Al-Khwarizmi Engineering Journal, Vol. 15, No. 2, June , (2019)
P.P. 13- 23

Al-Khwarizmi
Engineering
Journal

Motion Control of Three Links Robot Manipulator (Open Chain) with Spherical Wrist

Hassan Mohammad Alwan* **Zaid Hikmat Rashid****
** Department of Mechanical Engineering / University of Technology /Baghdad-Iraq*
*** Department of Mechanical Techniques / Al-Mussaib Technical Institute / ATU/ Babylon /Iraq*
Email: 20071@uotechnology.edu.iq
Email: me.21329@uotechnology.edu.iq

(Received 30 September 2018; accepted 17 December 2018)
https://doi.org/10.22153/kej.2019.12.001

Abstract

Robot manipulator is a multi-input multi-output system with high complex nonlinear dynamics, requiring an advanced controller in order to track a specific trajectory. In this work, forward and inverse kinematics are presented based on Denavit Hartenberg notation to convert the end effector planned path from cartesian space to joint space and vice versa where a cubic spline interpolation is used for trajectory segments to ensure the continuity in velocity and acceleration. Also, the derived mathematical dynamic model is based on Eular Lagrange energy method to contain the effect of friction and disturbance torques beside the inertia and Coriolis effect. Two types of controller are applied ; the nonlinear computed torque control (CTC) and the simpler form of its Proportional Derivative plus Gravity (PD+G) where they are designed to reduce the tracking trajectory errors which tend to zero where the used Kp and Kv gains are 900,60. Also, the RMS errors for tracking a step input of CTC were equal to [2.5E-14, 4.4E-14, 5.0E-14, -4.7E-14, -3.9E-14, -4.6E-14] (deg) and of PD+G were equal to [-1.77E-5, -1.22E-6, -4.28E-6, -8.97E-6, -1.32E-5, 1.05E-5] (deg) for joints one to six, respectively. The results show that CTC is more accurate but requires additional acceleration input and is more computationally extensive and PD+G controller is performed with acceptable tracking errors in manipulator position control applications.

Keywords: *Dynamic, kinematic, Manipulator, nonlinear control, trajectory tracking.*

1. Introduction

The precise dynamic model of a robot is an important step to achieve high performance robot control [1]. The linear classical controllers are commonly used to control industrial robot but they had limitation because the uncertainty in dynamic model and not sufficient response in different wide ranges so there is a need to applied a nonlinear controller to achieve suitable trajectory tracking by linearizing the system and cancelling the nonlinearly in the dynamic model. H. AL-Qahtani et al [2] presented the dynamics and control of a robotic having four links where the manual tuning PID gains obtains a satisfactory manipulator response, while Wathik and Wael [3] presented the modeling and control of the LabVolt 5250 robot arm 5 DOF by assuming each link is a homogenous cylinder, F. Pitan et al [4] presented a CTC method using matlab Simulink for puma 560 robot manipulator position, based on the simplified mathematical dynamic model of Armstrong et al[5], D. Receanu [6] presented the basics of nonlinear CTC application for a two link planner robot where the PD gain are selected for best performance (critical damping), MATLAB Simulink was used to simulate an arbitrary circular path to examine the analysis, H. AL-Dois et al [7] presented a comprehensive study of dynamic performance of serial robot manipulator where they presented a numerical example for PUMA 560 as an illustrative case, the kinematic manipulability and dynamic manipulability as end effector performance are measured. J. Kim et al

[8] present the dynamic modelling and motion control for an arbitrary three links robot manipulator using the PD-CTC as commanded joints torques and extract the input signals by using image processing technique where a pinhole camera is used to capture the motion of a user's arm as reference angle inputs and simulate the tracking motion.

In this work; motion control of a three link robot arm with spherical wrist (six DOF) as shown in Figure (1) is done by two stage the first one is a trajectory planning by specifying the cartesian positon and orientation of the end effector and transform this via points to joint space by application of inverse dynamic also the cubic polynomials are generated for each interval of time by using cubic spline interpolation, the second stage is the trajectory tracking by applying two type of controller CTC and the simpler form of its PD +G using MATLAB Simulink where the resulted joint response was inverted to cartesian space again by the forward kinematic transformation matrix.

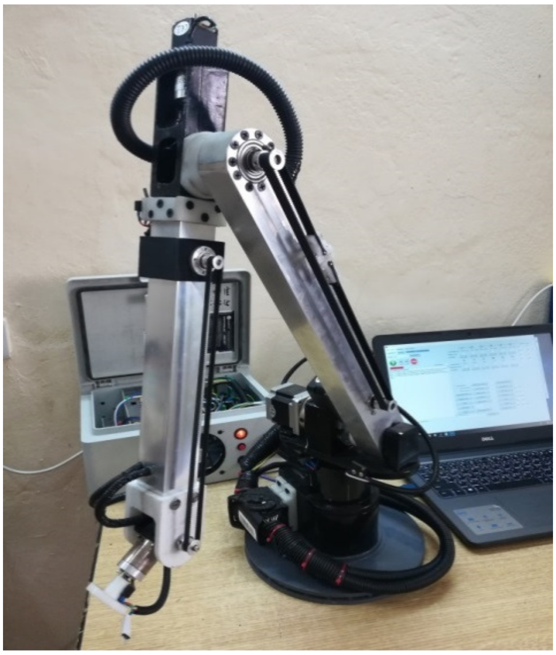


Fig. 1. Three Links Robot Manipulator 6 DOF.

The paper is organized sequentially as follow: manipulator forward and inverse kinematic, dynamic modelling, trajectory planning, controller design, result and discussion and conclusions.

2. Manipulator Forward and Invers kinematics

In this paper, Denavit Hartenberg (DH) convention is based to solve the forward kinematics of the manipulator by attaching coordinate frame system at each joint and specifying the four parameter of DH(α_i, a_i, θ_i and d_i)where: [9]

a_i: (link length) is the distance between z_{i-1} and z_i axes along the x_i axis.

