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 Simulation and control of Linear Delta Robot

In front the composed jury:

Dr. M LOUAMER	President
Dr. MILOUDI Khaled	Examiner
Dr. MANSOURI Khaled	Supervisor

presented by:

- REZAG SALEM Oussama
- SADOUN Yacine
- MANSOURI Arafat

2021-2022

Acknowledgment

Thank God so much for satisfying and after satisfaction because we agreed to reach this level of science

He then thanked Dr. Mansouri Khaled for standing with us, guiding and supporting us by providing us with information about our research. And we took advantage of it in other information, and all the professors who studied us at all levels.

Dedication

We are very pleased to guide this effort:

To our dear parents, who always gave us hope to live and who never stopped praying for us. They have always encouraged us to study and persevere.

to all our brothers and sisters,

And all our cousins and friends.

All our friends who shared the best moments of our lives.

Everyone we love and respect.

We also thank all the professors.

Table of contents

Lists figures

Introduction

Young people have acquired the word “system” to a very high extent during recent years, especially in systems sciences and engineering fields as it is used to fix problems, to implement discoveries, and to develop solutions in all disciplines. They use mathematical and scientific methods to achieve their goals as the designer.

Because robots are extremely important in this pandemic, especially as were search curing (or at least the patients first) and assisted in newly developing technologies and to learn more about the pathogenic agent. This is parallel robotics, which includes *robot*, which means various different sizes, has emerged and still continues to emerge, including social and technological robots. Next in this essay, we will discuss parallel robots. Perhap the strangest robots, for positive or negative, is the Biological robot. That is why so-called parallel robots or parallel kinematic model can now become the key to the current complex industrial processes that are happening because of access to better flexibility and options. As such, subsequent modeling of this kind and its dynamic control will contribute to finding more and more versatile solutions.

Dimensional attached-groove parallel robot can generate opposition movements that can be controlled through a mechanized "mechano-lever" chain to make ascratching movement. Here, we study the properties of this mechanism to better understand parallel robotic machines. This work can be an important step for the development of pegboard robots.

This work contains three parts that deal with the modeling of parallel robots: first part focuses on the state of the art of parallel robots, while another part deals with the reason of this study. The third part focuses on the use of parallel robots in several areas.

the second chapter specifies geometric description, kinematic, and the work space we used for the three-dimensional prototype of the industrial parallel robot .

Finally, the robot system used MATLAB, used simulation by using the information and relations in the previous Chapter and discussed results.

Chapter I: State of the art of dynamic parallel robot modelling

In normal human life and at any time he is used to making several movements and without thinking because it is part of his routine life such as lifting or grabbing objects and is helped by man's biological composition, but when we embody them on robots we find it difficult to do so.

This is why there is little difficulty modeling robots especially when we want to make them on the parallel manipulator or hybrid in connecting and moving chains but it is not impossible to design them in addition to some studies. Hence we focused on the parallel robot study.

In this study we will get closer to parallel robots and learn about them more, and then introduce a general concept to them to take a set of ideas on their role, and then deepen further the study of dynamic parallel modeling and the main methods we use in this model

I.1 Introduction

We can imagine the parallel robot theoretically, when the hand and fingertips hold a red apple (figure I 1). In this case palm is the foundation platform of the parallel robot and the arms operate serial robots associated with the base platform, thanks to which this formula deals with the apple collaboratively. Apple is a changing platform, and through this we can load it directly within the parallel robot[1, 2].

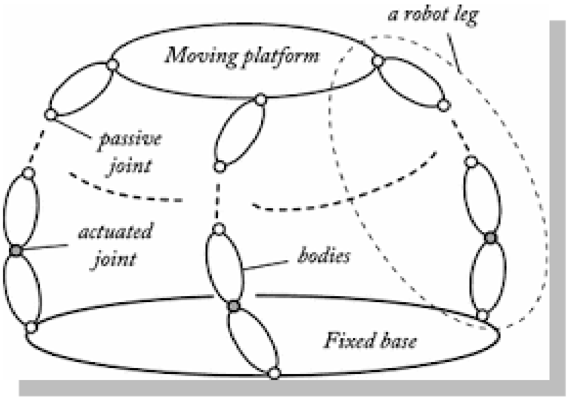
Figure I.1 – A hand clutching a red apple with its fingertips as a metaphor for a similar robot notion.

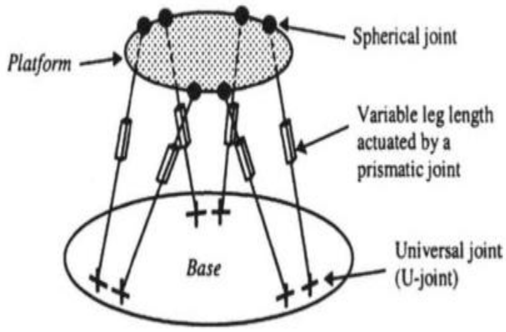
I.2 Definition

Parallel robots, also known as parallel manipulators or parallel kinematic machines (PKM), are a closed-loop mechanism that controls the motion of their end-effectors by connecting them to the base via at least two separate multiple linkages (kinematic chains). This loop is closed by an n-degree-of-freedom end-effector linked to the base by n separate chains with no more than tow links, each operated by a single prismatic or rotary actuator[3, 4].

The three components of the parallel robot are:

- Fixed component. It is the base of the robot.
- The part on which the end responder is generally placed
- The robot's legs, which connect the base to the platform, are also known as kinetic chains.





a) The gray joints denote the actuated joints b) Classic PKM platform (6DOF)

Figure 0.2: Simple drawing of general parallel robots

Thanks to the dynamic study of Stewart platforms, which have high torque capacity, high structural toughness, and low-motion mass, the starting point is to study parallel manipulators. Under basic criteria, the study was therefore focused on problems of oscillation or reverse dynamics. After a time we presented numerous research, for a more detailed study to solve the dynamic modeling of parallel manipulations using various mechanical formalities.

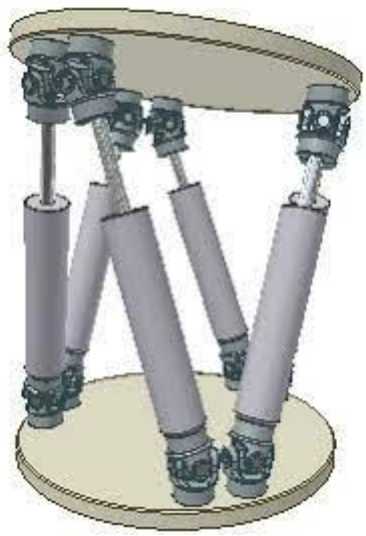


Figure 0.3: Stewart platform

I.3 Overview

Maximum of the robots utilized these days are sequential palms wherein various connections are connected through impelled joints, accordingly shaping an unmarried open kinematic chain from the base linked to the give end-effector. Such a course of action gives a huge workspace and rather natural kinematics. In any case, in a couple of uses, severe requests on payload, exactness, or dynamic execution forestall the utilization of sequential robots.

Because many legs of the system share the controlled weight, parallel robots are extremely appealing for a variety of applications. As a result, each kinematic chain carries just a portion of the overall weight, allowing for the construction of robots that are fundamentally more rigid. As a result, such topologies allow for a reduction in the bulk of the moveable links (all actuators are primarily fixed on the base, and many legs are strained by tension/compression efforts), allowing for the employment of less powerful actuators. Structures with such features claimed to have a high payload, great dynamic capabilities, and excellent precision. Parallel robots are being employed in a variety of applications[5].

I.4 General Comments

System: : The system consists of several different elements that can work together to achieve a specific objective.

The resulting variables are the main component variables.

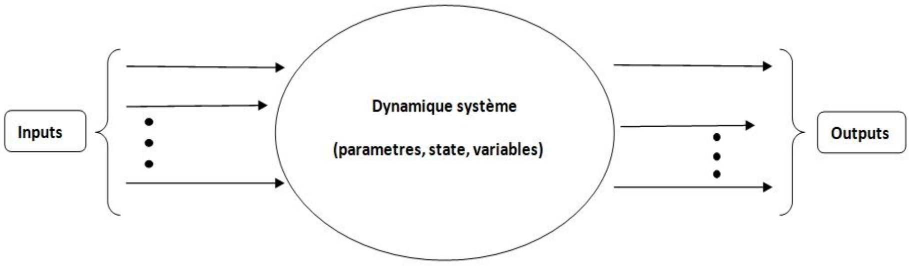


Figure 0.4- A dynamic system visual presentation.

A full collection of variables, termed State variables, is required for that system. The status variables are the smallest set of system variables needed to fully represent the state of the system at any given moment and are very important in the modelling and analysis of dynamic systems.

All moving systems on earth and in space are composed of various material bodies related by certain force laws of almost every type that indicate that forces are unified in single or multi-value rules. The material part of all these structures, however, are bodies, solid and elastic systems that are related to the needed performance in the optimum manner. In taking account of these systems, we thus have to make a preliminary decision on how to shape these forces, model them and eventually model the masses[6].

Dynamic system: The situation has changed over time. We have two main categories of dynamic systems in applications:

- ✓ Time changes separately
- ✓ Time is continuous.

Rigid-body dynamics: : Solid body robots provide joint links to this body. Given that solid bonds are fixed, robots usually have only a few numbers.

External forces are affected by the solid dynamics of the body and the movement of associated systems.

