Available online at http://www.iorajournal.org/index.php/ijgor/index

International Journal of Global Operations Research

e-ISSN: 2722-1016
p-ISSN: 2723-1739

Vol. 6, No. 4, pp. 282-289, 2025

Application of Conditional Trajectory Generation on Stewart Platform Robot as a CNC Machine Drive

Ahshonat Khoerunnisa[1*], Nur Jamiludin R[2], Aan Eko Setiawan[3], Siti Hadiaty Yuningsih[4], Hòe Nguyễn Đình[5], Suryaman[6]

[1]Unmanned Aerial System Engineering Technology,Manufacturing Automation and Mechatronics Engineering,Bandung Manufacturing Polytechnic, Bandung, Indonesia
[2,3]Automation System Engineering Technology,Manufacturing Automation and Mechatronics Engineering,Bandung Manufacturing Polytechnic, Bandung, Indonesia
[4]Machine maintenance,Manufacturing Engineering,Bandung Manufacturing Polytechnic, Bandung, Indonesia
[5]Hanoi University of Science and Technology
[6]Department of Mechanical Engineering, Universitas Kebangsaan Republik Indonesia, Bandung, Indonesia

[*]Corresponding author email: ahshonatk@polman-bandung.ac.id

Abstract

The development of industrial automation technology in recent decades has been very rapid. One of the technologies that supports industrial automation is robot manipulators. Robots can work with high precision, speed, and safety so that by using robots, industrial processes become more productive. The type of robot itself is divided into two, namely serial and parallel structures. Robots with parallel structures tend to be less studied, developed, and used in industry compared to serial structures even though there are several advantages of these parallel structures. Parallel structures have a kinematic configuration with a closed chain type, or it can be interpreted that each arm is connected to the point of origin. This relationship will result in robots having high precision and speed. Kinematic parallel manipulators perform better when compared to serial kinematics in terms of angular accuracy, acceleration at high speeds, and high stiffness. Therefore, this type of robot is very suitable for use in industries that require high-speed applications. In this study, a robot system was developed as a driving force for a CNC machine with its movements using a trajectory tracking control system. This system was chosen because this control has a point where each point contains position and speed information that is certainly needed for the CNC machine movement system.

Keywords: Robot Manipulator, Industrial Automation, Parallel Kinematics, CNC System, Trajectory Tracking Control

1. Introduction

The robot structure is divided into two, namely serial and parallel structures, parallel structures have kinematic movements with a closed chain type, or it can be interpreted that each arm is connected to the point of origin. This relationship will result in robots having high precision and speed. Kinematic parallel manipulators perform better when compared to serial kinematics in terms of angular accuracy, acceleration at high speeds, and high stiffness. Therefore, this type of robot is very suitable for use in industries that require high-speed applications. Such as pick and place and high-speed machining processes. This is certainly used in various industrial fields such as flight simulation systems, manufacturing, and medical applications.

One of the popular manipulators that is commonly used is the manipulator with 6 DOF (degree of freedom) Stewart Platform which was created by a scientist named Stewart as a flight simulator (Stewart, 1965). The manipulator consists of a top plate (moving plane) and a base plate (fixed plane), and 6 feet connected from the base plate to the top plate. The Stewart Platform uses the same architecture as the Gough mechanism (Merlet, 1999), which is a theory that studies a type of parallel manipulator also known as the Gough-Stewart theory.

Complex kinematic and dynamic mechanisms are often simplified in the model, which can lower accuracy. To overcome this, of course, it is necessary to identify the accuracy of kinematic and dynamic values. The kinematics and dynamics models on the Stewart Platform are more complicated when compared to serial robots. Basically, robot kinematics can be divided into two, namely forward kinematic and inverse kinematic. Forward kinematic on stewart platforms is very complex and difficult to solve due to the large number of nonlinear equations, therefore in general the problem of forward kinematic has more than one solution. As a result, many research papers that discuss parallel manipulators only focus on forward kinematics (Bonev and Ryu, 2000; Merlet, 2004; Harib and Srinivasan, 2003; Wang, 2007).

