathbf{m_{2m}} &= \left\{\tfrac{\sqrt{3}}{2} \quad \tfrac{1}{2} \quad 0\right\}^T \\ \mathbf{n_{2m}} &= \left\{\tfrac{-\sin\alpha_0}{2} \quad \tfrac{\sqrt{3}\sin\alpha_0}{2} \quad \cos\alpha_0\right\}^T \\ \mathbf{l_{2m}} &= \left\{\tfrac{\cos\alpha_0}{2} \quad \tfrac{-\sqrt{3}\cos\alpha_0}{2} \quad \sin\alpha_0\right\}^T \end{aligned} \tag{29}$$

$$\begin{aligned} \mathbf{m_{3m}} &= \left\{-\tfrac{\sqrt{3}}{2} \quad \tfrac{1}{2} \quad 0\right\}^T \\ \mathbf{n_{3m}} &= \left\{\tfrac{-\sin\alpha_0}{2} \quad \tfrac{-\sqrt{3}\sin\alpha_0}{2} \quad \cos\alpha_0\right\}^T \\ \mathbf{l_{3m}} &= \left\{\tfrac{\cos\alpha_0}{2} \quad \tfrac{\sqrt{3}\cos\alpha_0}{2} \quad \sin\alpha_0\right\}^T \end{aligned} \tag{30}$$

Being $\alpha_0 = 45^\circ$ the default value of α_i in the proposed case of study. Hence, each rotation matrix $\mathbf{R_{im}}$ has the following expression:

$$\mathbf{R_{im}} = \begin{bmatrix} \mathbf{m_{im}} & \mathbf{n_{im}} & \mathbf{l_{im}} \end{bmatrix} \qquad i = 1, 2, 3 \tag{31}$$

Substituting Eq. (31), Eq. (27) and Eq. (4) in Eq. (24), $\mathbf{R_{iSjoint}}$ is obtained. On the other hand, this matrix can be developed as

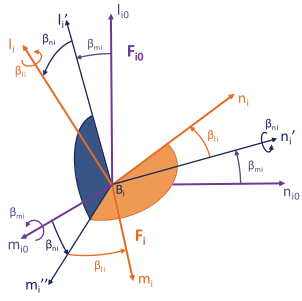


Fig. 4. Rotations performed between systems F_i and F_{i0}.

a function of three rotations around all the axes system $\mathbf{F_{i0}}$: β_{mi}, β_{ni}, and β_{li}, see Fig. 4. The sequence consists of a rotation around the m-axis, followed by a rotation around the n-axis and ending with a rotation around the l-axis. Therefore, the rotation matrix that relates the systems $\mathbf{F_i}$ with $\mathbf{F_{i0}}$ may also be expressed as the product of the above rotation matrices.

$$\mathbf{R_{iSjoint}} = \mathbf{R_{mi}R_{ni}R_{li}} = \begin{bmatrix} c\beta_{ni}\,c\beta_{li} & -c\beta_{ni}\,s\beta_{li} & s\beta_{ni} \\ s\beta_{mi}\,s\beta_{ni}\,c\beta_{li}+c\beta_{mi}\,s\beta_{li} & -s\beta_{mi}\,s\beta_{ni}\,s\beta_{li}+c\beta_{mi}\,c\beta_{li} & -s\beta_{mi}\,c\beta_{ni} \\ -c\beta_{mi}\,s\beta_{ni}\,c\beta_{li}+s\beta_{mi}\,s\beta_{li} & c\beta_{mi}\,s\beta_{ni}\,s\beta_{li}+s\beta_{mi}\,c\beta_{li} & c\beta_{mi}\,c\beta_{ni} \end{bmatrix} \tag{32}$$

Equaling the terms of the matrices in Eq. (24) and (32), the rotations in the spherical joints are obtained:

$$\begin{aligned} \beta_{mi} &= \operatorname{atan}(-\mathbf{R_{i-i0}}(2,3)/\mathbf{R_{i-i0}}(3,3)) \quad i=1,2,3 \\ \beta_{ni} &= \operatorname{asin}(\mathbf{R_{i-i0}}(1,3)) \\ \beta_{li} &= \operatorname{atan}(-\mathbf{R_{i-i0}}(1,2)/\mathbf{R_{i-i0}}(1,1)) \end{aligned} \tag{33}$$

3.4. Forward kinematics modeling

The forward kinematic problem determines the position and orientation of the platform with respect to the fixed frame, when the values of the actuated joints are known. Referring to Fig. 2, the following equivalences can be obtained by imposing a fixed length, L, for each limb of the mechanism:

$$\mathbf{OB_i} - \mathbf{OC_i} = \mathbf{OB_i} - (\mathbf{OA_i}+\mathbf{AC_i}) = \mathbf{C_iB_i} \quad i=1,2,3 \tag{34}$$

$$|\mathbf{OB_i} - \mathbf{OC_i}|^2 = L^2 \tag{35}$$

The developed expressions can be observed in the **Appendix A**. The previous calculated relationships together with the Eqs. (13)–(15) form a system of six equations which must be solved. As inputs for the system, the displacements of the actuators, $\mathbf{s_i}$, and the initial position of the platform $\{p_{x0}, p_{y0}, \phi_0\}$ are introduced. By means of an algebraic-loop solver in Matlab Simulink, the real position of the platform p_z, Ψ, θ is calculated. These solutions will be used as direct feedback to the resolution process, see Fig. 5.

4. Joint requirements

As stated in Section 2, the compliant 3-PRS must be able to achieve a Z travel of ±2 mm. With regard to the rotations around the X and Y axes, an objective has been set in ensuring minimum values of ±0.2° over the entire range of Z. After taking into account the whole footprint of the hybrid manipulator, the modal frequencies and the minimum static stiffness needed in the platform to minimize the legs deflection due to the cutting forces, the dimensions selected for the prototype are a length L of the legs of 109.215 mm, measured between the center points of the two flexure stages, a distance b from the platform axis to the center of the spherical joints of 47.91 mm, and a value of 45° for the three passive angles α in the default position. With 45°, no amplification happens, so the prismatic joints actuators must provide at least a travel range of ±2 mm. The section of the legs is $10 \times 10\ mm^2$. An aluminum alloy 7075 with a Young modulus of 72 GPa, a Poisson's ratio of 0.33 and a density of $2.81\ g/cm^3$ has been used for the first prototype.

These requirements have been introduced in the inverse kinematics problem (IKP) and the displacements on the actuators and the passive angles in the joints have been calculated. In Fig. 6, the studied positions are shown. As it can be seen in the first three graphs, the IKP has been solved with all the possible combinations of a Z displacement of ±2 mm and rotations ψ and θ of ±0.2°. As a result, the required displacements of the actuated joints, s_i, are achieved. They are comprised between −2.15 mm and 2.30 mm, see the fourth graph of Fig. 6. With regard to the rotation of the revolute joints, α_i must reach a variation of ±1.7° and regarding the rotations in the spherical hinges, the angles that must be reached are $\beta_m = \pm 1.9°$, $\beta_n = \pm 0.2°$ and $\beta_l = \pm 0.2°$, see Fig. 7.

5. FEM based design of the 3- PRS compliant mechanism joints

To design and analyze the performance of the prototype, simulations using ANSYS Workbench FEM software have been made and the main features of the flexure stage have been obtained. In Fig. 8 an example of the designed stage and the mesh is shown. The mesh applied to the mechanism consists of a tetrahedral mesh. The nodes size in the flexure joints has been reduced with a ratio 10:1 respect to the whole structure to obtain more accurate results in the areas with a high deformation and stress concentration. Quadratic tetrahedrons have been used for the mesh.

To obtain the final dimensions of the flexure joints, an optimization process has been developed. The first step has been to know the displacement requirements of the joints by means of the inverse kinematics, as explained in Section 3. Once this information is known, the design of the joints can be done [24], attending to the following conditions: a) the joints must be able to achieve the required movements, and b) the maximum stress has to be controlled to not enter in the plastic zone. The Al 7075 tensile yield strength is equal to 503 MPa. A safety factor has been established, so a stress greater than 400 MPa has not been overcome.

To characterize the revolute hinge, one side has been fixed and a moment has been applied in the other side. For the spherical joint, as in the previous case, one side has been fixed and two moments have been applied in the other side. In both cases, the fixed side and the applied loads have been located at a distance of 20 mm from the compliant bodies [25]. The aim is to avoid local disturbances by the boundary conditions in the points where the measurements have been performed and therefore, a reduction in the accuracy of the results. With these configurations, it is possible to measure both the maximum stresses and displacements supported in the joints, see Fig. 9.