In solid body dynamics there is kinetic energy and angular momentum are the two most important physical quantities. They lead to the concept of tensor inertia.

Modelling is the process of creating a model that represents the structure and operation of an interest system. A model is comparable, but simpler than the system. A key goal of a model is to allow the analyst to forecast the effect of system modifications. On the one hand, a model should be close to the actual system and incorporate the majority of its outstanding characteristics. Moreover, it should not be so complicated that it cannot be understood and experimented. The potential and greater equalization of realism and simplicity is a good model. Practices in simulation propose that iteratively increase the complexity of a model. Modeling validity is a significant problem. The validation of the model includes the simulations and comparisons between the model output and system output in known input circumstances.

A model for a simulation study is usually a mathematical model created with the aid of software for simulation. The classifications of the mathematical model include deterministic (fixed values for the input and output variables) or stochastic (probabilistic at least for one input or output); static (time is not taken into consideration) or dynamic (time-varying interactions among variables are taken into account). Simulation models often are dynamic and stochastic[6].

Robot control and simulation necessitate the use of a variety of mathematical models. Various modelling levels – geometric, cinematic, and dynamic – are needed based on the targets, job limitations and therefore the performance that is required.

It is not an easy process to obtain these models. The difficulty changes according on the complicated kinematics and degree of freedom of the mechanical structure.

Using these models on top of objects and simulation necessitates efficient and simple-to-use algorithms to estimate the geometric parameter values and, as a result, the robot's dynamic parameters. Furthermore, on-line application of an effect rule on a robot controller necessitates efficient models with fewer actions. These conditions are met by the approaches provided in this book. As a result, the dynamic element of the parallel manipulator will be the focus of this chapter's research.

I.5 Dynamics modeling

The dynamic model gives us the activity of the object over time. It is employed when this activity is a series of cases occurring in a given order.

The active forces that work on the robot with the accelerations they generate, or vice versa, are tied to the dynamic model. Active forces are likely to be moments of turnover and transitional forces.

Applying solid body dynamics to robots gives us robot dynamics, the body's steel system usually represents us robot mechanism.

I.5.1 Inverse Dynamic Model (IDM)

To control the movements and forces of robots the reverse dynamic model must be calculated the torque and/or forces the robot triggers must provide in order for the ultimate influencer to move in a given direction. It was applied to it..

I.5.2 Forward Dynamic Model (FDM)

The front dynamic model calculates the acceleration of state variables so as to obtain the specific forces, locations and speeds. It is also called direct dynamic model. It is mainly used in simulations.

I.5.3 Implicit Dynamic Model (ImplDM)

For robot movement equations (EOM). We use a dynamic model implicitly.

I.6 Why we interested in the Parallel Robots Dynamics?

Its dynamic model heavily influences robot design and control. The inverse dynamic model can be used to pick actuators for robot design, while the direct dynamic model is used to run simulations in order to test the robot's performance and compare the relative benefits of different control methods. The inverse dynamic model is used in robot control to calculate the actuator torques required to achieve a desired motion. It's also utilized to figure out what dynamic parameters are required for to achieve control and simulation applications.

The majority of works that can be utilized to compute the dynamic models of (flexible and/or rigid) parallel robots are broad works that define general equations for limited or closed-loop systems.[5]

However, the information deficiency is generally the following:

- They frequently fail to recognize, and do not give easy methods for computing, that Jacobian matrices needed for setting dynamic constraints in the dynamic model are not that simple.
- Most of these projects offer no efficient method for calculating dynamic models in terms of the reduction of the '+,' '−', 'after' and '/' operators needed to get the individual model expression. However, this optimization is important to model

production to anticipate and manage robot behavior and speed up the ideal robot design process till that have best possible control.

- The facts that (i) dynamic modeling may degenerate in the presence of specific types of singularities and (ii) this degeneration may be avoided by optimum planning of trajectories is entirely missing (optimal with respect to a criterion based on the dynamic model).
- They do not offer experimental evidence showing that parallel manipulators models can be extremely accurate even if they are complex.

I.7 Dynamic modelling of parallel robots

Due of their several closed loops, parallel robots are complicated multi-body systems that are challenging to describe. For design requirements and sophisticated control of parallel robots, dynamic modeling is required.

The dynamic and kinematic models of the legs are created utilizing serial robot-specific approaches and, finally, basic loops. As a result, the computational complexity of the suggested models can be lowered by employing approaches established for serial robots many years ago. The platform's dynamics are calculated using Newton-Euler equations, which calculate total forces and moments on a solid body. We present a new technique for calculating the inverse Jacobian matrix of parallel robots utilizing the Jacobian matrices of the legs since the robot's Jacobian matrix is required.[7]

Figure 0.5-Parallel Robot ABB IRB 360 Flex-picker

I.7.1 Inverse dynamic modelling of parallel robots

A parallel robot is a multi-body complicated system with several closed loops. It is made up of parallel legs that link a moving platform to a stationary foundation. Nonredundant robots are those in which the number of active joints is equal to the platform's degrees of freedom. The number of legs is represented by m, while the number of degrees of freedom (DOF) of the platform is represented by n. The frame $\sum_p$ is fastened to the platform, while the frame $\sum_p$ is fastened to the base. The forces and torques of motorized joints are calculated using the inverse dynamic model as a function of the mobile platform's intended trajectory.

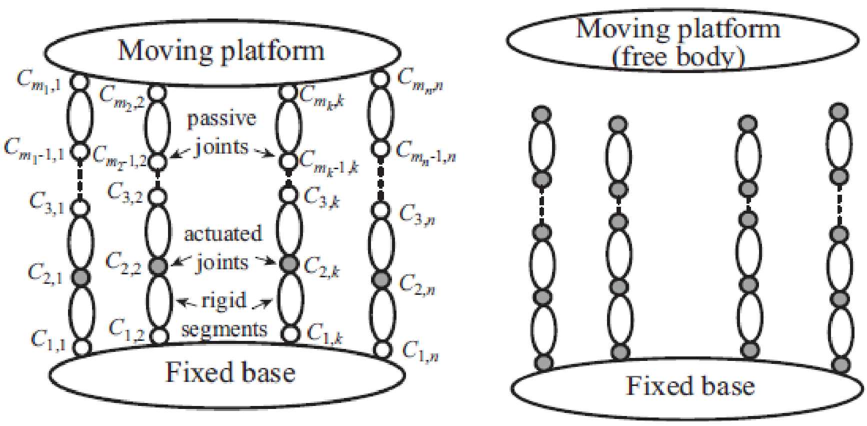
We propose using structural features of parallel robots to derive dynamic models by partitioning the system into two subsystems: the platform and the legs. The platform's dynamics are computed as a function of its Cartesian variables (spatial Cartesian location, velocity, and acceleration), whereas the legs' dynamics are calculated as a function of their joint variables $(q_i, \dot{q}_i, \ddot{q}_i)$. After projecting these dynamics on the active joint axis, the active joint torques are calculated by adding them together.

I.7.2 Direct dynamic model of parallel robots

Using the Dickarty location and platform speed we can represent the parallel robot. As a function of state variables, torque inputs and automated combined forces, the robot's direct dynamic model provides the Decarte acceleration platform.

We can illustrate the vision of parallel robots in dynamics as follows:

- Tree structure + platform.
- The rings are closed using restriction equations (Jacobiane matrices).



Kinematic chain b) virtual tree structure

Figure 0.6- Structures of parallel robot's dynamics

I.8 Methods of dynamic modelling

I.8.1 Dynamic Modelling Methods of serial Robots:

We need to know first about serial robots and their modeling methods because it is the beginning that led us to find dynamic modeling methods PKM before we talk about parallel robot modeling methods. Therefore, serial manipulator is an open chain motor structure consisting of many links associated in a series with different types of joints, especially rebellious and published joints. Such methods include Euler-Lagrange, Virtual Work and others. These methods are very important for dynamic modeling, whether sequenced or parallel robots.

I.8.2 Dynamic Modelling Methods of Parallel Robots:

For parallel robots, in fact, there is only one method specifically designed for it, which is **Khalil's method**, which he deduced from serial robot's methods. These are the most famous methodsThe Euler-Lagrange method; the Newton-Euler recursive method, the D'Alembert method, and Kane's method are all examples of these methods. We'll go through these approaches in more detail later[2].

I.8.2.1 Euler-Lagrange Method

The Euler Lagrange method uses the concept of energy conservation in a mechanism in order to obtain dynamic equations. Analytical and closed form equations are used. By distinguishing the Lagrange function, the Euler Lagrange equations are derived:

$$L(q,\dot{q}) = T(q,\dot{q}) - V(q) \tag{0-1}$$

Which yields:

$$\frac{d}{dt}\left(\frac{\partial \mathrm{L}}{\partial \mathrm{L}}\right) - \frac{\partial \mathrm{L}}{\partial \mathrm{q}} = \frac{d}{dt}\left(\frac{\partial \mathrm{T}}{\partial \dot{q}}\right) - \left(\frac{\partial \mathrm{T}}{\partial \mathrm{q}} - \frac{\partial \mathrm{V}}{\partial \mathrm{q}}\right) = \Gamma \tag{0-2}$$

T and V are the motor and potential energies This method is important for the study of dynamic properties as well as the analysis of control methods. However, they are less explicit in engineering terms because of their reliance on energy, and there are some studies that say they are computationally ineffective.