For the design and control of the stewart manipulator platform, a dynamic accuracy model is needed. The dynamic model in a parallel manipulator is a little complicated because the object has a closed loop structure, there is a relationship between parameter systems, and high non-linearity in the dynamic and kinematic system. Robot dynamics models can be divided into two topics: inverse and forward dynamic models. Dynamic Inverse models are important for control as long as their forward models are used for system simulation. To obtain the results of dynamic models on parallel manipulators, there are many literature studies that have been published. Analysis of dynamic models in parallel manipulators has been done classically with several different methods such as Newton's Euler Method, Lagrang concept, virtual working principle, and screw theory.

The approach with the Euler Newton method requires the computation of all the forces and moments between the links. One of the most important lessons presented by Dasgupta and Mruthyunjaya (1998) is the dynamic formulation of the Stewart Platform manipulator. In their study, a dynamic closed-form equation on 6-UPS SP in task-space and joint-space degradation was performed using this approach. The reduction of dynamic equations is implemented for dynamic inverse and forward dynamics in the SP manipulator and the resulting simulation is that the formulation provides a complete SP dynamic model. Moreover, the formulation demonstrates the Euler Newton approach applied to parallel manipulators and also shows how to lower dynamic equations through this formulation. This method is also used by Khalil and Ibrahim (2007). They present an easy solution to common equations in closed form for dynamic inverse and forward models on parallel robots. The method has been applied to two parallel robots with different structures. Harib and Srinivasan (2003) have presented kinematic and dynamic analysis on SP based on machine structure with inverse and forward kinematic, singularity, inverse and forward dynamics including dynamic calculation of friction force on joints and actuators. Newton's Euler formulation has been used to derive equations on rigid body dynamics. Do and Yang (1988) and Reboulet and Berthomieu, (1991) presented model dynamics on SP using Newton's Euler method approach. They introduced some simplifications to the leg model. Additional to their research, Guo and Li (2006), Carvahlo and Ceccarelli (2001) and the latter Riebe and Ulbrich (2003). They have also used an approach with Newton's Euler method.

Another method for deriving dynamic equations in parallel manipulators is the Lagrange method. This method has been used to describe the dynamics of mechanical systems from the concepts of work and energy. Abdellatif and Heimann (2009) have explicitly and in detail revealed differential equations on a set of 6 dimensions describing dynamic inverses in the kinematics of non-redundant parallel manipulators with 6 DOFs. They have demonstrated that a decline in explicit models is very possible if using the Lagrange method. Lee and Shah (1988) have derived a dynamic inverse model in joint space with 3DOF in parallel manipulator actuation using the Lagrange approach. Moreover, they have provided a numerical example of a helical path tracing to demonstrate the influence of link dynamics on its actuation force needs. Guo and Li (2006) have derived the equation of closed-form dynamics in an SP 6-DOF manipulator with a prismatic actuator using a combination method of Euler Newton and Lagrange formulations. To validate the formulation, they have studied numerical examples in other references. The results of the simulation show that the analysis of SP 6-DOF by combining the two methods. Lebert and his co-authors (1993) have studied dynamic equations in SP manipulators. These dynamics have been given in a step-by-step algorithm. Lin and Chen have presented an efficient computer symbol procedure for the SP manipulator equation model. They use the lagrange method to lower the equation. A learning object has been developed for MATLAB software based on an efficient algorithm for computation of dynamic equations of parallel robot link manipulators. And also they offer torque control to see the effectiveness of their dynamic equations. Motode Lagrange has also been used by others (Gregório and Parenti-Castelli, 2004; Beji and Pascal 1999; Liu et al., 1993).

For the dynamics of the parallel manipulator model, many approaches have been developed such as the principle of virtual work (Tsai, 2000, Wang and Gosselin, 1998; Geike and McPhee, 2003), screw theory (Gallardo et al., 2003), Kane method (Liu et al., 2000; Meng et al., 2010) and recursive matrix methods (Staicu and Zhang, 2008). Although the decrease in the dynamic equation of the parallel manipulator shows differences in complexity and computational load, the result of the actuator force/torque is processed with the same approach. The goal of this approach is to minimize the number of operations involved in the calculation of manipulator dynamics. It can be concluded that the dynamic equation in parallel manipulators is theoretically unproblematic. Moreover, in fact, attention must be focused on the accuracy and efficient calculation of the model.