α_i: (link twist) is the required rotation of z_{i-1} to z_i axes about the x_i axis.

d_i: (joint offset) is the distance between x_{i-1} and x_i axes along the z_{i-1} axis.

θ_i: (joint angle) is the required rotation of x_{i-1} to x_i axes about the z_{i-1}axis.

The transformation matrix of frame $\{i\}$ relative to previous frame {i-1} is:

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ...(1) $$

And the transformation matrix of nth coordinate frame to base coordinate frame is:

$$ {}^{0}_{n}T = {}^{0}_{1}T\,{}^{1}_{2}T \ldots {}^{n-1}_{n}T \quad ...(2) $$

DH parameters of the robot manipulator are defined according to the assigned frames that are shown in figure (2) and they are listed in table (1).

In other side the inverse kinematics is subdivided to end effector position problem and orientation problem [10], the position vector of transformation matrix from base to wrist center is:

$$ P_w = P_e - d_6 \{ r_{13} \quad r_{23} \quad r_{33} \}^T \quad ...(3) $$

Can be solved numerically by Newton – Raphson method:[11]

$$ q_{i+1} = q_i - J^{-1}(q_i) f(q_i) \quad ...(4) $$

Where:

i = arbitrary counter.

$$ q = [q_1 \quad q_2 \quad q_3]^T $$

$f(q) = [f_1(q) \quad f_2(q) \quad f_3(q)]^T$, Which can be taken from Pw vector in equation (3).

$$ J = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \frac{\partial f_1}{\partial q_2} & \frac{\partial f_1}{\partial q_3} \\ \frac{\partial f_2}{\partial q_1} & \frac{\partial f_2}{\partial q_2} & \frac{\partial f_2}{\partial q_3} \\ \frac{\partial f_3}{\partial q_1} & \frac{\partial f_3}{\partial q_2} & \frac{\partial f_3}{\partial q_3} \end{bmatrix} \quad [11] $$

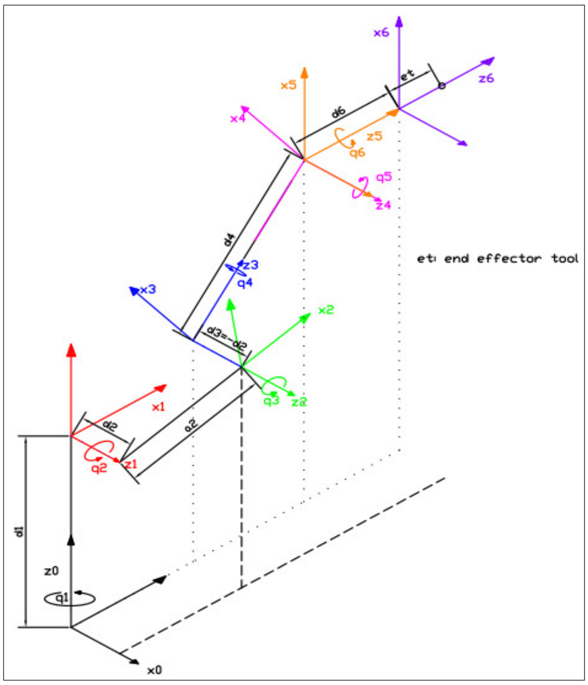


Fig. 2. The Attached Coordinate Frame Systems.

Table 1,
DH Parameters of the Robot Manipulator.

i	a_i(mm)	α_i(deg)	d_i(mm)	Θ_i
1	0	90	201	q_1
2	390	0	65	q_2
3	0	90	-65	q_3
4	0	-90	380	q_4
5	0	90	0	q_5
6	0	0	83	q_6

The orientation matrix of end effector may be calculated as:

$$ {}^3_6R = {}^0_3R^T\, {}^0_6R \qquad \text{...(5)} $$

And the solution is computed as ZYZ Euler angles when $\theta_5 \in [0, \pi]$as follow:[12]

$$ q_4 = a\tan 2(r_{23}, r_{13}) \qquad \text{...(6)} $$

$$ q_5 = a\tan 2(\sqrt{r_{13}^2 + r_{23}^2}, r_{33}) \qquad \text{...(7)} $$

$$ q_6 = a\tan 2(r_{32}, -r_{31}) \qquad \text{...(8)} $$

3. Manipulator Dynamics Modelling

By Eular - Lagrange formulation the equations of motion can be derived as:

$$ \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \zeta_i \qquad i=1,\ldots,6 \qquad \text{...(9)} $$

Where:

L = K- U

The kinetic energy of ith link and of ith motor is:

$$ K_{i(\ell)} = \frac{1}{2} v_{i(\ell)}^T m_{i(\ell)} v_{i(\ell)} + \frac{1}{2}\omega_{i(\ell)}^T\, {}^0I_{i(\ell)}\omega_{i(\ell)} \qquad \text{...(10)} $$

$$ K_{i(m)} = \frac{1}{2} v_{i(m)}^T m_{i(m)} v_{i(m)} + \frac{1}{2}\omega_{i(m)}^T\, {}^0I_{i(m)}\omega_{i(m)} \qquad \text{...(11)} $$

Where:

v and ω: linear and angular velocities of mass center respectively.

${}^0I$: mass moment in base frame coordinates.

