République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Superieure et de la Recherche Scientifique

Université Echahid Hamma Lakhdar d'El-Oued

FACULTE DE TECHNOLOGIE

DEPARTEMENT DE GENIE MECANIQUE

Mémoire de fin d'étude

Présenté pour l'obtention du diplôme de

MASTER ACADEMIQUE

Domaine : Sciences et Technologies
Filière : Electromécanique
Spécialité : Electromécanique

Thème

Dynamic analysis and Control of a Cartesian parallel manipulator

Devant le jury composé de :		**Présenté par :**
Dr. MENECEUR Noureddine	Président	- ZINE Aimane
Mme. MEZIANE Assia	Examinateur	- BOUKHEZNA Abdelmalek
Dr. MANSOURI Khaled	Encadreur	- KHATTAB Taleb mouhamed

2020-2021

Acknowledgment

We start our modest scientific research with the grace of God, the maximum amount of praise and thanks to him before and after that we have succeeded in reaching this degree of learning and development. Then, we thank our supervisor "Mr. MANSOURI KHALED" for the support and guidance he provided us on the topic of research and outside it, and all the professors who taught and supported us throughout our academic career and who were the reason for what we have reached today.

Dedication

We dedicate this work to whom do we prefer them over ourselves and who were and still a reason to facilitate the paths of life

To our parents

We dedicate it to our brothers and sisters, and we thank them for their encouragement and assistance, as well as to our family members. To all our friends from near and far.

Moreover, all those who stood beside us and helped us with everything they had.

We offer you this research

Table of contents

Lists of figures

List of abbreviations

PKM	Parallel kinematic machine
DOF	Degree of freedom
IDM	Invers dynamic model
FDM	Forward dynamic model
EOM	Equation of motion
GCI	Global condition index

Introduction

The field of robotics has witnessed significant advancements in recent years, with parallel manipulators being among the most popular types of robots due to their exceptional accuracy, rigidity, and speed. Cartesian parallel manipulators, in particular, have become increasingly popular due to their ability to perform precise and complex tasks in various applications, including manufacturing, aerospace, and medicine. However, the dynamic analysis and control of these robots remain a challenging task due to their complex kinematic structure and nonlinear dynamics. This dissertation aims to address these challenges by proposing novel techniques for the dynamic analysis and control of a Cartesian parallel manipulator. The research will focus on developing a comprehensive mathematical model of the manipulator's dynamics and control strategies that can improve its performance and accuracy. The results of this study are expected to provide valuable insights into the design and control of parallel manipulators and contribute to the development of efficient and reliable robotic systems.

In this dissertation, the focus is on the dynamic analysis and control of a specific type of parallel manipulator known as the Cartesian parallel manipulator. The Cartesian parallel manipulator is a highly versatile robot that uses parallel kinematics to achieve precise and rapid movements in all three Cartesian axes. However, the dynamic analysis and control of this robot are challenging due to the complex nature of its kinematic structure and nonlinear dynamics.

To address these challenges, the research in this dissertation will propose a techniques for the dynamic analysis and control of the Cartesian parallel manipulator. The research will begin by developing a comprehensive mathematical model of the robot's dynamics, accounting for all the significant factors that affect the robot's behavior. The model will consider various factors, including the robot's joint limits, actuator dynamics, and external disturbances that affect the robot's performance.

The results of this research are expected to provide valuable insights into the design and control of parallel manipulators and contribute to the development of efficient and reliable robotic systems. The proposed techniques for dynamic analysis and control of the Cartesian parallel manipulator can be extended to other types of parallel manipulators, opening up new avenues for research in the field of robotics.

Overall, this dissertation aims to contribute to the advancement of the field of robotics by addressing some of the challenges associated with the dynamic analysis and control of parallel manipulators. It is hoped that the proposed techniques will lead to the development of more precise and efficient robotic systems that can perform complex tasks in various applications, including manufacturing, aerospace, and medicine.

I Chapter I: Literature Review

I.1 Introduction

In this chapter, we will review the relevant literature on dynamic analysis and control of parallel manipulators, with a particular focus on Cartesian parallel manipulators. The chapter will begin by discussing the basic concepts of parallel manipulators and their kinematic structure. We will then examine the existing literature on dynamic modeling of parallel manipulators, including the different approaches used for modeling the robot's dynamics. The chapter will also review the various control strategies that have been proposed for parallel manipulators, including both traditional and advanced control techniques. Finally, we will discuss the limitations of the existing literature and identify the research gaps that need to be addressed in this dissertation. The literature review chapter will provide the foundation for the research work by establishing the current state of knowledge in the field and identifying the key research questions that need to be addressed.

I.2 Background and history of parallel manipulators

Most of the robots used today are designed with sequential arms, where multiple links are connected by actuated joints, forming a single open kinematic chain from the base to the end-effector. This arrangement provides a large workspace and relatively straightforward kinematics. However, in certain applications, stringent demands on payload, accuracy, or dynamic performance limit the use of serial robots.

Parallel robots are highly attractive for a variety of applications because the controlled load is shared among multiple legs of the system. As a result, each kinematic chain carries only a fraction of the overall load, enabling the construction of robots that are inherently more rigid. Such topologies allow for a reduction in the mass of the movable links (all actuators are primarily fixed on the base, and many legs are subject to tension/compression forces), enabling the use of less powerful actuators. Structures with these characteristics are claimed to offer high payload capacity, excellent dynamic capabilities, and superior precision. As a result, parallel robots are being utilized in a wide range of applications.[1]

I.3 General Comments

System: A system can be defined as a group of interconnected components that collaborate to accomplish a particular objective.

The output variables, which are the primary variables of the components, are collectively known as the component's main outputs...

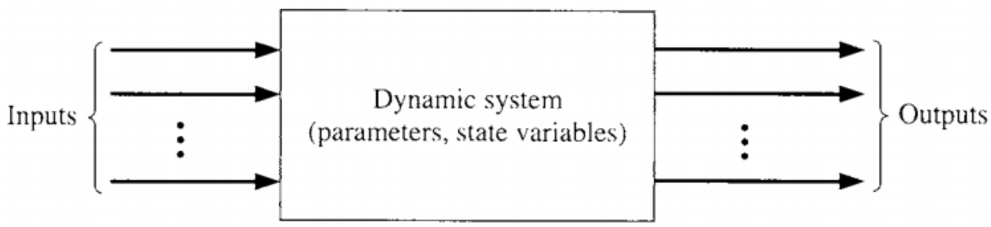


Figure I.1- A dynamic system visual presentation.

To completely describe a system, a complete set of variables known as state variables must be identified. Among these variables, the state variables are the minimum set required to comprehensively depict the system's state at any point in time. These state variables are critical for modeling and analyzing dynamic systems.

Every moving system, whether on the Earth or in space, comprises multiple material bodies that are interconnected through various force laws. These force laws exhibit a unified behavior governed by single or multiple rules. The physical components of these systems consist of solid and elastic bodies that are shaped to deliver optimal performance. To accurately account for these systems, we must first determine the appropriate format for representing the forces and masses through modeling [2].

Dynamic system: is one whose state change with the time. Two main types in dynamical systems are encountered in applications: those for which the time variable in discrete and those for which it is continuous.

Rigid-body dynamics: When constructing robots using a rigid body, linkages are attached to the body via joints.

Due to the constant connections of the rigid body, robots typically have a limited number of linkages.

The dynamics of the stiff body involve analyzing the movement of linked systems under the influence of external forces. In rigid body dynamics, the most important physical quantities are kinetic energy and angular momentum, which both lead to the concept of tensor inertia.

Modeling Modeling involves developing a simplified representation of a system's structure and operation. The goal of a model is to enable analysts to predict the impact of modifications to the system. A good model should closely resemble the actual system and incorporate most of its essential characteristics without being too complex to comprehend and experiment with. The optimal model strikes a balance between realism and simplicity. Simulation practices involve iteratively increasing the complexity of a model. The validity of a model is a significant concern, and its validation involves simulating and comparing the model output to the system output under known input conditions.

For simulation studies, a model is typically a mathematical model created using simulation software. Mathematical models can be classified as deterministic (with fixed values for input and output variables) or stochastic (probabilistic for at least one input or output); static (without considering time) or dynamic (considering time-varying interactions among variables). Simulation models are often dynamic and stochastic.

Robot control and simulation require various mathematical models at different levels, including geometric, kinematic, and dynamic models, depending on the targets, job limitations, and required performance. Obtaining these models is not a straightforward process and depends on the complexity of the kinematics and degree of freedom of the mechanical structure. Using these models for simulation necessitates the use of efficient and user-friendly algorithms to estimate the geometric and dynamic parameters of the robot. In addition, online application of an impact rule on a robot controller requires models with fewer actions. The approaches presented in this book address these requirements and focus on the dynamic element of the parallel manipulator in this chapter's research. Dynamics modeling

A dynamic model depicts an object's activity across time. It's utilized when an object's activity may be best characterized as a series of states that happen in a specific order.

A dynamic model connects the active forces that operate on a robot to the accelerations that they generate, or vice versa. These active forces might be both rotational moments and translational forces. Robot dynamics is the application of rigid-body dynamics to robotics where the robot mechanism is typically represented as a rigid-body system.

I.3.1 Inverse Dynamic Model (IDM)

The inverse dynamic model is a mathematical model that determines the torques and/or forces required by the robot's actuators to move the end-effector in a particular direction. This model has been utilized to regulate the motion and force of robots.

I.3.2 Forward Dynamic Model (FDM)

The forward dynamic model, also known as the direct dynamic model, calculates the accelerations of state variables given the forces, locations, and velocities. This model is primarily utilized in simulations.

I.3.3 Implicit Dynamic Model (ImplDM)

We provide an implicit dynamic model that is equivalent to a robot's equations of motion (EOM).

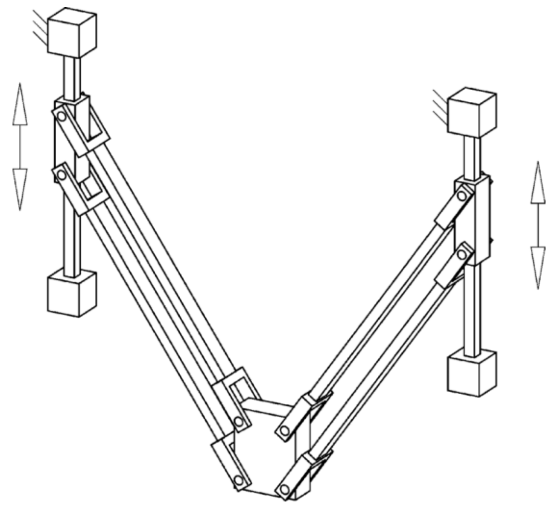
I.4 Classification of parallel manipulators.

Parallel manipulators can be classified based on several characteristics, such as the number of degrees of freedom, kinematic structures, and actuation systems. One common way to classify parallel manipulators is based on their kinematic structures, which can be further categorized into planar and spatial manipulators.

Planar manipulators have two degrees of freedom and operate in a two-dimensional plane, while spatial manipulators have three or more degrees of freedom and operate in three-dimensional space. Planar manipulators are further classified into serial-chain and parallel-chain manipulators, wherein serial-chain manipulators have a single chain of links and joints, while parallel-chain manipulators have multiple chains of links and joints.

Another classification of parallel manipulators is based on their actuation systems, which can be hydraulic, electric, or pneumatic. Additionally, they can be classified based on their application, such as micro/nano manipulators, medical manipulators, and industrial manipulators.

1. Degree of Freedom (DOF): Parallel manipulators can be classified based on their degrees of freedom, which refers to the number of independent motions that the manipulator can execute. Planar manipulators have two DOF and operate in a two-dimensional plane, while spatial manipulators have three or more DOF and operate in three-dimensional space.



Planar manipulators have 2DOF

Sspatial manipulators have 5 DOF

Figure I.2- Degree of Freedom (DOF)

2. Kinematic Structure: Parallel manipulators can also be classified based on their kinematic structures. A planar manipulator can be a serial-chain or parallel-chain manipulator. A serial-chain manipulator has a single chain of links and joints, while a parallel-chain manipulator has multiple chains of links and joints. A spatial manipulator can be a 3-RRR, 3-PRR, or 3-UPR manipulator, depending on their kinematic structure.

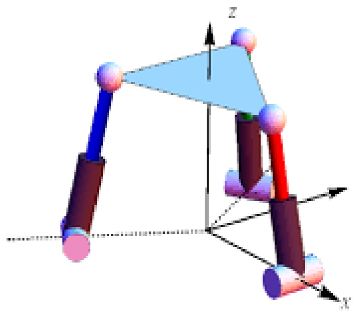
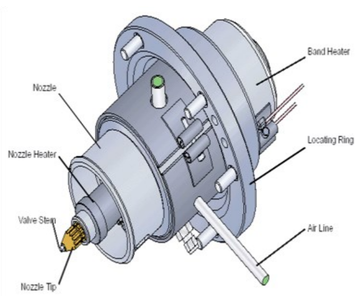
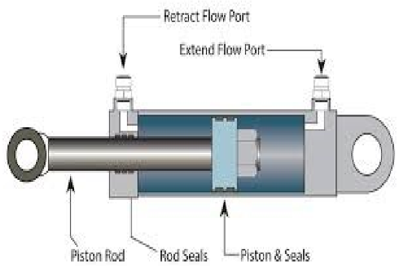


Figure I.3- Kinematics analysis of a novel over-constrained three degree-of-freedom spatial parallel manipulator

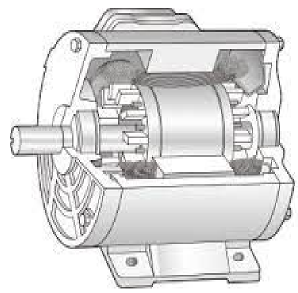
3. Actuation System: Parallel manipulators can be classified based on their actuation systems, which can be hydraulic, electric, or pneumatic. Hydraulic systems are commonly used for high-load applications, while electric systems are used for high-precision applications. Pneumatic systems are used when the manipulator needs to be lightweight and portable.



Parallel processors with pneumatic operating system



Parallel processors with hydraulic drive system



Parallel processors with electric drive system

Figure I.4- Actuation System

4. Application: Parallel manipulators can also be classified based on their application. For example, micro/nano manipulators are used in micro/nano-scale operations, medical manipulators are used in medical procedures, and industrial manipulators are usedin industrial applications such as assembly lines and material handling.

5. Symmetry: Parallel manipulators can also be classified based on their symmetry. Symmetrical manipulators have the same type of kinematic chains on both sides of the end-

effector, while unsymmetrical manipulators have different types of kinematic chains on each side of the end-effector.