Once the potential joints are obtained, they are introduced in the whole structure. It should be noted that their behavior can vary when they are subjected to the working conditions of the mechanism. For that reason, the next step has been to calculate the features of the whole structure. A static structural analysis has been performed, applying several force steps in the place of the actuators. Thanks to this, the relations between the applied forces and movements in the actuators and the displacements on the platform have been calculated. What is more, the maximum stress in the structure, which occurs in one of the flexure joints, has been controlled to avoid exceeding a stress greater than 400 MPa, as in the joints design. Also the maximum

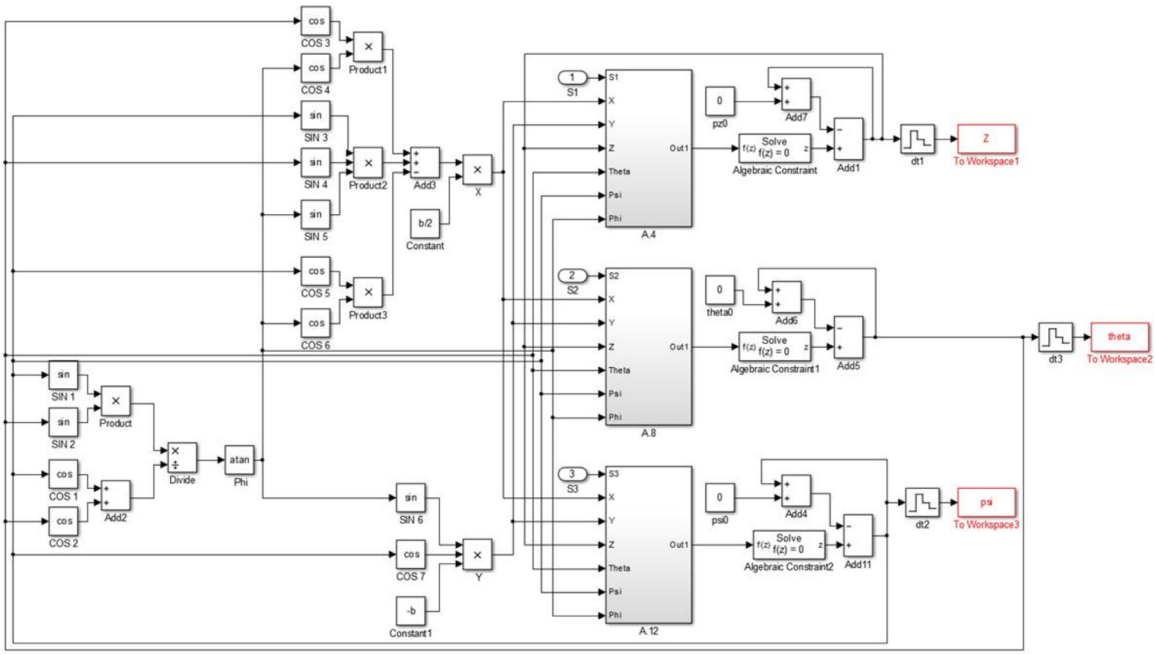


Fig. 5. Forward kinematic resolution scheme.

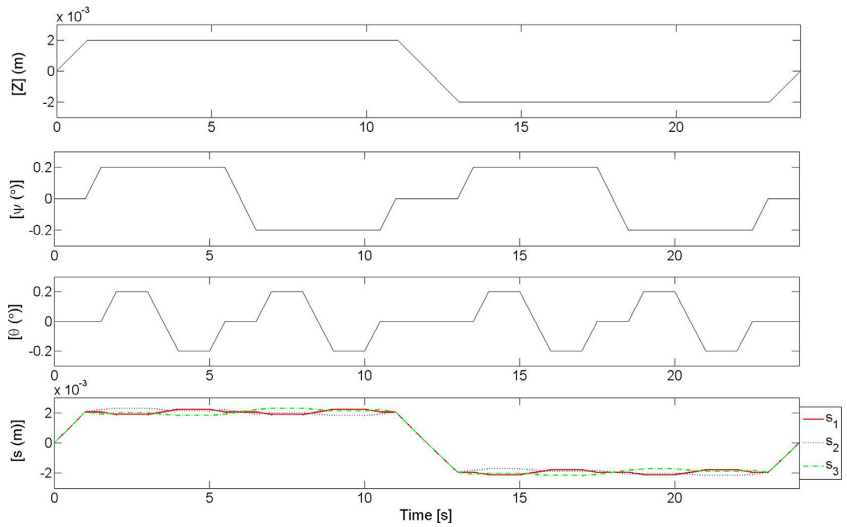


Fig. 6. Movements on the platform (z, ψ, θ) and displacements in the actuators (s_i).

stresses supported in the hinges have been controlled to achieve similar values in both revolute and spherical joints. The objective is to avoid weaknesses in the structure due to higher efforts on specific type of compliant hinge, see Fig. 10. A relation between the maximum stresses supported in the joints comprised between 1 ± 0.2 has been stablished as a design condition. Regarding the computational time, an average joint analysis has taken 384.3 s and an average analysis for the whole manipulator has taken 3846.2 s with an Intel-Core i5-2430 M with 2.4 GHz and 6 GB of RAM.

As design parameters, the radius and thickness for the revolute joints, and radius, thickness and length for the spherical joints have been chosen. Depending of the variation of any of these items, new features in the mechanism can be obtained. From the authors experience designing a compliant 3-PRS, the influences of all of them have been determined. Some useful indications to get a new mechanism $i+1$ starting from a previous case design i are shown in Table 1. As can be seen, for instance, an increase in the thickness of the revolute joint causes a significant increment in the stiffness of the structure, by a factor greater than 1.5, while the Z

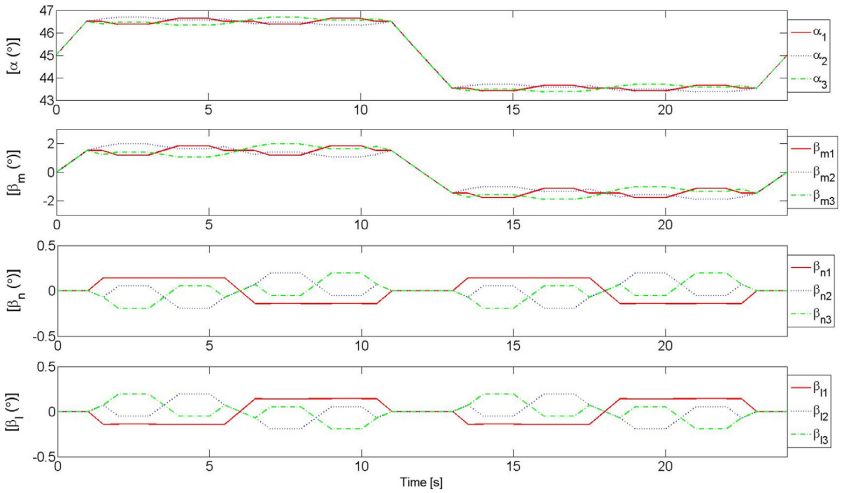


Fig. 7. Rotations in revolute joints (α_i) and in spherical joints (β_{mi}, β_{ni}, β_{li}).

Table 1
Influence of the design parameters in the structure.

	Positive increase	Structural Stiffness (n_{i+1}/ n_i)	Displacement Z direction (d_{i+1}/ d_i)	Relation Max. Stress in joints (T_{i+1}/ T_i) where T= T_{sph}/ T_{rev}
Revolute Joint	t_{rev}	↑↑↑	↓	↓
	R_{rev}	↓	↑	↑↑
Spherical Joint	t_{sph}	↑	↓	↑
	R_{sph}	↓	↑↑	↓↓
	l_{sph}	↓	↑↑↑	↓↓↓

ϵ 1 -1.2
ϵ 1.2 -1.5
> 1.5
ϵ 0.8 -1
ϵ 0.5 -0.8
< 0.5

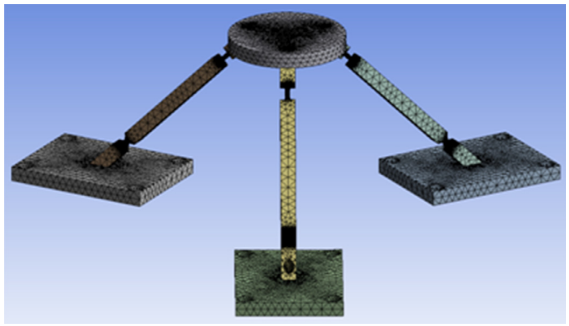


Fig. 8. 3-PRS compliant mechanism model.

displacement and the relation of maximum stresses in the joints suffer a slighter reduction with a factor of 0.8. By contrast, an increase in the length of the spherical joint causes a tinier reduction of the structural stiffness, an important increase of the Z displacement and a notable reduction of the relation of maximum stresses.

Different radii, lengths and thicknesses have been applied to the revolute and spherical flexure joints to obtain the desired motions in the platform and to fulfill the imposed conditions. An iterative process has been developed and as a result, the optimal structure has been obtained. The diagram illustrating the design process can be observed in Fig. 11.

For the revolute hinge, the achieved values have been $R_{rev} = 8$ mm and $t_{rev} = 2$ mm. Measuring the deformation produced in the joint, the rotational stiffness around the Y axis is $K_{Rrev} = 98.37$ Nm/rad. For the spherical joint, the selected dimensions have been $l_{sph} = 5$ mm, $t_{sph} = 3$ mm and $R_{sph} = 4$ mm. Also, the flexural stiffness and the torsional stiffness are $K_{Rsph} = 32.67$ Nm/rad and $K_{Tsph} = 24.46$ Nm/rad.

As an example, for a Z displacement of the platform, the obtained results in FEM analysis are shown in Table 2. Some force

Table 2
Static forces, displacements and stress at the joints.