The purpose of this paper is to present how the dynamics formulation on the SP 6 DOF manipulator works. The dynamic equation of the manipulator has been formulated by the Lagrangian method. Dynamic models include rigid body dynamics mechanisms as well as actuator dynamics. The Jacobian matrix is derived in two different ways. An accurate Jacobian matrix is very important if the simulation model is accurate. Ultimately, the dynamic equations of the manipulator can be simulated on MATLAB-simulics and verified on a plant system. This paper consists of 5 chapters, namely chapter 1 will discuss the introduction, chapter 2 on kinematics analysis and Jacobian matrix, chapter 3 on the dynamic equations of manipulator 6 DOF SP, chapter 4 on dynamic simulation and experimental results and the last one on Conclusion

2.1 Description and Modeling of the System

2.1.1 Structure Description

The SP manipulator used in this study is a manipulator with 6 DOF consisting of movable rigid plates, the plates are connected to the base plate through 6 feet. The legs have identical kinematics, each leg contains a precision screw ball and a DC motor. The length of the manipulator's legs can be controlled to perform platform movements.

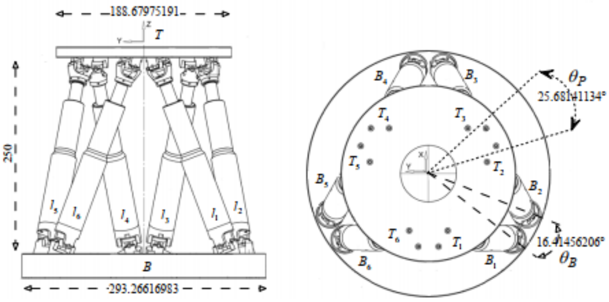

Figure 1. Manipulator Stewart Platform

2.1.2 Invers kinematic

To describe the movement of the platform movement clearly, the coordinate system is illustrated in Figure 2. The coordinate system ($B_{XYZ)}$ is attached to a fixed base and the other coordinates i.e. the coordinates ($Txyz$) are located at the center of mass of the moving platform. Points B_i and Ti, respectively connect the points to the moving base and platform. These points are placed fixed platforms and moving platforms (Figure 2.a).

Also, the angle of separation between points (T_2 and T3, T4 and T5, T1 and T6) is denoted by θ_p as shown in Figure 2.b. In the same way, the angle of separation between points (B_1 and B2, B3 and B4, B5 and B6) is denoted by θ_b.

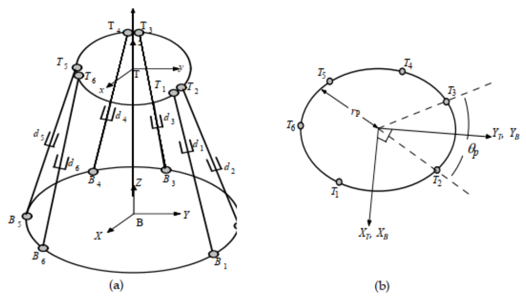

Figure 2. Schematic diagram manipulator SP

From figure 2b, the location *of ith* marked by the point (Ti) on the moving platform can be found equation 1. R_p and *rbase* are the fingers of the displacement of the platform and the fixed base, respectively. Using the same approach, the *location of ith* marked with a point (Bi) on the base platform can be obtained from the following equation.

$$^{B}R_T = R_Z(\gamma)R_Y(\beta)R_X(\alpha) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$= \begin{bmatrix} \cos\beta\cos\gamma & \cos\gamma\sin\alpha\sin\beta - \cos\alpha\sin\gamma & \sin\alpha\sin\gamma + \cos\alpha\cos\gamma\sin\beta \\ \cos\beta\sin\gamma & \cos\alpha\cos\gamma + \sin\alpha\sin\beta\sin\gamma & \cos\alpha\sin\beta\sin\gamma - \cos\gamma\sin\alpha \\ -\sin\beta & \cos\beta\sin\alpha & \cos\alpha\cos\beta \end{bmatrix}$$

