On the modeling and control of the cartesian parallel manipulator

Ayssam Y. Elkady, Sarwat N. Hanna and Galal A. Elkobrosy
Mathematics and Physics Dept. Faculty of Engineering, Alexandria University, Alexandria, Egypt
Email: Ayssam@gmail.com

The Cartesian Parallel Manipulator (CPM) which was proposed by Han Sung Kim, and Lung-Wen Tsai [1] consists of a moving platform that is connected to a fixed base by three limbs. Each limb is made up of one prismatic and three revolute joints and all joint axes are parallel to one another. In this way, each limb provides two rotational constraints to the moving platform and the combined effects of the three limbs lead to an over-constrained mechanism with three translational degrees of freedom. The manipulator behaves like a conventional X-Y-Z Cartesian machine due to the orthogonal arrangement of the three limbs. In this paper, the inverse dynamics of the CPM has been presented using Lagrangian multiplier approach to give a more complete characterization of the model dynamics. The dynamic equation of the CPM has a form similar to that of a serial manipulator. So, the vast control literature developed for serial manipulators can be easily extended to this class of manipulators. Based on this approach, four control algorithms; simple PD control with reference position and velocity only, PD control with gravity compensation, PD control with full dynamic feedforward terms, and computed torque control, are formulated. Then, the simulations are performed using Matlab and Simulink to evaluate the performance of the four control algorithms.

تنامت التطورات النظرية والتطبيقات العملية في مجال الروبوت المتوازى Parallel Manipulator خلال السنوات القليلة الماضية لما يمنحه من مميزات عدة مثل التحمل الزائد والجساءة العالية بالاضافة الى السرعة العالية . يتناول هذا البحث دراسة الروبوت المتوازى الكارتيزى (CPM) والتحكم الآلى فيه. ذلك الروبوت الذى قام بتصميمه كل من Han Sung Kim و Lung-Wen Tsai هذا الروبوت يوفر العديد من المزايا مثل التحمل الزائد والجساءة العالية بالاضافة الى السرعة العالية. ولهذا الروبوت ثلاث درجات حرية ويتكون من قاعدة ثابتة وأخرى متحركة حركة إنتقال متوازى ، وتتصل القاعدتان معا عن طريق ثلاثة أذرع متحركة متماثلة التركيب ، يتركب الذراع الواحد من عضوين يتحركان على امتداد أحد إتجاهات المحاور ويدوران حول نفس المحور ويحقق هذا الروبوت سلوك الروبوت الكارتيزى التقليدى كنتيجة للترتيب المتعامد لأذرعه الثلاثة. واستطاع هذا البحث التوصل الى النَموذج الرياضى الذى يعطى وصفا كاملا لديناميكا هذا الروبوت معتمدة على طريقة لاجرانج. ولقد تبين أن هذا النموذج الرياضى يشبه مثيله للروبوت المتسلسل لذا امكن استخدام بعض طرق التحكم الالى المستخدمة فى الروبوت المتسلسل ويعتبر تطبيق بعض طرق التحكم الالى أحد الأهداف الأساسية فى هذا البحث لذا أجريت دراسة على بعض مُتحكِمات الحركة للحصول على القوة اللازمة لاحداث الحركة المطلوبة وهذا يستوجب الإلمام بديناميكا الروبوت لتحسين النتائج المتحصل عليها كما تم تنفيذ المحاكاة عن طريق تحديد مسار لاختبار نتائج الطّرق المقترحة . كما أجريت دراسة على بعض طرق التحكم الالى مثل :

- PD control with position and velocity reference.
- PD control with gravity compensation.
- PD control with full dynamics feedforward terms.
- Computed torque control.

Keywords: Cartesian manipulator, Parallel manipulator, Dynamics, Control, Simulation.

1. Introduction

Parallel manipulators are robotic devices that differ from the more traditional serial robotic manipulators by virtue of their kinematic structure. Parallel manipulators are composed of multiple closed kinematic loops. Typically, these kinematic loops are formed by two or more kinematic chains that connect a moving platform to a base. This kinematic structure allows parallel manipulators to be driven by actuators positioned on or near the base of the manipulator. In contrast, serial manipulators do not have closed kinematic

loops and are usually actuated at each joint along the serial linkage. Accordingly, the actuators that are located at each joint along the serial linkage can account for a significant portion of the loading experienced by the manipulator. This allows the parallel manipulator links to be made lighter than the links of an analogous serial manipulator. Hence, parallel manipulators can enjoy the potential benefits associated with light weight construction such as high-speed operation [2].