6. Singularity: Parallel manipulators can also be classified based on their singularity. Singularity refers to a configuration of the manipulator where it loses one or more degrees of freedom. Manipulators that are prone to singularity are called singular manipulators, while manipulators that are less prone to singularity are called non-singular manipulators.

These are just some of the ways that parallel manipulators can be classified. The classification of parallel manipulators is important because it helps in selecting the appropriate manipulator for a specific application and in designing control strategies for the manipulator.[3]

I.5 Kinematic and dynamic analysis of parallel manipulators

Kinematic analysis of parallel manipulators involves studying the motion of the manipulator's end-effector and its relationship to the motion of its joints. This analysis is essential for determining the manipulator's workspace, dexterity, singularity, and the inverse kinematics problem. The kinematic analysis of parallel manipulators involves determining the forward kinematics to calculate the position, orientation, and velocity of the end-effector and the inverse kinematics to calculate the joint angles necessary to achieve a desired end-effector position and orientation.

Dynamic analysis of parallel manipulators involves studying the forces and torques acting on the manipulator's links and joints. This analysis is essential for understanding the manipulator's behavior under the influence of external forces, such as payloads and disturbances. The dynamic analysis of parallel manipulators involves determining the inverse dynamics to calculate the required joint torques and forces to achieve a desired end-effector trajectory and the forward dynamics to predict the motion of the manipulator under the influence of external forces.

The kinematic and dynamic analyses of parallel manipulators are interrelated and are essential for designing and controlling the manipulator. The kinematic analysis provides information on the manipulator's motion, while the dynamic analysis provides information on the forces and torques acting on the manipulator. The kinematic and dynamic analyses of parallel manipulators are complex due to the manipulator's multiple degrees of freedom, nonlinearity, and singularity. Therefore, advanced mathematical tools, such as robotics software and simulation tools, are necessary to perform accurate and efficient kinematic and dynamic analyses of parallel manipulators.

In addition to the forward and inverse kinematics and dynamics analyses, other essential analyses of parallel manipulators include workspace analysis, dexterity analysis, and singularity analysis. Workspace analysis involves determining the volume of space that the manipulator can reach with its end-effector. Dexterity analysis involves evaluating the manipulator's ability to reach any point in its workspace with a certain level of accuracy and with minimum joint motion. Singularity analysis involves identifying the configurations of the manipulator where it loses one or more degrees of freedom.

Overall, the kinematic and dynamic analyses of parallel manipulators are crucial for designing and controlling these complex robots. These analyses enable engineers to predict the manipulator's behavior under different loading and motion conditions, optimize its performance, and ensure its safety and reliability.

1. Kinematic Analysis: In the kinematic analysis of parallel manipulators, the first step is to determine the forward kinematics, which involves calculating the position, orientation, and velocity of the end-effector given the joint angles. The inverse kinematics is then determined to calculate the joint angles needed to achieve a desired end-effector position and orientation. The kinematic analysis also involves determining the Jacobean matrix, which relates the joint velocities to the end-effector velocity, and the singularity analysis, which identifies the configurations where the manipulator loses one or more degrees of freedom.

2. Dynamic Analysis: In the dynamic analysis of parallel manipulators, the first step is to determine the inverse dynamics, which involves calculating the required joint torques and forces to achieve a desired end-effector trajectory. The forward dynamics is then determined to predict the motion of the manipulator under the influence of external forces, such as payloads and disturbances. The dynamic analysis also involves determining the mass, inertia, and friction parameters of the manipulator and identifying the configurations where the manipulator experiences singularities.

3. Workspace Analysis: The workspace analysis involves determining the volume of space that the manipulator can reach with its end-effector. This analysis is essential for selecting the appropriate manipulator for a given application and for optimizing the manipulator's performance.

4. Dexterity Analysis: The dexterity analysis involves evaluating the manipulator's ability to reach any point in its workspace with a certain level of accuracy and with minimum joint motion.

This analysis is essential for determining the manipulator's performance and for optimizing its control strategies.

5. Singularity Analysis: The singularity analysis involves identifying the configurations of the manipulator where it loses one or more degrees of freedom. This analysis is essential for avoiding these configurations and for ensuring the manipulator's safety and reliability.

6. Simulation Tools: The kinematic and dynamic analyses of parallel manipulators are complex and require advanced mathematical tools, such as robotics software and simulation tools, to perform accurate and efficient analyses. These tools enable engineers to model the manipulator and its environment, simulate its motion under different loading and motion conditions, and evaluate its performance.

Overall, the kinematic and dynamic analysis of parallel manipulators is a crucial step in the design and control of these complex robots. These analyses enable engineers to predict the manipulator's behavior under different loading and motion conditions, optimize its performance, and ensure its safety and reliability. The development of advanced simulation tools has made these analyses more accessible and efficient, enabling engineers to design and control more complex and sophisticated parallel manipulators.[4]

I.6 Methods of dynamic modeling

Control systems for parallel manipulators are essential for regulating their motion and ensuring their safety and performance. These control systems can be classified into two main categories: open-loop control and closed-loop control.

I.6.1 Open-Loop Control :

Open-loop control, also known as feed forward control, involves preprogramming the manipulator's motion based on a desired trajectory. This control method does not rely on feedback from sensors and assumes that the manipulator will follow the desired trajectory accurately. Open-loop control is simple and efficient, but it is not suitable for applications where the manipulator is subject to external disturbances or uncertainties.

I.6.2 Closed-Loop Control:

Closed-loop control, also known as feedback control, involves measuring the manipulator's motion using sensors and adjusting its motion based on the feedback. This control method is more complex than open-loop control but is more robust and reliable. Closed-loop control can be further classified into position, velocity, and force control.

I.6.2.1 Position Control:

Position control involves regulating the manipulator's end-effector position to follow a desired trajectory. This control method is suitable for applications that require high precision and accuracy[5].

I.6.2.2 Velocity Control:

Velocity control involves regulating the manipulator's joint velocities to follow a desired trajectory. This control method is suitable for applications that require high-speed motion and smooth trajectory tracking[6].

I.6.2.3 Force Control:

Force control involves regulating the manipulator's end-effector forces to achieve a desired force or impedance. This control method is suitable for applications that require contact force control, such as assembly or grinding. [7]

Various control strategies can be used to implement closed-loop control for parallel manipulators, such as proportional-integral-derivative (PID) control, adaptive control, and model-based control. These control strategies differ in their complexity, performance, and robustness and are selected based on the manipulator's application and requirements.

In addition to position, velocity, and force control, other control systems can be used for parallel manipulators, such as hybrid control, fuzzy control, and neural network control. These control systems utilize advanced mathematical and computational techniques to optimize the manipulator's performance and improve its adaptability to external disturbances and uncertainties.

1. Proportional-Integral-Derivative (PID) Control: PID control is a widely used control strategy for parallel manipulators. It involves measuring the error between the desired and actual motion of the manipulator and adjusting the control input based on the proportional, integral, and derivative terms. PID control is simple, reliable, and effective for many applications.

2. Adaptive Control: Adaptive control is a control strategy that adjusts the control input based on the manipulator's changing dynamics and uncertainties. It relies on a model of the manipulator's dynamics and uses adaptive algorithms to estimate the model parameters and adjust the control input accordingly. Adaptive control is suitable for applications that involve changing dynamics, uncertainties, and disturbances.

3. Model-Based Control: Model-based control is a control strategy that uses a mathematical model of the manipulator's dynamics to design a control law. The model is derived based on the manipulator's kinematics, dynamics, and actuation system and is used to predict the manipulator's behavior under different operating conditions. Model-based control is suitable for applications that require high precision and accuracy.

4. Hybrid Control: Hybrid control is a control strategy that combines multiple control systems to achieve a desired performance. For example, position control can be used for high-precision motion, while force control can be used for contact force regulation. Hybrid control is suitable for applications that require a combination of different control strategies.

5. Fuzzy Control: Fuzzy control is a control strategy that uses fuzzy logic to map the manipulator's inputs and outputs. It involves defining fuzzy sets and rules to relate the inputs and outputs and adjust the control input accordingly. Fuzzy control is suitable for applications that involve uncertainties and imprecise data.

6. Neural Network Control: Neural network control is a control strategy that uses artificial neural networks to learn the manipulator's dynamics and control law. It involves training the neural network using data from the manipulator's motion and adjusting the control input based on the network output. Neural network control is suitable for applications that involve nonlinear dynamics and uncertainties [8].

The selection of a suitable control system for parallel manipulators depends on the manipulator's application, requirements, and complexity. The control system should be able to regulate the manipulator's motion accurately, efficiently, and robustly and accommodate external disturbances and uncertainties. The development of advanced control strategies and computational tools has made it possible to design and control more complex and sophisticated parallel manipulators with higher performance and safety.

Overall, control systems for parallel manipulators are crucial for regulating their motion and ensuring their safety and performance. The selection of a suitable control system depends on the manipulator's application, requirements, and complexity. The development of advanced control strategies and computational tools has made it possible to design and control more complex and sophisticated parallel manipulators.

I.6.2.4 Existing research on dynamic analysis and control of Cartesian parallel manipulators

Research on dynamic analysis and control of Cartesian parallel manipulators has been an active area of research for many years. This research aims to understand the dynamic behavior of the robot under various loading conditions and develop control strategies to improve its performance. Here are some key findings from the existing research in this area:

Dynamic Analysis:

- Researchers have developed various dynamic models to analyze the motion of Cartesian parallel manipulators. These models include the Lagrangian formulation, the Newton-Euler formulation, and the Kane's formulation. Each of these models has its own advantages and limitations, and researchers have used them to study different aspects of the robot's dynamic behavior.

- One important aspect of dynamic analysis is the study of singularities. Singularities are configurations in which the robot loses one or more degrees of freedom, and they can have a significant impact on the robot's performance. Researchers have developed methods to identify and avoid singularities in Cartesian parallel manipulators, including workspace analysis, Jacobean analysis, and singularity avoidance algorithms.

- Another important aspect of dynamic analysis is the study of the robot's stability. Researchers have investigated the stability of Cartesian parallel manipulators under different loading conditions and control strategies. They have developed methods to assess the stability of the robot, such as Lyapunov stability analysis, and have used these methods to design more stable control strategies.

Control:

- Researchers have developed various control strategies for Cartesian parallel manipulators, including position control, force control, hybrid control, and adaptive control. These strategies aim to achieve precise and efficient motion of the robot under different loading conditions and task requirements.

- In position control, the robot's joint angles are controlled to achieve a desired position and orientation of the end-effector. This strategy is commonly used for pick-and-place operations and other tasks that require precise positioning.

- In force control, the robot's end-effector is controlled to apply a desired force or torque to the environment. This strategy is often used in tasks that require contact force control, such as assembly or machining.

- Hybrid control combines position and force control to achieve both precise positioning and force control. This strategy is often used in tasks that require both positioning and force control, such as robotic surgery.

- Adaptive control involves adjusting the control parameters in real-time based on changes in the robot's environment or task requirements. This strategy can improve the robot's ability to handle unexpected disturbances or changes in the task.

Overall, the research on dynamic analysis and control of Cartesian parallel manipulators has led to significant advancements in the field of robotics. These advancements have enabled the development of robots with improved performance and versatility, and have expanded the range of applications for these robots in industry and medicine. Ongoing research in this area is focused on further improving the performance of these robots, developing new control strategies for more complex tasks, and exploring new applications for them in various industries.

I.7 Conclusion

. In conclusion, the literature review on dynamic analysis and control of Cartesian parallel manipulators has revealed the significant advancements made in the field of robotics. The research has focused on understanding the dynamic behavior of the robot under various loading conditions and developing control strategies to improve its performance. The research findings have led to the development of advanced control strategies for Cartesian parallel manipulators that can handle more complex tasks and achieve high precision and accuracy. Decentralized control, feedback and feed forward control, and adaptive control strategies have been investigated to improve the robot's performance and stability in tasks where disturbances or changes in the environment are common.

The research on dynamic analysis and control of Cartesian parallel manipulators has led to the development of robots that can perform a wide range of tasks with high precision and accuracy, making them an important technology for many industries. These robots are used in manufacturing, aerospace, and medical robotics, among other industries. Ongoing research in this area is focused on further improving the performance of these robots, developing new control strategies for more complex tasks, and exploring new applications for them in various industries.

Overall, the research on dynamic analysis and control of Cartesian parallel manipulators has made significant contributions to the field of robotics and has paved the way for the development of more advanced and sophisticated robots with the potential to revolutionize many industries and improve people's lives in numerous ways.

II Chapter II: Kinematic and Dynamic Analysis

II.1 Introduction

Kinematic and dynamic analysis are critical components of the study of robots and automation. The design, control, and performance of robotic systems depend heavily on the kinematic and dynamic analysis. In particular, the kinematic analysis involves the study of the motion of a robot without considering the forces that cause the motion. Kinematic analysis is essential in determining the robot's position, velocity, and acceleration, which are necessary for developing control strategies for the robot.

Dynamic analysis, on the other hand, is the study of the forces that cause motion in the robot. The dynamic analysis is critical for developing control strategies that improve the robot's performance and stability. Dynamic analysis involves the study of the motion of a robot under different loads and external forces, which is essential for understanding the robot's behavior in various environments.

In this context, Cartesian parallel manipulators are an important area of focus for kinematic and dynamic analysis. These robots have a parallel structure, which provides many advantages over other types of robots, such as high stiffness, accuracy, and speed. The kinematic and dynamic analysis of Cartesian parallel manipulators is essential for improving their performance and versatility in various applications, including manufacturing, aerospace, and medical robotics.

The ongoing research in kinematic and dynamic analysis of Cartesian parallel manipulators is focused on developing new methods and models to improve the accuracy and efficiency of the analysis. The research aims to develop advanced control strategies that can handle more complex tasks and achieve high precision and accuracy. Overall, the kinematic and dynamic analysis of Cartesian parallel manipulators is a critical area of research in robotics, and the advancements made in this area have the potential to revolutionize many industries and improve people's lives in numerous ways.

II.2 Manipulator description and kinematics

A manipulator is a machine or device that is designed to manipulate or move objects. In the context of robotics, manipulators are typically robotic arms that are designed to perform various tasks such as assembly, welding, painting, and material handling. The kinematics of a manipulator refers to the study of its motion, including its position, velocity, and acceleration.

The kinematics of a manipulator can be described in terms of its joint angles and link lengths. A manipulator typically consists of a series of rigid links connected by joints, which can be either revolute (rotational) or prismatic (linear). The joint angles and link lengths determine the position and orientation of the end-effector of the manipulator, which is the part of the manipulator that interacts with the environment.

The kinematics of a manipulator can be modeled using forward and inverse kinematics. Forward kinematics involves computing the position and orientation of the end-effector given the joint angles and link lengths. Inverse kinematics involves computing the joint angles required to achieve a desired position and orientation of the end-effector.