$$P = \begin{bmatrix} P_x & P_y & P_z \end{bmatrix}^T$$

The position of the moving platform can be represented by the position vector, P and rotation of $^{\text{the BRT matrix}}$. The rotation matrix is determined by the angle of roll, pitch and yaw i.e. rotation α regarding the fixed x-axis, R_X (α), followed by rotation β i.e. fixed y-axis, RY (β) and rotation γ i.e. fixed z-axis, R_Z (γ). In this way, the rotation matrix of the moving platform relates to the basic platform coordinate system obtained. The position of the P vector indicates the translation vector of the moving platform that corresponds to the base platform. Thus, the rotation matrix and position vector are described as follows.

$$GT_i = \begin{bmatrix} GT_{xi} \\ GT_{yi} \\ GT_{zi} \end{bmatrix} = \begin{bmatrix} r_p \cos(\lambda_i) \\ r_p \sin(\lambda_i) \\ 0 \end{bmatrix}, \quad \begin{matrix} \lambda_i = \frac{i\pi}{3} - \frac{\theta_p}{2} & i = 1,3,5 \\ \lambda_i = \lambda_{i-1} + \theta_p & i = 2,4,6 \end{matrix}$$

$$B_i = \begin{bmatrix} B_{xi} \\ B_{yi} \\ B_{zi} \end{bmatrix} = \begin{bmatrix} r_{base} \cos(\upsilon_i) \\ r_{base} \sin(\upsilon_i) \\ 0 \end{bmatrix}, \quad \begin{matrix} \upsilon_i = \frac{i\pi}{3} - \frac{\theta_b}{2} & i = 1,3,5 \\ \upsilon_i = \upsilon_{i-1} + \theta_b & i = 2,4,6 \end{matrix}$$

Going back to figure 2, the GT_I and Bi vectors above have been selected as the position vectors. Vector L_i on link I can be calculated with the formula

$$L_i = R_{XYZ} GT_i + P - B_i \qquad i=1,2, \ldots ,6.$$

Where the position and orientation of the moving platform is

$$X_{p-o} = \begin{bmatrix} P_x & P_y & P_z & \alpha & \beta & \gamma \end{bmatrix}^T$$

The length of each leg can be calculated by the following formula

$$l_i^2 = \left(P_x - B_{xi} + GT_{xi} r_{11} + GT_{yi} r_{12}\right)^2$$
$$+ \left(P_y - B_{yi} + GT_{xi} r_{21} + GT_{yi} r_{22}\right)^2 + \left(P_z + GT_{xi} r_{31} + GT_{yi} r_{32}\right)^2$$

The length of the actua $l_i = \|L_i\|$

2.1.3 Jacobian matrix

For moving platforms, in general, the Jacobian matrix is very closely related to the speed of each joint (actuator). For parallel manipulators, the commonly used equations are as follows:

$$\dot{L} = J \dot{X}$$

Where in order $\dot{L}$ and is the speed of the feet and the moving platform. In this case, there are two different derivations of the Jacobian matrix developed. The first derivation is the general expression of the given Jacobian matrix and the second derivation is the Jacobian matrix which will be developed from equation 7. The equation can be rewritten to see the relationship between the speed of the actuator and the speed of the moving platform, namely by writing the following formula$\dot{X}\dot{L}\dot{X}_{P-0}$

$$\dot{L} = J_A \dot{X}_{p-0} = J_{IA} \vec{V}_{T_J}$$

In general, the speed at which the platform moves is written with formulations

$$\vec{V}_{T_J} = J_{IIA} \dot{X}_{p-o}$$

Where $\vec{V}_{T_J}$ is the speed of the connection point of the platform legs. Figure 3 shows the schematics of one leg of the SP manipulator.