Han Sung Kim and Lung-Wen Tsai [1] presented a parallel manipulator called CPM fig. 1 that employs only revolute and prismatic joints to achieve translational motion of the moving platform. They described its kinematic architecture and discussed two actuation methods. For the rotary actuation method, the inverse kinematics provides two solutions per limb, and the forward kinematics leads to an eighth-degree polynomial. Also, the rotary actuation method results in many singular points within the workspace. On the other hand, for the linear actuation method, there exists a one-to-one correspondence between the input and output displacements of the manipulator. Also, they discussed the effect of misalignment of the linear actuators on the motion of the moving platform. They suggested a method to maximize the stiffness in order to minimize the deflection at the joints caused by the bending moment because each limb structure is exposed to a bending moment induced by the external force exerted on the end-effector.

In this paper using Lagrange formulation, we develop the dynamic equation of the CPM. Based on the dynamical model, we reformulate four basic control algorithms simple PD control with reference position and velocity only, PD control with gravity compensation, PD control with full dynamic feedforward terms, and computed torque control. Then the simulation is used to evaluate the performance of each control algorithm.

The paper is organized as follows. In section 2, the kinematic relations are developed. In section 3, we use the Lagrange's equations of motion to derive the dynamic equation of the CPM. Four control algorithms are reviewed in section 4. In section 5, Simulation is described. Simulation results are presented in section 6, followed by concluding remarks in section 7.

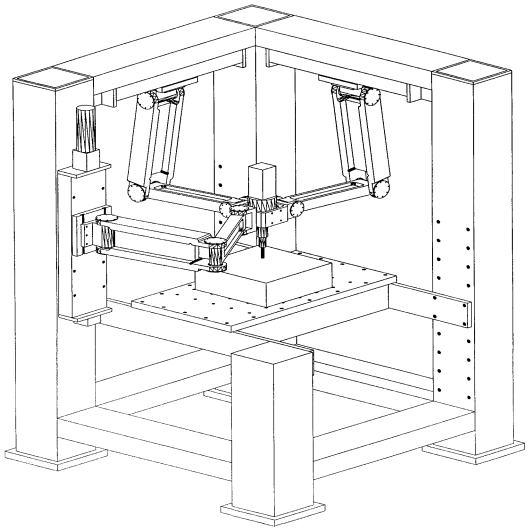

Fig. 1. Assembly drawing of the CPM.

2. Problem formulation

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-revolute-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 B_1, B_2, and B_3 and the fixed base is defined by three guide rods passing through A_1, A_2, and A_3. The three revolute joint axes in each limb are located at points A_i, M_i, and B_i and are parallel to the ground-connected prismatic joint axis. 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 L_1, and L_2.The starting point of a prismatic joint is defined by d_{0i} and the sliding distance is defined by $d_i - d_{0i}$.

The schematic diagrams of the three limbs of the CPM are sketched in fig. 3. The relationships for the three limbs are written for the position P [x, y, z] in the coordinate frame XYZ and differentiation with respect to time yields eq. 1.

$$\begin{bmatrix} \dot{\theta}_{11} \\ \dot{\theta}_{21} \\ \dot{\theta}_{31} \end{bmatrix} = \Gamma \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}. \qquad (1)$$

where

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

where $\dot{\theta}_{11}$, $\dot{\theta}_{21}$, and $\dot{\theta}_{31}$ are the derivatives of θ_{11} , θ_{21} , and θ_{31} with respect to the time and $\dot{x}$, $\dot{y}$, and $\dot{z}$ are the X,Y, and Z components of the velocity of point P on the moving platform in the reference frame. Differentiation of equation 1 with respect to the time gives,

$$\begin{bmatrix} \ddot{\theta}_{11} \\ \ddot{\theta}_{21} \\ \ddot{\theta}_{31} \end{bmatrix} = \Gamma \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + \frac{d\Gamma}{dt} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}. \qquad (2)$$

3. Dynamics of closed-chain mechanism

The Lagrange formulation [4] is used to find the actuator forces required to generate a desired trajectory of the manipulator. In general, the Lagrange multiplier approach involves solving the following system of equations:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} = Q_j + \sum_{i=1}^{k}\left(\lambda_i \frac{\partial f_i}{\partial q_j}\right). \qquad (3)$$

For j =1 to n, 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 are θ_{11}, θ_{21}, and θ_{31} beside the generalized coordinates x, y, and z. Thus we have θ_{11}, θ_{21}, θ_{31}, x, y, and z as the generalized coordinates. Eq. (3) represents a system of six equations in six variables, where the six variables are λ_i for i = 1, 2, and 3, and the three actuator forces, Q_j for j = 4, 5, and 6. The external generalized forces, Q_j 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.