To solve the forward kinematics problem, the Denavit-Hartenberg (DH) convention is commonly used. This convention involves assigning a coordinate frame to each joint of the manipulator, and using these frames to compute the position and orientation of the end-effector. The DH parameters include the link lengths, joint angles, and joint offsets, and are used to calculate the transformation matrix between adjacent frames. Once the transformation matrices are calculated, they can be multiplied together to obtain the transformation matrix from the base frame to the end-effector frame. This transformation matrix can then be used to compute the position and orientation of the end-effector.

Inverse kinematics is a more challenging problem, as it involves finding the joint angles that will achieve a desired position and orientation of the end-effector. There are various methods for solving the inverse kinematics problem, including iterative numerical methods and closed-form solutions for specific types of manipulators.

In addition to forward and inverse kinematics, the kinematics of a manipulator can also be described in terms of its Jacobean matrix. The Jacobean matrix relates the velocity of the end-effector to the velocities of the joints, and can be used to compute the manipulator's velocity and acceleration.

Overall, the kinematics of a manipulator is an important aspect of its design and operation, as it determines the manipulator's ability to perform tasks in a given environment. By understanding the kinematics of a manipulator, engineers and researchers can optimize the design and control of the manipulator for specific applications.

II.2.1 Manipulator structure

A manipulator is a type of robot that is designed to manipulate or move objects in its environment. The structure of a manipulator typically consists of several key components, including:

1. Base: This is the foundation of the manipulator, which provides stability and support.

2. Arm: The arm is the main component of the manipulator, which is responsible for reaching and moving objects. It is typically composed of several segments or links that are connected by joints, allowing it to move in various directions.

3. Joints: Joints are the connections between the links of the manipulator arm, which allow it to move in different directions and orientations. There are several types of joints, including revolute (rotational), prismatic (linear), and spherical (ball-and-socket).

4. End effector: The end effector is the tool or device at the end of the manipulator arm, which is used to grasp or manipulate objects. Examples of end effectors include grippers, suction cups, and welding torches.

5. Actuators: Actuators are the components that provide the force and motion required to move the manipulator arm and end effector. They can be electric, hydraulic, or pneumatic, depending on the specific application and requirements.

Overall, the structure of a manipulator is designed to provide flexibility, precision, and control, allowing it to perform a wide range of tasks in various environments and industries.

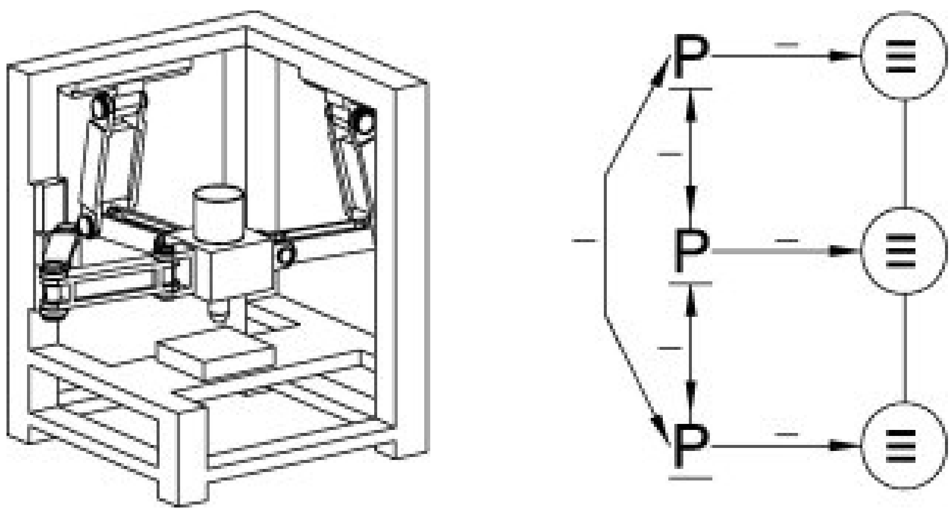


Figure II.1- Topology of serial and parallel manipulators and topological diagrams

II.2.2 Kinematic structure

The kinematic structure of the CPM is shown in figure 2 where a moving platform is connected to a fixed base by three PRRR (Prismatic Revolute) limbs. The origin of the fixed coordinate frame is located at point O and a reference frame XYZ is attached to the fixed base at this point. The moving platform is symbolically represented by a square whose length side is 2L defined by B1, B2, and B3 and the fixed base is defined by three guide rods passing through A1, A2, and A3, respectively. The three revolute joint axes in each limb are located at points Ai, Mi., and Bi, respectively, and are parallel to the ground connected prismatic joint axis. Furthermore, the three prismatic joint axes, passing through point Ai, for i = 1, 2, and 3, are parallel to the X, Y, and Z axes, respectively. Specifically, the first prismatic joint axis lies on the X-axis; the second prismatic joint axis lies on the Y axis; and the third prismatic joint axis is parallel to the Z axis. Point P represents the center of the moving platform. The link lengths are L1, and L2. The starting point of a prismatic joint is defined by d0i and the sliding distance is defined by di - d0i for i = 1, 2, and 3

II.2.3 Kinematics constraints

For this analysis, the position of the end-effector is considered known, and is given by the position vector P= [x, y, z] which defines the location of P at the center of the moving platform in the XYZ coordinate frame. The inverse kinematics analysis produces a set of two joint angles for each limb (θi1 and θi2 for the itch limb) that define the possible postures for each limb for the given position of the moving platform.

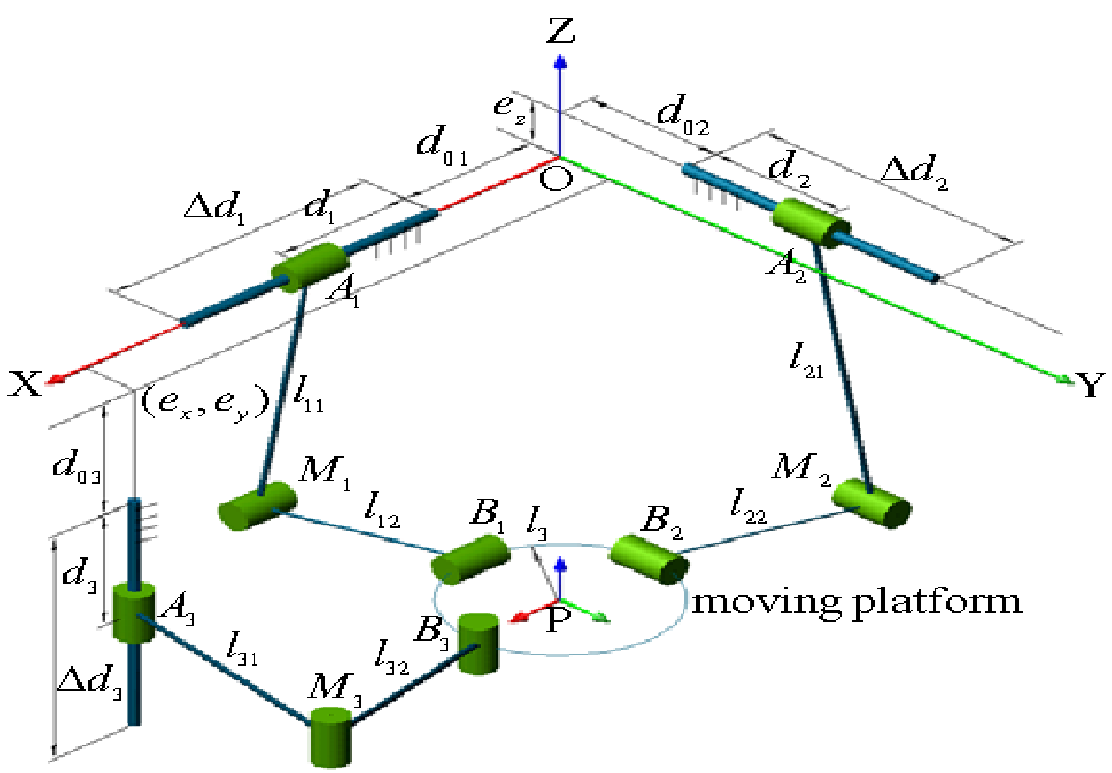


Figure II.2- Spatial 3-PRRR parallel manipulator.

II.2.3.1 The first limb

A schematic diagram of the first limb of the CPM is sketched in figure 3, and then the relationships for the first limb are written for the position P[x, y, z] in the coordinate frame XYZ.

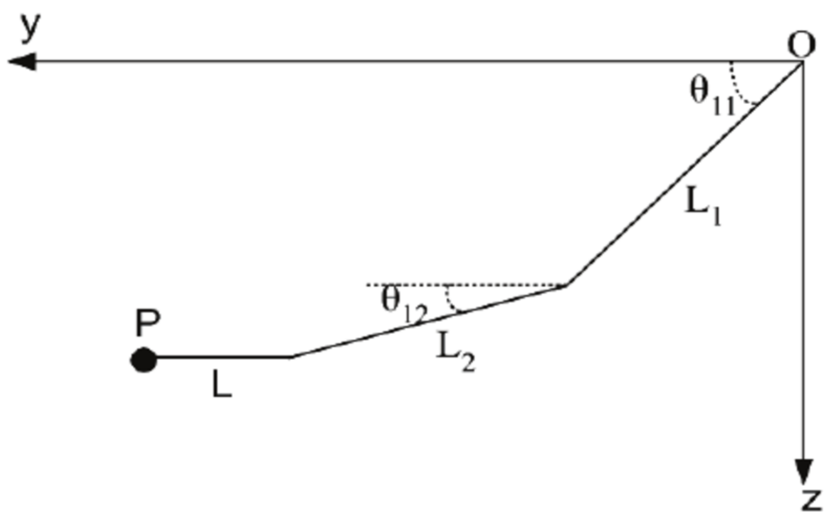


Figure II.3- Description of the joint angles and link lengths for the first limb.

$$y = L_1 cos\theta_{11} + L_2 cos\theta_{12} + L \tag{II-1}$$

$$z = L_1 sin\theta_{11} + L_2 sin\theta_{12} \tag{II-2}$$

$$L_1^2 = (y - L_1 cos\theta_{11} - L)^2 + (z - L_1 sin\theta_{11})^2 \tag{II-3}$$

II.2.3.2 The second limb

A schematic diagram of the second limb of the CPM is sketched in figure 4, and then the relationships for the second limb are written for the position P[x, y, z] in the coordinate

frame XYZ.

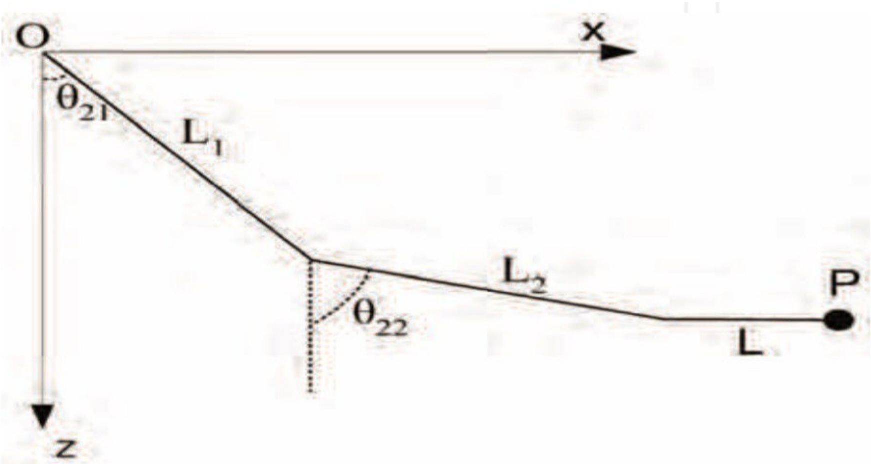


Figure II.4 : Description of the joint angles and link for the second limb.

$$z = L_1 cos\theta_{21} + L_2 cos\theta_{22} \tag{II-4}$$

$$x = L_1 sin\theta_{21} + L_2 sin\theta_{22} + L \tag{II-5}$$

$$L_2^2 = (z - L_1 cos\theta_{21})^2 + (x - L_1 sin\theta_{21} - L)^2 \tag{II-6}$$

II.2.3.3 The third limb

A schematic diagram of the third limb of the CPM is sketched in figure 5, and then the relationships for the third limb are written for the position P[x, y, z] in the coordinate frame XYZ.

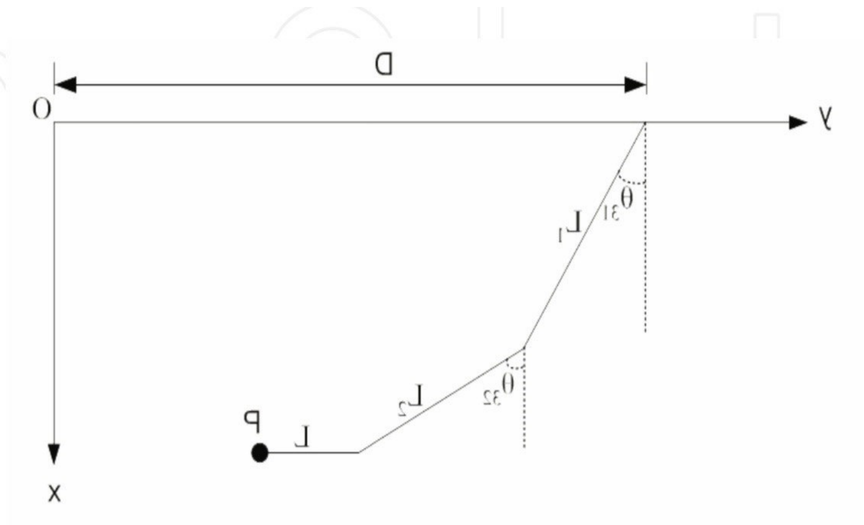


Figure II.5 Description of the joint angles and link lengths for the third limb.

$$x = L_1 cos\theta_{31} + L_2 cos\theta_{32} \quad \text{(II-7)}$$

$$y = D - L - L_1 sin\theta_{31} - L_2 sin\theta_{32} \quad \text{(II-8)}$$

$$L_2^2 = (X - L_1 cos\theta_{31})^2 + (y - D + L_1 sin\theta_{31} + L)^2 \quad \text{(II-9)}$$

II.2.4 Linear actuation method

As described by Han Sung Kim and Lung-Wen Tsai (Kim & Tsai, 2002), for the linear actuation method, a linear actuator drives the prismatic joint in each limb whereas all the other joints are passive. This method has the advantage of having all actuators installed on the fixed base. The forward and inverse kinematic analyses are trivial since there exists a one-to-one correspondence between the end-effector position and the input joint displacements. Referring to figure 2, each limb constrains point P to lie on a plane which passes through point Ai and is perpendicular to the axis of the linear actuator. Consequently, the location of P is determined by the intersection of three planes. A simple kinematic relation can be written as

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \quad \text{(II-10)}$$

II.2.5. Analysis of manipulator dynamics

Having an understanding of manipulator dynamics is crucial from multiple perspectives. Firstly, it is necessary for appropriately sizing the manipulator components and actuators. In the absence of a manipulator dynamics model, it becomes challenging to predict the actuator force requirements and subsequently difficult to select the actuators properly. Secondly, a dynamics model is useful for developing a control scheme, enabling the design of a controller with better performance characteristics than those typically obtained through heuristic methods after the manipulator's construction. Additionally, some control schemes like the computed torque controller rely directly on the dynamics model to anticipate the desired actuator force for feed forward purposes. Thirdly, a dynamical model can be used for computer simulation of a robotic system, allowing examination of various manufacturing automation tasks without the need for a real system.