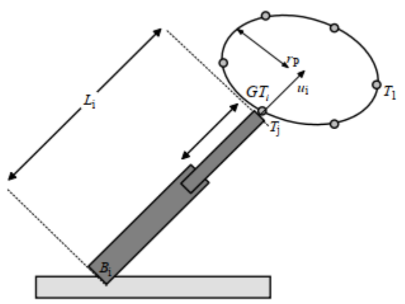

Figure 3. Schematic leg i^{th} manipulator parallel

$$J_{IIA}=\begin{bmatrix} I_{3x3} & R_Y(\beta)S(X)R_X(\alpha)R_Z(\gamma)GT_1 & S(Y)R_Y(\beta)R_X(\alpha)R_Z(\gamma)GT_1 & R_Y(\beta)R_X(\alpha)S(Z)R_Z(\gamma)GT_1 \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ I_{3x3} & R_Y(\beta)S(X)R_X(\alpha)R_Z(\gamma)GT_6 & S(Y)R_Y(\beta)R_X(\alpha)R_Z(\gamma)GT_6 & R_Y(\beta)R_X(\alpha)S(Z)R_Z(\gamma)GT_6 \end{bmatrix}_{18x6}$$

Then the substitution of equations 8 and 9 will form a new equation, namely

$$\dot{L}=J_A\,\dot{X}_{p-o}=J_{IA}\,J_{IIA}\,\dot{X}_{p-o}$$

The second Jacobian matrix equation is found in equation 10 so that the following calculation formula is obtained

$$\vec{u}_i=\frac{B_iT_j}{|L_i|}=\frac{L_i}{l_i},\begin{cases} j=\dfrac{i+1}{2} & if\ i\ is\ odd \\ j=\dfrac{i}{2} & if\ i\ is\ even \end{cases}$$

$$S=\begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$$

Where I_{3x3} the identity matrix 3 x 3 and S shows the symmetric screw with the 3 x 3 matrix associated with the vector $a=[a_x\ a_y\ a_z]^T$,

$$\dot{L}_i=\vec{V}_{T_j}\vec{u}_i=\begin{bmatrix}\dot{P}_x & \dot{P}_y & \dot{P}_z\end{bmatrix}^T\cdot\vec{u}_i+\omega\times\left({}^BR_T\,GT_i\right)\cdot\vec{u}_i=\dot{x}\cdot\vec{u}_i+\omega\times\left({}^BR_T\,GT_i\right)\cdot\vec{u}_i$$

The first Jacobian matrix on the SP manipulator is defined as follows

$$J_A=J_{IA}\,J_{IIA}$$

The second Jacobian matrix on the SP manipulator is defined as follows

$$J_B=J_{IB}\,J_{IIB}$$

It is known on a platform that moves with a cue on the coordinate base () so that it is obtained$GT_i=[GT_{xi}\ \ GT_{yi}\ \ GT_{zi}]^T, T_jB_{xyz}$

The velocity at the point is obtained by subtracting the equation of 17 times the following equation is obtainedT_j

$$T_j=\begin{bmatrix}P_x & P_y & P_z\end{bmatrix}^T+{}^BR_TGT_i=x+{}^BR_TGT_i$$

$$\vec{V}_{T_j}=\begin{bmatrix}\dot{P}_x & \dot{P}_y & \dot{P}_z\end{bmatrix}^T+\omega\times{}^BR_TGT_i=\dot{x}+\omega\times{}^BR_TGT_i$$

Where is the angular velocity on a moving platform that refers to the base platform. The following omega equation is obtained$\omega=(\omega_x,\omega_y,\omega_z)$

Since the projection velocity vector () on the prismatic joint axis of the i link produces the extension rate of the I link, the active joint velocity () can be calculated from $T_i L_i$

Equation is rewritten in the following matrix format

$$\dot{L}_i = J_B \begin{bmatrix} \dot{x} \\ \omega \end{bmatrix} = J_B \dot{X}_{p-o} = J_{IB} J_{IIB} \dot{X}_{p-o}$$

$$J_{IB} = \begin{bmatrix} u_{x1} & u_{y1} & u_{z1} & \left({}^{B}R_T GT_1 \, x \, \vec{u}_1\right)^T \\ \vdots & \vdots & \vdots & \vdots \\ u_{x6} & u_{y6} & u_{z6} & \left({}^{B}R_T GT_6 \, x \, \vec{u}_6\right)^T \end{bmatrix}_{6x6}$$