It can be assumed that the first link of each limb is a uniform rod and its mass is m_1. The mass of the second rod of each limb is evenly divided between and concentrated at

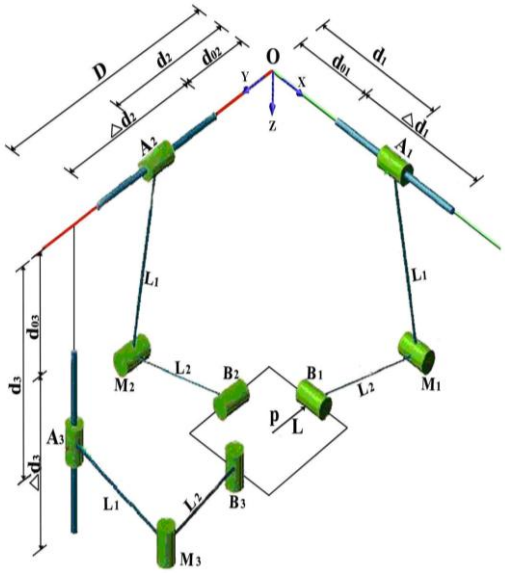

Fig. 2. Spatial 3-*PRRR* parallel manipulator.

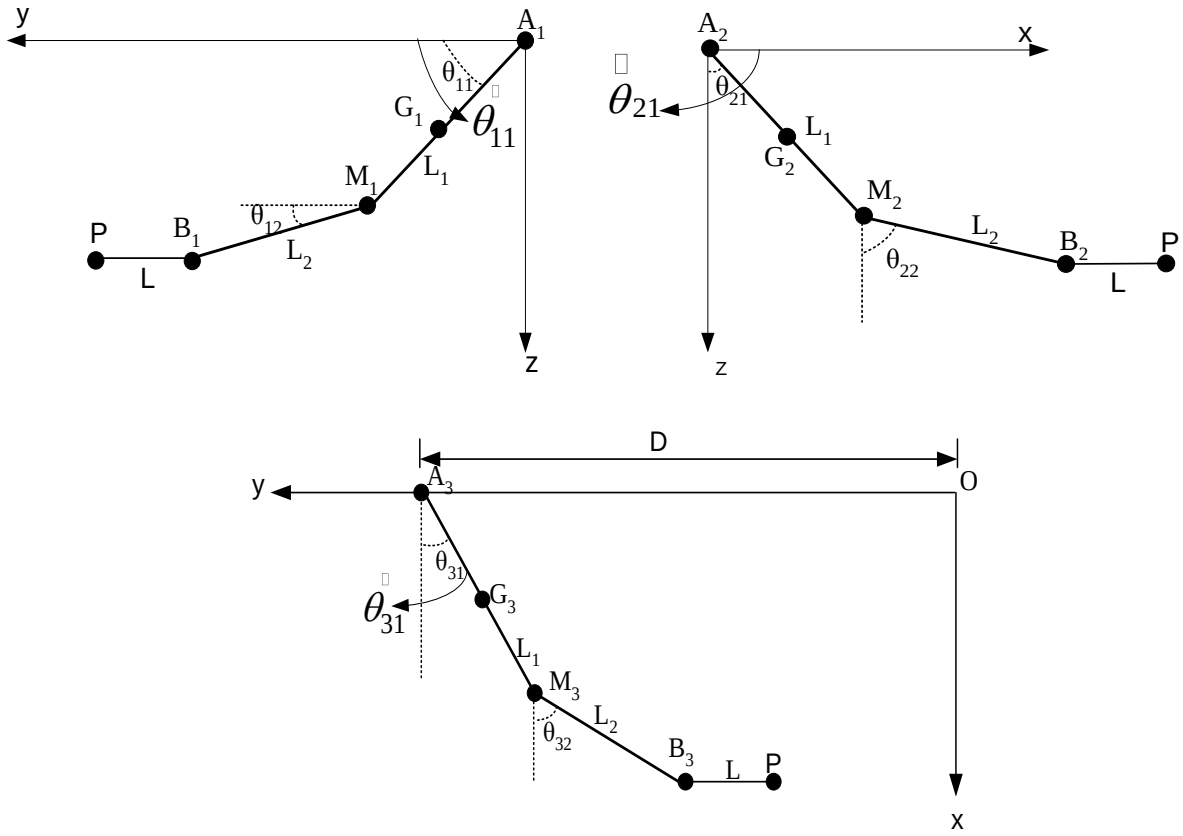

Fig. 3. Description of the joint angles and link lengths for the three limbs.