Actuator 1	F_1 (N)	20	40	60
	s_1 (mm)	0.935	1.899	2.865
Actuator 2	F_2 (N)	20	40	60
	s_2 (mm)	0.935	1.899	2.865
Actuator 3	F_3 (N)	20	40	60
	s_3 (mm)	0.935	1.899	2.865
Platform	z (mm)	0.933	1.896	2.859
Joints	Max. stress (MPa)	169.25	344.01	518.78

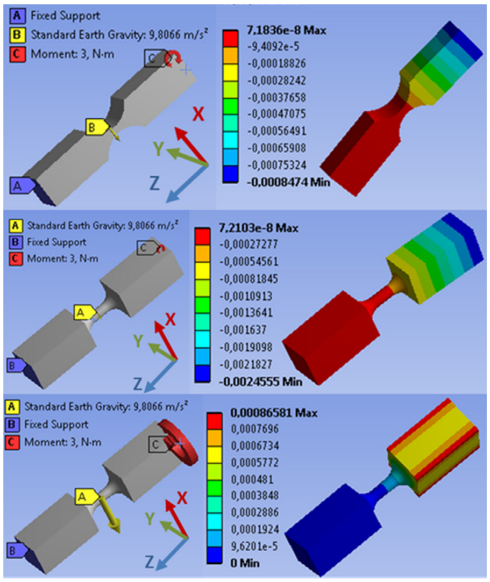


Fig. 9. Loading and deformation of the joints. Above: bending of the revolute joint; middle and below: bending and torsion of the spherical joint.

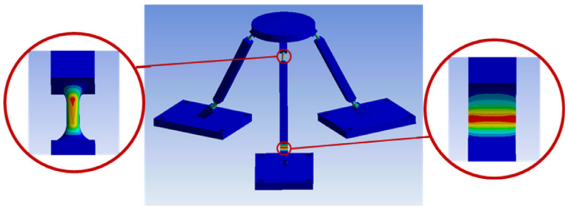


Fig. 10. Stressed areas of the joints during a Z displacement.

steps have been applied to the actuators and the occurred displacements in them and on the platform and the maximum stress in the mechanism have been measured. Attending to the linear behavior of the values, for a stress of 400 MPa, a displacement in z direction equal to 2.205 mm should be achieved.

6. Experimental methodology

To validate experimentally the results obtained in the previous sections, a prototype has been built, see Fig. 12. The actuation system of the mechanism consists of three identical actuators based on a RE-40 Maxon DC motor with a GP-32A gearbox of 14:1 gear ratio. Flexible couplings are used to connect the output shafts of the gearboxes to three Igus ZLW-1040-02-S-100 linear belt drives.

Regarding the control of the device, a PID cascaded position, velocity and current control has been used, see Fig. 11. The position is controlled in the first stage on a NI-PXIe 1032 with a Real-Time operating system. The position control loop is based on a proportional gain with a value of $2.5s^{-1}$ and a 5 ms cycle time. The velocity reference, which is the output of this stage, is sent to the motor driver Maxon EPOS2 50/5 through a CANOPEN interface, where the velocity and current regulation is performed. As for the velocity control loop, a PI velocity regulator is employed, whose proportional gain value is of $0.9Arad^{-1}$ and the integral gain is of 0.001 $Asrad^{-1}$, and 1 ms has been stablished as cycle time. A current reference, that is the output of the velocity controller, is handled by

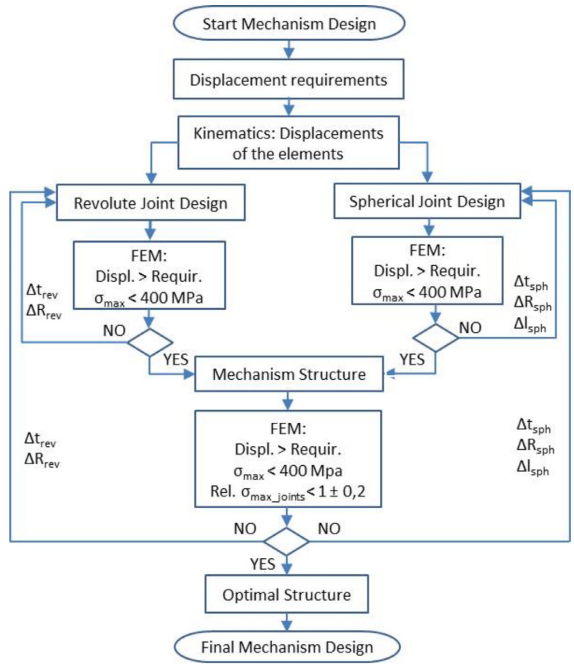


Fig. 11. Scheme of the design process.

a PI current controller. The torque constant of the RE-40 motors is 30.2×10^{-3} Nm/A. The end platform position commands are converted to a θ reference into the motors using the IKP and the θ control is performed. To measure the real position of the manipulator, an external position measuring device has been used, that is, a coordinate measuring machine (CMM) ZEISS MC850 with software ZEISS CALYPSO.

6.1. Dimensional verification

Here, the default position of all components has been measured [26]. The selected measurements to perform have been: 1) angles between the guides, Fig. 1: $OA_i - OA_{i+1}$; 2) position and orientation of each base, Fig. 1: points C_i; 3) heights and overhead planes of legs, Fig. 1: $L_i \cdot \sin\alpha_0$; 4) position and orientation of the platform, Fig. 1: point P; and actuators position, Fig. 1: points A_i. In the following paragraphs, the procedure to obtain all of them is explained.

Firstly, the angular position between the guides in XY plane was obtained. To do that, several points on each side of the guides were measured at the same height, as can be seen in Fig. 13. With the coordinates of these points, two equations for each guide were obtained, providing the angles between the guides. The measured angles can be observed in Table 3.

Secondly, the position and orientation of each base was measured. The purpose was to know if the supports where the legs of the prototype were attached were in the same plane, see Fig. 13. The reference plane (RF) to measure was placed on the surface of the CMM to avoid possible inclinations. Several points in each base were measured to obtain the average height and the plane in which were located. With these planes, tilt and location errors could be accounted. In Table 3, the heights and the angles around the X and Y axes of each base are shown. The average height of

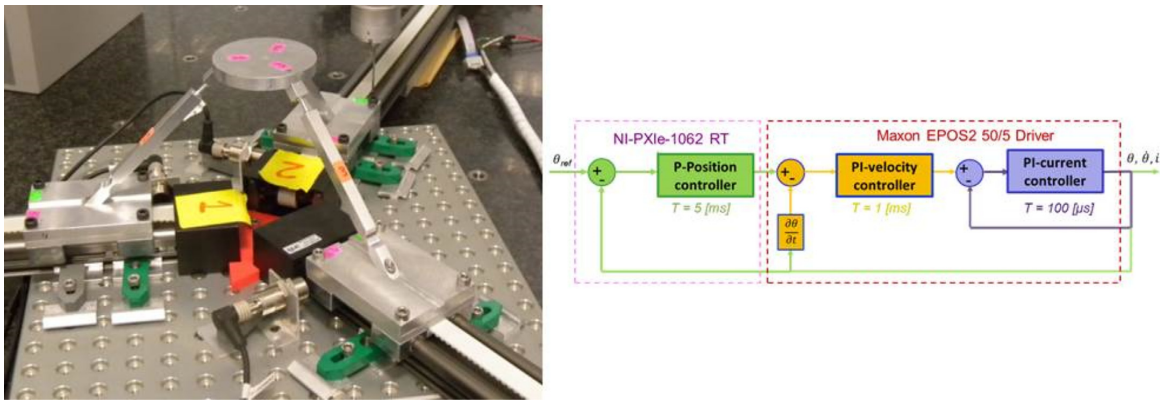


Fig. 12. Left) developed prototype; Right) control architecture for the motor of the actuation system.

Fig. 13. Dimensional verification.

them is 77.472 mm. It will be necessary to calculate the height of the prototype.

Thirdly, the prototype was assembled without the upper platform and the heights and the overhead planes of the legs were measured, see Fig. 13. The average height of each leg can be observed in Table 3. The next step was to perform the complete assembly of the prototype and measure both the height and the inclination of the platform. To do that, several points were measured in the upper surface and in the contour of the platform, see Fig. 13, and the plane surface and the axis of the cylinder of the platform were obtained. Intersecting these two elements, the coordinate center and the height of this point were obtained. Also, knowing the equation of the plane, the inclination of the platform around the X and Y axes is determined. These results are shown in Table 3.

Finally, the position of the actuators relative to the fixed frame was also measured. These values allow knowing the real displacements of the actuators when a movement is programmed to the mechanism. To do that, the backplane location of each base was measured, see Fig. 13. Knowing the equations of the planes and the place of coordinate center, the position of the actuators can be achieved. These magnitudes can be observed in Table 3.

In view of the results, the average deviations have been: $\varepsilon_{OAi\text{-}OAi+1} = 0^{o}3'12''$ in the angles between the guides; $\varepsilon_{C} = 0.254$ mm, $\varepsilon_{\psi c} = 0^{o}14'3''$ and $\varepsilon_{\theta c} = 0^{o}8'19''$ in the position and orientation of the basis; $\varepsilon_{L\cdot Sin\alpha} = 0.452$ mm in the height of the legs; $\varepsilon_{A} = 0.134$ mm in the position of the actuators, and $\varepsilon_{p} = 0.164$ mm, $\varepsilon_{\psi p} = 0^{o}2'9''$ and $\varepsilon_{\theta p} = 0^{o}7'48''$ in the position and orientation of the platform. These differences between the design parameters and the real values are due to the introduced

Table 3
Dimensional verification results. Distances in millimeters angles in DMS notation.