Several methods have been used to characterize the dynamics of parallel manipulators, with the most common based on the Newton-Euler formulations and Lagrange's equations of motion (Tsai, 1999). The traditional Newton-Euler formulation necessitates the equations of motion to be written for each manipulator body, resulting in a large number of equations and poor computational efficiency. The Lagrangian formulation, on the other hand, eliminates unwanted reaction forces and moments from the outset, making it more efficient than the Newton-Euler formulation. However, deriving explicit equations of motion in terms of a set of independent generalized coordinates becomes a prohibitive task due to the numerous constraints imposed by a manipulator's closed loops. To simplify the problem, additional coordinates and a set of Lagrangian multipliers are often introduced.

II.2.5.1 Lagrange based dynamic analysis

It can be assumed that the first rod of each limb is a uniform and its mass is m1. The mass of second rod of each limb is evenly divided between and concentrated at joints Mi. and Bi. This assumption can be made without significantly compromising the accuracy of the model since the concentrated mass model of the connecting rods does capture some of the dynamics of the rods. Also, the damping at the actuator is disregarded since the Lagrangian model does not readily accommodate viscous damping as is assumed for the actuators. The Lagrangian equations are written in terms of a set of redundant coordinates. Therefore, the formulation requires a set of constraint equations derived from the kinematics of a mechanism. These constraint equations and their derivatives must be adjoined to the equations of motion to produce a number of equations that is equal to the number of unknowns. In general, the Lagrange multiplier approach involves solving the following system of equations :

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} = Q_j + \sum_{i=1}^{k}\left(\lambda_i \frac{\partial f_i}{\partial q_j}\right) \qquad \text{(II-11)}$$

For j =1 ton, where

j : is the generalized coordinate index,

n: is the number of generalized coordinates,

i : is the constraint index,

q_j: is the j^{th} generalized coordinate,

k : is the number of constraint functions,

L : is the Lagrange function, where L= T− V,

T : is the total kinetic energy of the manipulator,

V : is the total potential energy of the manipulator,

f_i: is a constraint equation,

Q_j: is a generalized external force, and

λ_i: is the Lagrange multiplier.

Theoretically, the dynamic analysis can be accomplished by using just three generalized coordinates since this is a 3 DOF manipulator. However, this would lead to a cumbersome expression for the Lagrange function, due to the complex kinematics of the manipulator. So we choose three redundant coordinates which areθ_{11}, $\theta_{21}$and θ_{31}beside the generalized coordinates x, y, and z. Thus we haveθ_{11}, θ_{21}, θ_{31}, , x, y, and z as the generalized coordinates. Equation 11 represents a system of six equations in six variables, where the six variables are $\lambda_i$for i = 1, 2, and 3, and the three actuator forces, $Q_i$for j = 4, 5, and 6. The external generalized forces, $Q_i$for j=1, 2, and 3, are zero since the revolute joints are passive. This formulation requires three constraint equations, $f_i$for i = 1, 2, and 3, that are written in terms of the generalized coordinates.

II.3 Kinematics and dynamics equations for the CPM

The forward and inverse kinematics and dynamics equations for the Cartesian parallel manipulator can be derived using various methods, including the Lagrangian formulation, the Newton-Euler formulation, and the Kane's formulation. Here is a brief overview of each method:

II.3.1 Lagrangian Formulation:

The Lagrangian formulation is a mathematical method for deriving the equations of motion for a mechanical system. In this method, the Lagrangian of the system, which is the difference between the kinetic and potential energies of the system, is used to derive the equations of motion.

To derive the forward kinematics equations using the Lagrangian formulation, the position and orientation of the moving platform are expressed as functions of the joint angles. The Lagrangian of the system is then derived using the position and orientation functions, and the equations of motion are obtained by differentiating the Lagrangian with respect to time.

To derive the inverse kinematics equations using the Lagrangian formulation, the joint angles are expressed as functions of the position and orientation of the moving platform. The Lagrangian of the system is then derived using the joint angle functions, and the equations of motion are obtained by differentiating the Lagrangian with respect to time.

II.3.2 Newton-Euler Formulation:

The Newton-Euler formulation is a method for deriving the equations of motion for a mechanical system based on Newton's laws of motion and Euler's equations of motion.

To derive the forward kinematics equations using the Newton-Euler formulation, the position and orientation of the moving platform are expressed as functions of the joint angles. The forces and torques acting on the system are then calculated using Newton's laws of motion, and the equations of motion are obtained using Euler's equations of motion.

To derive the inverse kinematics equations using the Newton-Euler formulation, the joint angles are expressed as functions of the position and orientation of the moving platform. The forces and torques acting on the system are then calculated using Newton's laws of motion, and the equations of motion are obtained using Euler's equations of motion.

II.3.3 Kane's Formulation:

Kane's formulation is a method for deriving the equations of motion for a mechanical system based on the principle of virtual work. In this method, the virtual work done by the forces and torques acting on the system is equated to zero to derive the equations of motion.

To derive the forward kinematics equations using Kane's formulation, the position and orientation of the moving platform are expressed as functions of the joint angles. The virtual work done by the forces and torques acting on the system is then calculated, and the equations of motion are obtained by equating the virtual work to zero.

To derive the inverse kinematics equations using Kane's formulation, the joint angles are expressed as functions of the position and orientation of the moving platform. The virtual work done by the forces and torques acting on the system is then calculated, and the equations of motion are obtained by equating the virtual work to zero.

In summary, the forward and inverse kinematics and dynamics equations for the Cartesian parallel manipulator can be derived using various mathematical methods, including the Lagrangian formulation, the Newton-Euler formulation, and Kane's formulation. These equations are essential for understanding the behavior of the robot and developing control strategies that improve its performance and versatility. The ongoing research in this area is focused on developing new methods and models to improve the accuracy and efficiency of the analysis and control strategies that can handle more complex tasks and achieve high precision and accuracy [9]

II.3.4 Kinematics

Kinematic modeling includes the study of manipulator positions without consideration of the external forces and moments. Inverse kinematics is the inverse of forward kinematics i.e. obtaining the joint angles for the necessary and desired position of the end effector. Figure 3a and 3b represents the schematic diagram of 3PUU PM. For the kinematic analysis, Cartesian coordinate system has been assigned to the kinematic model as shown in Figure 3a. The inverse kinematics with the tool is shown in Figure 3b. The fixed base is denoted by a global reference system O{x, y, z} which is located at the center of the base platform. Similarly, the moving platform is assigned with a moving Cartesian frame O' {x', y', z'} which is located at the center of moving platform. Where, b1, b2 and b3 are the vertices of the base joints and a1, a2 and a3 are the vertices of moving joints. The global reference system and the moving frame are parallel to each other

II.3.5 Derivation of the manipulator's dynamics

II.3.5.1 The kinetic and potential energy of the first limb

Referring to figure 6, the velocities of A1 (the prismatic joint of the first limb), A2 and A3 are $\dot{x}$, $\dot{y}$ and $\dot{z}$. The angular velocity of the rod A1 M1 is $\dot{\theta}_{11}$. We can consider the moment of inertia of rods A1 M1, A2 M2, and A3 M3 is $I = \frac{m_1}{12} L_1^2$. m3 is the mass of each prismatic joint (A1, A2, and A3). So, the total kinetic energy of the first limb, T1, is

$$T_1 = [m_1 + m_2 + m_3]\frac{\dot{x}^2}{2} + \frac{m_2}{2}(\dot{y}^2 + \dot{z}^2) + (\frac{m_1}{6} + \frac{m_2}{4})L_1^2\dot{\theta}_{11}^2 \qquad \text{(II-12)}$$

The total potential energy of the first limb, V1,is

$$V_1 = -\frac{m_1 + m_2}{2} gL_2 sin\theta_{11} - \frac{m_2}{2} gz \qquad \text{(II-13)}$$

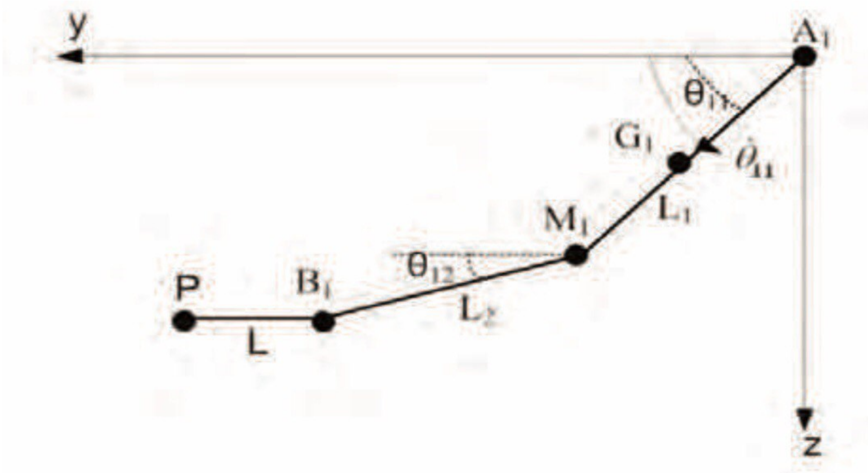


Figure II.6: Schematic diagram of the first limb for the dynamic analysis.

II.3.5.2 The kinetic and potential energy of the second limb

Referring to figure 7, if the angular velocity of the rod A2 M2 is θ_{21}, the total kinetic energy of the second limb, T2 is

$$T_2 = [m_1 + m_2 + m_3]\frac{y^2}{2} + \frac{m_2}{2}(\dot{x}^2 + \dot{z}^2) + (\frac{m_1}{6} + \frac{m_2}{4})L_1^2\dot{\theta}_{21}^2 \qquad \text{(II-14)}$$

The total potential energy of the second limb, V2, is given by

$$V_2 = -\frac{m_1 + m_2}{2} gL_1 sin\theta_{21} - \frac{m_2}{2} gz \qquad \text{(II-15)}$$

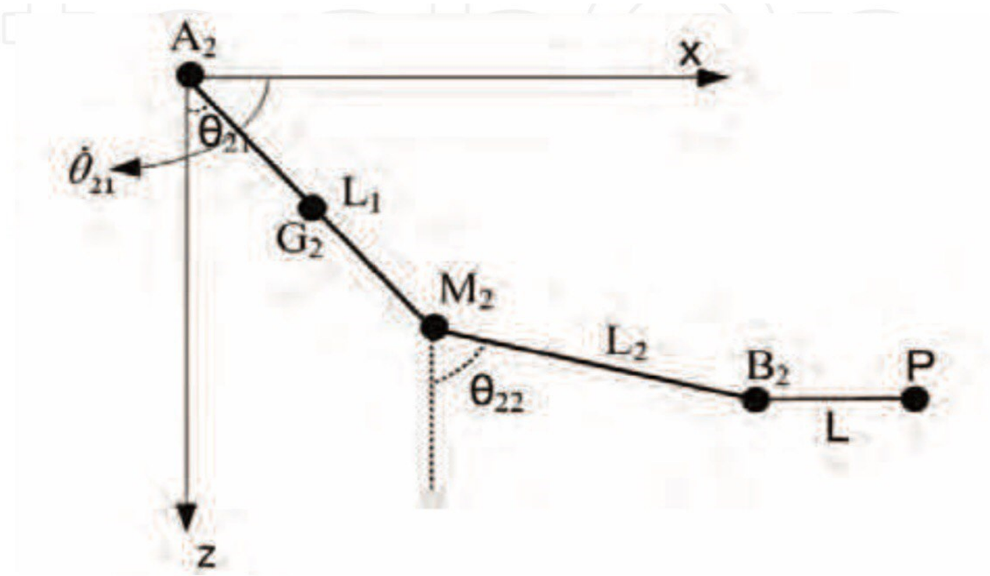


Figure II.7: Schematic diagram of the second limb for the dynamic analysis

II.3.5.3 The kinetic and potential energy of the third limb

Referring to figure 8, the total kinetic energy of the second limb, T3 is

$$T_3 = [m_1 + m_2 + m_3]\frac{\dot{z}^2}{2} + \frac{m_2}{4}(\dot{x}^2 + \dot{y}^2) + \left(\frac{m_1}{6} + \frac{m_2}{4}\right)L_1^2\dot{\theta}_{31}^2 \qquad \text{(II-16)}$$

The total potential energy of the third limb, V3, is

$$V_3 = -[m_1 + m_2 + m_3]gz \qquad \text{(II-17)}$$

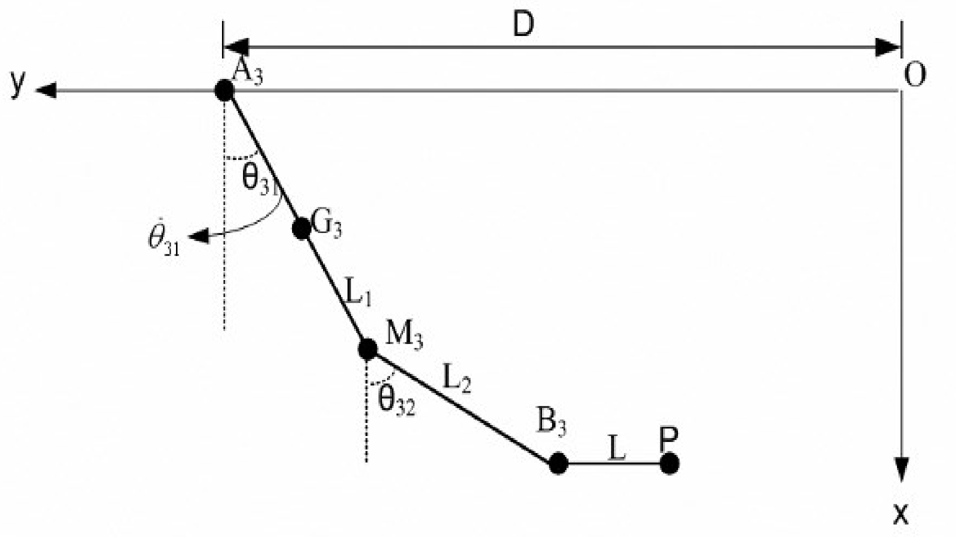


Figure II.8 Schematic diagram of the third limb for the dynamic analysis..