Since:

$$ {}^0v_i = J_p\dot{q}_i \qquad \text{...(12)} $$

$$ {}^0\omega_i = J_o\dot{q}_i \qquad \text{...(13)} $$

And the rotation of principle inertia axes of ith link is:

$$ {}^0I_i = {}^0_iR\; {}^iI_i\; {}^0_iR^T \qquad \text{...(14)} $$

As shown in figure (3), the total kinetic energy for the assigned robot manipulator is:

$$ K = \sum_{i=1}^{3} K_{i(\ell)} + \sum_{i=2}^{6} K_{i(m)} \qquad \text{...(15)} $$

Substituting equations (12-14) in (15):

$$ K = \frac{1}{2}\sum_{i=1}^{3} (J_{p(i)}\dot{q}_i)^T m_{i(\ell)} (J_{p(i)}\dot{q}_i) + $$
$$ \frac{1}{2}\sum_{i=1}^{3} (J_{o(i)}\dot{q}_i)^T ({}^0_iR\,{}^iI_{i(\ell)}\,{}^0_iR^T)(J_{o(i)}\dot{q}_i) $$
$$ +\frac{1}{2}\sum_{i=2}^{6} (J_{p(i)}\dot{q}_i)^T m_{i(m)} (J_{p(i)}\dot{q}_i) + $$
$$ \frac{1}{2}\sum_{i=2}^{6} (J_{o(i)}\dot{q}_i)^T ({}^0_iR\,{}^iI_{i(m)}\,{}^0_iR^T)(J_{o(i)}\dot{q}_i) \qquad \text{...(16)} $$

$$ K = \frac{1}{2}\dot{q}^T B(q)\dot{q} = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}(q)\;\dot{q}_i\dot{q}_j \qquad ,n=6 \qquad \text{...(17)} $$

Where (in matrix form):

$$ B(q) = \sum_{i=1}^{3} (J_{p\ell(i)}^T m_i J_{p\ell(i)} + J_{o\ell(i)}^T\, {}^0_iR\, {}^iI_{\ell(i)}\, {}^0_iR^T J_{o\ell(i)}) $$
$$ + \sum_{i=2}^{6} (J_{pm(i)}^T m_i J_{pm(i)} + J_{om(i)}^T\, {}^0_iR\, {}^iI_{m(i)}\, {}^0_iR^T J_{om(i)}) \qquad \text{...(18)} $$

And the potential energy of ith link and ith actuator is:

$$ U_{i(\ell)} = -m_{i(\ell)}\, {}^0g^T\, {}^0_ir_{(\ell)} \qquad \text{...(19)} $$

$$ U_{i(m)} = -m_{i(m)}\, {}^0g^T\, {}^0_ir_{(m)} \qquad \text{...(20)} $$

Where:

g: gravity acceleration vector in base coordinate frame which is $[0 \quad 0 \quad -9.81]^T$

${}^0r_i$: The distance of ith link mass center to base coordinates frame, i.e

$$ {}^0r_i = {}^0_iT\, {}^ir_i \qquad \text{...(21)} $$

And the total potential energy of the robot manipulator:

$$U = \sum_{1}^{3} U_{i(\ell)} + \sum_{2}^{6} U_{i(m)} \qquad ...(22)$$

By noticing that U doesn't depend on $\dot{q}$ then equation (9) can be written as:

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_i}\right) - \frac{\partial K}{\partial q_i} + \frac{\partial U}{\partial q_i} = \zeta_i \qquad ...(23)$$

By substituting equations (17) and (22) into equation (23), getting that:

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_i}\right) = \sum_{j=1}^{n} b_{ij}(q)\ddot{q}_j + \sum_{j=1}^{n} \frac{db_{ij}(q)}{dt}\dot{q}_j \qquad ...(24)$$

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_i}\right) = \sum_{j=1}^{n} b_{ij}(q)\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial b_{ij}(q)}{\partial q_k}\dot{q}_k\dot{q}_j \qquad ...(25)$$

Also

$$\frac{\partial K}{\partial q_i} = \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial b_{jk}(q)}{\partial q_i}\dot{q}_k\dot{q}_j \qquad ...(26)$$

And

$$\frac{\partial U}{\partial q_i} = -\left(\sum_{i=1}^{3} m_{i(\ell)}\,{}^0g^T \frac{\partial\, {}^0r_{i(\ell)}}{\partial q_i} + \sum_{i=2}^{6} m_{i(m)}\,{}^0g^T \frac{\partial\, {}^0r_{i(m)}}{\partial q_i}\right) \qquad ...(27)$$

$$\frac{\partial U}{\partial q_i} = -\left(\sum_{i=1}^{3} m_{i(\ell)}\,{}^0g^T J_{p(\ell i)}(q) + \sum_{i=2}^{6} m_{i(m)}\,{}^0g^T J_{p(mi)}(q)\right) = G_i(q) \qquad ...(28)$$

As a result the equations of motion are:

$$\sum_{j=1}^{n} b_{ij}(q)\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}(q)\dot{q}_k\dot{q}_j + G_i(q) = \zeta_i$$

i=1,2,..,6 ...(29)

Where:

$$h_{ijk} = \frac{\partial b_{ij}}{\partial q_k} - \frac{1}{2}\frac{\partial b_{jk}}{\partial q_i}$$

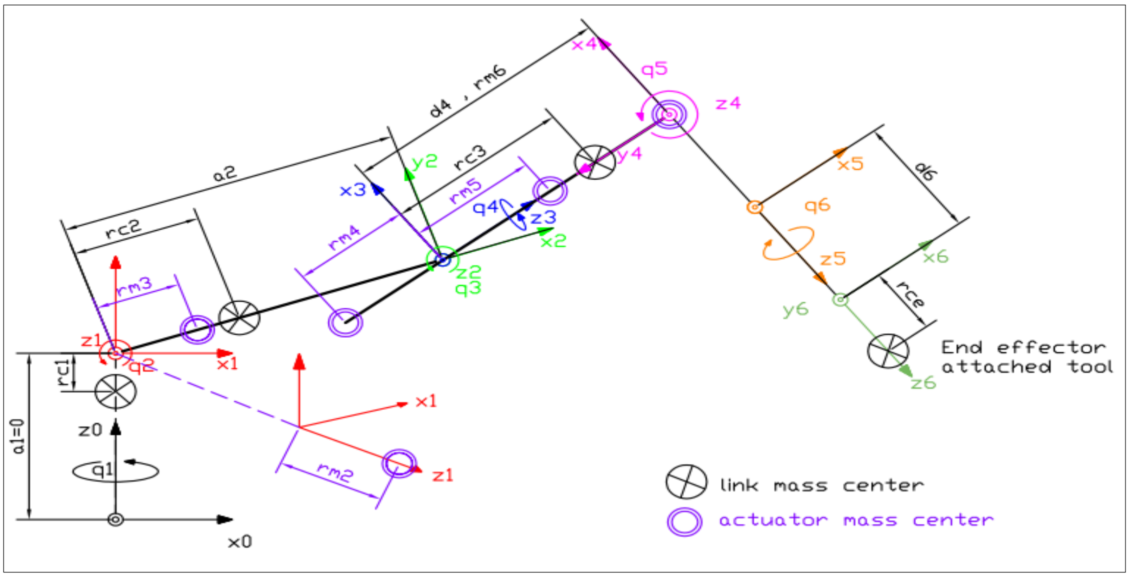


Fig. 3. Schematic Diagram of Dynamic Model.