Guides			Basis						
Axes	Angles		Plane	Height		ψ		θ	
	Design	Error		Design	Error	Design	Error	Design	Error
OA_1–OA_2	120°	−0° 5′ 24″	C1	77.500	−0.080	0°	−0° 0′ 3″	0°	−0° 0′ 10″
OA_2–OA_3	120°	−0° 1′ 12″	C2	77.500	−0.343	0°	0° 23′ 49″	0°	−0° 9′ 21″
OA_3–OA_1	120°	0° 3′ 0″	C3	77.500	0.338	0°	0° 17′ 34″	0°	0° 15′ 25″

Legs					Actuators		
Leg	Height to RF		Leg height		Actuator	Plane A_i	
	Design	Error	Design	Error		Design	Error
H_{leg1}	189,728	−0.031	112.228	0.049	Act. 1	194.232	0.069
H_{leg2}	189,728	0.478	112.228	0.822	Act. 2	194.232	0.069
H_{leg2}	189,728	0.822	112.228	0.484	Act. 3	194.232	0.265

PLATFORM								
Plane	Height to RF		Prototype height		ψ		θ	
	Design	Error	Design	Error	Design	Error	Design	Error
P	197.228	0.136	119.728	0.164	0°	0° 2′ 9″	0°	0° 7′ 48″

Table 4
Comparison of CMM, FEM and FKP measurements.

Δs (μm)			Δz (μm)			$\Delta\psi$ (DMS)			$\Delta\theta$ (DMS)		
Δs_1	Δs_2	Δs_3	CMM	FEM	FKP	CMM	FEM	FKP	CMM	FEM	FKP
1395	1531	1306	1471	1408	1384	7′47″	4′5″	8′59″	5′13″	0′33″	1′4″
777	685	725	714	727	722	2′41″	−0′45″	−1′38″	0′16″	−1′30″	−3′23″
−676	−494	−706	−636	−624	−631	−2′17″	3′53″	8′55″	−3′34″	1′35″	3′41″
−1485	−1175	−1781	−1437	−1478	−1511	9′8″	11′5″	26′6″	4′21″	0′6″	0′11″

errors during the manufacturing and assembly of the prototype. All of them will be considered as starting errors in the following developments because a positioning error is introduced in the mechanism due to these deviations with regard to the design parameters.

6.2. Kinematics verification

Here, a series of movements were programmed and the real displacements at the platform and actuators were measured by means of the CMM measuring external device. Two movements in the positive direction of the actuator and other two movements in the negative direction were scheduled.

To perform this task, it was necessary to measure several points on the top surface of the end platform and in the backplane of each base. Knowing the default position of the prototype and the planes containing these measured points and their position, the real displacement and rotation of the platform and the displacements of the actuators were achieved.

In Table 4, in the first three columns, the real displacements Δs_1, Δs_2, Δs_3, are shown. These measured movements in the actuators were introduced in a FEM analysis of the prototype. As a result, the displacements on the platform during the FEM analysis, z_{FEM}, ψ_{FEM}, θ_{FEM}, were compared with those measured with the CMM, z_{CMM}, ψ_{CMM}, θ_{CMM}. Also, to check the validity of the rigid body kinematics, the real displacements of the actuators have also been introduced into the forward kinematic problem (FKP) and its solutions, z_{FKP}, ψ_{FKP}, θ_{FKP}, have been compared with the measurements from the CMM. All the results can be observed in Table 4.

In Table 5, the absolute errors of the FEM and FKP predictions regarding the CMM measurements are shown. Comparing the position of the end platform measured by the CMM and the FEM predictions, the linear errors in Z direction are always under 70 μm, and the angular errors are less than 7′, see Table 5. Also, while the linear error is somewhat proportional to the displacement, growing for higher displacements, the angular error behavior is more erratic. Due to the fact that the deformation of the mechanism is relatively small and linear, the Z displacement is a 1,3% of the length of the limbs, the FEM estimations calculated with a static analysis should be considered as a good reference, being the main uncertainty the quality of the mesh and the parameters of the material. So, the deviations observed can be attributed mainly to the measured errors in the prototype due to the manufacturing and assembling stage.

Comparing the CMM measurements with the FKP predictions, similar trends are observed. However, it must be noted that the linear error in Z grows even higher for larger displacements. It is something predictable, as the hypothesis of considering the kinematics of the compliant mechanism as equal to a conventional one is known to lose validity for larger deformations of the flexure joints. Due to the length of the spherical joints, their deflection in large displacements introduces not only a rotation around their central point, but also a translation of the central point itself, which is not considered in the rigid body kinematics. Here, the limit for that hypothesis is observed to be around the millimeter in Z displacement, approximately a 0.9% of the length of the limbs. Below that limit, it must be noted that the error of the FKP predictions in Z translation is always below 10 μm, very close to the predictions of the FEM.

In order to avoid the prototype imperfections in the calculations, it is possible to consider the FEM predictions as optimal results to verify the FKP solutions. The comparison between both cases is shown in Table 6. As it was commented previously, similar results are obtained for small displacements, with deviations fewer than 8 μm for Z displacements, increasing these differences when the introduced movements are larger. Therefore, the validity of the FKP model for designing a 3-PRS compliant parallel mechanism under small displacements is demonstrated, and the limit for

Table 5
Absolute errors in the CMM measurements compared to the FEM and FKP predictions.

Δz (μm)			Δψ (DMS)			Δθ (DMS)		
CMM	FEM	FKP	CMM	FEM	FKP	CMM	FEM	FKP
	$\lvert\varepsilon_{abs}\rvert$	$\lvert\varepsilon_{abs}\rvert$		$\lvert\varepsilon_{abs}\rvert$	$\lvert\varepsilon_{abs}\rvert$		$\lvert\varepsilon_{abs}\rvert$	$\lvert\varepsilon_{abs}\rvert$
1471	**63**	**87**	7′47″	**3′42″**	**1′12″**	5′13″	**4′40″**	**4′9″**
714	**13**	**8**	2′41″	**3′25″**	**4′18″**	0′16″	**1′46″**	**3′39″**
−636	**12**	**5**	−2′17″	**6′11″**	**11′12″**	−3′34″	**5′9″**	**7′15″**
−1437	**41**	**74**	9′8″	**1′57″**	**16′58″**	4′21″	**4′15″**	**4′10″**

Table 6
Comparison of FEM and FKP predictions.

Δz (μm)		Δψ (DMS)		Δθ (DMS)	
FEM	FKP	FEM	FKP	FEM	FKP
	$\lvert\varepsilon_{abs}\rvert$		$\lvert\varepsilon_{abs}\rvert$		$\lvert\varepsilon_{abs}\rvert$
1408	**24**	4′5″	**4′54″**	0′33″	**0′31″**
727	**5**	−0′45″	**0′53″**	−1′30″	**1′53″**
−624	**7**	3′53″	**5′2″**	1′35″	**2′6″**
−1478	**33**	11′5″	**15′1″**	0′6″	**0′5″**

its assumptions can be quantified in a 0.9% of the length of the limbs.

7. Conclusions

In the present work, a procedure for the development, validation and evaluation of a 3-PRS compliant parallel manipulator has been proposed. To do that, first a study of the solid body kinematics of the mechanism has been performed under the assumption of small displacements, solving both the inverse and forward kinematic problem. The equations of the parasitic displacements as well as the necessary rotations on the revolute and spherical flexure joints are also provided.

Then, a design approach based on FEM analysis for the design of compliant parallel manipulators has been detailed, dividing the mechanical design in two stages, first, the design of the flexure joints, and then, the evaluation of the whole mechanism. The iterative process to achieve the optimal solution for a 3-PRS compliant mechanism has been shown, providing useful guidelines that relate the overall stiffness, displacement and stresses with the geometrical parameters of the flexure joints.

An experimental validation has been done on a prototype, using an external measurement of both the end platform and the actuated joints position by means of a coordinate measuring machine, verifying first the dimensions of the prototype and then the displacements performed. Those results have been compared with the FKP and FEM estimations during the design stage. The hypothesis of using the solid body kinematics for a compliant mechanism whenever the displacements are small has been experimentally tested, demonstrating that it provides reliable results if the end platform linear displacement is below a 0.9% of the length of the limbs. Above that limit, only FEM calculations must be trusted.

Acknowledgments

The authors of this paper wish to acknowledge the financial support received from the Spanish Government via the Ministerio de Educación y Ciencia (Project DPI2011-22955) and Ministerio de Economía y Competitividad (Project DPI2015-64450-R), the ERDF of the European Union, the Government of the Basque Country (Project GIC07/78, IT445-10 and SAIOTEK 2013 SAI13/245, SPC13UN011), and the University of the Basque Country (Project EHUA13/30 and Zabalduz-2012). Thanks are also addressed to Dr. Jorge Presa and Alfonso Urzainki from Egile Corporation XXI for their valuable contributions.