II.3.6 Derivation of the Lagrange equation

From equations 12, 14, and 16, the total kinetic energy of the manipulator T is given by:

$$T = \frac{1}{2}[m_1 + m_2 + m_3 + m_4](\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + (\frac{m_1}{6} + \frac{m_2}{4})L_1^2(\dot{\theta}_{11}^2 + \dot{\theta}_{21}^2 + \dot{\theta}_{31}^2) \qquad \text{(II-18)}$$

where m4 is the mass of the tool. From equations 13, 15, and 17, the total potential energy V of the manipulator is calculated relative to the plane of the stationary platform of the manipulator, and is found to be:

$$V = -\frac{m_1 + m_2}{2}gL_1(sin\theta_{11} + cos\theta_{21}) - (m_1 + m_2 + m_3 + m_4)gz \qquad \text{(II-19)}$$

The Lagrange function is defined as the difference between the total kinetic energy, T, and the total potential energy V: L= T− V

$$L = A(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + B(\dot{\theta}_{11}^2 + \dot{\theta}_{21}^2 + \dot{\theta}_{31}^2) + C(sin\theta_{11} + cos\theta_{21}) + Ez \qquad \text{(II-20)}$$

Where:

$$A = \frac{1}{2}[m_1 + 2m_2 + m_3 + m_4], B = \left(\frac{m_1}{6} + \frac{m_2}{4}\right)L_1^2, C = \frac{m_1+m_2}{2} gL_1 \text{ And } E = [m_1 + 2m_2 + m_3 + m_4]g \tag{II-21}$$

II.3.6.1 The constraint equations

Differentiation of equation 3 with respect to time yields

$$0 = \frac{(y-L_1\cos\theta_{11}-L)}{((y-L)\sin\theta_{11}-z\cos\theta_{11})L_1}\dot{y} + \frac{(y-L_1\sin\theta_{11})}{((y-L)\sin\theta_{11}-z\cos\theta_{11})L_1}\dot{z} + \dot{\theta}_{11} \tag{II-22}$$

Differentiation of equation 6 with respect to time yields

$$0 = \frac{(z-L_1 cos\theta_{21})}{L_1(zsin\theta_{21}+(L-x)cos\theta_{21})}\dot{z} + \frac{(x-L_1 sin\theta_{21}-L)}{L_1(zsin\theta_{21}+(L-x)cos\theta_{21})}\dot{x} + \dot{\theta}_{11} \tag{II-23}$$

Differentiation of equation 9 with respect to time yields

$$0 = \frac{(x-L_1 sin\theta_{31})}{(xsin\theta_{31}+cos\theta_{31}(y-D+L))L_1}\dot{x} + \frac{(x-D+L_1 sin\theta_{31}+L)}{(xsin\theta_{31}+cos\theta_{31}(y-D+L))L_1}\dot{y} + \dot{\theta}_{31} \tag{II-24}$$

The equations 27, 28, and 29 are rearranged so as to produce:

$$\begin{bmatrix}\dot{\theta}_{11}\\ \dot{\theta}_{21}\\ \dot{\theta}_{31}\end{bmatrix} = \Gamma \begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} \tag{II-25}$$

Where

$$\Gamma = -\begin{bmatrix} 0 & \dfrac{(y - L_1 cos\theta_{11} - L)}{(y-L)sin\theta_{11} - zcos\theta_{11})L_1} & \dfrac{(y - L_1 sin\theta_{11})}{(y-L)sin\theta_{11} - zcos\theta_{11})L_1} \\ \dfrac{(x - L_1 sin\theta_{21} - L)}{L_1(zsin\theta_{21} + (L-x)cos\theta_{21})} & 0 & \dfrac{(z - L_1 cos\theta_{21})}{L_1(zsin\theta_{21} + (L-x)cos\theta_{21})} \\ \dfrac{(x - L_1 sin\theta_{31})}{\big(xsin\theta_{31} + cos\theta_{31}(y - D + L)\big)L_1} & \dfrac{(x - D + L_1 sin\theta_{31} + L)}{\big(xsin\theta_{31} + cos\theta_{31}(y - D + L)\big)L_1} & 0 \end{bmatrix}$$

(II-26)

II.3.6.2 Taking the derivatives of the Lagrange function with respect to θ_{11}

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}_{11}}\right) = 2B\ddot{\theta}_{11}, \frac{\partial L}{\partial\theta_{11}} = C\ cos\theta_{11} \tag{II-27}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}_{11}}\right) - \frac{\partial L}{\partial\theta_{11}} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial\theta_{11}}\right) + Q_1 \tag{II-28}$$

(Q1, Q2 and Q3 =0 since the revolute joints are passive)

$$2B\ddot{\theta}_{11} - C\ cos\theta_{11} = \lambda_1 \tag{II-29}$$

II.3.6.3 Taking the derivatives of the Lagrange function with respect to $\boldsymbol{\theta_{21}}$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{21}}\right) = 2B\ddot{\theta}_{21}, \frac{\partial L}{\partial \theta_{21}} = -C\ sin\theta_{21} \quad \text{(II-30)}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{21}}\right) - \frac{\partial L}{\partial \theta_{21}} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial \theta_{21}}\right) + Q_2 \quad \text{(II-31)}$$

$$2B\ddot{\theta}_{21} + C\ sin\theta_{21} = \lambda_2 \quad \text{(II-32)}$$

II.3.6.4 Taking the derivatives of the Lagrange function with respect to $\boldsymbol{\theta_{31}}$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{31}}\right) = 2B\ddot{\theta}_{31}, \frac{\partial L}{\partial \theta_{31}} = 0 \quad \text{(II-33)}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_{31}}\right) - \frac{\partial L}{\partial \theta_{31}} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial \theta_{31}}\right) + Q_3 \quad \text{(II-34)}$$

$$2B\ddot{\theta}_{31} = \lambda_3 \quad \text{(II-35)}$$

Rearrangement of equations 31, 32, and 33 produces:

$$2B\begin{bmatrix}\ddot{\theta}_{11}\\ \ddot{\theta}_{21}\\ \ddot{\theta}_{31}\end{bmatrix} + C\begin{bmatrix}-cos\theta_{11}\\ sin\theta_{21}\\ 0\end{bmatrix} = \Gamma\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} \quad \text{(II-36)}$$

Differentiation equation 30 with respect to time yields

$$\begin{bmatrix}\ddot{\theta}_{11}\\ \ddot{\theta}_{21}\\ \ddot{\theta}_{31}\end{bmatrix} = \Gamma\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} + \frac{d\Gamma}{dt}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} \quad \text{(II-37)}$$

Substituting into equation 34 yields

$$\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} = 2B\Gamma\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} + 2B\frac{d\Gamma}{dt}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} + C\begin{bmatrix}-cos\theta_{11}\\ sin\theta_{21}\\ 0\end{bmatrix} \quad \text{(II-38)}$$

II.3.6.5 Taking the derivatives of the Lagrange function with respect to X

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = 2A\ddot{x}, \frac{\partial L}{\partial x} = 0 \quad \text{(II-39)}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \frac{\partial L}{\partial x} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial x}\right) + Q_4 \quad \text{(II-40)}$$

$$2A\ddot{x} = F_x - \Gamma_{11}\lambda_1 - \Gamma_{21}\lambda_2 - \Gamma_{31}\lambda_3 \quad \text{(II-41)}$$

where F_x, $F_x$and $F_x$are the forces applied by the actuator for the first, second and third limbs. $\Gamma_{ij}$is the (i, j) element of the Γ matrix

II.3.6.6 Taking the derivatives of the Lagrange function with respect to Y

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}}\right) = 2A\ddot{y}, \frac{\partial L}{\partial y} = 0 \tag{II-42}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}}\right) - \frac{\partial L}{\partial y} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial y}\right) + Q_3 \tag{II-43}$$

$$2A\ddot{y} = F_y - \Gamma_{12}\lambda_1 - \Gamma_{22}\lambda_2 - \Gamma_{32}\lambda_3 \tag{II-44}$$

II.3.6.7 acing the derivatives of the Lagrange function with respect to Z

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{z}}\right) = 2A\ddot{z}, \frac{\partial L}{\partial z} = E \tag{II-45}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{z}}\right) - \frac{\partial L}{\partial z} = \sum_{i=1}^{3}\left(\lambda_i \frac{\partial f_i}{\partial z}\right) + Q_6 \tag{II-46}$$

$$2A\ddot{z} - E = F_z - \Gamma_{13}\lambda_1 - \Gamma_{23}\lambda_2 - \Gamma_{33}\lambda_3 \tag{II-47}$$

Rearrangement of equations 36, 37, and 38 produces:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = 2A\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ E \end{bmatrix} + \Gamma^T \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} \tag{II-48}$$

Substituting into equation 35 yields

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = -\begin{bmatrix} 0 \\ 0 \\ E \end{bmatrix} + \Gamma^T C \begin{bmatrix} -cos\theta_{11} \\ sin\theta_{21} \\ 0 \end{bmatrix} + (2AI + \Gamma^T 2B\Gamma \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + \Gamma^T 2B \frac{d\Gamma}{dt}\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{II-49}$$

The dynamic equation of the whole system can be written as

$$F = M(q)\ddot{q} + G(q,\dot{q})\dot{q} + K(q) \tag{II-50}$$

Where

$$F = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}, q = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix}, q = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}, q = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, M(q) = 2AI + \Gamma^T 2B\Gamma, \mathrm{G}(q,\dot{q}) = 2B\frac{d\Gamma}{dt} \tag{II-51}$$

$$K(q) = -\begin{bmatrix} 0 \\ 0 \\ E \end{bmatrix} + \Gamma^T C \begin{bmatrix} -cos\theta_{11} \\ sin\theta_{21} \\ 0 \end{bmatrix} \tag{II-52}$$

where q is the 3×1 vector of joint displacements, $\dot{q}$ is the 3×1 vector of joint velocities, F is the 3×1 vector of applied force inputs, M(q) is the manipulator inertia matrix, G (q ,$\dot{q}$)$ is the manipulator centripetal and carioles matrix which is the 3×3 matrix of centripetal and carioles forces, and K(q) is the vector of gravitational forces which is the 3×1 vector of gravitational forces due to gravity. The Lagrangian dynamics equation, equation 39, possess important properties that facilities analysis and control system design. Among these are the M(q) is a 3×3 symmetric and positive definite matrix and th matrix W(q ,$\dot{q}$)= $\dot{M}$(q)-2G(q ,$\dot{q}$) is a skew symmetric matrix.

II.4 Controller design

Introduction:

The control problem for robot manipulators is the problem of determining the time history of joint inputs required to cause the end-effector to execute a commanded motion. The joint inputs may be joint forces or torques depending on the model used for controller design. Position control and trajectory tracking are the most common tasks for robot manipulators; given a desired trajectory, the joint force is chosen so that the manipulator follows that trajectory. The control strategy should be robust with respect to initial condition errors, sensor noise, and modeling errors. The primary goal of motion control in joint space is to make the robot joints q track a given time varying desired joint positionq_d. Rigorously, we say that the motion control objective in joint space is achieved provided that $\lim_{t\to+\infty} e(t) = 0$

where e(t)=q_d(t) - q(t) denotes the joint position error. Although the dynamics of the manipulator‘s equation is complicated, it nevertheless is an idealization, and there are number of dynamic effects that are not included in this equation. For example, friction at the joints is not accounted for in this equation and may be significant for some manipulators. Also, no physical body is completely rigid. A more detailed analysis of robot dynamics would include various sources of flexibility, such as elastic deformation of bearings and gears, deflection of the links under load, and vibrations.

II.4.1 PID control versus model based control:

The PID controller is a single-input/single-output (SISO) controller that produces a control signal that is a sum of three terms. The first term is proportional (P) to the positioning error, the second term is proportional to the integral (I) of the error, and the third is proportional to the derivative (D) of the error. The PID (or PD) type is usually employed in industrial robot manipulators because it is easy to implement and requires little computation time during real time operation. This approach views each actuator of the manipulator independently, and essentially ignores the highly coupled and nonlinear nature of the manipulator. The error between the actual and desired joint position is used as feedback to control the actuator associated with each joint. However, independent joint controllers cannot achieve a satisfactory performance due to their inherent low rejection of disturbances and parameter variations. Because of such limitation, model based control algorithms were proposed that have the potential to perform better than independent joint controllers that do not account for manipulator dynamics. However, the difficulty with the model based controller is that it requires a good model of the manipulator inverse dynamics and good estimates of the model parameters, making this controller more complex and difficult to implement than the non-model based controller. The model based control scheme was intensively experimentally tested for example the experimental evaluation of computed torque control was presented in (Griffiths et al., 1989)

II.4.2 PD control with position and velocity reference:

The first PD control law is based on the position and velocity error of each actuator in the joint space. To implement the joint control architecture, the values for the joint position error and the joint rate error of the closed chain system are used to compute the joint control force F (Spong & Vidyasagar, 1989).

$$F = k_p e + k_D \dot{e} \tag{II-53}$$

Where $e = q_d - q$,which is the vector of position error of the individual actuated joints, and $\dot{e} = \dot{q}_d - \dot{q}$, which is the vector of velocity error of the individual actuated joints. Where $\dot{q}_d$and $q_d$are the desired joint velocities and positions, and $K_d$and $K_p$are 3 ×3 diagonal matrices of velocity and position gains. Although this type of controller is suitable for real time control since it has very few computations compared to the complicated nonlinear dynamic equations, there are a few downsides to this controller. It needs high update rate to achieve reasonable accuracy. Using local PD feedback law at each joint independently does not consider the couplings of dynamics between robot links. As a result, this controller can cause the motor to overwork compared to other controllers presented next.

II.4.3 PD Control with gravity compensation

This is a slightly more sophisticated version of PD control with a gravitational feed forward term. Consider the case when a constant equilibrium posture is assigned for the system as the reference input vector q_d. It is desired to find the structure of the controller which ensures global asymptotic stability of the above posture. The control law F is given by (Spong & Vidyasagar, 1989):

$$F = k_p e + k_D \dot{e} + K(q_d) \tag{II-54}$$

It has been shown (Spong, 1996) that the system is asymptotically stable but it is only proven with constant reference trajectories. Although with varying desired trajectories, this type of controller cannot guarantee perfect tracking performance. Hence, more dynamic modeling information is needed to incorporate into the controller.