And the final matrix form of equations can be written as:[13]

$$B(q)\{\ddot{q}\} + C(q,\dot{q})\{\dot{q}\} + F_v\{\dot{q}\} + F_s\{\operatorname{sgn}(\dot{q})\}$$
$$+G^T(q) + J_e^T F_e = \tau \qquad ...(30)$$

Where:

$B(q)$: Inertia matrix symmetric (6x6).

$C(q,\dot{q})$: Centrifugal and coriolies forces matrix (6x6).

$$c_{ij} = \sum_{i=1}^{k} h_{ijk}\dot{q}_k$$

$$h_{ijk} = \frac{1}{2}\left(\frac{\partial b_{ij}}{\partial q_k} + \frac{\partial b_{ik}}{\partial q_j} - \frac{\partial b_{jk}}{\partial q_i}\right)$$ Christoffel symbol first type[14]

F_v : Viscous friction coefficients, diagonal matrix (6x6).

F_s : Coulomb friction (static torque friction), diagonal matrix (6x6).

$\operatorname{sgn}(\dot{q})$: vector of sign function of joints velocities (6x1).

G: Gravitational torques matrix (1x6).

g: gravity acceleration vector in base coordinate frame which is [0 0 -9.81]^T.

F_e: Vector of forces and moments exerted by the end effector on the environment (*6*x*1*).

τ: Actuator torque vector (*6*x*1*).

An inverse dynamic method is applied to find the dynamic response of joints variables $q, \dot{q}$.

From equation (30):

$$B(q)\{\ddot{q}\}+N=\tau \qquad \text{...(31)}$$

Where

$$N=C(q,\dot{q})\{\dot{q}\}+F_V\{\dot{q}\}+F_S\{\text{sgn}(\dot{q})\}+G^T(q)+J_e^T F_e$$

And

$$\ddot{q}=B^{-1}(\tau-N) \qquad \text{...(32)}$$

In order to solve the arm dynamic, equation (32) may be converted to ordinary differential equation system (ODEs). So the state space representation of the robot manipulator (MIMO system like) can be written as:[15]

$$\{\dot{x}\}=\begin{bmatrix}[0] & [I]\\ [0] & [0]\end{bmatrix}_{2nx2n}\begin{Bmatrix}q\\ \dot{q}\end{Bmatrix}_{2nx1}+\begin{bmatrix}[0]\\ [I]\end{bmatrix}_{2nxn}\{M^{-1}(\tau-N)\}_{nx1} \qquad \text{...(33)}$$

$$y=[I]_{2nx2n}\{x\} \qquad \text{...(34)}$$

Where:

$$\{x\}=\{q,\dot{q}\}_{2nx1}^T$$

4. Trajectory Planning

The time behavior of a pre-planned path called trajectory planning, a cubic polynomial is the lowest order that can be used to satisfy the position and velocity constrains at the initial and final joint points. where each joint starts and ends its motion at the same time, each via points of a divided path segments are converted into a set of desired joints angles by application of the inverse kinematics where cubic spline interpolation schemes is used to compute the functions for each interval as shown in Figure (4).

Since each pair of knots is connected by cubic function so the second derivative within each interval is straight line and equal zero for ends knots (conditions), the function of interior interval can be written as: [12]

$$\Pi_i(t)=\frac{\ddot{\Pi}(t_{i-1})}{6(t_i-t_{i-1})}(t_i-t)^3+\left(\frac{\Pi(t_{i-1})}{t_i-t_{i-1}}-\frac{\ddot{\Pi}(t_{i-1})(t_i-t_{i-1})}{6}\right)(t_i-t)$$
$$+\frac{\ddot{\Pi}(t_i)}{6(t_i-t_{i-1})}(t-t_{i-1})^3+\left(\frac{\Pi(t_i)}{t_i-t_{i-1}}-\frac{\ddot{\Pi}(t_i)(t_i-t_{i-1})}{6}\right)(t-t_{i-1}) \qquad \text{...(35)}$$

Where $\ddot{\Pi}$ can be evaluated from equation (16) which is based on the fact that first derivative at knots must be continuous.

$$(t_i-t_{i-1})\ddot{\Pi}(t_{i-1})+2(t_{i+1}-t_{i-1})\ddot{\Pi}(t)+$$
$$(t_{i+1}-t_i)\ddot{\Pi}(t_{i+1})=\frac{6}{(t_{i+1}-t_i)}(\ddot{\Pi}(t_{i+1})-\ddot{\Pi}(t_i))$$
$$\frac{6}{(t_i-t_{i-1})}(\ddot{\Pi}(t_{i-1})-\ddot{\Pi}(t_i)) \qquad \text{...(36)}$$

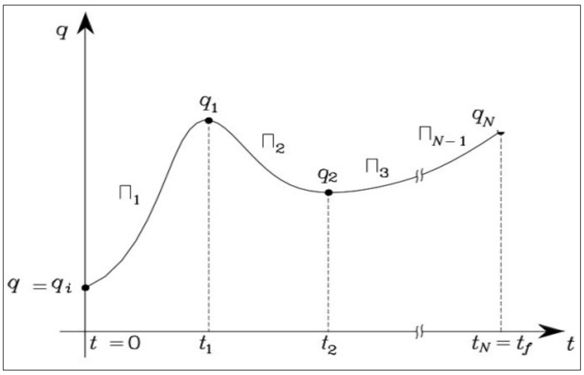


Fig. 4. Cubic Spline Interpolated Trajectory [12]**.**

5. Controller Design

The system is termed as an open loop system by applying a feed forward torque and closed loop system by applying feedback control torques as illustrated in figure (5).