joints M_i and B_i. So, the two particles M_i and B_i have the same mass which is $\frac{m_2}{2}$. 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.

The total kinetic energy T of the manipulator is given by:

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

where the mass of the tool is m_4 and m_3 is the mass of the prismatic joint A_i and its actuator. 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:

$$V = -\frac{m_1 + m_2}{2} gL_1(\sin\theta_{11} + \cos\theta_{21}) - (m_1 + 2m_2 + m_3 + m_4)gz \quad (5)$$

The Lagrange function is $L = T - V$ where

$$L = A(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + B(\dot{\theta}_{11}^2 + \dot{\theta}_{21}^2 + \dot{\theta}_{31}^2) + C(\sin\theta_{11} + \cos\theta_{21}) + Ez \quad (6)$$

Where: $A = \frac{1}{2}[m_1 + 2m_2 + m_3 + m_4]$ $B = (\frac{m_1}{6} + \frac{m_2}{4})L_1^2$ $C = \frac{m_1 + m_2}{2} gL_1$ $E = (m_1 + 2m_2 + m_3 + m_4)g$.

Taking the partial derivatives of the Lagrange function with respect to the three generalized coordinates θ_{11}, θ_{21} , and θ_{31}, we obtain

$$2B\ddot{\theta}_{11} - C\cos\theta_{11} = \lambda_1, \quad (7)$$

$$2B\ddot{\theta}_{21} + C\sin\theta_{21} = \lambda_2, \quad (8)$$

$$2B\ddot{\theta}_{31} = \lambda_3. \quad (9)$$

Rearrangement of eqs. (7-9) then by substituting into eq. (2) yields

$$\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} = 2B\Gamma\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} + 2B\frac{d\Gamma}{dt}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} + C\begin{bmatrix}-\cos\theta_{11}\\ \sin\theta_{21}\\ 0\end{bmatrix}. \quad (10)$$

Taking the partial derivatives of the Lagrange function with respect to the three generalized coordinates x, y, and z, we obtain

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

$$2A\ddot{y} = F_y - \Gamma_{12}\lambda_1 - \Gamma_{22}\lambda_2 - \Gamma_{32}\lambda_3, \quad (12)$$

$$2A\ddot{z} - E = F_z - \Gamma_{13}\lambda_1 - \Gamma_{23}\lambda_2 - \Gamma_{33}\lambda_3, \quad (13)$$

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

Rearrangement of eqs. (11-13) then using eq. (10) yields

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

The dynamic equation of the whole system can be written as

$$F = M(q)\ddot{q} + G(q,\dot{q})\dot{q} + K(q), \quad (14)$$

where

$$F = \begin{bmatrix}F_x\\ F_y\\ F_z\end{bmatrix},\ \ddot{q} = \begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix},\ \dot{q} = \begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix},\ q = \begin{bmatrix}x\\ y\\ z\end{bmatrix},$$

$$M(q) = 2AI + \Gamma^T 2B\Gamma$$

$G(q,\dot{q}) = \Gamma^T 2B \dfrac{d\Gamma}{dt}$, and

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

Where q is the vector of joint displacement, $\dot{q}$ is the vector of joint velocities, F is the vector of applied force inputs, $M(q)$ is the manipulator inertia matrix, $G(q,\dot{q})$ is the manipulator centripetal and coriolis matrix, and $K(q)$ is the vector of gravitational forces.

4. Controller design

In this section, we review four basic control algorithms for control of the CPM:

4.1. PD control with position and velocity reference

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.

$$F = K_P e + K_D \dot{e}\,, \tag{15}$$

where $e = q_d - q$, is the vector of position error of the individual actuated joints, $\dot{e} = \dot{q}_d - \dot{q}$, is the vector of velocity error of the individual actuated joints, $\dot{q}_d$ 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. 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.