II.4.4 PD control with full dynamics feed forward terms

This type of controller augments the basic PD controller by compensating for the manipulator dynamics in the feed forward way. It assumes the full knowledge of the robot parameters. The key idea for this type of controller is that if the full dynamics is correct, the resulting force generated by the controller will also be perfect. The controller is given by

(Gullayanon , 2005

$$F = M(q_d)\ddot{q}_d + \mathrm{G}(q, \dot{q})\dot{q}_d + K(q_d) + k_p e + k_D \dot{e} \tag{II-55}$$

If the dynamic knowledge of the manipulator is accurate, and the position and velocity error terms are initially zero, the applied force $F = M(q_d)\ddot{q}_d + \mathrm{G}(q_d, \dot{q})\dot{q}_d + K(q_d)$ is sufficient to maintain zero tracking error during motion. This controller is very similar to the computed torque controller, which is presented next. The difference between these two controllers is the location of the position and velocity correction terms. This controller is less sensitive to any mass changes in the system. For example, if the robot picks up a heavy load in the middle of its operation, this controller is likely to respond to this change slower compared to computed torque controller. The advantage of this controller is that once the desired trajectory for a given task has been specified, then the feed forward terms relying on the robot dynamics $M(q_d)\ddot{q}_d + \mathrm{G}(q_d, \dot{q})\dot{q}_d + K(q_d)$ can be computed offline to reduce the computational burden.

II.4.5 Computed torque control:

This controller is developed for the manipulator to examine if it is possible to improve the performance of the trajectory tracking of the manipulator by utilizing a more complete understanding of the manipulator dynamics in the controller design. This controller employs a computed torque control approach, and it uses a model of the manipulator dynamics to estimate the actuator forces that will result in the desired trajectory. Since this type of controller takes into account the nonlinear and coupled nature of the manipulator, the potential performance of this type of controller should be quite good. The disadvantage of this approach is that it requires a reasonably accurate and computationally efficient model of the inverse dynamics of the manipulator to function as a real time controller. The controller computes the dynamics online, using the sampled joint position and velocity data. The key idea is to find an input vector, F, as a function of the system states, which is capable to realize an input/output relationship of linear type. It is desired to perform not a local linearization but a global linearization of system dynamics obtained by means of a nonlinear state feedback. Using the computed torque approach with a proportional-derivative (PD) outer control loop, the applied actuator forces are calculated at each time step using the following force law as described by Lewis, 1993:

$$F = M(q)\left[\ddot{q}_d + k_D\dot{e} + k_p e\right] + \mathrm{G}(q,\dot{q})\dot{q} + K(q) \qquad \text{(II-56)}$$

where F is the force applied to input links, $K_D$is the diagonal matrix of the derivative gains, K_P is the diagonal matrix of the proportional gains, and e is the vector of the position errors of the input links,$e = q_d - q$ To show that the computed torque control scheme linearizes the controlled system, the force computed by equation 43 is substituted into equation 39, yielding:$M(q) = M(q)\ddot{q}_d + M(q)[K_d\dot{e} + K_P e]$. Then multiplying each term by M^{-1}(q), and substituting the relationship,$\ddot{e} = \ddot{q_d} - \ddot{q}$, provides the following linear relationship for the error:

$$\ddot{e} + k_D\dot{e} + k_p e = 0 \qquad \text{(II-57)}$$

This relationship can be used to select the gains to give the desired nature of the closed loop error response since the solution of equation 44 provides a second order damped system with a natural frequency of ω_n, and a damping ratio of ζ where:

$$\omega_n = \sqrt{k_p}, \zeta = \frac{k_D}{2\sqrt{k_p}} \qquad \text{(II-58)}$$

Since the equation 44 is linear, it is easy to choose $K_D$and $K_P$so that the overall system is stable and e → 0 exponentially as t→∞. Usually, if $K_D$and $K_P$are positive diagonal matrices, the control law 43 applied to the system 39 results in exponentially trajectory tracking. It is customary in robot applications to take the damping ration ζ =1 so that the response is critically damped. This produces the fastest non-oscillatory response. The natural frequency $\omega_n$determines the speed of the response. So, the values for the gain matrices K_D and $K_P$are determined by setting the gains to maintain the following relationship:

$$k_D = 2\sqrt{k_p} \qquad \text{(II-59)}$$

If the error response is critically damped. Hence, the general solution of equation 44 is:

$$e(t) = (c_1 + c_2 t)e^{-\frac{k_D}{2}t} \qquad \text{(II-60)}$$

where C1 and C2 are constants

II.5 Kinematic study

In the kinematics analysis of a 3-DoF PKM, the position and velocity of the end-effector are determined as functions of the joint angles. The position analysis involves determining the position of the end-effector in three-dimensional space.

The vector method for position analysis involves representing the position of the end-effector as a vector from the base to the end-effector. This vector is then decomposed into its x, y, and z components. The joint angles can be used to determine the length and orientation of the vector, which allows for the determination of the end-effector position.

The geometric method for position analysis involves constructing a geometric model of the PKM. This model includes the base, the links, and the end-effector. The joint angles can be used to determine the position of each link, which allows for the determination of the end-effector position.

Both methods can be used to determine the position of the end-effector, but the choice of method may depend on the specific requirements of the application. For example, the vector method may be more appropriate for applications that require a quick calculation of the end-effector position, while the geometric method may be more suitable for applications that require a more detailed analysis of the PKM's kinematics.

Once the position of the end-effector is determined, the velocity can be calculated using the time derivative of the position. The velocity analysis provides information about the speed and direction of the end-effector motion, which is important for control and trajectory planning

The Jacobin matrix can also be used to analyze the kinematics of a 3-DoF PKM. The Jacobin matrix relates the end-effector velocity to the joint velocities and provides information about the robot's motion. The Jacobin matrix can be computed using the partial derivatives of the forward kinematics equations with respect to the joint angles.

In summary, the kinematics analysis of a 3-DoF PKM involves analyzing its position and velocity. Both the vector and geometric methods can be used for position analysis, and the choice of method may depend on the specific requirements of the application. Velocity analysis provides information about the end-effector motion, while the Jacobin matrix can be used to relate the end-effector velocity to joint velocities. Understanding the kinematics of a 3-DoF PKM is important for trajectory planning, control, and simulation.[10, 11]

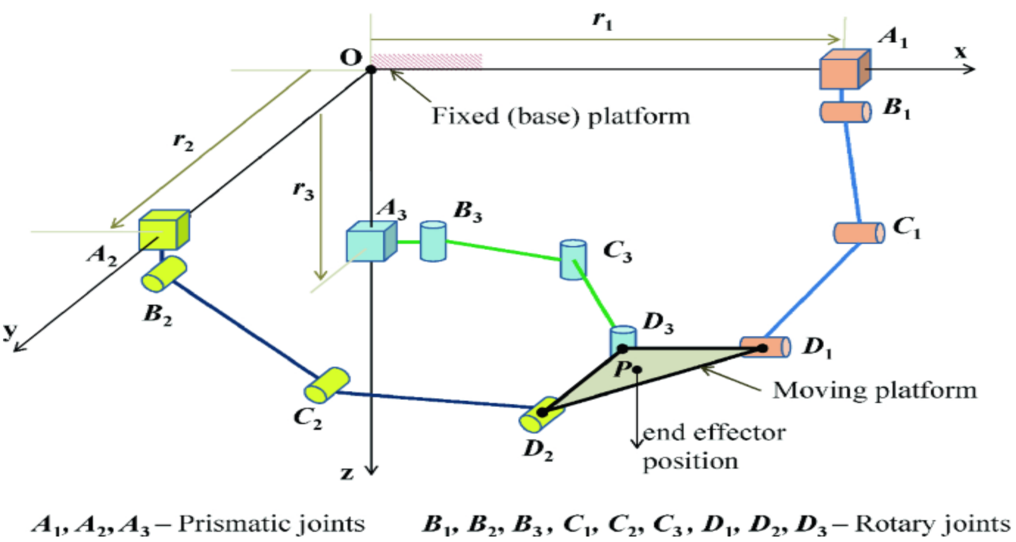


Figure II.9- Conceptual Design and Control of a Sitting-Type Lower-Limb Rehabilitation System Established on a Spatial 3-PRRR Parallel Manipulator | SpringerLin

II.6 Workspace

The workspace of a 3-DOF parallel plane manipulator, also known as a planar parallel manipulator, is a region in the three-dimensional space where the end-effector can reach. The manipulator consists of three arms or links that are connected to a fixed base and a movable platform in a parallel configuration.

The shape and size of the workspace of a 3-DOF parallel plane manipulator depend on the geometric and kinematic constraints of the manipulator, as well as its physical limitations. The workspace is typically bounded by the maximum and minimum values of the manipulator's joint angles and the physical constraints of the manipulator's arms, such as their length and range of motion.

The workspace of a 3-DOF parallel plane manipulator is often visualized as a three-dimensional region or volume. In general, the shape of the workspace is cylindrical or spherical, depending on the geometry of the manipulator. The shape of the workspace can provide insight into the manipulator's capabilities and limitations, and it can be used to optimize the design and control of the manipulator for specific tasks

The workspace of a 3-DOF parallel plane manipulator can be calculated using forward kinematics equations that relate the joint angles to the position and orientation of the end-effector. This allows for the determination of the reachable workspace of the manipulator for a given set of joint angles. The maximum workspace of the manipulator can be determined by finding the optimal joint angles that allow the end-effector to reach the furthest possible points in space.

The range of motion of the joints is an important factor in determining the workspace of the manipulator. If one of the joints has a limited range of motion, it can restrict the ability of the manipulator to reach certain points in space. Similarly, the maximum and minimum values of the joint angles can limit the size and shape of the reachable workspace [12].

In general, a 3-DOF parallel plane manipulator can achieve a large workspace with good accuracy and precision, making it suitable for a wide range of applications such as pick and place operations, assembly, and machining. However, the manipulator's workspace is limited to a two-dimensional plane, which can be a disadvantage in some applications that require three-dimensional motion {Carretero, 2000 #28}

II.7 Conclusion

kinematic and dynamic analysis are indispensable tools for the design, control, and optimization of robotic systems. By studying the motion, position, and forces involved in a robot's operation, engineers can create more efficient and precise systems that achieve desired performance while ensuring safety. These analyses play a crucial role in advancing the capabilities and applications of robotics.

III Chapter III: Simulation and Experimental Validation

III.1 Introduction

Simulation and experimental validation are two important methods used in engineering and scientific research to test and verify the performance and functionality of various systems and devices. Simulation involves creating a virtual model of a system and testing it under controlled conditions, while experimental validation involves testing the physical system in real-world scenarios.

Both simulation and experimental validation play crucial roles in the design and optimization of various systems, including mechanical, electrical, and software systems. They enable researchers and engineers to refine and optimize designs before implementation, resulting in more efficient and effective systems. By combining these methods, it is possible to better understand the behavior of a system, predict its performance, and optimize its design.

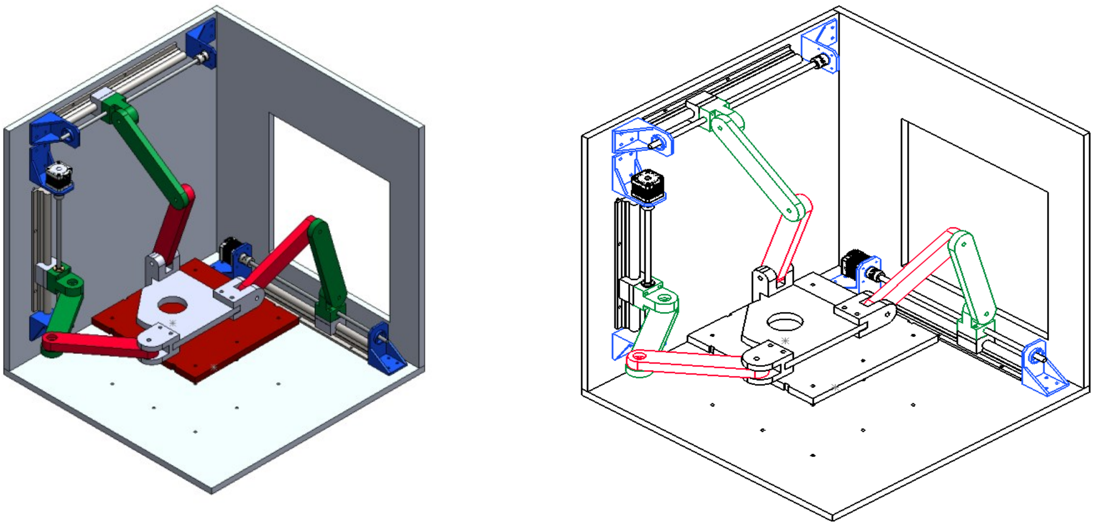


Figure III.1- Robot structure by Solidworks

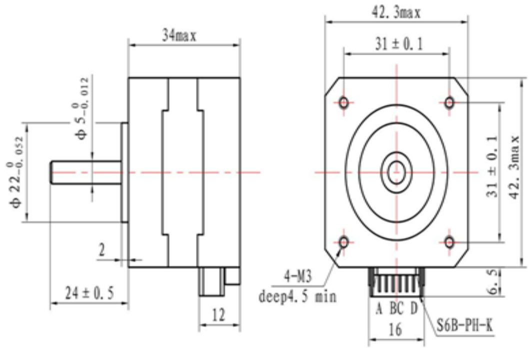
III.2 Conception et realization critique le robot hybrid

A hybrid robot is an assembly of parallel and serial structures. We show below the design and practical implementation of a hybrid robot

- Stepper motor
- Bearing
- Precision Trapezoidal Threaded Rod
- Flexible coupler
- Shoulder nut
- Structure components from Alico material

Stepper motor:

A hybrid stepper motor combines features of both permanent magnet and variable reluctance motors. It is essentially a variable reluctance motor with a permanent magnet rotor. This design offers the advantage of providing many steps for precise control of the motor.



A- Real image ***B- Solid Work photo***

Figure III.2 : Stepper motor

Bearing:

A deep groove ball bearing is a type of rolling-element bearing that uses balls to maintain separation between the inner and outer races of the bearing. Single row deep groove ball bearings are the simplest and most common type of deep groove ball bearings. They are commonly used in a wide range of industrial applications, such as electric motors, pumps, and conveyors.

One of the main advantages of single row deep groove ball bearings is their versatility. They can operate at high or very high speeds, making them suitable for a wide range of applications. Additionally, they are robust during operation and require little maintenance.

The racetrack grooves of the bearing and the interference fit between the grooves and the ball bearings allow the bearing to accommodate both radial and axial loads. This makes them suitable for applications where both types of loads are present, such as in electric motors.