I.8.2.2 Newton-Euler Method / Luh-Walker-Paul's Algorithm

The dynamic equations are derived using the Newton-Euler method, which takes use of the balance of forces and torques in a Mechanism. Numeric and recursive-form equations are used. Newton's equations of motion are used in this method:

$$f = ma \quad , \quad \tau = \mathcal{L}^T \dot{\omega} + \omega \times (\mathcal{L}^T \omega) \tag{0-3}$$

where f, m, a, and L represent the body's linear momentum, angular momentum, mass, linear acceleration, angular velocity, and inertia, respectively. The dynamics are calculated using two loops in the method:

–Forward Loop: moves from base to end to evaluate the velocities and the accelerations.

– Backward Loop: moves from end to base to compute the forces and torques.

It is systematic and efficient for real time implementation of the control schemes.

I.8.2.3 D'Alembert Method / Principle of Virtual Work

To obtain the dynamic equations, this method uses the idea of virtual work conservation in a mechanism. The total of variations in work arising from virtual forces operating via a real displacement or real forces acting via a virtual displacement equals zero, according to this formula. The displacement is tiny and conforms to the system's limitations:

$$0 = \sum_i (f_i - m_i a_i)^T \, \delta x_i = 0 \tag{0-4}$$

Where f_i represents an applied force, f_i represents a particle's mass, a_i represents a particle's acceleration, δx_i represents an infinitesimal displacement compatible with the restrictions, and I represents a specific particle in the system [2].

I.8.2.4 Kane's Method

While deriving the equations of motion of a system, Kane's method has the Lagrange form of D'Alembert principle and gives numerous advantages. It does not necessitate the employment of energy functions or, as a result, the difficulty of their differentiation. It employs generalized forces, with non-contributing forces being removed directly by projection. It allows you to use variables other than generalized coordinates, which can have a big impact on the equations of motion that arise. For multi-body systems, this method is also more useful.

Parallel robot'smethods: All the methods for serial robots can be adapted to parallel robots. In addition to those methods, there exist one method designed for parallel robots in particula, it's **Khalil's Method**.

I.8.2.5 Khalil's Method:

Khalil proposed[8] to obtain the equations of motion of a parallel robot by extending the systematic approach of the modelling of a serial robot. This approach proceeds as follows:

- Each of a parallel robot's kinematic legs is treated as a separate serial robot, and the inverse dynamic model of this kinematic leg is constructed using one of the approaches described for serial robot modeling.
- On the moving platform, the equilibrium of all efforts (torques and forces) is determined. These efforts originate from each of the kinematic legs, the moving platform's acceleration, and external forces (weight, contact, etc.).
- On the moving platform, the whole effort is transferred onto the active joints.

Because each leg contributes to the overall effort on the moving platform, this technique is extremely straightforward for dealing with kinematic limitations:

$$\Gamma = FDKM_{robot}^{T}\left(W_{platform} + \sum_{i=1}^{N_{legs}}\left(J_i^T IDKM_{leg(i)}^{T} IDM_{leg(i)}\right)\right) \qquad (0\text{-}5)$$

Where $FDKM_{robot}^{T}$ is the parallel robot's forward differential kinematic model, $W_{platform}$ is the wrench vector for the moving platform's dynamics, J_i^T is the Jacobian matrix relating the terminal point of the i^{th} leg's velocity to the end-effector velocity, $IDKM_{leg\,(i)}^{T}$ is the inverse differential kinematic model of the i^{th} leg, and $IDM_{leg\,(i)}$ is the inverse dynamic model of the i^{th} leg. Any of the current serial robot algorithms may be used to calculate the $IDKM_{leg\,(i)}^{T}$.

I.9 Conclusion

When studying parallel robots, we should have a background on the closed loop mechanism that controls the movement of final effects, because this helps us to reach the last result.

The modeling of robotics parallel to dynamics is a case of current knowledge of issues and is also an important work to be discussed and applied to a large framework because most of the solid body presentation bonds have a constant form, as the linkages are connected through joints in order to be able to form and model robots dynamically.

We can conclude the state of robots (positioning, inertia, speed...) According to dynamic model type. Dynamic modeling is crucial for design and control and represents robot behaviors as a solid body and final effects.

We can apply Khalil's method, which is a way of modeling serial robots and by which we find an equivalent to the movement of parallel manipulations.

Chapter II: Kinematic and dynamic models of Delta robot

The Inverse Geometric Model (MGI) links the robot's q R m articular coordinates to the platform's xθ R n operational coordinates. In other words, the MGI can calculate q = [q1,q2,...,qm] T knowing the xθ installation = [p T o T] T of the platform as q = f (xθ) Reymond Clavel is the creator of Delta Robot in 1985 [8],[9].

We have in shape II-1, the original Delta Robot which has three rotational input axes. The three arms relate to three lower arms consisting of two parallel rods, each forming parallel ribs. The parallel ribs determined the movement of the final responder with three degrees of transformative freedom.[8].

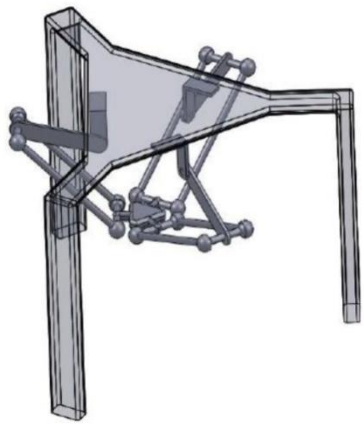


Figure II-1: Clavel's Delta Robot

Recently with the advent of 3D printing, the Delta Robot has been revisited. Its design excels at this task because of its three translational degrees of freedom, and its speed [10]. Hobbyists and engineers have developed many different designs for the Delta Robot for 3D printing, most importantly using linear inputs driven by timing belts instead of rotational inputs. Figure II-2 demonstrates how the end effector of the linear input Delta Robot moves with three translational degrees of freedom as a result of three linear inputs driving carriages with two spherical or universal joints connecting two parallel rods to the end effector. The linear input Delta Robot has become a common choice when purchasing a 3D printer.

Figure II-2: Linear Input Delta Robot

II-1 Parallel vs Serial Robots

The Delta Robot belongs to a group called parallel robots[11]. Parallel robots use multiple links attached to the end effector in order to move. In most cases, the motors used to drive parallel robots are mounted to the stationary frame of the robot, and do not move [11]. Parallel robots excel at pick and place operations where speed is important[11],[10],[12]. Just as the first Delta Robot was used to move chocolates from a conveyor belt to their packaging[8], today's parallel robots are used for similar tasks (pick and place operations, and 3D printing).

Robots that are not parallel are known as serial robots. Serial robots have only one link attached to the end effector, and often the motors are attached to moving parts of the robot [11]. The traditional robotic arm is an example of a serial robot as shown in Figure II- 3.

Other serial robots include the traditional 3D printer which consists of a moving gantry with two degrees of freedom and a build plate with one degree of freedom as shown in Figure II-4.

Figure II- 3: Fanuc Robotic Arm [9]

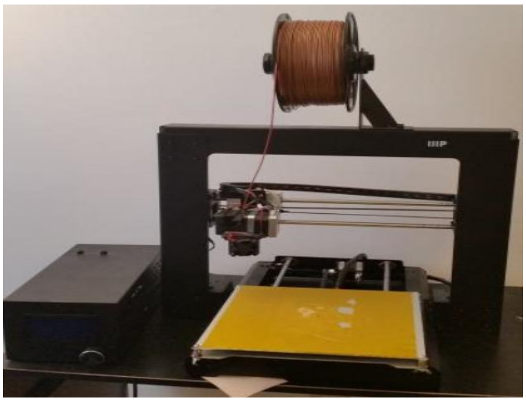


Figure II-4: Traditional Cartesian 3D Printer

Serial robots have both strengths and weaknesses. They have a large and simple workspace [11], they also have high dexterity often in the form of six degrees of freedom[11].

This makes serial robots effective in manufacturing environments where a task needs carried out in a relatively large work cell (cage where the robot is enclosed for safety).

However serial robots also have numerous disadvantages. Each motor must overcome the weight and inertia of the motors downstream of it, this results in slow speeds and a low payload/weight ratio [11]. Along with this the error associated with each link is compounding, resulting in large error in the location of the end effector [11].

Parallel robots also have many advantages and disadvantages. Their biggest disadvantage is a small and complex work space [11],[10]. Because of this, they can only be used when the task can be completed in a very small area. However the advantages make parallel robots the optimal choice in many situations. Because their motors are mounted to the frame of the robot, the weight and inertia of the moving parts is low. This results in higher top speeds and accelerations [11],[8], [10]. Along with their speed capabilities, parallel robots have a high payload/weight ratio, this is a result of multiple links sharing the load of the end effector [11], [10]. Lastly, parallel robots are accurate[11].

This is because unlike serial robots, the error associated with the end effector does not compound, but averages. Along with this, several links support the end effector resulting in less strain. A more complete comparison of the advantages and disadvantages of serial and parallel robots is provided in Figure II- 5.