4.2. PD Control with gravity compensation

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:

$$F = K_P e + K_D \dot{e} + K(q_d)\,. \tag{16}$$

4.3. PD control with full dynamics feedforward terms

This type of controller augments the basic PD controller by compensating for the manipulator dynamics in the feedforward 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 in the form

$$F = M(q_d)\ddot{q}_d + G(q_d,\dot{q}_d)\dot{q}_d + K(q_d) + K_P e + K_D \dot{e}\,. \tag{17}$$

If the dynamic knowledge of the manipulator is accurate, and the position and velocity error terms are initially zero, the applied force F is sufficient to maintain zero tracking error during motion.

4.4. Computed torque control

This controller 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, using the following force law as described by Lewis [3], 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.

$$F = M(q)[\ddot{q}_d + K_D\dot{e} + K_p e] + G(q,\dot{q})\dot{q} + K(q)\,. \quad (18)$$

To show that the computed torque control scheme linearizes the controlled system, the force computed by equation 18 is substituted into equation 14, yielding:

$$M(q)\ddot{q} = M(q)\ddot{q}_d + M(q)[K_D\dot{e} + K_p e]$$

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:

$$\ddot{e} + k_D\dot{e} + k_P e = 0\,. \quad (19)$$

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 19 provides a second order damped system with a natural frequency of ω_n, and a damping ratio of ζ where:

$$\omega_n = \sqrt{K_P}\,,\ \zeta = \frac{K_D}{2\sqrt{K_P}}\,. \quad (20)$$

The natural frequency ω_n determines the speed of the response. It is customary in robot applications to take the damping ratio $\zeta = 1$ so that the response is critically damped. This produces the fastest non-oscillatory response. So, the values for the gain matrices K_D and K_P are determined by setting the gains to maintain the following relationship:

$$K_D = 2\sqrt{K_P}\ . \quad (21)$$

5. Simulation

In controlling the manipulator, any sudden changes in desired joint angle, velocity, or acceleration can result in sudden changes of the commanded force. This can result in damages of the motors and the manipulator. Here, the manipulator is given a task to move along careful preplanned trajectories without any external disturbances or no interaction with environment.

The sample trajectory of the end-effector is chosen to be a circular path (see fig. 4) with the radius of 0.175 meters and its center is O(0.425 ,0.425 ,0.3). This path is designed to be completed in 4 seconds when the end-effector reaches the starting point P1 (0.6, 0.425, 0.3) again with constant angular velocity $\omega = \frac{\pi}{2}$ rad/sec. The desired end-effector position along x-axis is $x = 0.425 + 0.175\cos(\omega t)$ meters, along y-axis is $y = 0.425 + 0.175\sin(\omega t)$ meters, and along z-axis is $z = const. = 0.3$ meters where the time t is in seconds.

The performance of each control method is evaluated by comparing the tracking accuracy of the end-effector. The tracking accuracy is evaluated by the *Root Square Mean Error* (RSME). The end-effector error is defined as

$$E_{xyz} = \sqrt{(e_x^2 + e_y^2 + e_z^2)}\,, \quad (22)$$

where e_x, e_y , and e_z are the position errors in *x*-, *y*-, and *z*-axis given in manipulator's workspace coordinates.

$$RSME = \sqrt{\frac{\sum E_{xyz}^2}{n}}\,, \quad (23)$$

where n is the number of the samples. The simulation is used to find a set of minimum proportional gain K_P and derivative gain K_D that minimized *RSME*. It must be considered that the actuators cannot generate forces larger than 120 Newtons.

The values of the physical kinematic and dynamic parameters of the CPM are given in table 1 and table 2.

Table 1
Kinematic parameters of the CPM

Parameters	L (m)	L_1 (m)	L_2 (m)	D (m)
Values	0.105	0.5	0.373	0.9144

Table 2
Dynamic parameters of the CPM

Parameters	m_1 (kg)	m_2 (kg)	m_3 (kg)	m_4 (kg)
Values	1.892994	0.695528	0.2	0.3

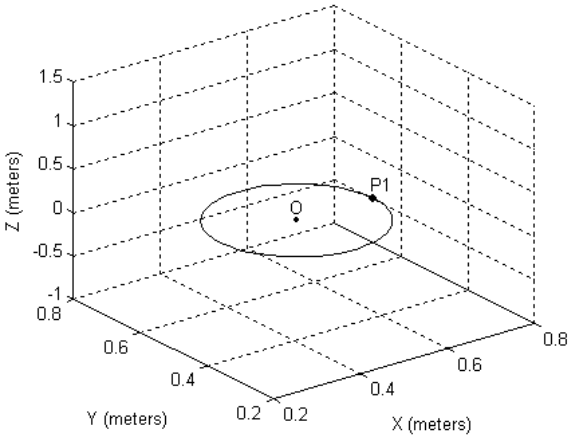

Fig. 4: End-effector path for the circular trajectory.