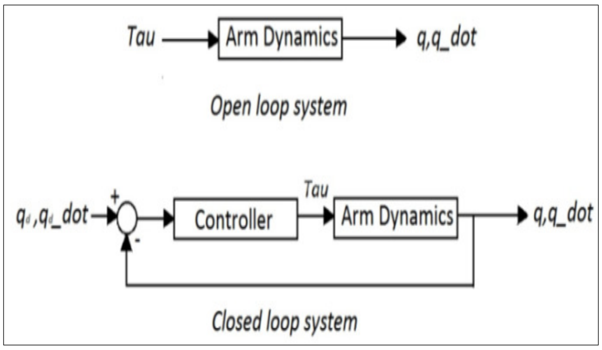


Fig. 5. Control System I/O.

If and only if the dynamic model of the robot is exact to real and there be no initial errors then the joint torques can be computed as eq. (31) and the open loop controller satisfy the trajectory tracking.

Since is too difficult to cover the exact dynamic model then the closed loop controller ensure to vanishing the dynamics errors.

To track trajectory, the joint variable error may defined as:

$$e=q_{des}-q \qquad \text{...(37)}$$

Also The computed torque command low is:[15]

$$\tau = B(\ddot{q}_d - u) + N \qquad ...(38)$$

Where u is the auxiliary control input.
By letting

$$u = -K_p e - K_v \dot{e} \qquad ...(39)$$

Where:

$$K_p = diag\{k_{p1} \quad ... \quad k_{p6}\}$$

$$K_v = diag\{k_{v1} \quad ... \quad k_{v6}\}$$

The contribution of equations (37-39) with equation (31) leads to second order differential equation of error dynamic:

$$\ddot{e} + K_p\ \dot{e} + K_v\ e = 0 \qquad ...(40)$$

This is equivalent to the characteristics equation:

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0 \qquad ...(41)$$

Where ω_n is the natural frequency and ζ is the damping ratio, the equation has three solutions. The critical damping solution represents the desired performance of the robot i.e the PD gains may be as ($Kp = \omega_n^2$) and ($Kv = 2\,\omega_n$) where ω_n is higher as possible to increase the robot response. [15]

The overall computed torque input become as shown in figure (6):

$$\tau = B(\ddot{q}_d + K_p e + K_v \dot{e}) + N \qquad ...(42)$$

or

$$\tau = [B(\ddot{q}_d) + N] + [B(K_p e + K_v \dot{e})] \qquad ...(43)$$

The 1st termed feed forward component and the 2nd termed feedback component.

In this controller, the manipulator track a desired trajectory $q_d, \dot{q}_d$ & $\ddot{q}_d$ which are additional computation of inverse kinematics, inverse jacobian and derivatives, thus a simpler controller PD plus gravity may use especially for tracking a continuous path with via points were it is a classical PD added by the most effective term the gravity torques .

$$\tau = K_p e + K_v \dot{e} + G \qquad ...(44)$$

6. Results and Discussions

Matlab m-files and Matlab Simulink is used to embrace the theoretical work as shown in figure (7)

Figures (8 & 9) show the dynamic response from rest to rest of a step input with zero friction and external forces where the PD gains are subjected on manual tuning with value ω_n = 30 rad/s, the settling errors for joints one to six respectively are [2.5E-14, 4.4E-14, 5.0E-14, -4.7E-14, -3.9E-14, -4.6E-14] (deg) which may be considered as zero errors. Also figure (10) shows the computed torques which start with relative high due to high error and settling on [-4.22E-12, -2.74, -4.33, -1.56E-3, -1.58, 4.19E-2] (N.m), that show the maximum torque at joint 2 and 3 because the gravity play roll in increasing the holding torques in vertical plane.

Figure (11) shows the dynamic response of step input using the simpler PD +Gravity controller were the RMS of errors were [-1.77E-5, -1.22E-6, -4.28E-6, -8.97E-6, -1.32E-5, 1.05E-5] (deg). The tracking results were satisfactory relative to the proposed model, also PD +Gravity easier in application since there is no need to feed a reference inputs of velocity and acceleration.

Figure (12) shows the joint space tracking trajectory of a pick - place planned path using CTC and the tracking RMS errors [4.04E-2, 5.4E-3, 1.3E-3, 3.1E-14, 1.9E-2, 3.06E-1] (deg), which are acceptable but relative high because the max peek of error for all is high at 0.026 s That because $\ddot{q}_{des}$ represent the 2nd order derivative of q_{des} is too high at starting that lead to high input commanded torque as shown in figures (13,14) (for joint six: 300 rad/s^2 and max error 2.65 (deg)).

While the RMS errors of PD+G controller were [2.94E-4, 4.17E-3, 1.60E-3, 2.43E-4, 1.32E-3, 1.07E-4] (deg) ,therefore, for pick and place the PD+G perform better than the CTC in tracking trajectory with absence of disturbance.

A simscape multibody simulation were developed in Matlab can be watched on link https://drive.google.com/open?id=15J3Ph3Yy_gB2LIuYBLz0Lq1Ipo6ual44.