Feature	Serial robot	Parallel robot
Workspace	Large	Small and complex
Solving forward kinematics	Easy	Very difficult
Solving inverse kinematics	Difficult	Easy
Position error	Accumulates	Averages
Force error	Averages	Accumulates
Maximum force	Limited by minimum actuator force	Summation of all actuator forces
Stiffness	Low	High
Dynamics characteristics	Poor, especially with increasing the size	Very high
Modelling and solving dynamics	Relatively simple	Very complex
Inertia	Large	Small
Areas of application	A great number in different areas, especially in industry	Currently limited, especially in industry
Payload/weight ratio	Low	High
Speed and acceleration	Low	High
Accuracy	Low	High
Uniformity of components	Low	High
Calibration	Relatively simple	Complicated
Workspace/robot size ratio	High	Low

Figure II-5: Serial and Parallel Robot Comparison [3]

Current Delta Robots

There are several different Delta Robot models currently in circulation. Figure II-6 shows information about many popular models. The robots shown in Figure II-6 are depicted in Figure II-7 and Figure II-8. It can be seen that most Delta Robots have a mechanical accuracy of approximately .1mm. This sets a standard that will be the goal of the robot designed in this thesis. Achieving a resolution on the order of magnitude of.1mm will be considered a success due to the prototype nature of the designed robot.

	Build Area	Footprint	Height (in)	Accuracy	Speed (mm/s)
Mini Kossel	6" Ø, 8.2" height	12"x12"	25.5"	.1mm	300
Orion Delta	6" Ø, 9" height	14"x14"	24"	.05mm	300
Deltaprintr	7" Ø, 10" height	11"x11"	24"	.1mm	200
Deltamaker	9.5" Ø, 10.2" height	16"x16"	27"	.1mm	300
Fanuc M-1iA/.5	11" Ø, 4" height	20"x17"	24.75"	.02mm	3000
Thesis Robot	12"Ø, 15" height	27"x27"	36"	Studied	Studied

Figure II-6: Delta Robot Information

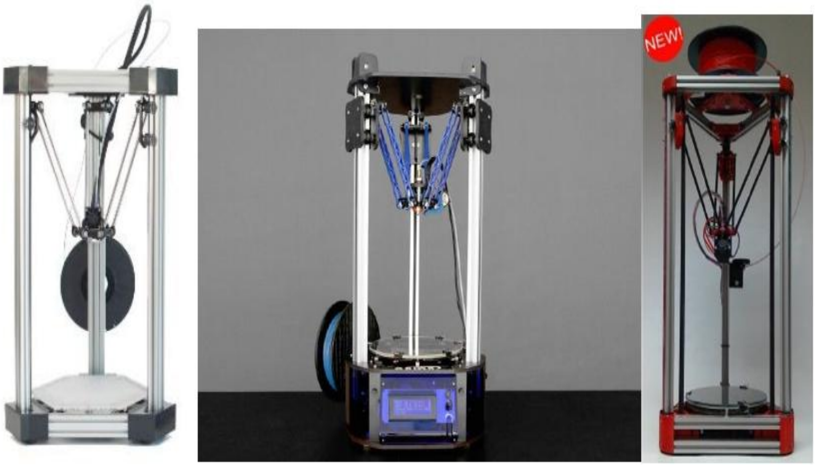


Figure II-7: A- Delta maker 3D Printer [40]

B- ORION 3D Printer[23]

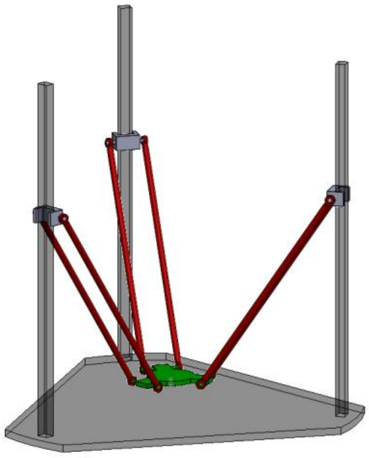
C- Kossel Mini 3D Printer [24]

Figure II-8: A- Delta printer 3D Printer [25]

B- Fanuc M-1iA Delta Robot [26]

Most Delta Robots have top speeds of 200-300 mm/s as shown in Figure II-6. Although the robots are capable of high speeds, often they are operated at much slower speeds. The 3D printer used to construct many parts of the Delta Robot designed in this thesis gave the best results at speeds of 20mm/s and lower. Due to the complicated nature of 5th order polynomial control, reaching speeds of 200-300 mm/s is unlikely and unnecessary for the purposes of this thesis. As technology increases, low cost microcontrollers will increase in computational power. This will result in lower calculation times and higher top speeds for the robot being controlled via the presented method. Because of this, high speeds will be attainable using the same methods presented in this thesis when more powerful hardware becomes available.

The DELTA robot with linear actuators is shown in (**Erreur ! Source du renvoi introuvable.**), and the geometric parameters are shown in (



), where the moving platform is connected to the base by three identical serial chains. Each of the three chains contains one spatial parallelogram, the vertices of which are four spherical joints. Each parallelogram is connected to the base by a prismatic joint. The moving platform of the robot has three translational Do Fs with respect to the base. And the output can be obtained through the combination of the actuation to the three prismatic joints [13, 14].

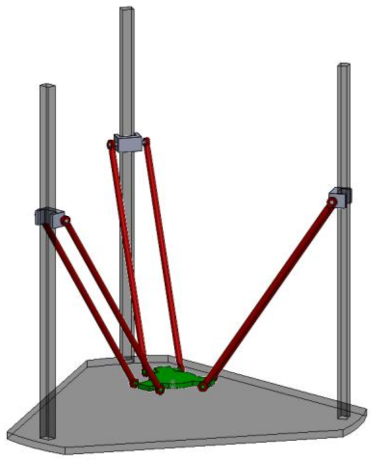


Figure II-9: A DELTA robot with linear actuators.

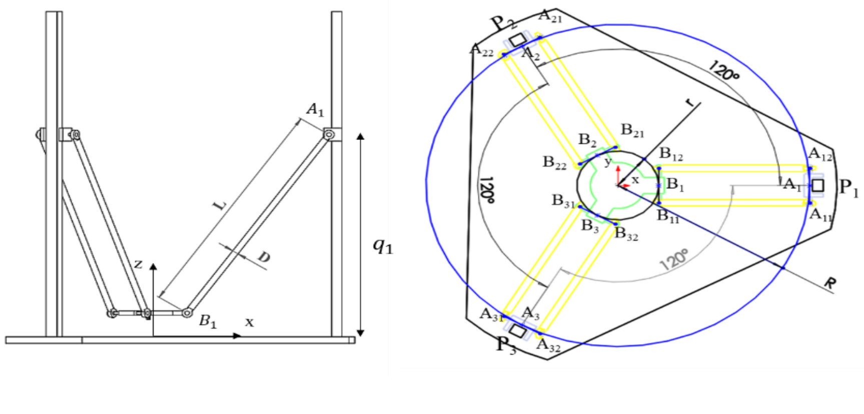
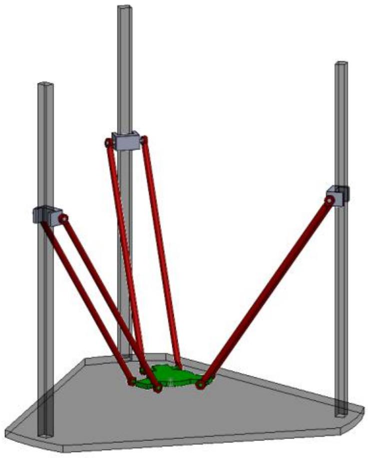


Figure II-10: Geometrical parameters of robot DELTA.

II-2 Inverse And Forward Position Relationship

$q = [q_1 , \ q_2, \ q_3]^T$ and $x = [x, y, z]^T$ respectively stand for the set of joint positions and the vector of Cartesian position. The geometrical parameters are defined in



. The coordinate of the point$A_i and B_i$ in the frame R

$$[A_i]_R = [Rcos(\theta_i), Rsin(\theta_i), q_i]^T \ , i = 1, 2, 3. \quad (0\text{-}1)$$

$$[B_i]_R = [rcos(\theta_i) + x, rsin(\theta_i) + y, z]^T \ , \qquad i = 1, 2, 3. \quad (0\text{-}2)$$

Where $\theta_i = 2(i-1)\frac{\pi}{3}$, $i = 1, 2, 3.$

Then, the inverse kinematics of the DELTA robot can be solved by writing the following constraint equations:

$$\|[B_i - A_i]_R\| = L \quad (0\text{-}2)$$

That is,

$$(x - (R - r)cos(\theta_i))^2 + (y - (R - r)sin(\theta_i))^2 + (z - q_i)^2 = L^2, \quad i = 1, 2, 3. \quad (0\text{-}3)$$

The inverse and forward kinematics for the Delta robot respectively given by Eq **Erreur ! Source du renvoi introuvable.** and Eq **Erreur ! Source du renvoi introuvable.**:

$$\begin{cases} q_1 = z + \sqrt{L^2 - (r - R + x)^2 - y^2} \\ q_2 = z + \sqrt{L^2 - \left((R - r)\frac{1}{2} + x\right)^2 - \left((r - R)\frac{\sqrt{3}}{2} + y\right)^2} \\ q_3 = z + \sqrt{L^2 - \left((R - r)\frac{1}{2} + x\right)^2 - \left((R - r)\frac{\sqrt{3}}{2} + y\right)^2} \end{cases} \tag{0-4}$$

$$\begin{cases} (A^2 + C^2 + 1)z^2 + 2((C(D - (R - r)) + A * B - q\_1\,)\,z + (B^2 + \left(D - (R - r)\right)^2 + q_1^2 - L^2 = 0 \\ y = Az + B \\ x = Cz + D \end{cases} \tag{0-5}$$