6. Simulation results

In this section, some results are presented for the four control algorithms implemented on the CPM. The simulation results are presented in table 3.

6.1. PD control with position and velocity reference

It was required that the robot achieved the desired trajectory with a position error less than $3\times10^{-3}m$ after 0.3 seconds. Although this controller is easy to implement and no knowledge of the system is needed to develop this type of controller, the tracking ability is very poor (especially along *z*-axis because of the limbs weight) compared to the rest of the controllers used in this paper. The position and velocity errors of the end-effector obtained from this controller are shown in figures 5 and 6. To improve the performance, the proportional gain K_P must be increased but it is impossible because of the limitation of the actuators.

6.2. PD Control with gravity compensation

It was required that the robot achieved the desired trajectory with a position error less than $3\times10^{-4}m$ after 0.3 seconds. The implementation of the PD controller with gravity compensation requires partial dynamic modeling information incorporated into the controller. The simulation results show a significant improvement in tracking ability from a simple PD controller (see figs. 7-8).

6.3. PD control with full dynamics feedforward terms

It was required that the robot achieved the desired trajectory with a position error less than $10^{-5}m$ after 0.3 seconds. The model based controllers such as this type and computed torque controller can generate force commands more intelligently and accurately than simple non-model based controllers. After 0.4 seconds, the position errors are approximately zeros but the velocity errors are approximately zeros after 0.3 seconds (see figs. 9 and 10).

6.4. Computed torque control

The initial conditions of the error and its derivative of our sample trajectory of the end-effector are $e(0)=[0\ \ 0\ \ 0]^t$, and $\dot{e}(0)=[0\ \ \dot{e}_0\ \ 0]^t$ then the solution of eq. (19) is:

$$e=\dot{e}_0 te^{-0.5K_D t}. \qquad (24)$$

Eq. (24) suggests that the derivative gain K_D should be a maximum value to achieve the desired critical damping but the actuator force cannot exceed more than 120 Newtons.

According to eq. (24), the position errors in *x*-, *z*-axis are zeros because the initial velocity errors in *x*-, *z*-axis are zeros. After 0.2 seconds, the position and velocity errors are approximately zeros (see figs. 11 and 12). The simulation results show that the computed torque controller gives the best performance. This is a result of the computed torques canceling the nonlinear components of the controlled system.

Table 3
The performance of various controllers

Controller	K_P	K_D	Position RSME	Velocity RSME
Pd control with position and velocity reference	12691	436	2.7×10^{-3}	0.0223
Pd control with gravity compensation	8507	436	3.4804×10^{-4}	0.021
Pd control with full dynamics feedforward	7053	436	3.0256×10^{-4}	0.0182
Computed torque control	2550.25	101	2.3469×10^{-4}	0.0161

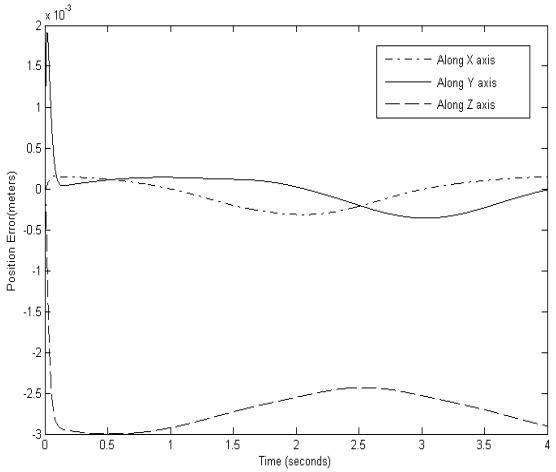

Fig. 5. Position error of the end-effector obtained from the simple PD controller.