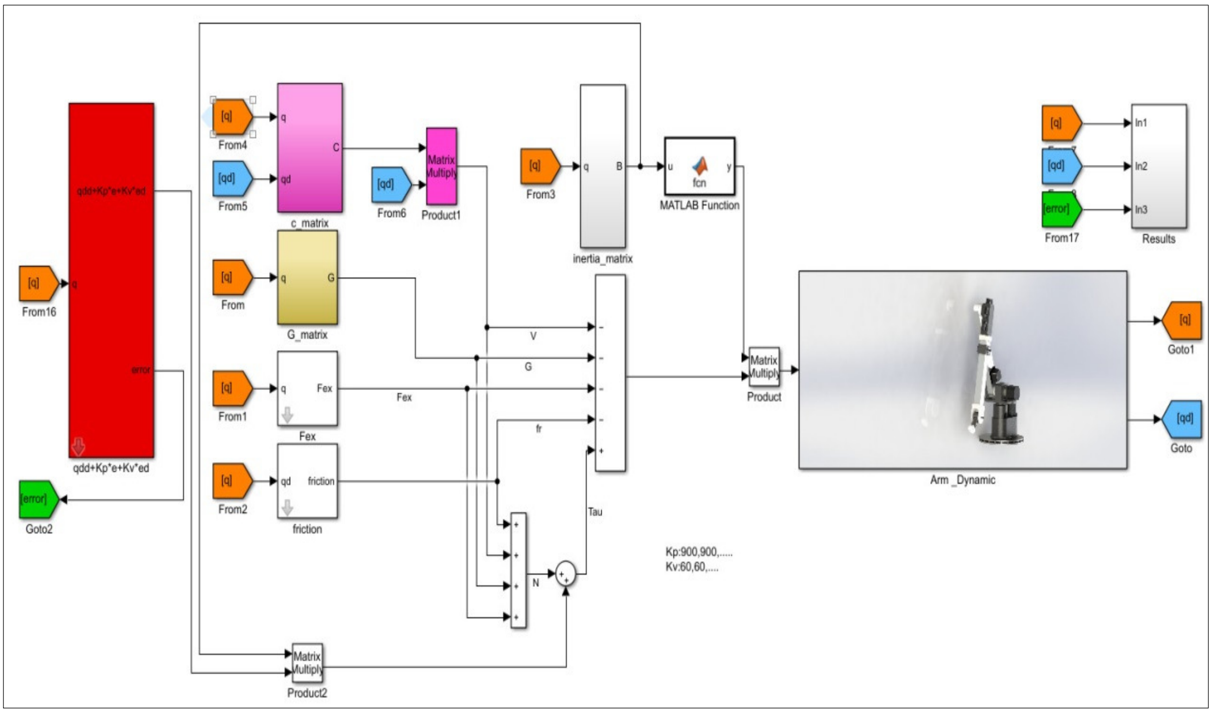


Fig. 7. Matlab Simulink System.

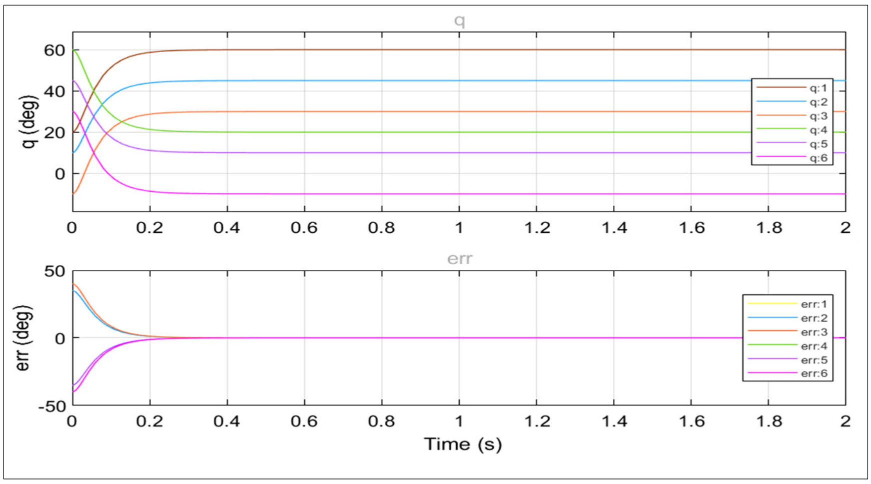


Fig. 8. Joints Trajectory of Step Input /Non-Linear Controller.

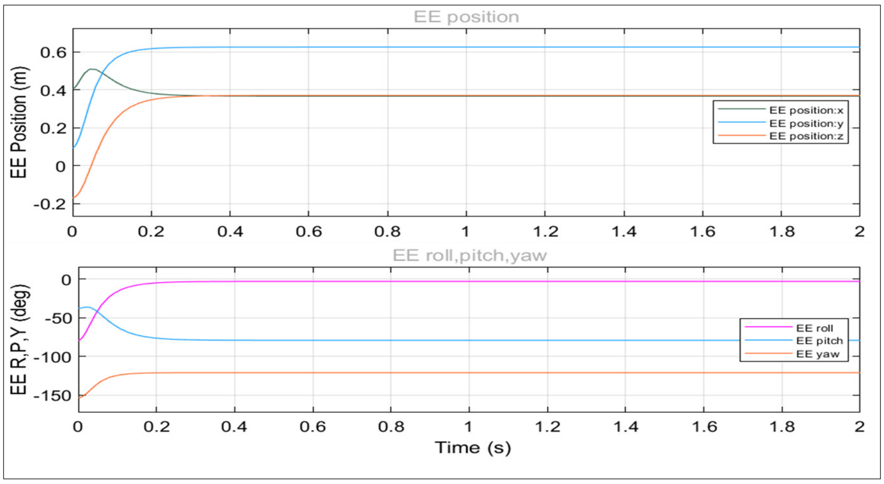


Fig. 9. End Effector Cartesian Position and Orientation.

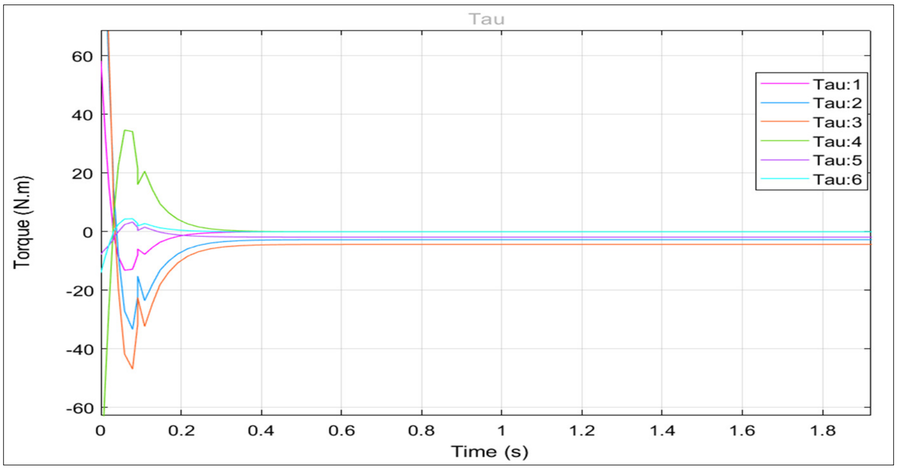


Fig. 10. Joints Torques of Step Input.