Where: $A = \frac{(q_2 - q_3)}{\sqrt{3}(r - R)}$ $B = \frac{q_3^2 - q_2^2}{\sqrt{3}(r - R)}$ $C = \frac{2(q_2 - q_1) - A\sqrt{3}(r - R)}{3(R - r)}$ $D = \frac{q_1^2 - q_2^2 - B\sqrt{3}(r - R)}{3(R - r)}$

II-3 Velocity relationships

Eq (0-3) can be differentiated with respect to time to obtain the velocity equations, which leads to

$$\dot{q} = J_q^{-1} J_x \dot{x} = J^{-1} \dot{x} \tag{0-6}$$

Where the Jacobian matrix J is $J = J_x^{-1} J_q$

$$J_x = \begin{bmatrix} r - R + x & y & z - q_1 \\ \frac{1}{2}(R - r) + x & \frac{\sqrt{3}}{2}(r - R) + y & z - q_2 \\ \frac{1}{2}(R - r) + x & \frac{\sqrt{3}}{2}(R - r) + y & z - q_3 \end{bmatrix} and J_q = \begin{bmatrix} z - q_1 & 0 & 0 \\ 0 & z - q_2 & 0 \\ 0 & 0 & z - q_3 \end{bmatrix} \tag{0-7}$$

II-4 Condition number and singularity analysis

The condition number of the inverse Jacobian matrix is

$$\kappa_J = \|J^{-1}\| \|J\| \tag{0-8}$$

The condition number is quite often used as an index to describe first the accuracy/dexterity of a robot and, second, the closeness of a pose to a singularity[15]. Actually, the condition number κ_J is such an all-around index that can evaluate the dexterity, isotropy, as well as the static stiffness of a robot [16, 17].

- *Serial and Parallel singularities:* to eliminate the serial ($|J_q| = 0$) and parallel ($|J_x| = 0$) singularities it is necessary to choose $\left(L > (R - r)\right)$

II-5 Acceleration relationship

This section is dedicated to the relation between operational space acceleration and velocity and the joint space velocities. Establishing the derivative of kinematics Eq. (0-6) with respect to time leads to

$$\ddot{q} = J^{-1}\ddot{x} + J_q^{-1}(\dot{J}_x - \dot{J}_q J^{-1})\dot{x} \tag{0-9}$$

Where

$$\dot{J}_x = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} - \dot{q}_1 \\ \dot{x} & \dot{y} & \dot{z} - \dot{q}_2 \\ \dot{x} & \dot{y} & \dot{z} - \dot{q}_3 \end{bmatrix} and\, \dot{J}_q = \begin{bmatrix} \dot{z} - \dot{q}_1 & 0 & 0 \\ 0 & \dot{z} - \dot{q}_2 & 0 \\ 0 & 0 & \dot{z} - \dot{q}_3 \end{bmatrix} \tag{0-10}$$

By deriving the direct motor model with respect to the time we get , we can consider x¨ as a function of q¨:

$$\ddot{x} = J\ddot{q} + \dot{J}\dot{q} \tag{0-11}$$

II-6 Inverse dynamics of linear Delta robot

The dynamics analysis of parallel mechanics machines is difficult due to the presence of multiple closed chains. many approaches are proposed, including the Newton–Euler formulation, Lagrangian formulation, the principle of virtual work principle.

For the inverse dynamics problem, a preferred trajectory of the moving platform is given, and the problem is to determine the input forces required to provide that movement. To obtain this goal, the dynamical equations of movement are formulated through the principle of virtual work.

We can find the Jacobian matrix through the kinematic model, which is a 3 × 3 matrix and describes the relationship between the speed of the triggers and the speed of the mobile platform. From Eq. (II-7), we can generate

$$\delta x = J\delta q \tag{0-12}$$

Let$\tau = [\tau_1, \tau_2, \tau_3]^T$ be the force vector of the actuator and $\delta q = [\delta q_1, \delta q_2, \delta q_3]^T$ the corresponding virtual displacement vector. Let $Fe_N = [F_x, F_y, F_z]^T$ be the external forces actuated on the nacelle. As the DELTA Robot is a parallel mechanism with three DOFs, the mobile platform can only move along x, y and z. Suppose that $\delta x = [\delta d_x, \delta d_y, \delta d_z]^T$ is the corresponding virtual displacement vector. Then, adopting the principle of virtual work, the following equation can be derived:

$$\tau^T.\delta q + {G_m}^T.\delta q + {Fe_N}^T.\delta x + {G_N}^T.\delta x - {f_m}^T.\delta q - {f_N}^T.\delta x - w_b = 0 \qquad (0\text{-}13)$$

- ✓ G_m, G_N the force of gravity of the motors and the force of gravity of the nacelle.
- ✓ f_m, f_N vectors represent of the force of inertia of the motors and of the nacelle.
- ✓ w_b is the virtual work of the bars. Or

$$G_m = M_{mot}.g.[1,1,1]^T, \qquad G_N = [0,0,M_{nac}.g]^T$$

$$f_m = \overline{M_{mot}}.\ddot{q}, \qquad f_N = \overline{M_{nac}}.\ddot{x}$$

$$\overline{M_{mot}} = \begin{bmatrix} M_{mot} & 0 & 0 \\ 0 & M_{mot} & 0 \\ 0 & 0 & M_{mot} \end{bmatrix}, \qquad \overline{M_{nac}} = \begin{bmatrix} M_{nac} & 0 & 0 \\ 0 & M_{nac} & 0 \\ 0 & 0 & M_{nac} \end{bmatrix}$$

M_{mot} : Actuator mass

M_{nac} : Nacelle mass

- Virtual work of a rigid bar (w_b):

It is enough to know the virtual displacement vectors and the acceleration vectors at the 2 ends of a rigid bar to determine the virtual work necessary to obtain this virtual displacement. We suppose that

$$\delta_{q1} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 1\,0\,0 \end{bmatrix} \delta q = J_{q1}\delta q, \qquad \delta_{q2} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 0\,1\,0 \end{bmatrix} \delta q = J_{q2}\delta q, \qquad \delta_{q3} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 0\,0\,1 \end{bmatrix} \delta q = J_{q3}\delta q$$

$$\ddot{q_1} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 1\,0\,0 \end{bmatrix} \ddot{q} = J_{q1}\ddot{q}, \qquad \ddot{q_2} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 0\,1\,0 \end{bmatrix} \ddot{q} = J_{q2}\ddot{q}, \qquad \ddot{q_3} = \begin{bmatrix} 0\,0\,0 \\ 0\,0\,0 \\ 0\,0\,1 \end{bmatrix} \ddot{q} = J_{q3}\ddot{q}$$

Where δq, $\ddot{q}$ are the virtual displacement and the acceleration vectors of the bars ends next to actuators, δx, $\ddot{x}$ are the virtual displacement and the acceleration vectors of the bars ends next the mobile platform.

The virtual work of a bar given by

$$w_{b_i} = \frac{M_b}{3}\left[{\ddot{q}_i}^T\delta_{qi} + \ddot{x}^T\delta x + \frac{1}{2}\left({\ddot{q}_i}^T\delta x + \ddot{x}^T\delta_{qi}\right)\right] i = 1,2,3 \qquad (0\text{-}14)$$

Replacing Eq (0-12) and (0-11) in Eq (0-14) gives

$$w_{b_i} = \frac{M_b}{3}\left[{\ddot{q}_i}^T\delta_{qi} + (\ddot{q}^T J^T + \dot{q}^T \dot{J}^T)J\delta q + \frac{1}{2}\left({\ddot{q}_i}^T J\delta q + (\ddot{q}^T J^T + \dot{q}^T \dot{J}^T)\delta_{qi}\right)\right]$$

$$w_{b_i} = \frac{M_b}{3}\left[{\ddot{q}_i}^T\delta_{qi} + \ddot{q}^T J^T J\delta q + \dot{q}^T \dot{J}^T J\delta q + \frac{1}{2}\left({\ddot{q}_i}^T J\delta q + \ddot{q}^T J^T\delta_{qi} + \dot{q}^T \dot{J}^T\delta_{qi}\right)\right] \qquad (0\text{-}15)$$

$$w_{b_i} = \frac{M_b}{3}\left[\ddot{q}^T\left(J_{qi}^T J_{qi} + J^T J + \frac{1}{2}\left(J_{qi}^T J + J^T J_{qi}\right)\right) + \dot{q}^T\left(\dot{J}^T J + \frac{1}{2}\dot{J}^T J_{qi}\right)\right]\delta q$$

When

$$\Gamma_{\ddot{q}_i}^T = J_{qi}^T J_{qi} + J^T J + \frac{1}{2}\left(J_{qi}^T J + J^T J_{qi}\right) \Leftrightarrow \Gamma_{\ddot{q}_i} = \left(J_{qi}^T J_{qi} + JJ^T + \frac{1}{2}\left(J^T J_{qi} + J_{qi}^T J\right)\right)$$
$$\Gamma_{\dot{q}_i}^T = \dot{J}^T J + \frac{1}{2}\dot{J}^T J_{qi} \Leftrightarrow \Gamma_{\dot{q}_i} = \left(J^T \dot{J} + \frac{1}{2}J_{qi}^T \dot{J}\right) \quad (0\text{-}16)$$

So

$$w_b = \sum_{i=1}^{3} w_{b_i} = \frac{M_b}{3}\sum_{i=1}^{3}\left[\ddot{q}^T \Gamma_{\ddot{q}_i}^T + \dot{q}^T \Gamma_{\dot{q}_i}^T\right]\delta q \quad (0\text{-}17)$$