Appendix

A. Forward Kinematics Modeling

Developing Eq. (35) for each leg, the resulting expressions can be obtained:

Expressions for Leg 1

$$(OB_1 - OC_1)_x^2 + (OB_1 - OC_1)_y^2 + (OB_1 - OC_1)_z^2 - L^2 = 0 \tag{A.1}$$

The expressions for the components of $\mathbf{OB_1}$ can be observed in Eq. (6). On the other hand, the vector $\mathbf{OC_1}$ can be defined as:

$$\mathbf{OC_1} = \begin{Bmatrix} a - s_1 \\ 0 \\ 0 \end{Bmatrix} \tag{A.2}$$

Therefore, substituting in Eq (A.1), the following equation is reached:

$$(p_x + b \cdot u_x - a + s_1)^2 + (p_y + b \cdot u_y)^2 + (p_z + b \cdot u_z)^2 - L^2 = 0 \tag{A.3}$$

Finally, substituting the expressions of the components of the rotation matrix showed in Eq. (4), the resulting equation for Leg 1 is obtained:

$$\begin{aligned}&(p_x + b(c\theta c\phi + s\psi s\theta s\phi) - a + s_1)^2 + (p_y + bc\psi s\phi)^2 \\ &+ (p_z + b(-s\theta c\phi + s\psi c\theta s\phi))^2 - L^2 = 0\end{aligned} \tag{A.4}$$

Expressions for Leg 2

$$(OB_2 - OC_2)_x^2 + (OB_2 - OC_2)_y^2 + (OB_2 - OC_2)_z^2 - L^2 = 0 \tag{A.5}$$

The expressions for the components of $\mathbf{OB_2}$ can be observed in Eq. (6). On the other hand, the vector $\mathbf{OC_2}$ can be defined as:

$$\mathbf{OC_2} = \begin{Bmatrix} (a - s_2) \cdot \left(-\frac{1}{2}\right) \\ (a - s_2) \cdot \frac{\sqrt{3}}{2} \\ 0 \end{Bmatrix} \tag{A.6}$$

Therefore, substituting in Eq. (A.5) , the following equation is reached:

$$\begin{aligned}&\left(p_x - \tfrac{b}{2} \cdot u_x + \tfrac{\sqrt{3}b}{2} \cdot v_x + \tfrac{a - s_2}{2}\right)^2 + \left(p_y - \tfrac{b}{2} \cdot u_y + \tfrac{\sqrt{3}b}{2} \cdot v_y - \tfrac{\sqrt{3}(a - s_2)}{2}\right)^2 \\ &+ \left(p_z - \tfrac{b}{2} \cdot u_z + \tfrac{\sqrt{3}b}{2} \cdot v_z\right)^2 - L^2 = 0\end{aligned} \tag{A.7}$$

Finally, substituting the expressions of the components of the rotation matrix showed in Eq. (4), the resulting equation for Leg 2

is obtained:

$$\left(p_x - \tfrac{b}{2}(c\theta c\phi + s\psi s\theta s\phi) + \tfrac{\sqrt{3}b}{2}(-c\theta s\phi + s\psi s\theta c\phi) + \tfrac{a-s_2}{2}\right)^2 + \left(p_y - \tfrac{b}{2}(c\psi s\phi) + \tfrac{\sqrt{3}b}{2}(c\psi c\phi) - \tfrac{\sqrt{3}(a-s_2)}{2}\right)^2 + \left(p_z - \tfrac{b}{2}(-s\theta c\phi + s\psi c\theta s\phi) + \tfrac{\sqrt{3}b}{2}(s\theta s\phi + s\psi c\theta c\phi)\right)^2 - L^2 = 0 \tag{A.8}$$

Expressions for Leg 3

$$(OB_3 - OC_3)_x^2 + (OB_3 - OC_3)_y^2 + (OB_3 - OC_3)_z^2 - L^2 = 0 \tag{A.9}$$

The expressions for the components of $\mathbf{OB_3}$ can be observed in Eq. (6). On the other hand, the vector $\mathbf{OC_3}$ can be defined as:

$$\mathbf{OC_3} = \begin{Bmatrix} (a - s_3)\cdot\left(-\frac{1}{2}\right) \\ (a - s_3)\cdot\left(-\frac{\sqrt{3}}{2}\right) \\ 0 \end{Bmatrix} \tag{A.10}$$

Therefore, substituting in Eq. (A.9) , the following equation is reached:

$$\left(p_x - \tfrac{b}{2}\cdot u_x - \tfrac{\sqrt{3}b}{2}\cdot v_x + \tfrac{a-s_3}{2}\right)^2 + \left(p_y - \tfrac{b}{2}\cdot u_y - \tfrac{\sqrt{3}b}{2}\cdot v_y + \tfrac{\sqrt{3}(a-s_3)}{2}\right)^2 + \left(p_z - \tfrac{b}{2}\cdot u_z - \tfrac{\sqrt{3}b}{2}\cdot v_z\right)^2 - L^2 = 0 \tag{A.11}$$

Finally, substituting the expressions of the components of the rotation matrix showed in Eq. (4), the resulting equation for Leg 3 is obtained:

$$\left(p_x - \tfrac{b}{2}(c\theta c\phi + s\psi s\theta s\phi) - \tfrac{\sqrt{3}b}{2}(-c\theta s\phi + s\psi s\theta c\phi) + \tfrac{a-s_3}{2}\right)^2 + \left(p_y - \tfrac{b}{2}(c\psi s\phi) - \tfrac{\sqrt{3}b}{2}(c\psi c\phi) + \tfrac{\sqrt{3}(a-s_3)}{2}\right)^2 + \left(p_z - \tfrac{b}{2}(-s\theta c\phi + s\psi c\theta s\phi) - \tfrac{\sqrt{3}b}{2}(s\theta s\phi + s\psi c\theta c\phi)\right)^2 - L^2 = 0 \tag{A.12}$$

References

[1] Howell LL. Compliant mechanisms. New York: Wiley; 2001.
[2] Yue Y, Gao F, Zhao X, Jeffrey Ge Q. Relationship among input-force, payload, stiffness and displacement of a 3-DOF perpendicular parallel micro-manipulator. Mech Mach Theory 2010;45:756–71. doi:10.1016/j.mechmachtheory.2009.12.006.
[3] Merlet J-P. Parallel robots, vol. 208; 2006. doi:10.1007/1-4020-4133-0.
[4] Wu J, Wang J, Wang L, Li T. Dynamics and control of a planar 3-DOF parallel manipulator with actuation redundancy. Mech Mach Theory 2009;44:835–49. doi:10.1016/j.mechmachtheory.2008.04.002.
[5] Wu J, Chen X, Li T, Wang L. Optimal design of a 2-DOF parallel manipulator with actuation redundancy considering kinematics and natural frequency. Robot Comput Integr Manuf 2013;29:80–5. doi:10.1016/j.rcim.2012.07.005.
[6] Wu J, Li T, Wang J, Wang L. Stiffness and natural frequency of a 3-DOF parallel manipulator with consideration of additional leg candidates. Rob Auton Syst 2013;61:868–75. doi:10.1016/j.robot.2013.03.001.
[7] Kim JJ, Choi YM, Ahn D, Hwang B, Gweon DG, Jeong J. A millimeter-range flexure-based nano-positioning stage using a self-guided displacement amplification mechanism. Mech Mach Theory 2012;50:109–20. doi:10.1016/j.mechmachtheory.2011.11.012.
[8] Shiou F-J, Chen C-J, Chiang C-J, Liou K-J, Liao S-C, Liou H-C. Development of a real-time closed-loop micro-/nano-positioning system embedded with a capacitive sensor. Meas Sci Technol 2010;21:054007. doi:10.1088/0957-0233/21/5/054007.
[9] Yeom T, Simon TW, Zhang M, North MT, Cui T. High frequency, large displacement, and low power consumption piezoelectric translational actuator based on an oval loop shell. Sensor Actuat A Phys 2012;176:99–109. doi:10.1016/j.sna.2012.01.001.
[10] Muraoka M, Sanada S. Displacement amplifier for piezoelectric actuator based on honeycomb link mechanism. Sensor Actuat A Phys 2010;157:84–90. doi:10.1016/j.sna.2009.10.024.
[11] Juuti J, Kordás K, Lonnakko R, Moilanen V-P, Leppävuori S. Mechanically amplified large displacement piezoelectric actuators. Sensor Actuat A Phys 2005;120:225–31. doi:10.1016/j.sna.2004.11.016.
[12] Wang W, Han C, Choi H. 2-DOF kinematic XY stage design based on flexure element. In: IEEE Int conf mechatronics autom 2011; 2011. p. 1412–27. doi:10.1109/ICMA.2011.5985783.
[13] Yong YK, Aphale SS, Moheimani SOR. Design, identification, and control of a flexure-based XY stage for fast nanoscale positioning. IEEE Trans Nanotechnol 2009;8:46–54.
[14] Kim H-Y, Ahn D-H, Chun B-S, Gweon D-G. Development and Optimization of a Novel 3-DOF Precision Flexure Stage. In: Proc 10th IEEE int conf nanotechnol Jt symp with nano Korea; 2010. p. 903–6.
[15] Kenton BJ, Leang KK. Design and control of a three-axis serial-kinematic high-bandwidth nanopositioner. IEEE/ASME Trans Mechatronics 2012;17:356–69. doi:10.1109/TMECH.2011.2105499.
[16] Hao G, Kong X. Design and modeling of a large-range modular XYZ compliant parallel manipulator using identical spatial modules. J Mech Robot 2012;4:021009. doi:10.1115/1.4006188.
[17] Yao Q, Dong J, Ferreira PM. A novel parallel-kinematics mechanisms for integrated, multi-axis nanopositioning. Precis Eng 2008;32:7–19. doi:10.1016/j.precisioneng.2007.03.001.
[18] Liang Q, Zhang D, Chi Z, Song Q, Ge Y, Ge Y. Six-DOF micro-manipulator based on compliant parallel mechanism with integrated force sensor. Robot Comput Integr Manuf 2011;27:124–34. doi:10.1016/j.rcim.2010.06.018.
[19] Dunning AG, Tolou N, Herder JL. A compact low-stiffness six degrees of freedom compliant precision stage. Precis Eng 2013;37:380–8. doi:10.1016/j.precisioneng.2012.10.007.
[20] Huijts M, Brouwer D, Dijk J. Design, modeling and control of an elastic parallel kinematic 6-DOFs manipulator. MiKroniek 2009;49:420–47.
[21] Shi H, Su H-J, Dagalakis N. A stiffness model for control and analysis of a MEMS hexapod nanopositioner. Mech Mach Theory 2014;80:246–64. doi:10.1016/j.mechmachtheory.2014.05.004.
[22] Li Y, Xu Q. Kinematic analysis of a 3-PRS parallel manipulator. Robot Comput Integr Manuf 2007;23:395–408. doi:10.1016/j.rcim.2006.04.007.
[23] Bonev IA, Gosselin CM. Geometric analysis of parallel mechanisms. Québec: Université Laval; 2002.
[24] Lobontiu N. Compliant mechanisms: design of flexure hinges. Boca Raton, FL: CRC Press; 2002.
[25] Yong YK, Lu TF. Comparison of circular flexure hinge design equations and the derivation of empirical stiffness formulations. In: IEEE/ASME int conf adv intell mechatronics, 32. AIM; 2009. p. 510–15. doi:10.1109/AIM.2009.5229961.
[26] Ruiz A, Campa FJ, Roldán C, Altuzarra O. Experimental validation of the kinematics of a 3-PRS compliant mechanism for micromilling. Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf., Boston, Massachusetts, USA: 2015. ASME; 2015.