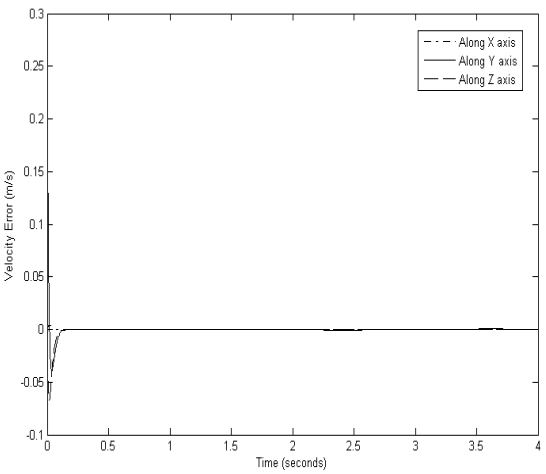

Fig. 6. Velocity error of the end-effector obtained from the simple PD controller.

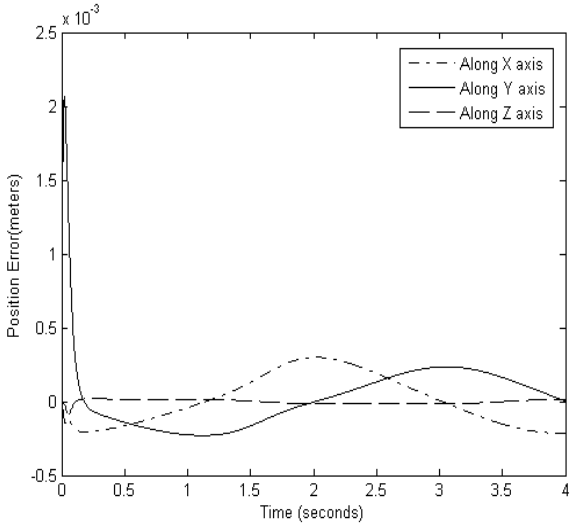

Fig. 7. Position error of the end-effector obtained from the PD controller with gravity compensation.

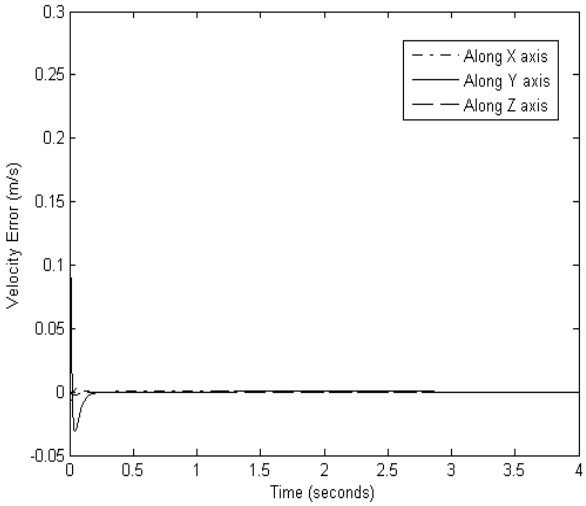

Fig. 8. Velocity error of the end-effector obtained from the PD controller with gravity compensation.

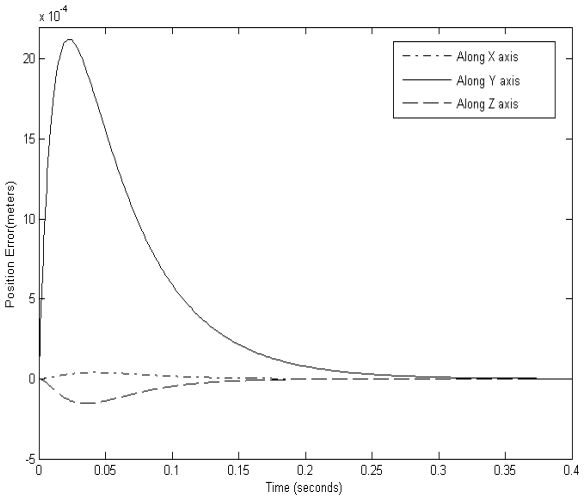

Fig. 9. Position error of the end-effector obtained from the PD controller with full dynamics feedforward terms within the first 0.4 seconds.