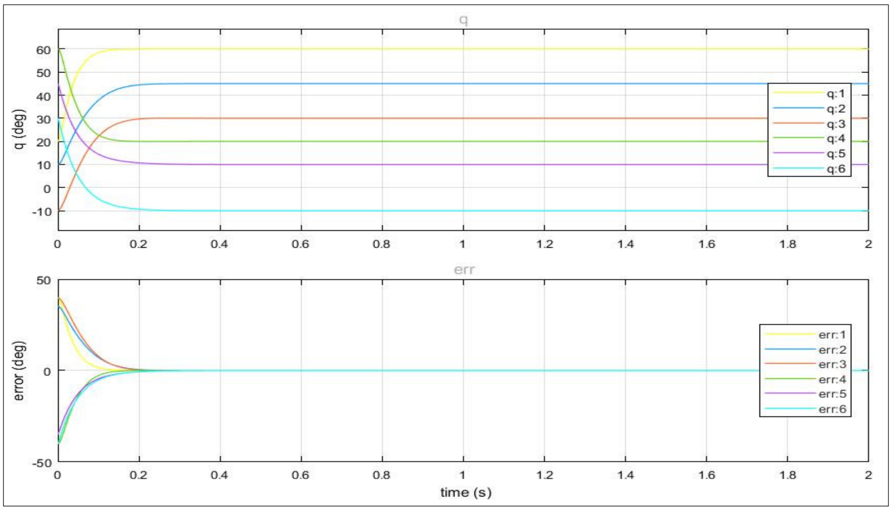


Fig. 11. Joints Trajectory of Step Input / PD+G Controller.

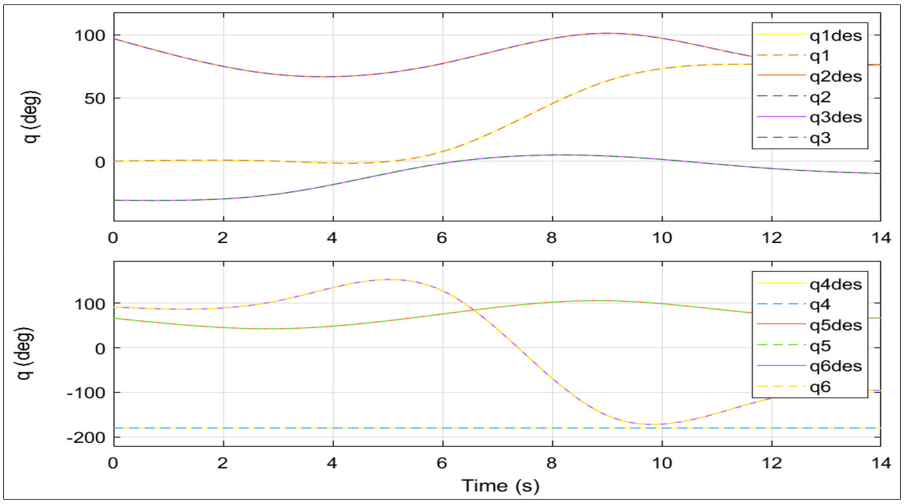


Fig. 12. Joint Space Tracking Trajectory/ PD-CTC.

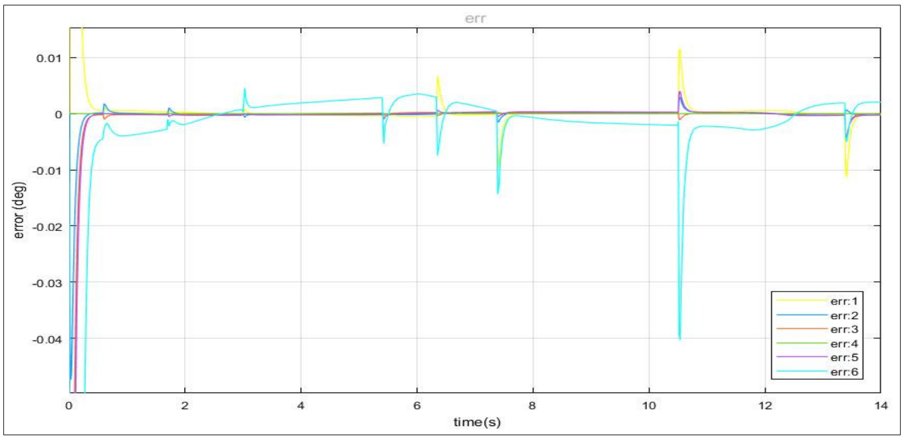
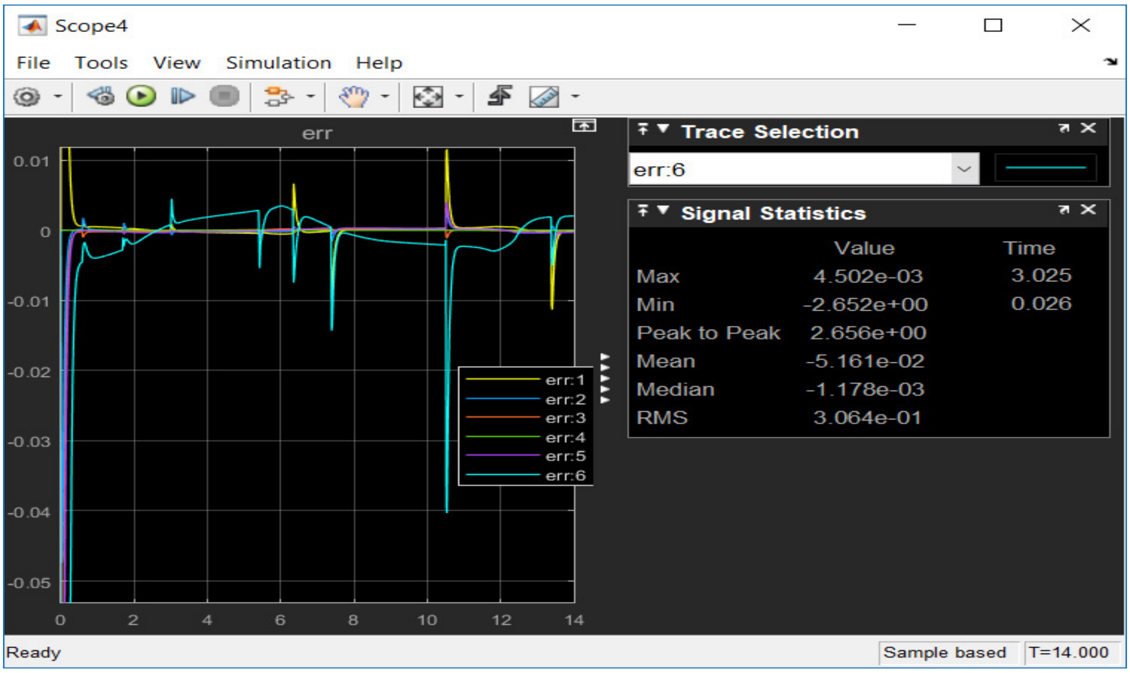