Part VI

End Matter

Bibliography

[1] KUKA Robotics. *KUKA Robotics web site*. URL: http://www.kuka-robotics.com/en/ (see p. 5).

[2] *Adept Technology, Inc.* Last visited on October 14th. Adept Technology, Inc. 2015. URL: http://www.adept.com/products/robots (see pp. 5, 7).

[3] Jean-Pierre Merlet. *Parallel Robots*. Ed. by G.M.L. Gladwell. 2Ed. Vol. 128. Solid Mechanics and Its Applications. Springer Netherlands, 2006. DOI: 10.1007/1-4020-4133-0 (see pp. 5, 6, 46).

[4] *PKMtricept SL*. Last visited on October 15th. 2015. URL: http://www.pkmtricept.com/ (see p. 7).

[5] *Physik Instrumente (PI) GmbH & Co. KG*. Last visited on October 14th. Physik Instrumente (PI). 2015. URL: http://www.physikinstrumente.com/products/parallel-kinematic-hexapods.html (see p. 7).

[6] *Moog Inc.* Last visited on October 15th. 2015. URL: http://www.moog.com/products/hexapods-positioning-systems/ (see p. 7).

[7] Jean-Pierre Merlet. "Keynote: Still a long way to go on the road for parallel mechanisms". In: *Proceedings of the ASME 27th Biennal Mechanism and Robotics Conference*. Montréal, Quebec, 2002 (see pp. 8, 9).

[8] Saioa Herrero Villalibre. “Design Methodology for MAST-type parallel Manipulators Based on Kinematic, Dynamic and Stiffness Criteria: Theoretical and Experimental Application to the 2PRU-1PRS”. PhD thesis. Superior Technical School of Enginering Bilbao, UPV-EHU, July 2016 (see pp. 10, 39, 50, 125).

[9] Thomas Kölling, Michael Krees, Mathias Hüsing, and Burkhard Corves. “Estimating Ergonomic Comfort During the Process of Mechanism Design by Interaction with a Haptic Feedback-System”. In: *Digital Human Modeling. Applications in Health, Safety, Ergonomics and Risk Management: Ergonomics and Health: 6th International Conference, DHM 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2-7, 2015, Proceedings, Part II*. Ed. by Vincent G. Duffy. Cham: Springer International Publishing, 2015, pp. 62–73. DOI: 10.1007/978-3-319-21070-4_7 (see pp. 10, 183).

[10] Yaskawa. *Yaskawa Global Website*. English. Yaskawa Electric Corporation. Aug. 2016. URL: https://www.yaskawa-global.com/ (see pp. 13, 32).

[11] Jack Dinsdale. “Mechatronics and Asics”. In: *CIRP Annals - Manufacturing Technology* 38.2 (1989), pp. 627–634. DOI: http://dx.doi.org/10.1016/S0007-8506(07)61130-5 (see p. 13).

[12] Jim Hewit. “Mechatronics design — The key to performance enhancement”. In: *Robotics and Autonomous Systems* 19.2 (1996), pp. 135–142. DOI: http://dx.doi.org/10.1016/S0921-8890(96)00041-3 (see p. 13).

[13] Jacob Buur and Mogens Myrup Andreasen. “A theoretical approach to mechatronics design”. PhD thesis. 1990 (see p. 14).

[14] Joshua Hurst. *Personal website*. English. Rensselaer Polytechnic Institute. URL: http://homepages.rpi.edu/~hurstj2/ (see pp. 14, 15).

[15] *Website of the Association of German Engineers*. 2016. URL: http://www.vdi.eu/ (see p. 15).

[16] J Gausemeier and S Moehringer. "New Guideline Vdi 2206 - A Flexible Procedure Model For The Design Of Mechatronic Systems". In: *Proceedings of ICED 03*. Ed. by A Folkeson, K Gralen, M Norell, and U Sellgren. 2003 (see p. 15).

[17] G. Reinhart and M. Weissenberger. "Multibody simulation of machine tools as mechatronic systems for optimization of motion dynamics in the design process". In: *1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No.99TH8399)*. 1999, pp. 605–610. DOI: 10.1109/AIM.1999.803237 (see p. 16).

[18] T.H. Yan, X.D. Chen, and R.M. Lin. "Servo system modeling and reduction of mechatronic system through finite element analysis for control design". In: *Mechatronics* 18.9 (2008), pp. 466–474. DOI: http://dx.doi.org/10.1016/j.mechatronics.2008.04.007 (see p. 16).

[19] Dehong Huo, Kai Cheng, and Frank Wardle. "A holistic integrated dynamic design and modelling approach applied to the development of ultraprecision micro-milling machines". In: *International Journal of Machine Tools and Manufacture* 50.4 (2010). Design of Ultraprecision and Micro Machine Tools and their Key Enabling Technologies, pp. 335–343. DOI: http://dx.doi.org/10.1016/j.ijmachtools.2009.10.009 (see p. 17).

[20] C. Brecher, T. Ostermann, and D.A. Friedrich. "Control concept for {PKM} considering the mechanical coupling between actors". In: *International Journal of Machine Tools and Manufacture* 48.3–4 (2008), pp. 427–436. DOI: http://dx.doi.org/10.1016/j.ijmachtools.2007.09.002 (see pp. 17, 41).

[21] David Prévost, Sylvain Lavernhe, Claire Lartigue, and Didier Dumur. "Feed drive modelling for the simulation of tool path tracking in multi-axis high speed machining". In: *International Journal of Mechatronics and Manufacturing Systems* (2011). DOI: 10.1504/IJMMS.2011.041472 (see pp. 17, 79).

[22] Richard Crowder. "Chapter 8 - Stepper motors". In: *Electric Drives and Electromechanical Systems*. Ed. by Richard Crowder. Oxford: Newnes, 2005, pp. 215–233. DOI: http://dx.doi.org/10.1016/B978-075066740-1/50009-0 (see p. 19).

[23] T.E. Marlin. *Process control: designing processes and control systems for dynamic performance*. McGraw-Hill chemical engineering series. McGraw-Hill, 2000 (see pp. 23, 24).

[24] George Ellis. *Control System Design Guide (Third Edition)*. Ed. by George Ellis. Third Edition. Burlington: Academic Press, 2004 (see p. 25).

[25] Clarence W. de Silva. *Mechatronics a Foundation Course*. CRC Press, 2010 (see pp. 27, 28, 32).

[26] Charles P. Pinney and William E. Baker. "Measurement, Instrumentation, and Sensors Handbook". In: ed. by Jhon G. Webster and Halit Eren. Second Edition. CRC Press, 2014. Chap. 35-Velocity Measurement, pp. 1–18 (see p. 28).

[27] William Bolton. *Instrumentation and Control Systems*. Ed. by Sonnini R. Yura. Second Edition. Newnes, Elsevier, 2015 (see p. 28).

[28] Maxon. *Maxon motor online catalog*. Maxon Motor. 2016. URL: http://www.maxonmotor.com/maxon/view/catalog/ (see pp. 31, 129, 132).

[29] Pololu. *Pololu Robotics and Electronics*. Pololu Robotics and Electronics. 2017. URL: https://www.pololu.com/ (see p. 31).

[30] Siemens. *Siemens on-line catalog*. Siemens. 2017. URL: http://w3.siemens.com/mcms/mc-solutions/en/motors/motion-control-motors/simotics-s-servomotors/simotics-s-1fk7/pages/simotics-s-1fk7.aspx (see p. 32).

[31] Clarence W. de Silva. *Mechatonics: An Integrated Approach*. CRC Press, 2005 (see p. 35).

[32] HarmonicDrive. *C. Walton Musser website*. Harmonic Drive LLC. 2016. URL: http://www.waltmusser.org/ (see p. 35).

[33] Igus-GmbH. *Drylin ZLW -1040 Toothed belt axis online catalog*. Igus-GmbH. 2016. URL: http://www.igus.eu/wpck/4773/zlw_1040 (see pp. 37, 129).

[34] *Ipirsite Husillos website*. Ipiranga Husillos. URL: http://www.ipirangahusillos.com/ (see p. 38).

[35] *NSK Americas web site*. NSK. URL: http://www.nskamericas.com/ (see p. 38).