Single row deep groove ball bearings are available in various sizes and designs to meet different application requirements. For example, there are deep groove ball bearings with a larger internal clearance that are suitable for high-temperature applications, and bearings with a snap ring groove that facilitate easy installation and removal

Figure III.3- Bearing

Precision Trapezoidal Threaded Rod:

This particular trapezoidal threaded rod features two unthreaded 8mm forged ends that can be connected to the shaft of a stepper motor via flexible couplers. The trapezoidal thread profile has the advantage of withstanding heat treatment better due to its wider crest. This makes it possible to create screws with a harder surface, which are more resistant to wear and have less friction. As a result, this type of threaded rod is suitable for use in applications where the nut slides constantly, such as in vices, presses, screw clamps, jacks, and machine tools. It is typically used in combination with a nut made from a material with good friction qualities, such as a copper alloy or acetyl. For fixing screws, where the nut only slides during tightening, other types of threaded rods may be more appropriate.

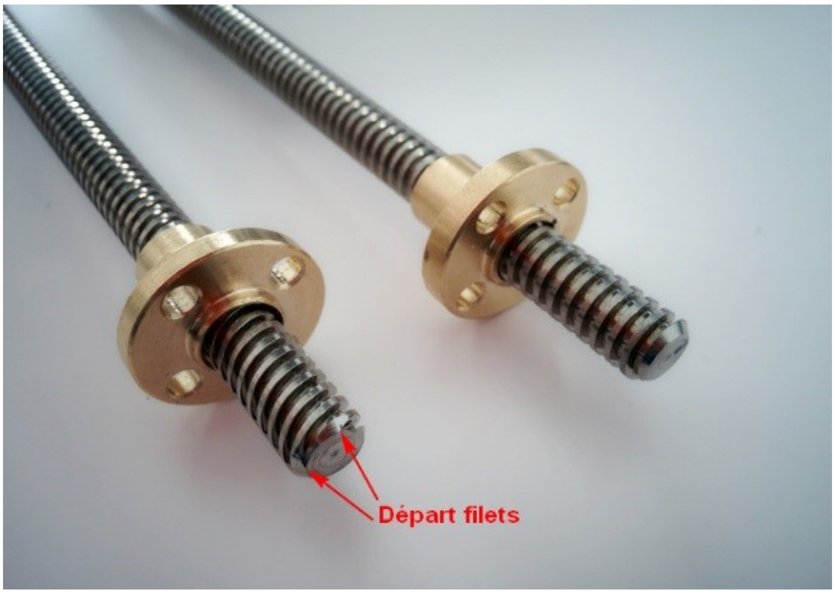


Figure III.4- Tige Precision Trapezoidal Threaded Rod

Flexible coupler

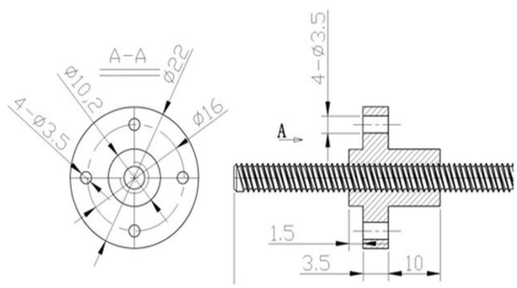
By using a flexible coupler, it becomes feasible to compensate for any misalignments that may exist between the motor shaft and the driven shaft (in this case, the screw). This compensation helps in reducing the forces exerted on the motor bearings, thereby increasing the lifespan of the motor. As a result, the screw is free to move without dragging the carriage along with it.

It is worth noting that there are alternative methods to solve this issue. For instance, if a straight screw is employed, the problem would not arise in the first place. Another option is to make the nut float instead of the screw, but this approach can be more challenging to implement since the nut must remain capable of moving while being locked in rotation.

Figure III.5- Coupleur souple

Shoulder nut:

This shouldered nut is designed to fit perfectly with all of our 8 mm trapezoidal threaded rods, ensuring complete compatibility.



A :real image ***B : Solid Work photo***

Figure III.6- Shoulder nut

Structure components from Alico material:

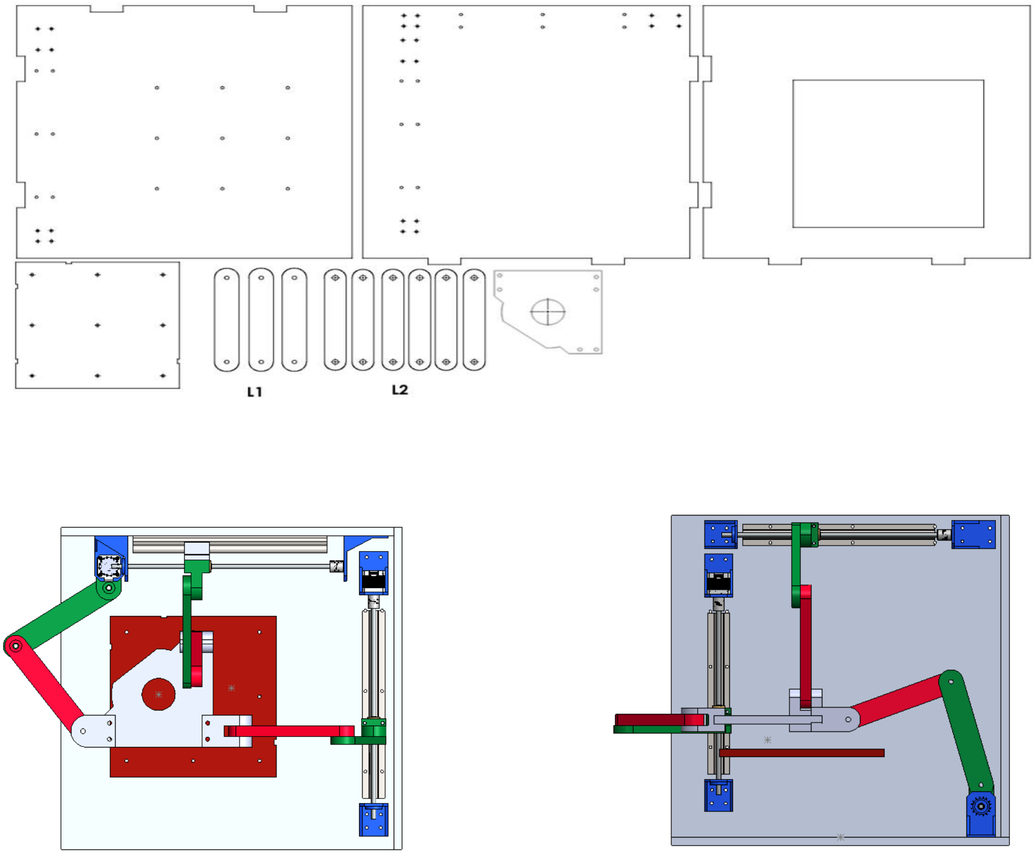


Figure III.7- Structure components

Table III-1 - Structure parameters

Design Variables	Optimum Values(cm)
d_{0p}	150.00
d_{0z}	150.00
e_x	0.00
e_y	0.00
l_{p1}	19.20
l_{m2}	19.80
l_{z1}	19.00
l_{l22}	19.50
l_3	20.00

III.3 Electrical and electronic components:

In these paragraphs we talk about the practical electrical and electronic components of the different parts of the hybrid robot:

III.3.1 The command box:

- Adriano Mega 2560 CH340
- RepRap Arduino Mega shield
- A4988 Stepper Motor Driver Module

Arduino Mega 2560 CH340

Arduino Mega 2560 is a microcontroller board based on the ATmega2560, and the CH340 is a USB-to-serial converter chip that is commonly used with Arduino boards to provide a serial communication interface between the board and a computer.

The Arduino Mega 2560 is an updated version of the Arduino Mega, with more memory and more input/output pins. It has 54 digital input/output pins, 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It is compatible with most shields designed for the Arduino Mega and can be programmed using the Arduino IDE.

The CH340 is a popular USB-to-serial converter chip used with Arduino boards and other microcontroller boards. It is a low-cost alternative to the more expensive FTDI chips and provides a reliable serial communication interface between the board and a computer.

To use the Arduino Mega 2560 with the CH340 chip, you need to install the CH340 driver on your computer. The driver can be downloaded from the manufacturer's website or from the Arduino website. Once the driver is installed, you can connect the Arduino Mega 2560 to your computer using a USB cable and program it using the Arduino IDE.

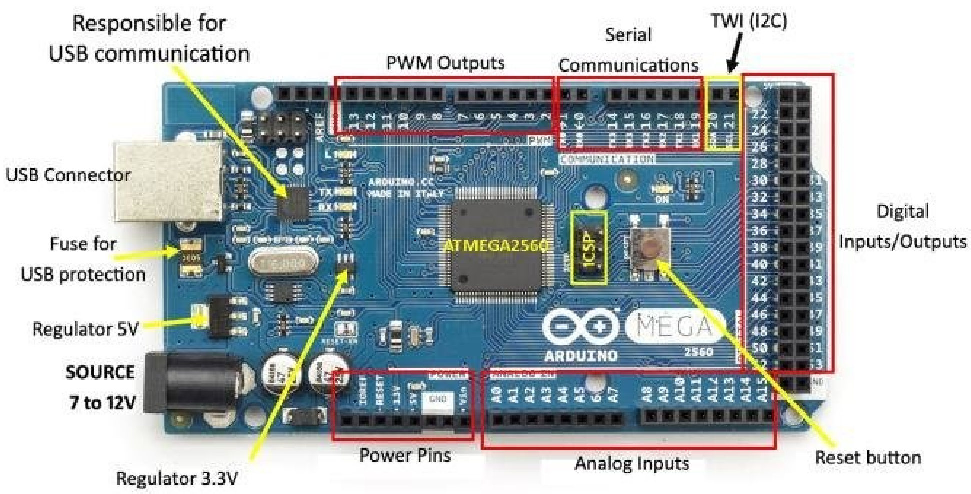


Figure III.8- Arduino Mega 2560 CH340

RepRap Arduino Mega shield

Ramps is short for RepRap Arduino mega pololu shield, it is mainly designed for the purpose ofusing pololu stepper driven board (similar to 4988 driven board). Ramps can only work when connected to its mother board Mega 2560 and 4988/DRV8825. Owning to its stability in operation and great compatibility with most 3Dprinter (all reprap-model such as pursa i2 and i3). The combination of Ramps1.4+MEGA2560+A4988/DRV8825 is becoming a mainstream of DIY 3D printer control board.

Features:

1. Standard interfaces (as that of extruder)

2. Reserved GCI like I2C and RS232

3 MOSFET 3 MOSFET are applied to the heater/ fan and thermistor circuit.

4. Adding another 5A to protect the component parts.

5. An 11A fuse is added to the hotbed

6. Support 5 stepper drive board

7. The adoption of Pin Header as pololu makes it more convenient to repair or change.

8. I2C and SPI are reserved for expanding

9. All the MOSFET can be controlled by PWM

10. Use the interface of servo motor to adjust the level of printing platform automatically.

11. Adding a SD module for SD ramps module.

12. LED can indicate the status of the heater (the open and close of MOS).

13. 2 stepper motor for Z axis in parallel.

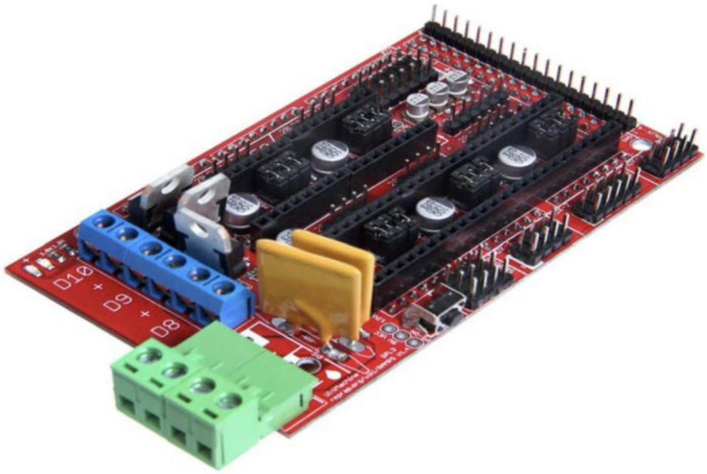


Figure III.9- reprap Arduino mega pololu shield

Interface Layout

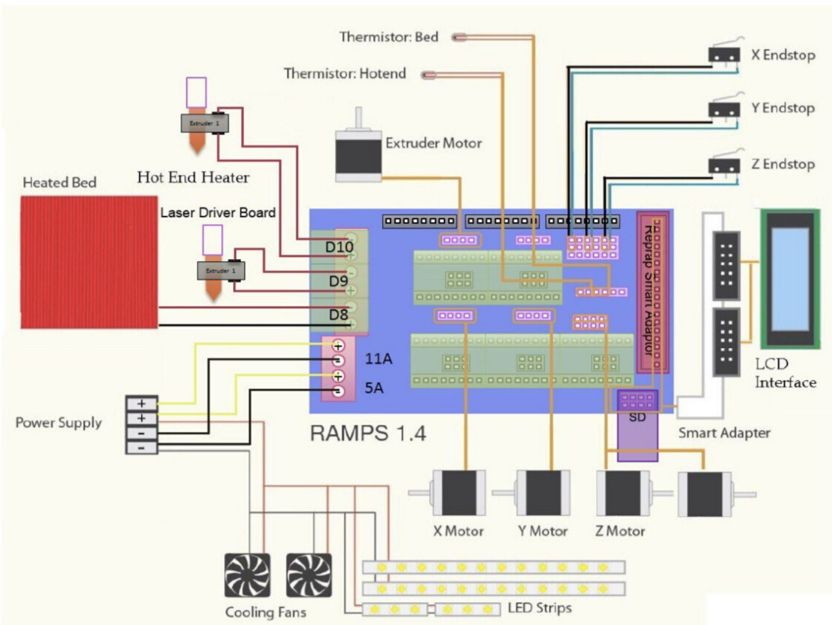


Figure III.10- Interface Layout

Interface specifications

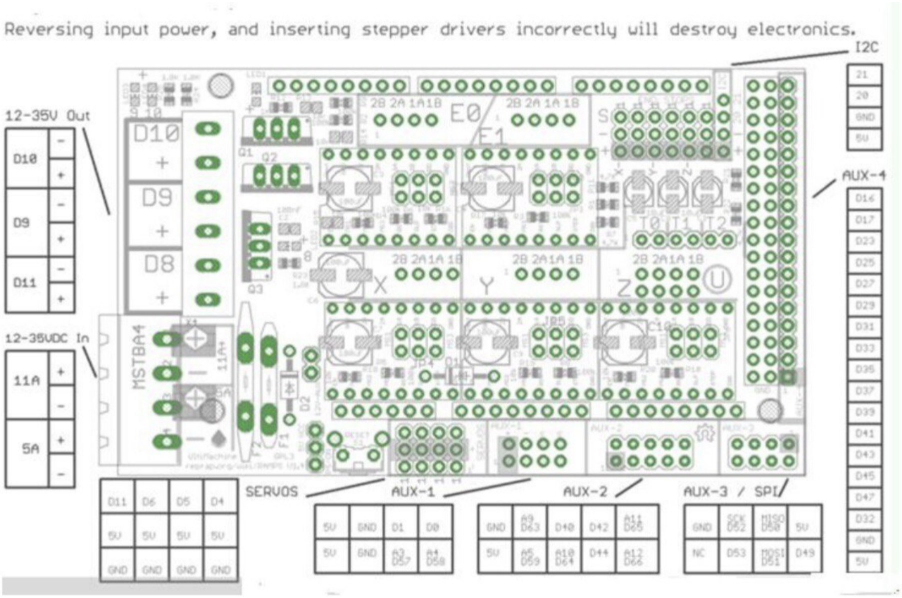


Figure III.11- Interface specifications

2 power interfaces 12v 11a/12v 5a

6 motors (one for X axis, one for Y axis, 2 for Z axis, 2 for extruder

2 interfaces for LCD&SD

6 interfaces for end stop（X/Y/Z min, X/Y/Z max）

3 interfaces for PWM (one for hotbed, one for fan and one for extruder)

3 interfaces for thermistor

A4988 Stepper Motor Driver Module

The A4988 stepper motor driver module is a small electronic circuit board that is used to control the movement of a stepper motor. It is commonly used in 3D printers, CNC machines, and other automated equipment.