Replacing Eqs (0-17) and (0-12) in Eq.(0-13) gives

$$(\tau^T + G_m{}^T + Fe_N{}^T.J + G_N{}^T.J - f_m{}^T - f_N{}^T.J - \frac{M_b}{3}\sum_{i=1}^{3}\left[\ddot{q}^T \Gamma_{\ddot{q}_i}^T + \dot{q}^T \Gamma_{\dot{q}_i}^T\right]).\delta q = 0 \quad (0\text{-}18)$$

Since this equation is valid for any virtual displacement δq, we can obtain

$$\tau^T + G_m{}^T + Fe_N{}^T.J + G_N{}^T.J - f_m{}^T - f_N{}^T.J - \frac{M_b}{3}\sum_{i=1}^{3}\left[\ddot{q}^T \Gamma_{\ddot{q}_i}^T + \dot{q}^T \Gamma_{\dot{q}_i}^T\right] = 0 \quad (0\text{-}19)$$

By Taking transfer Eq (II-20) and arranging it to replace inertial forces gives

$$\tau + G_m + J^T.Fe_N + J^T.G_N - \overline{M_{mot}}.\ddot{q} - J^T\overline{M_{nac}}.\ddot{x} - \frac{M_b}{3}\sum_{i=1}^{3}\left[\Gamma_{\ddot{q}_i}\ddot{q} + \Gamma_{\dot{q}_i}\dot{q}\right] = 0 \quad (0\text{-}20)$$

So,

$$\tau = \left(\overline{M_{mot}} + J^T\overline{M_{nac}}J + \frac{M_b}{3}\sum_{i=1}^{3}\Gamma_{\ddot{q}_i}\right)\ddot{q} + \left(J^T\overline{M_{nac}}\dot{J} + \frac{M_b}{3}\sum_{i=1}^{3}\Gamma_{\dot{q}_i}\right)\dot{q} - G_m - J^T.G_N - J^T.Fe_N \quad (0\text{-}21)$$

We can give the mass matrix to the our device by the next equation:

$$M = \overline{M_{mot}} + J^T\overline{M_{nac}}J + \frac{M_b}{3}\sum_{i=1}^{3}\Gamma_{\ddot{q}_i} \quad (0\text{-}22)$$

Chapter III: Results and discussion

Simulation is an essential step that takes place after mathematical modelling of any system, regardless of its level of complexity. Simulations are carried out to find out the effectiveness of the system and its portability in reality.

To make this simulation it should be use a simulator software to extract the results and apply it on the robot, and from these simulators MATLAB Simulink that we will use them.

In this chapter, we will simulate a parallel robot 3dof, by means of the inverse geometric model that gives the relationship of the position of the three motors in terms of the position of the end-effector. Then we simulate the speed of the three motors. Then the acceleration of the three motors acceleration. As these three steps take place simultaneously in real time. Finally, extracting the force applied to the movements.

This chapter introduces the mathematical model of three coplanar motors so that the relative positions and accelerations of these motors are able to create an understanding of the simulated movements. The speed and acceleration of the motors are then simulated, and we demonstrate the forces applied to the driving motors are the cause of these forces acting, according to this model, on the load-bearing object. All of these calculations are made in real time so that we can observe how the interventions of the motors affect the force applied to the object.

III-1 Simulation schematic of parallel robot 3DOF

The simulation of the robot using the MATLAB SIMULINK program is important for the study of the forces and how they make the robot move. In this way, we started from a specific path that the robot must move to arrive to the dynamic life model. As such, this simulation allows us study the effect of pressures, gravity, and suit able motor movements.

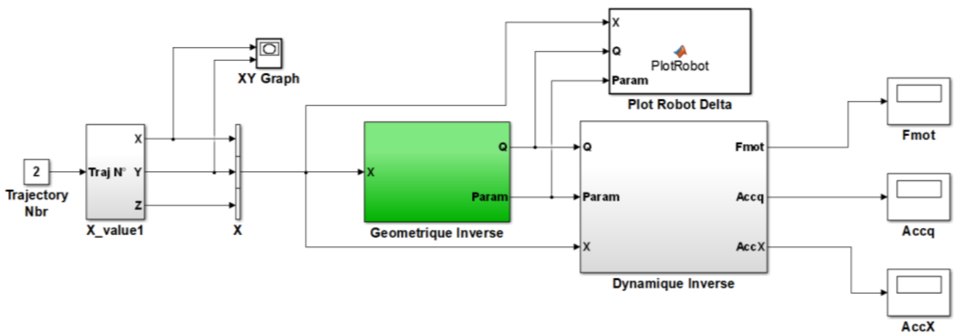


Figure 0.1 – Global simulation model using SIMULINK

III-1-1Parameters schematic

Through the subsystem we insert the parameter on this parallel processor, which are the dimensions and blocks of the parallel processor formed by it, and the external forces applied to the final responder.

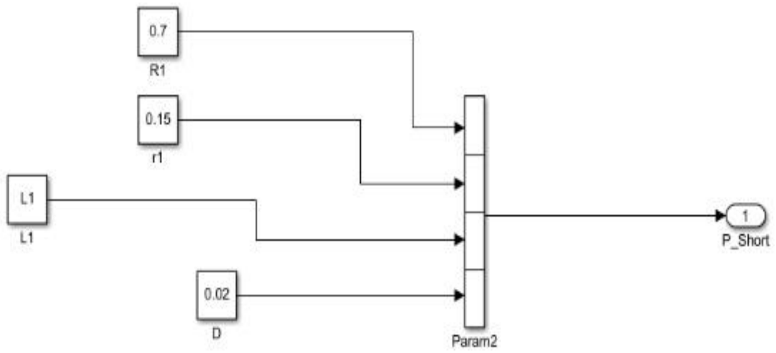


Figure 0.2 - Parameters schematic sub-system

Parameter table

Rayon de Base R (m)	Rayon de Plateforme mobile r (m)	Longueur des barres L (m)
0.7	0.15	0.99

The values in this table are used on the schema block for obtain the result dynamic forces and torque its mean the dynamic model results.

III-1-2 Trajectory schematic

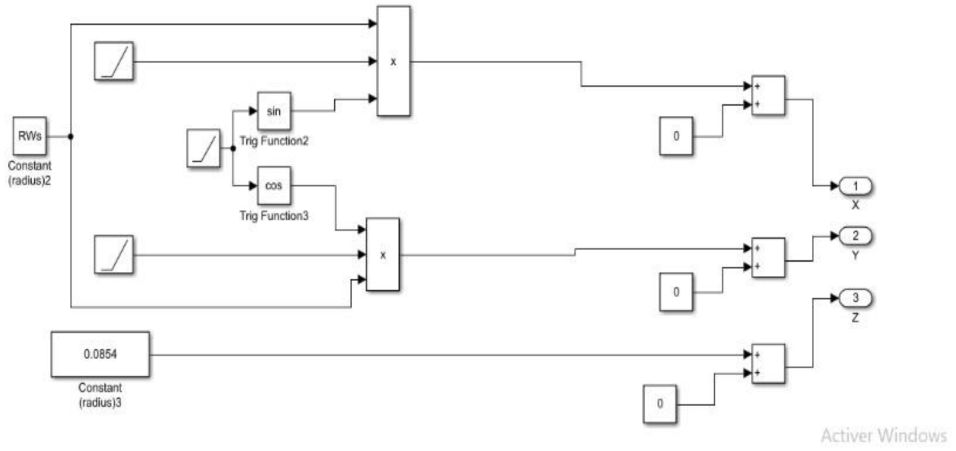


Figure 0.3 - Ttrajectory sub-system schematic

We chose a trajectory like a Circle (Figure III-4) from the center of the mobile platform to find the positions of the gliders (joint coordinates), its speeds and accelerations.

III-2 Results and graphs discussion

To be able to more accurately calculate the forces applied to the three gears B1.B2 and B3, We do the following simulations, extract curves and comment on them.

III-2-1 Trajectories

This graph shows model trajectories for the geometric and dynamic models of a 3DOF robot that were constructed for the purposes of this study.