[36] R.T Severn. "Development of European shaking tables". In: *Earthquake Engineering Research Centre,* (2009) (see p. 40).

[37] A.J. Clark. "Dynamic characteristics of large multiple degree of freedom shaking tables". In: *Earthquake Engineering Tenth World Conference*. 1992 (see p. 40).

[38] Curt Nelson. *Vibration test evolution: single axis, single shaker to 6DoF*. Tech. rep. Team Corporation, 2002 (see p. 40).

[39] Nicolas A. Pouliot, Clé, ment M. Gosselin, and Meyer A. Nahon. "Motion Simulation Capabilities of Three-Degree-of-Freedom Flight Simulators". In: *Journal of Aircraft* 35.1 (Jan. 1998), pp. 9–17. DOI: 10.2514/2.2283 (see p. 40).

[40] S. E. Salcudean, P. A. Drexel, D. Ben-Dov, A. J. Taylor, and P. D. Lawrence. "A six degree-of-freedom, hydraulic, one person motion simulator". In: *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*. May 1994, 2437–2443 vol.3. DOI: 10.1109/ROBOT.1994.351146 (see p. 40).

[41] Flavien Paccot, Nicolas Andreff, and Philippe Martinet. "A Review on the Dynamic Control of Parallel Kinematic Machines: Theory and Experiments". In: *The International Journal of Robotics Research* 28.3 (2009), pp. 395–416. eprint: http://ijr.sagepub.com/content/28/3/395.full.pdf+html (see pp. 40–43, 45–47, 87).

[42] Wisama Khalil and Etienne Dombre. *Modeling, Identification and Control of Robots*. Ed. by Taylor. 3rd. Bristol, PA, USA: Taylor & Francis, Inc., 2002 (see pp. 41, 42, 65).

[43] G. Barrette and C.M. Gosselin. "Determination of the Dynamic Workspace of Cable-Driven Planar Parallel Mechanisms." In: *ASME. J. Mech.* 127.2 (2005), pp. 242–248. DOI: 10.1115/1.1830045. (see p. 41).

[44] H. Abdellatif and B. Heimann. "Adapted Time-Optimal Trajectory Planning for Parallel Manipulators with Full Dynamic Modelling". In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Apr. 2005, pp. 411–416. DOI: 10.1109/ROBOT.2005.1570153 (see p. 41).

[45] Paul Lambrechts, Matthijs Boerlage, and Maarten Steinbuch. "Trajectory planning and feedforward design for electromechanical motion systems". In: *Control Engineering Practice* 13.2 (2005), pp. 145–157. DOI: http://dx.doi.org/10.1016/j.conengprac.2004.02.010 (see p. 41).

[46] Jinsong Wang, Jun Wu, Liping Wang, and Zheng You. "Dynamic feed-forward control of a parallel kinematic machine". In: *Mechatronics* 19.3 (2009), pp. 313–324. DOI: http://dx.doi.org/10.1016/j.mechatronics.2008.11.004 (see p. 41).

[47] J. Luh, M. Walker, and R. Paul. "Resolved-acceleration control of mechanical manipulators". In: *IEEE Transactions on Automatic Control* 25.3 (June 1980), pp. 468–474. DOI: 10.1109/TAC.1980.1102367 (see p. 41).

[48] Ka-Tjun Oen and Li-Chun T. Wang. "Optimal dynamic trajectory planning for linearly actuated platform type parallel manipulators having task space redundant degree of freedom". In: *Mechanism and Machine Theory* 42.6 (2007), pp. 727–750. DOI: http://dx.doi.org/10.1016/j.mechmachtheory.2006.05.006 (see p. 42).

[49] S. Kock and W. Schumacher. "A mixed elastic and rigid-body dynamic model of an actuation redundant parallel robot with high-reduction gears". In: *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*. Vol. 2. 2000, 1918–1923 vol.2. DOI: 10.1109/ROBOT.2000.844875 (see p. 43).

[50] J. Swevers, C. Ganseman, D. B. Tukel, J. de Schutter, and H. Van Brussel. "Optimal robot excitation and identification". In: *IEEE Transactions on Robotics and Automation* 13.5 (Oct. 1997), pp. 730–740. DOI: 10.1109/70.631234 (see p. 43).

[51] Miguel Díaz-Rodríguez, Vicente Mata, Ángel Valera, and Álvaro Page. "A methodology for dynamic parameters identification of 3-DOF parallel robots in terms of relevant parameters". In: *Mechanism and Machine Theory* 45.9 (2010), pp. 1337–1356. DOI: http://dx.doi.org/10.1016/j.mechmachtheory.2010.04.007 (see pp. 43, 63, 65, 154, 157).

[52] T. Dallej, N. Andreff, Y. Mezouar, and P. Martinet. "3D Pose Visual Servoing Relieves Parallel Robot Control from Joint Sensing". In: *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Oct. 2006, pp. 4291–4296. DOI: 10.1109/IROS.2006.281959 (see p. 43).

[53] Massimo Callegari, Matteo-Claudio Palpacelli, and Marco Principi. "Dynamics modelling and control of the 3-RCC translational platform". In: *Mechatronics* 16.10 (2006), pp. 589–605 (see pp. 43, 48, 50).

[54] Jiri Tlusty, John Ziegert, and Shannon Ridgeway. "Fundamental Comparison of the Use of Serial and Parallel Kinematics for Machines Tools". In: *CIRP Annals - Manufacturing Technology* 48.1 (1999), pp. 351–356. DOI: http://dx.doi.org/10.1016/S0007-8506(07)63200-4 (see p. 46).

[55] J. Wang and O. Masory. "On the accuracy of a Stewart platform. I. The effect of manufacturing tolerances". In: *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*. May 1993, 114–120 vol.1. DOI: 10.1109/ROBOT.1993.291970 (see p. 46).

[56] Leon Žlajpah. "Simulation in robotics". In: *Mathematics and Computers in Simulation* 79.4 (2008). 5th Vienna International Conference on Mathematical Modelling/Workshop on Scientific Computing

in Electronic Engineering of the 2006 International Conference on Computational Science/Structural Dynamical Systems: Computational Aspects, pp. 879–897 (see p. 47).

[57] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. *Robotic: Modelling, Planning and Control*. Ed. by Michael J. Grimble and Michael A. Johnson. Advanced Textbooks in Control and Signal Processing. Springer-Verlag London Limited, 2010 (see p. 47).

[58] Jiegao Wang, ClémentM. Gosselin, and Li Cheng. "Modeling and Simulation of Robotic Systems with Closed Kinematic Chains Using the Virtual Spring Approach". English. In: *Multibody System Dynamics* 7.2 (2002), pp. 145–170 (see p. 47).

[59] Yunjiang Lou, Zhibin Li, Yingying Zhong, Jiangang Li, and Zexiang Li. "Dynamics and contouring control of a 3-DoF parallel kinematics machine". In: *Mechatronics* 21.1 (2011), pp. 215–226. DOI: http://dx.doi.org/10.1016/j.mechatronics.2010.10.007 (see p. 48).

[60] Alain Codourey. "Dynamic Modeling of Parallel Robots for Computed-Torque Control Implementation". In: *The International Journal of Robotics Research* 17.12 (1998), pp. 1325–1336. DOI: 10.1177/027836499801701205. eprint: http://ijr.sagepub.com/content/17/12/1325.full.pdf+html (see p. 48).

[61] Chifu Yang, Qitao Huang, and Junwei Han. "Computed force and velocity control for spatial multi-DOF electro-hydraulic parallel manipulator". In: *Mechatronics* 22.6 (2012). Special Issue on Intelligent Mechatronics (LSMS2010 & ICSEE2010), pp. 715–722 (see p. 48).

[62] Ho Seong Lee and M. Tomizuka. "Robust motion controller design for high-accuracy positioning systems". In: *IEEE Transactions on Industrial Electronics* 43.1 (Feb. 1996), pp. 48–55. DOI: 10.1109/41.481407 (see pp. 48, 154).

[63] Reza N. Jazar. *Theory of Applied Robotics*. Ed. by Reza N. Jazar. Second Edition. Springer, 2010. DOI: 10.1007/978-1-4419-1750-8 (see p. 49).

[64] I. Ebert-Uphoff and C.M. Gosselin. "Dynamic modeling of a class of spatial statically-balanced parallel platform mechanisms". In: *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*. Vol. 2. 1999, 881–888 vol.2 (see p. 50).

[65] R. Oftadeh, M.M. Aref, and H.D. Taghirad. "Explicit dynamics formulation of Stewart-Gough platform: A Newton-Euler approach". In: *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. Oct. 2010, pp. 2772–2777 (see p. 50).

[66] K. Ebert-Uphoff I; Kozak. "Review of the Role of Quasi-Coordinates for the Kinematic and Dynamic Modeling of Parallel Manipulators." In: *Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*. Quebec, 2002, pp. 328–338 (see pp. 50, 52, 97, 98, 104).

[67] Jiegao Wang and Clément M. Gosselin. "A New Approach for the Dynamic Analysis of Parallel Manipulators". In: *Multibody System Dynamics* 2.3 (1998), pp. 317–334. DOI: 10.1023/A:1009740326195 (see p. 50).

[68] Houssem Abdellatif and Bodo Heimann. "Computational efficient inverse dynamics of 6-DOF fully parallel manipulators by using the Lagrangian formalism". In: *Mechanism and Machine Theory* 44.1 (2009), pp. 192–207 (see pp. 51–54, 92, 98, 104).