The A4988 driver module uses pulse width modulation (PWM) to control the current that is sent to the stepper motor. It can handle up to 2A of current per coil and has several features that make it useful for controlling stepper motors

To use the A4988 driver module, you need to connect it to an Adriano or other microcontroller and program it is using the appropriate software. You also need to connect the stepper motor to the driver module, as well as a power source to power the motor and the driver.

The driver module can be connected to the microcontroller using a set of pins, and the stepper motor can be connected to the driver module using a set of wires.

Overall, the A4988 stepper motor driver module is an affordable and versatile solution for controlling the movement of stepper motors. Its ability to support micro stepping, thermal protection, and adjustable current make it ideal for a wide range of applications.

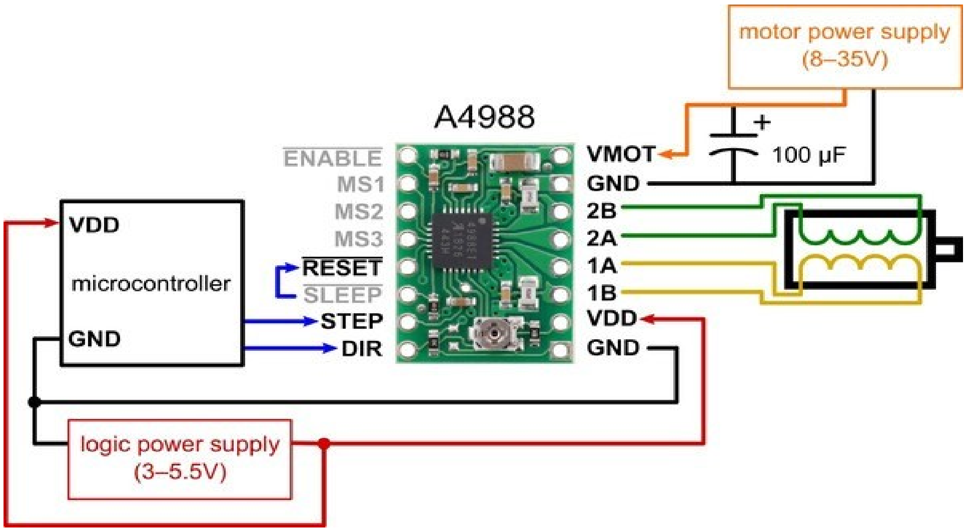


Figure III.12- 8825 Stepper Motor Driver Module

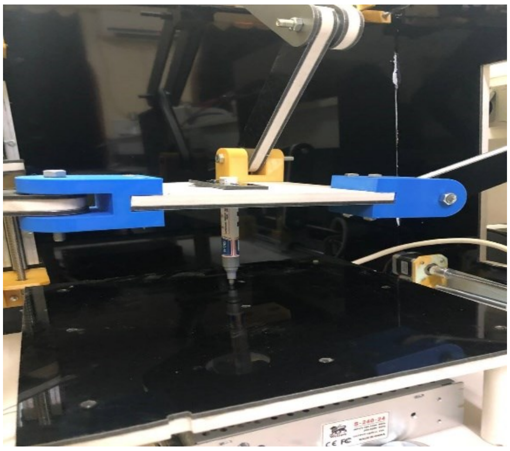
III.3.2 POWER SOURCE

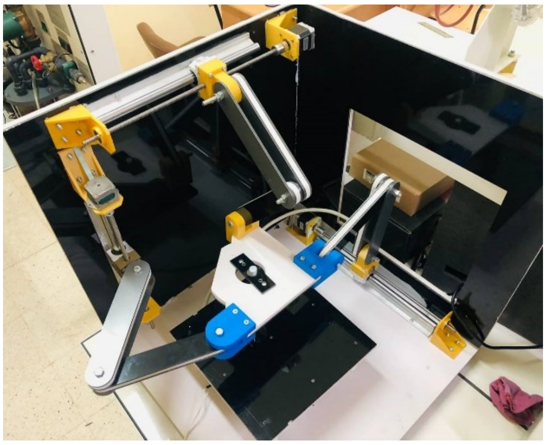
The PSU, or power supply unit, is an essential component of a computer that provides the electrical energy required for all its components to operate dynamically. It is denoted by the symbol (PSU) and not only supplies electricity but also performs an important regulatory function for the computer. The power supply converts alternating current (AC) from the standard electrical switch into a constant direct current (DC) to provide the computer with consistent voltages (3.3V, 5V, 12V, -12V) necessary for optimal performance.

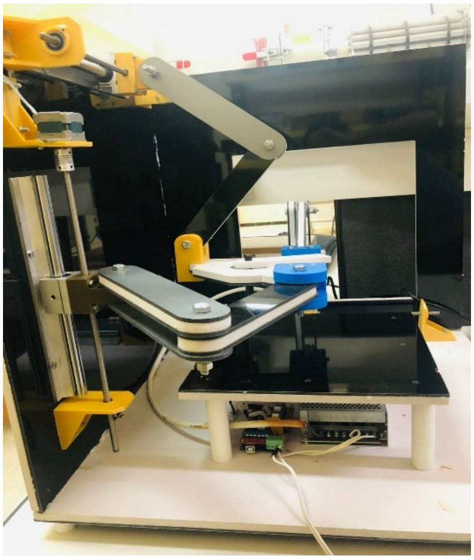
Figure III.13- POWER SOURCE

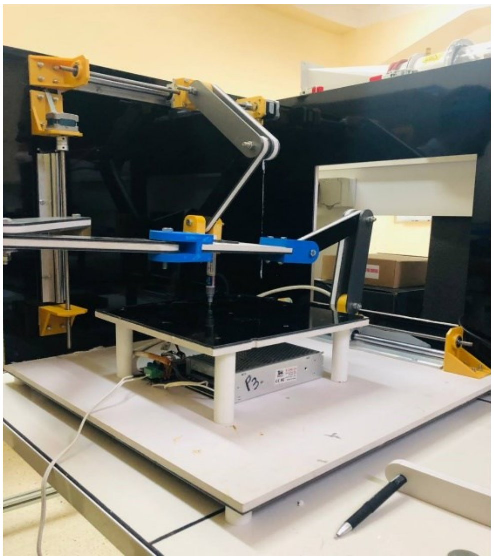
III.4 Experimentation and Results:

The collection of pictures displayed below illustrates the utilization of the robot control graphical user interface (GUI) across multiple programs.









III.5 Conclusion

simulation and experimental validation are important methods for testing and verifying the performance and functionality of various systems and devices. Simulation allows for testing under controlled conditions, while experimental validation provides real-world results.

By using simulation and experimental validation together, it is possible to refine and optimize designs before they are implemented in the real world. This can lead to more efficient and effective systems, as well as reduced costs and risks associated with trial and error.

. simulation and experimental validation play critical roles in engineering and scientific research, enabling researchers and engineers to develop and test new ideas and technologies in a safe and controlled environment

General conclusion

the dynamic analysis and control of a Cartesian parallel manipulator is a complex and challenging task that requires a combination of theoretical modelling, simulation, and experimental validation. Through careful analysis and testing, it is possible to better understand the behaviour of the manipulator and develop effective control strategies to achieve desired performance.

The development of accurate mathematical models and simulations is essential for predicting the behaviour of the manipulator under different conditions and optimizing its performance. Additionally, experimental validation is crucial for verifying the accuracy of the models and simulations, as well as for testing the manipulator in real-world scenarios.

Effective control strategies are also critical for achieving optimal performance of the manipulator. These control strategies can be developed using various techniques, including feedback control, feed forward control, and adaptive control. By selecting the appropriate control strategy and implementing it effectively, the manipulator can achieve high accuracy, speed, and reliability.

The goal of the dynamic analysis and control of a Cartesian parallel manipulator is to provide a detailed understanding of the theoretical modelling, simulation, experimental validation, and control strategies involved in optimizing the performance of a Cartesian parallel manipulator. The note aims to educate readers about the complexity of this multidisciplinary field and the importance of accurate mathematical models, simulations, and experimental validation to predict the manipulator's behaviour under different conditions.

References

[1] A. Vivas and P. Poignet, "Predictive functional control of a parallel robot," *Control Engineering Practice,* vol. 13, no. 7, pp. 863-874, 2005.

[2] F. Pfeiffer and C. Glocker, *Multibody dynamics with unilateral contacts*. John Wiley & Sons, 1996.

[3] L-.W. Tsai, *Robot analysis: the mechanics of serial and parallel manipulators*. John Wiley & Sons, 1999.

[4] K. Sugimoto, "Kinematic and dynamic analysis of parallel manipulators by means of motor algebra," 1987.

[5] N. Kumar, V. Panwar, N. Sukavanam, S. P .Sharma, and J.-H. Borm, "Neural network based hybrid force/position control for robot manipulators," *International Journal of Precision Engineering and Manufacturing,* vol. 12, pp. 419-426, 2011.

[6] F. Garelli, L. Gracia, A. Sala, and P. Albertos, "Sliding mode speed auto-regulation technique for robotic tracking," *Robotics and Autonomous Systems,* vol. 59, no. 7-8, pp. 519-529, 2011.

[7] W. Shang and S. Cong, "Motion control of parallel manipulators using acceleration feedback," *IEEE Transactions on Control Systems Technology,* vol. 22, no. 1, pp. 314-321, 2013.

[8] B. Borovic, A. Liu, D. Popa, H. Cai, and F. Lewis, "Open-loop versus closed-loop control of MEMS devices: choices and issues," *Journal of Micromechanics and Microengineering,* vol. 15, no. 10, p.2005 ,1917 .

[9] P.-L. Yen and C.-C. Lai, "Dynamic modeling and control of a 3-DOF Cartesian parallel manipulator," *Mechatronics,* vol. 19, no. 3, pp. 390-398, 2009.

[10] A. Müller, "Dynamics modeling of topologically simple parallel kinematic manipulators: A geometric approach," *Applied Mechanics Reviews,* vol. 72, no. 3, p. 030801, 2020.

[11] A. A. Selvakumar and M. A. Kumar, "Experimental investigation on position analysis of 3–DOF parallel manipulators," *Procedia Engineering,* vol. 97, pp. 1126-1134, 20.14

[12] C. Wang, Y. Fang, S. Guo, and C. Zhou, "Design and kinematic analysis of redundantly actuated parallel mechanisms for ankle rehabilitation," *Robotica,* vol. 33, no. 2, pp. 366-384, 2015.

Abstract

Report title: Dynamic modelling of parallel robot two DOF.

Master: Electromechanical.

Keywords: parallel robot, dynamic, modelling, kinematic, workspace, geometric model, virtual work.

Authors: Hadjaidji Abdelmounaim, Fezzai Ramzi, Hamlaoui Chakib Abdenour**.**

Abstract:

Dynamics modelling parallel robot 2dof is the application of rigid-body dynamics to this robots. Using two modelling types: inverse (calculate the torques and/or forces), and forward (calculate the state variables accelerations). With the help of some methods one of them virtual work method. That this project used on, where we started from the geometric, than the kinematic down to the dynamic model that give the relationship between the two motors and acceleration relation of end-effector. In addition found the force that calculated by virtual work, finally the torque.

Titre du mémoire : Modélisation dynamique du robot parallèle 2 degré de liberté.

Mots clés : robot parallèle, dynamique, modélisation, cinématique, espace de travail, modèle géométrique, travail virtuel.

Résumé :

Modélisation du dynamique robot parallèle 2 degrés de liberté. est l'application de la dynamique des corps rigides à ce robot en utilisant deux types de modélisation : inverse (calcule les couples et/ou forces) et avant (calcule les accélérations des variables d'état) à l'aide de quelques méthodes l'une des la méthode de travail virtuelle utilisée par ce projet, où nous sommes partis de la géométrie, de la cinématique jusqu'au modèle dynamique qui donne la relation entre les deux moteurs et la relation d'accélération de nacelle. En plus on retrouve la force que l'on calcule par travail virtuel, enfin le couple**.**

عنوان المذكرة: النمذجة الديناميكية للروبوت المتوازي درجتين من الحرية.

الكلمات المفتاحية: روبوت متوازي، ديناميكي، نمذجة، حركي، مساحة عمل، نموذج هندسي، عمل افتراضي

الملخص:

النمذجة الديناميكية للروبوت المتوازي درجتين من الحرية هو تطبيق ديناميكيات الجسم الصلب على هذه الروبوتات باستخدام نوعين من النمذجة: معكوس (يحسب عزم الدوران و / أو القوى) والأمامي (يحسب تسارع متغيرات الحالة) بمساعدة بعض الطرق، إحداها طريقة العمل الافتراضية التي استخدمها هذا البحث، حيث بدأنا من الشكل الهندسي، ثم الحركي وصولاً إلى النموذج الديناميكي الذي يعطي العلاقة بين المحركين وعلاقة تسارع المؤثر النهائي. بالإضافة إلى ذلك، ايجاد القوة التي تم حسابها من خلال العمل الافتراضي، وأخيراً عزم الدوران.