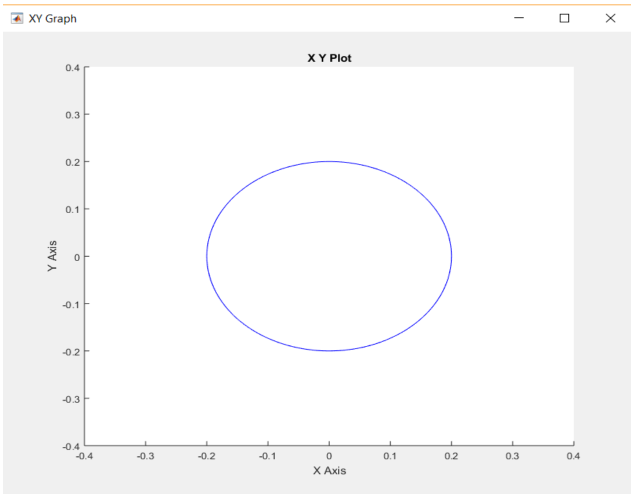


Figure 0.4 – circle trajectory of end-effector

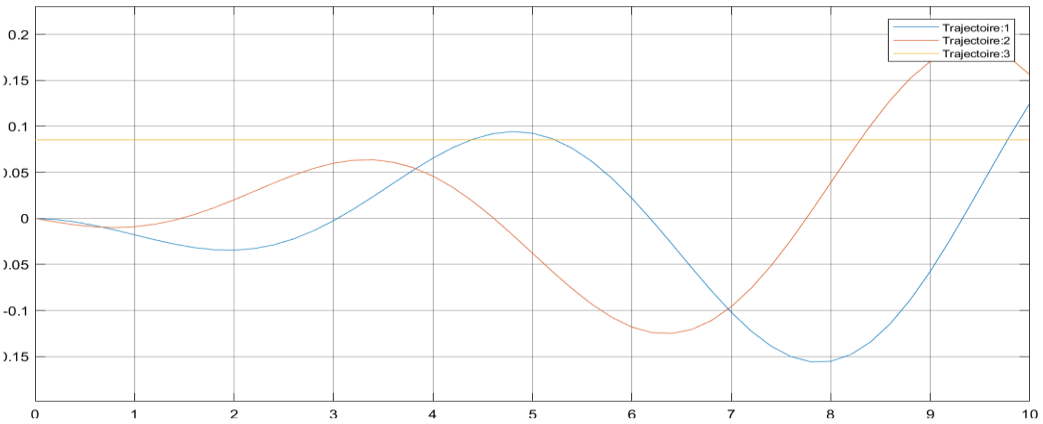


Figure 0.5 – End-effector position graph

III-2-2 End-effector velocity and acceleration

The end-effector position goes higher, while the end-effector velocity goes lower. The equation of the end-effector position could be written using three cosine functions. Three sine functions are used along the edges of the end-effector position, as shown in Figure III-5.

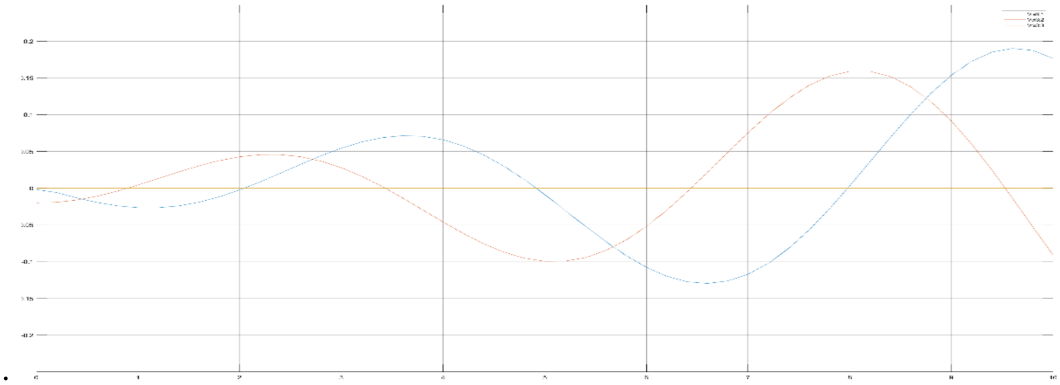


Figure 0.6 – End-effector velocity graph.

The graph below represents the acceleration of the final effector on the x-and y-axes, and it also depicts the Sine function. This acceleration represents its speed and direction, as well.

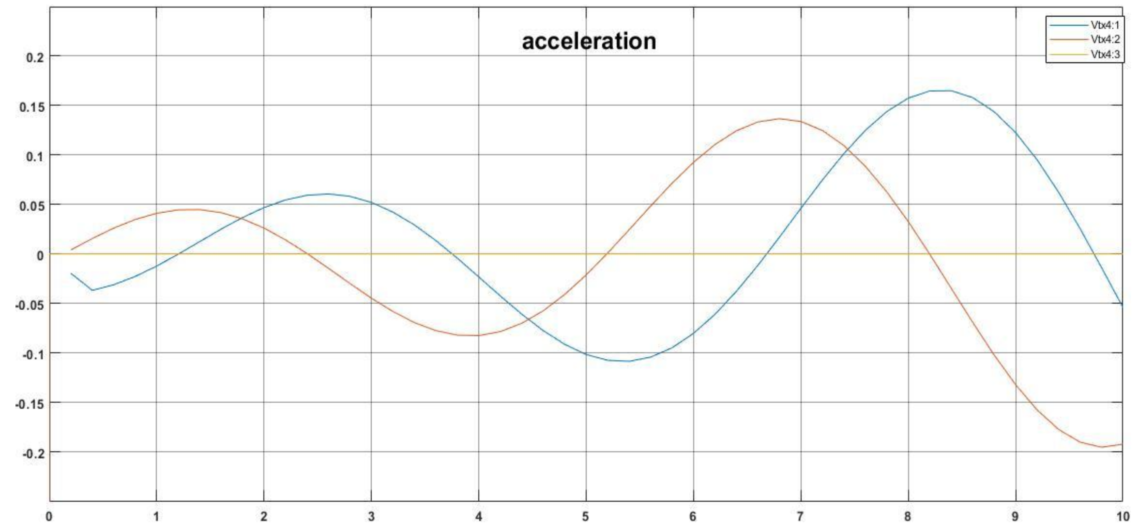


Figure 0.7 – The end-effector acceleration in X and Y-axes

III-2-3 Motors position, velocity and acceleration

The location of q1, q2 and q3 sliders changed in terms of time according to the first track such as the following format, the B2 engine moves first and the second B1 moves a few seconds later. Finally moving B3.

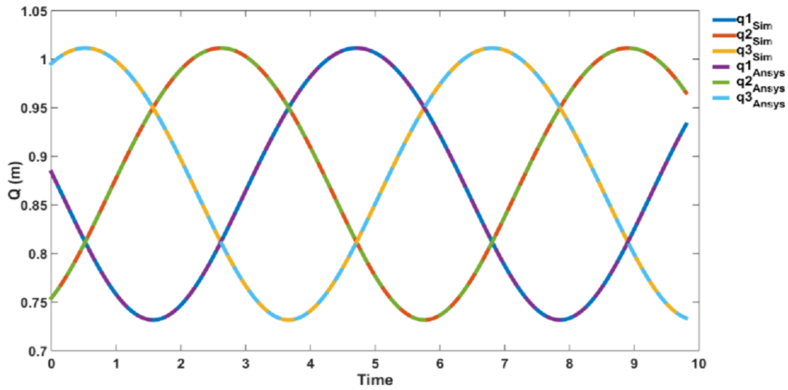


Figure 0.8– The position vector of sliders

The velocity $\dot{q}_1$ $\dot{q}_2$ and $\dot{q}_3$ change also according the positions and trajectory that we want design. In the zero (s), the three sliders start moving to get the specific point that it started on

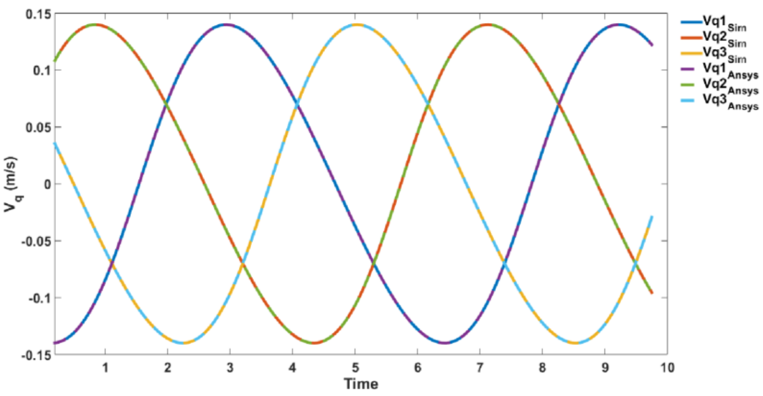


Figure 0.9 – Slider velocity graph

Here we see the acceleration of the drag slide is across the track. For each all, we notice the slides accelerate by approximately 1.1 m/s2. The first dipisquicker than the two others. On the other hand, if the design work is reversed, they reflect the acceleration (the first drop is fast, the second one is slow and the third one is slower than the second) of acceleration (the great than the beginning and still have a different acceleration).

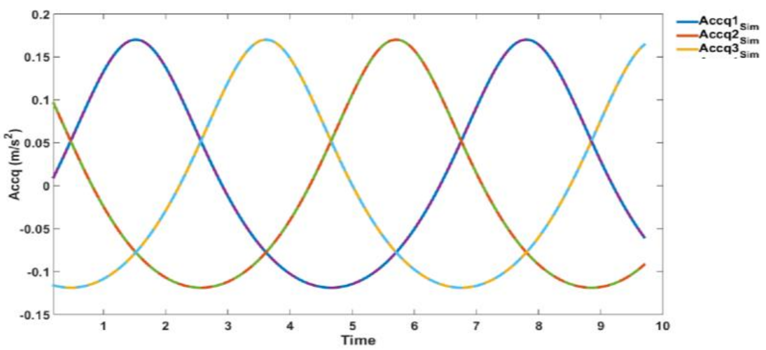


Figure III.10 – The acceleration of sliders, which linked to motors

III-2-4 Motors load force:

In order to calculate the forces applied to the three motors B1.B2.B3, we used the force equations to extract the results shown in and (Figure III.11).