2.1.4 Model Dynamic

Dynamic analysis on SP manipulators is much more difficult when compared to serial manipulators because the kinematics on each leg are connected to the platform. Several methods are used to describe the problem and obtain a dynamic model on the manipulator. In literature, there is no best formulation that discusses manipulator dynamics. The lagrange method has been used in this regard because it has a regular structure of Analysis. To get the dynamic equation on the SP manipulator, the entire system is separated into two parts: the moving platform and the legs. The kinetic and potential energies for these two parts are calculated from the following energy equation decrease

$$I_{(mf)} = \begin{bmatrix} I_X & 0 & 0 \\ 0 & I_Y & 0 \\ 0 & 0 & I_Z \end{bmatrix} \tag{26}$$

$$\vec{\Omega}_{up(mf)} = R_Z(\gamma)^T R_X(\alpha)^T R_Y(\beta)^T \vec{\Omega}_{up(ff)} \tag{27}$$

2.1.5 Kinetic energy and potential of the manipulator foot

Each of the manipulator's legs consists of two parts: the moving part and the stationary part (figure 4). At the bottom there is a fixed part of the leg that is connected to the base platform through a universal joint, while at the top there is a

$$\vec{\Omega}_{up(ff)} = \dot{\alpha} R_Y(\beta)\vec{X} + \dot{\beta}\vec{Y} + \dot{\gamma} R_X(\alpha) R_Z(\gamma)\vec{Z}$$
$$= \left(\begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right.$$
$$\left. + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & 0 \\ 0 & 1 & -\sin\alpha \\ -\sin\beta & 0 & \cos\alpha \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix}$$

moving part that is connected to the moving platform that is also connected to the universal joint.

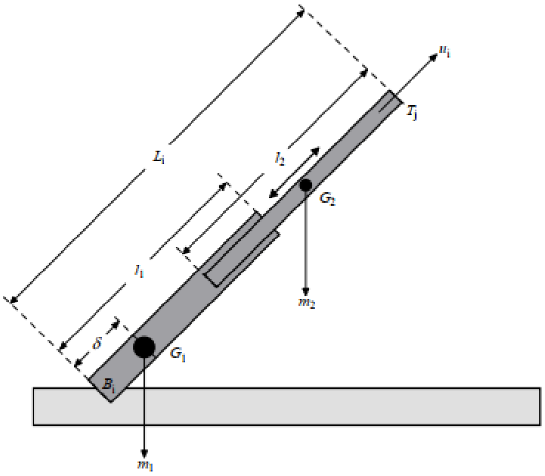

Figure 4. SP manipulator legs

As shown in the image above, the center of mass i.e. on each foot () has been taken into account. displays the center of mass on a fixed part of the leg. and in sequence is the value of Length and mass on a fixed part and is the distance between and . For the moving part of the leg, it shows the center of mass. and in order are the values of Length and mass.$G_i Leg_i = 1 \dots 6 G_{1i} l_1 m_1 \delta B_i G_i G_{2i} l_2 m_2$

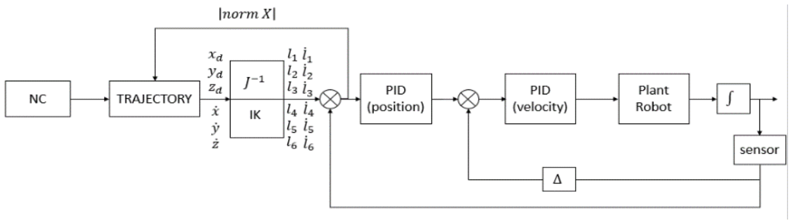

Figure 5. System diagram blocks

In this block of diagrams, it can be explained that the movement of the robot is based on the trajectory that has been previously translated from the G-Code language. Then we'll get the x, y, and z points and also the velocity on all three axes. These six elements were then processed with kinematic and jacobian inverse formulas so that the arm length and arm speed of each robot were obtained. After obtaining data on arm position and arm speed, both are processed using the PID system to match the robot's movements.