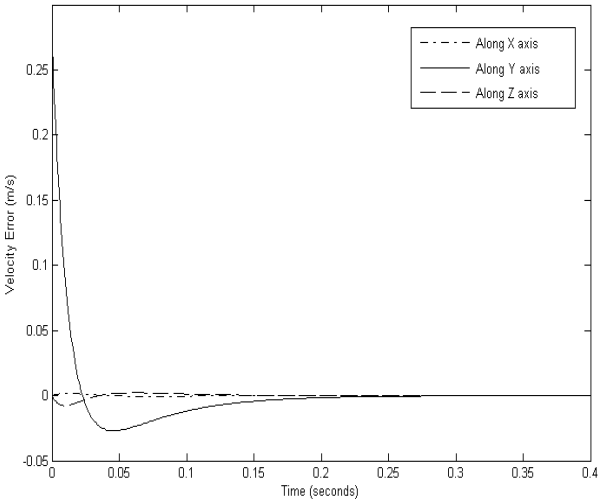

Fig. 10. Velocity error of the end-effector obtained from the PD controller with full dynamics feedforward terms within the first 0.4 Seconds.

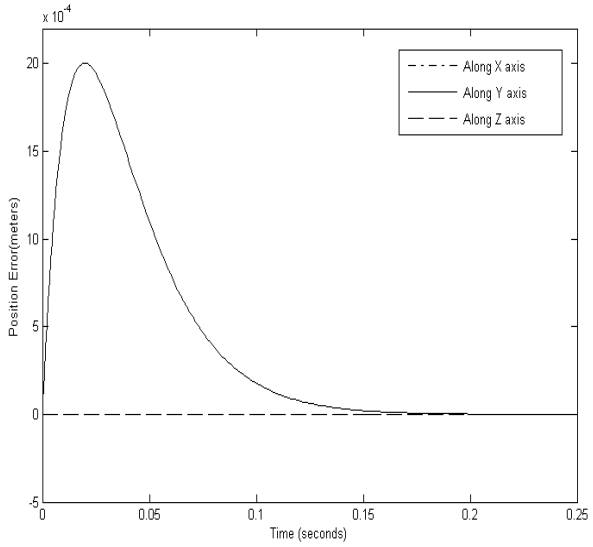

Fig. 11. Position error of the end-effector obtained from the computed torque controller within the first 0.25 seconds.

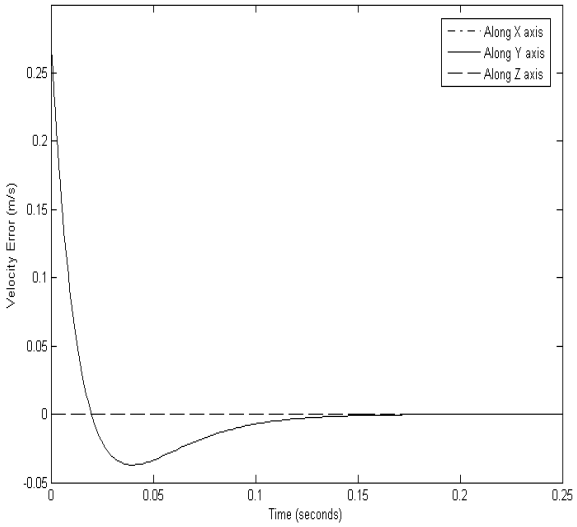

Fig. 12. Velocity error of the end-effector obtained from the computed torque controller within the first 0.25 seconds.

7. Conclusions

In this paper, using Lagrangian multiplier approach, a model for the dynamics of the manipulator is developed which has a form similar to that of a serial manipulator. Then we have presented four control algorithms on the CPM. The performance of these controllers are studied and compared.

As expected, complete mathematical modeling knowledge is needed to give the controller complete advantage in motion control. The model based control schemes perform better than non-model based controllers. Hence, the need for studying dynamics of robot manipulator as well as having a good understanding of various basic motion controller theories are important in

designing and controlling motion of the robot to achieve the highest quality and quantity of work.

Acknowldgement

The authors would like to thank Han Sung Kim for his valuable suggestions during this work.

References

[1] H.S. Kim and L. Tsai, "Design Optimization of a Cartesian Parallel Manipulator", Proceedings of DETC'02, ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference Montreal, Canada, September 29 - October 2 (2002).

[2] R.E. Stamper, "A Three Degree of Freedom Parallel Manipulator with Only Translational Degrees of Freedom", Dissertation submitted to the Faculty of the Graduate School of the University of Maryland at College Park for the degree of PhD (1997).

[3] F. Lewis, C. Abdallah and D. Dawson, "Control of Robot Manipulators", MacMillan Publishing Company (1993).

[4] L.W. Tsai, "Robot Analysis: the Mechanics of Serial and Parallel Manipulators", John Wiley and Sons (1999).

Received December 2, 2006
Accepted May 14, 2007