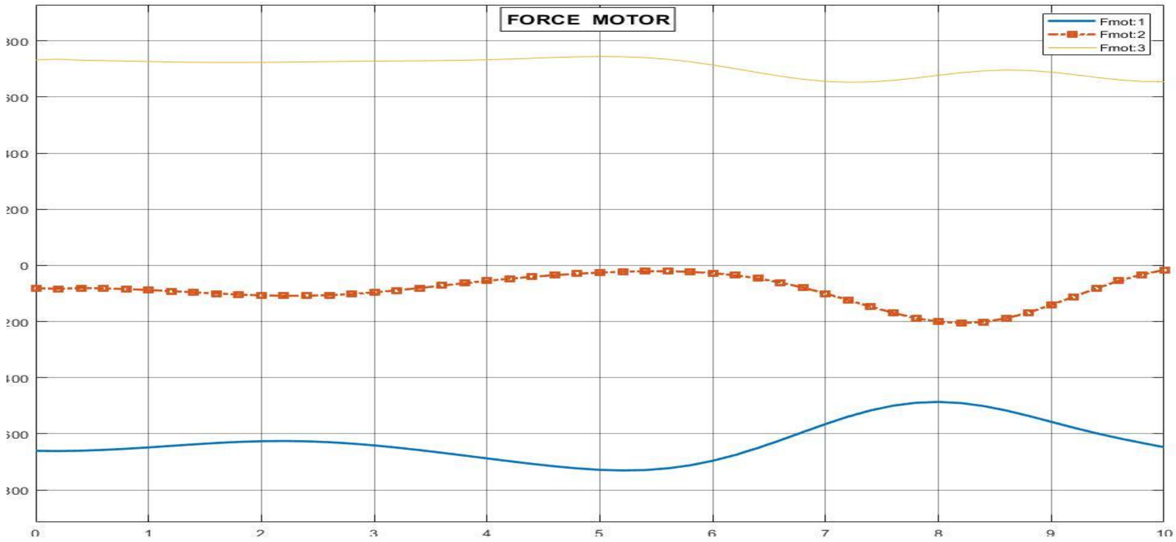


Figure 0.11 – The force load applied on motors

The force applied to the three motors produces the torque multiply the radius. In this case, it was assumed that the motor rotates a screw with an advance of 10 mm and an efficiency of 0.9 and the results obtained are shown in (figure III.11).

III-3 Conclusion

We believe that MATLAB Simulink can help us model robot motion more professionally and apply it on reality.

Simulations proved the importance of using this tool in the field of robotics, especially about the displacement changes, velocities, and acceleration values, as well as the relation between those elements. The simulator also can facilitate the deduction of the nature of the forces applied to the motors and create an idea of the best dynamic model for the robot.

General conclusion

The theory of software testing clearly shows that it is efficient to model a robot for tasks with multiple software modules. Although it is possible to build a dynamic model that can simulate the dynamics of the robot, no closed form can adequate control. For this reason, one can make assumptions about the robot dynamics in order to simulate the efficient motions, even though this approach will not accurately represent the entire behavior. We describe the simplifying hypothesis that leads to simulations that allow us to calculate the robot's behaviors. The research proves that this approach will accurately calculate the robot's motions when testing two modules of software. Expert university researchers developed a new robotic arm, which is parallel. The reason the robot can be used for scientists to develop humanoid robots.

Dynamic modeling of parallel robots is essential for the simulation of the parallel robot. In order to develop the initial mathematical modeling of the manipulator before its real creation, this is extremely important in the detection of an error in the model before applying it in the Reality. A: You did a great development. A few notes: If there is something not clear, you can ask about it. You speak clearly in very formal sentences. There is not that much, really,

Constructing this configuration represents the entire system as a dynamic model, which gives the relationship between the velocity, acceleration and position vectors for all three. This representation newly adds to the global position system in such a way that motors continue to move in a platform, really show casing that particular coupling. It not the same as the traditional Jacobian matrix, which linearizes the position so that motors move at the same rate as the platform. It also applies to the compatibility layer, which provides a dynamic equivalent to the necessary information to move the motor system.

This model can be verified by repetition of experiments. For the robot, the parameters such as gravity, weight of the parts, etc., are entered, producing a time-based dynamic model that predicts threat to operating motor. Then, the principal goal of this dynamic modelling is to calculate the hit forces applied by the motors, which are calculated in real time, to deduce the required torque, where hypotheses are put to simplify the model and use virtual work to reach the required modelling.

References

[1] J. Lenarcic and P. Wenger, *Advances in robot kinematics: analysis and design*. Springer, 2008.

[2] J.-P. Merlet, "Singular configurations of parallel manipulators and Grassmann geometry," *The international journal of robotics research,* vol. 8, no. 5, pp. 45-56, 1989.

[3] S. Küçük, *Serial and parallel robot manipulators: Kinematics, dynamics, control and optimization*. BoD–Books on Demand, 2012.

[4] J.-P. Merlet, "Parallel manipulators: state of the art and perspectives," *Advanced Robotics,* vol. 8, no. 6, pp. 589-596, 1993.

[5] S. Briot and W. Khalil, "Dynamics of parallel robots," *From rigid bodies to flexible elements. Springer,* 2015.

[6] A. Maria, "Introduction to modeling and simulation," in *Proceedings of the 29th conference on Winter simulation*, 1997, pp. 7-13.

[7] رمزي, فزاعي, ع. المنعم, حجايجي, ش. ع. النور, and حملاوي, "Dynamic modelling of parallel robot 2dof," 2021.

[8] R. Clavel, "A fast robot with parallel geometry," in *Proc. Int. Symposium on Industrial Robots*, 1988, pp. 91-100.

[9] R. Clavel, "Device for the movement and positioning of an element in space," ed: Google Patents, 1990.

[10] R. Williams, "The delta parallel robot: Kinematics solutions. Internet publication," ed, 2015.

[11] Z. Pandilov and V. Dukovski, "COMPARISON OF THE CHARACTERISTICS BETWEEN SERIAL AND PARALLEL ROBOTS," *Acta Technica Corviniensis-Bulletin of Engineering*, vol. 7, no. 1, 2014.

[12] X. Yang, Z. Feng, C. Liu, and X. Ren, "A geometric method for kinematics of delta robot and its path tracking control," in *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*, 2014: IEEE, pp. 509-514.

[13] O. Company and F. Pierrot, "Modelling and design issues of a 3-axis parallel machine-tool," *Mechanism and Machine Theory,* vol. 37, no. 11, pp. 1325-1345, 2002, doi: http://dx.doi.org/10.1016/S0094-114X(02)00040-X.

[14] X.-J. Liu, J. Wang, K.-K. Oh, and J. Kim, "A New Approach to the Design of a DELTA Robot with a Desired Workspace," *Journal of Intelligent and Robotic Systems,* vol. 39, no. 2, pp. 209-225, 2004.

[15] J.-P. Merlet, "Jacobian, manipulability, condition number, and accuracy of parallel robots," *Journal of Mechanical Design,* vol. 128, no. 1, pp. 199-206, 2006.

[16] J. K. Salisbury and J. J. Craig, "Articulated hands force control and kinematic Issues," *The International Journal of Robotics Research,* vol. 1, no. 1, pp. 4-17, 1982.

[17] X.-J. Liu, Z.-L. Jin, and F. Gao, "Optimum design of 3-DOF spherical parallel manipulators with respect to the conditioning and stiffness indices," *Mechanism and Machine Theory,* vol. 35, no. 9, pp. 1257-1267, 2000.

Abstract:

The design of innovative models for smart machines dedicated to high speed manufacturing, requires the implementation of analytical and numerical models to improve the motor and dynamic behavior of the machine, taking into account the component cutting distortions and the level of control of the machine.

In the context of multi-target improvement, Part I is concerned with identifying the parameters and variables inherent in each component of the thoughtful DELTA robot-type machine, the purpose of which is to improve the core elements of its structure. This requires the formulation of a multi-target problem by expressing objective functions, limitations and corresponding research spaces, as well as solving the problem through the use of high-performance mathematical methods and tools (genetic algorithms, etc.).

In the second part, we will focus on the problem of effective improvement, where the device will be considered for all its components (control, structure,...) The problem of improvement (objective functions and limitations, controls and variables) will be formulated and resolved, after determining the basic components of the specifications (limitations, disciplines, solutions).

In the third part, we took specific values (length and mass of the rod, engine block, rotation diameter....) And we applied it with MATLAB, we got the speed of the three engines and the power...

Keywords: multi-joint system, engineering and motor modeling, multi-target improvement, effective improvement.

ملخص:

يتطلب تصميم نماذج مبتكرة ل لروبوتات الذكية المخصصة للتصنيع عالي السرعة تنفيذ نماذج تحليلية ورقمية لتحسين السلوك الحركي والديناميكي ل هذه الروبوتات، مع الأخذ في الاعتبار تغيرات القطع المكونات ومستوى التحكم في الجهاز.

في سياق التحسين المتعدد الأهداف، يهتم الجزء الأول لأحدث ما توصلت إليه نمذجة الروبوتات الموازية الديناميكية ،

في الجزء الثاني، سنركز على نماذج حركية وديناميكية لروبوت دلتا.

في الجزء الثالث، أخذنا قيم محددة (طول وكتلة القضيب، كتلة المحرك، قطر الدوران....) وقمنا بتطبيقه مع MATLAB ، وحصلنا على سرعة المحركات الثلاثة وقيمة العزم.

الكلمات الرئيسية: أجهزة حركية متوازية، نمذجة الهندسة والمحركات، تحسين متعدد الأهداف، تحسين فعال.