Here is a simulation of the stewart robot platform that has been created

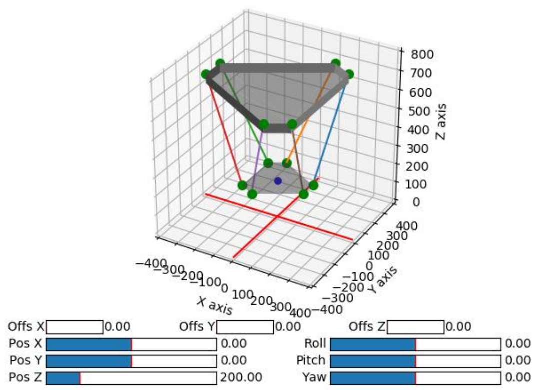

Figure 6. Simulation of robot movements

In the image above, it refers to the settings for the input of x, y, and z values. That way we can know how it moves, besides that we can also set the movement of the orientation of the frame by adjusting the roll, pitch, and yaw values. In this simulation, it has been adjusted to the original robot plant so that if the input axis cannot be achieved by the robot, then this simulation will give a caption that the robot cannot reach the desired point.

To process the system diagram block to match the desired system, of course data is needed that shows the speed graph of each motor and the linear potentiometer graph as the sensor. For the speed graph of each motorcycle is shown with a pwm graph of time while for the graph of each potentiometer it is shown with a graph of the length of the resistance. If both graphs have been made, what I do is a linearization process which will then be processed by the control system. The motor speed graph and potentiometer graph are shown in the following figure.

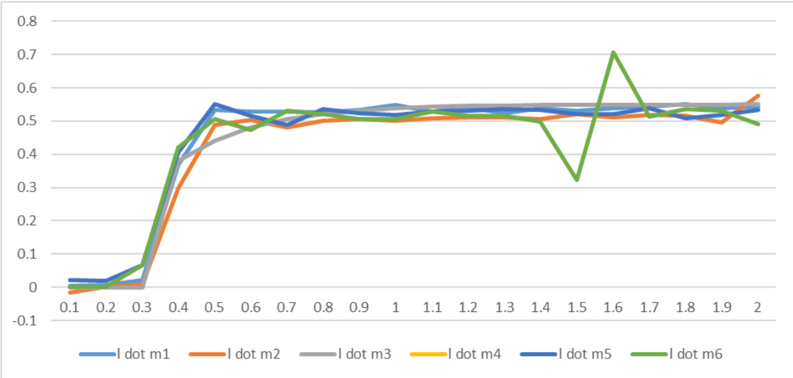

Figure 7. Motor dynamics graph 1-6

2. Conclussion

This research has successfully designed and built a control system for the Stewart Platform 6 DOF, utilizing it as a high-speed CNC machine driver, based on the advantages of parallel kinematics in terms of precision and stiffness compared to serial structures. The implemented control system is a trajectory tracking control using PID control on a Raspberry Pi, which processes position and velocity data from G-Code. The platform dynamics model is derived comprehensively using the Lagrangian method, while the actuator (robot leg) position feedback is obtained from a linear potentiometer sensor, confirming the feasibility of this platform as a robust and stable solution for manufacturing applications requiring high speed and accuracy.

References

Bingul, Zafer. Karahan Oguzhan. 2012. *Dynamic Modelling and Simulation of Stewart Platform.* Paper. Mechatronics Engineering: Kocaeli University Turkey.

Harib, Khalifa. Srinivisan, Krisnaswamy. 2003. *Kinematic and dynamic analysis of Stewart platform-based machine tool structures*. Paper. Research Gate.

Holy, Anita. 2009. *Design and Testing of Stewart Platform Robots.* Thesis. Mechanical Engineering, FTMD: ITB.

Jazztyan Indra P, Muhammad. 2014. *Design and Build Stewart Platform Control System.* Thesis. Mechanical Engineering, FTMD: ITB.

Mahalik, Nitaiguor Premchand. 2006. *Micromanufacturing and Nanotechnology*. Paper. University College of Engineering, Burla: India.

Promise, Firm. 2011. *Calculation of Reachable Workspace on the Stewart Platform Robot.* Thesis. Mechanical Engineering, FTMD: ITB.

Tosello, Guido. 2017. *Micro/ Nano Manufacturing*. Paper. Departement of Mechanical Engineering: Technical University of Denmark.