[69] A.I. Lurie. *Analytical Mechanics*. English. Foundations of Engineering Mechanics. Springer Berlin Heidelberg, 2002, pp. 449–522 (see pp. 54, 93, 113).

[70] H. Olsson, K.J. Åström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky. "Friction Models and Friction Compensation". In: *European Journal of Control* 4.3 (1998), pp. 176–195. DOI: http://dx.doi.org/10.1016/S0947-3580(98)70113-X (see pp. 55–58, 114, 154).

[71] Brian Armstrong-Hélouvry, Pierre Dupont, and Carlos Canudas De Wit. "A survey of models, analysis tools and compensation methods for the control of machines with friction". In: *Automatica* 30.7 (1994), pp. 1083–1138. DOI: http://dx.doi.org/10.1016/0005-1098(94)90209-7 (see pp. 55–57, 59, 114, 154).

[72] B Friedland and Y. J. Park. "On adaptive friction compensation". In: *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on*. Dec. 1991, 2899–2902 vol.3. DOI: 10.1109/CDC.1991.261068 (see p. 57).

[73] A.J. Morin. "New friction experiments carried out at Metz in 1831-1833". In: *Proceedings of the French Royal Academy of Sciences*. Vol. 4. 1833, pp. 1–128 (see p. 57).

[74] Dean Karnopp. "Computer Simulation of Stick-Slip Friction in Mechanical Dynamic Systems". In: *Journal of Dynamic Systems, Measurement, and Control* 107.1 (1985), pp. 100–103. DOI: 10.1115/1.3140698 (see p. 58).

[75] Erik Wernholt and Svante Gunnarsson. "Nonlinear identification of physically parameterized robot model 1". In: *IFAC Proceedings Volumes* 39.1 (2006), pp. 143–148. DOI: http://dx.doi.org/10.3182/20060329-3-AU-2901.00016 (see pp. 58, 67, 115, 154).

[76] F. Al-Bender, V. Lampaert, and J. Swevers. "The generalized Maxwell-slip model: a novel model for friction Simulation and compensation". In: *IEEE Transactions on Automatic Control* 50.11 (Nov. 2005), pp. 1883–1887. DOI: 10.1109/TAC.2005.858676 (see pp. 58, 60, 154).

[77] P.R. Dahl. *A solid friction model*. Tech. rep. TOR-0158(3107-18)-1. El Segundo, California 90245: The Aerospace Corporation, May 1968 (see p. 59).

[78] F. Al-Bender, V. Lampaert, and J. Swevers. "A Novel Generic Model at Asperity Level for Dry Friction Force Dynamics". In: *Tribology Letters* 16.1 (2004), pp. 81–93. DOI: 10.1023/B:TRIL.0000009718.60501.74 (see p. 60).

[79] T. Tjahjowidodo, F. Al-Bender, and H. Van Brussel. "Friction identification and compensation in a DC motor". In: *IFAC Proceedings Volumes* 38.1 (2005). 16th {IFAC} World Congress, pp. 554–559. DOI: http://dx.doi.org/10.3182/20050703-6-CZ-1902.00093 (see p. 61).

[80] J. Swevers, W. Verdonck, and J. De Schutter. "Dynamic Model Identification for Industrial Robots". In: *IEEE Control Systems* 27.5 (Oct. 2007), pp. 58–71. DOI: 10.1109/MCS.2007.904659 (see pp. 62, 63, 148).

[81] Lennart Ljung. *System Identification (2Nd Ed.): Theory for the User*. Ed. by Lennart Ljung. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1999 (see p. 62).

[82] Erik Wernholt and Svante Gunnarsson. "Nonlinear grey-box identification of industrial robots containing flexibilities". In: *IFAC Proceedings Volumes* 38.1 (2005). 16th {IFAC} World Congress, pp. 356–361. DOI: http://dx.doi.org/10.3182/20050703-6-CZ-1902.00060 (see pp. 63, 67, 84).

[83] Jun Wu, Jinsong Wang, and Zheng You. "An overview of dynamic parameter identification of robots". In: *Robotics and Computer-Integrated Manufacturing* 26.5 (2010), pp. 414–419. DOI: http://dx.doi.org/10.1016/j.rcim.2010.03.013 (see pp. 64, 65).

[84] J. Swevers, C. Ganseman, J. De Schutter, and H. Van Brussel. "Experimental robot identification using optimised periodic trajectories". In: *Mechanical Systems and Signal Processing* 10.5 (1996), pp. 561–577. DOI: http://dx.doi.org/10.1006/mssp.1996.0039 (see pp. 65, 159, 174).

[85] M. Gautier. "Numerical calculation of the base inertial parameters of robots". In: *Proceedings., IEEE International Conference on Robotics and Automation*. May 1990, 1020–1025 vol.2. DOI: 10.1109/ROBOT.1990.126126 (see pp. 65, 157).

[86] S. Guegan, W. Khalil, and P. Lemoine. "Identification of the dynamic parameters of the Orthoglide". In: *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*. Vol. 3. Sept. 2003, 3272–3277 vol.3. DOI: 10.1109/ROBOT.2003.1242095 (see p. 65).

[87] A. Vivas, P. Poignet, F. Marquet, F. Pierrot, and M. Gautier. "Experimental dynamic identification of a fully parallel robot". In: *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*. Vol. 3. Sept. 2003, 3278–3283 vol.3. DOI: 10.1109/ROBOT.2003.1242096 (see p. 65).

[88] J. Cazalilla, M. Vallés, V. Mata, M. Díaz-Rodríguez, and A. Valera. "Adaptive control of a 3-DOF parallel manipulator considering payload handling and relevant parameter models". In: *Robotics and Computer-Integrated Manufacturing* 30.5 (2014), pp. 468–477. DOI: http://dx.doi.org/10.1016/j.rcim.2014.02.003 (see p. 66).

[89] M. Honegger, A. Codourey, and E. Burdet. "Adaptive control of the Hexaglide, a 6 dof parallel manipulator". In: *Proceedings of International Conference on Robotics and Automation*. Vol. 1. Apr. 1997, 543–548 vol.1. DOI: 10.1109/ROBOT.1997.620093 (see p. 66).

[90] Brian D.O. Anderson. "Failures of adaptive control theory and their resolution". In: *Communication in Information and Systems* 5 (2005), pp. 1–20 (see p. 66).

[91] Naira Hovakimyan and Chengyu Cao. $\mathcal{L}_\infty$ *Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Ed. by Ralph C. Smith. SIAM, 2010 (see p. 66).

[92] G. H. Golub and V. Pereyra. "The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate". In: *SIAM Journal on Numerical Analysis* 10.2 (1973), pp. 413–432. DOI: 10.1137/0710036. eprint: http://dx.doi.org/10.1137/0710036 (see p. 67).

[93] Ramu Krishnan. *Electric Motor Drives: Modeling, Analysis, and Control*. Prentice Hall, 2001 (see p. 78).

[94] B.S. Guru and H.R. Hiziroglu. *Electric Machinery and Transformers*. The Oxford Series in Electrical and Computer Engineering Series. OUP USA, 2000 (see p. 78).

[95] Ali Bekir Yildiz. "Electrical equivalent circuit based modeling and analysis of direct current motors". In: *International Journal of Electrical Power & Energy Systems* 43.1 (2012), pp. 1043–1047. DOI: http://dx.doi.org/10.1016/j.ijepes.2012.06.063 (see p. 78).

[96] Leonard Meirovitch. *Methods of Analytical Dynamics*. Ed. by Irving Shames. McGraw-Hill, 1970 (see p. 99).

[97] Constantino Roldán, F.J. Campa, Oscar Altuzarra, and Enrique Amezua. "Automatic identification of the inertia and friction of an electromechanical actuator." In: *New Advances in Mechanisms, Transmissions and Applications: Proceedings of the Second Conference MeTrApp 2013, Bilbao, Spain*. 2013, pp. 409–417 (see p. 115).

[98] Daniel Lichtblau and Eric W. Weisstein. *Condition Number*. MathWorld. URL: http://mathworld.wolfram.com/ConditionNumber.html (see p. 159).

[99] Miguel Díaz-Rodríguez. "Identificación de Parámetros Dinámicos de Robots Paralelos Basada en un Conjunto de Parámetros Significativos". PhD thesis. Dpto. Ingeniería Mecánica y de Materiales UPV, 2009 (see pp. 159, 174).

[100] Matlab. *What is Direct Search?* Matlab. URL: https://es.mathworks.com/help/gads/what-is-direct-search.html (see p. 160).

[101] C. M. Pham and M. Gautier. "Essential parameters of robots". In: *[1991] Proceedings of the 30th IEEE Conference on Decision and Control*. Dec. 1991, 2769–2774 vol.3. DOI: 10.1109/CDC.1991.261862 (see p. 179).

[102] M. Holcombe. "Machines and semigroups". In: *Algebraic Automata Theory:* Cambridge: Cambridge University Press, Aug. 19, 1982, pp. 25–75. DOI: 10.1017/CBO9780511525889.003 (see p. 216).

[103] *What is a Real-Time Operating System (RTOS)?* National Instruments. Nov. 2013. URL: `http://www.ni.com/white-paper/3938/en/` (see p. 224).

[104] J.A. Carretero, R.P. Podhorodeski, M.A. Nahon, and C.M. Gosselin. "Kinematic Analysis and Optimization of a New Three Degree-of-Freedom Spatial Parallel Manipulator." In: *ASME. J. Mech. Des.* 122.1 (1999), pp. 17–24 (see p. 233).

Index