Fig. 13. Tracking Error/ PD-CTC.



err:6

	Value	Time
Max	4.502e-03	3.025
Min	-2.652e+00	0.026
Peak to Peak	2.656e+00	
Mean	-5.161e-02	
Median	-1.178e-03	
RMS	3.064e-01	

Fig. 14. Error Statistics/ PD-CTC.

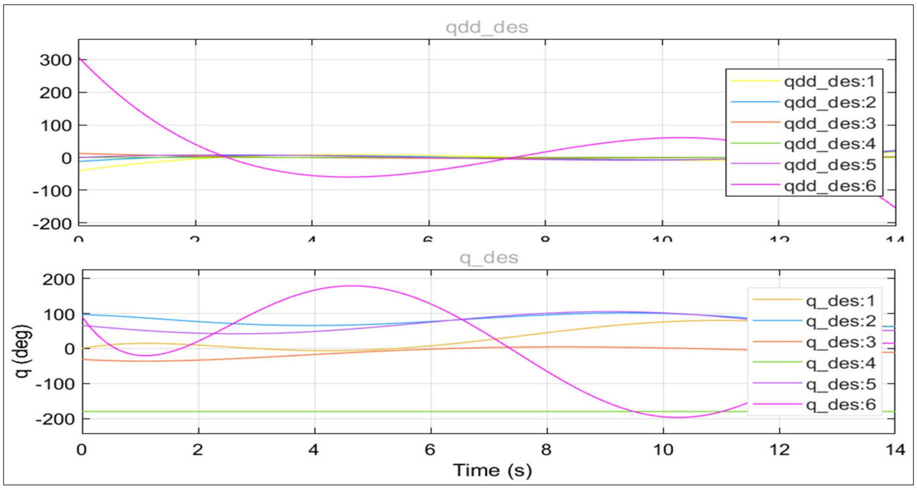


Fig. 15. Desired Joint and Acceleration/ reference inputs.

7. Conclusions

From the control point of view, the computed torque control technique guaranteed to reduce the tracking error tends to zero through linearizing the system, and specifying the desired trajectory $q_{des}, \dot{q}_{des}$ & $\ddot{q}_{des}$. The results show that CTC more accurate but required additional acceleration input and computational extensive while PD+G controller is perform with acceptable tracking errors in manipulator position control applications like pick and place. So the PD+G can be used to track a continuous path with via points with acceptable errors and satisfactory robot performance.

Nomenclatures

b_{ij}	Element of inertia matrix.
c_{ij}	Element of centrifugal and coriolies matrix.
e	Error.
J	Jacobian matrix.
J_p	Linear velocity jacobian matrix.
J_o	Angular velocity jacobian matrix.
K	kinetic energy.
m	Mass.
P	Position vector.
q_i	Joint variable.
R	Rotation matrix (3x3).
r_{ij}	Element of rotation matrix.
T	Transformation matrix (4x4).
U	Potential energy.
ζ_i	Generalized moment.

Sub Script

A^T	Transpose matrix.
$A_{\ell(i)}$	Refer to ith link.
$A_{m(i)}$	Refer to ith motor.
${}^iA$	Refer to coordinate frame that an item is described in.
A_e	Refer to end effector.
A_w	Refer to wrist point.
A_{des}	Desired.

8. References

[1] K. M. Lynch and F. C. Park, Introduction To Modern Robotics Mechanics, Planning, and Control, no. May. 2016.

[2] H. M. Al-Qahtani, A. A. Mohammed, and M. Sunar, “Dynamics and Control of a Robotic Arm Having Four Links,” Arab. J. Sci. Eng., vol. 42, no. 5, pp. 1841–1852, 2017.

[3] M. W. M. H. Hadi and D. R. W. I. M. AL-Tameemi, “Modeling and Control of 5250 Lab-Volt 5 DoF Robot Manipulator,” IRAQI J. Comput. Commun. Control Syst. Eng., vol. 15, no. 2, pp. 34–46, 2015.

[4] F. Piltan and M. Yarmahmoudi, “PUMA-560 robot manipulator position computed torque control methods using MATLAB/SIMULINK and their integration into graduate nonlinear control and MATLAB courses,” Int. J. Robot. Autom., vol. 3, no. 3, pp. 167–191, 2012.

[5] B. Armstrong, O. Khatib, and J. Burdick, “The explicit dynamic model and inertial parameters of the PUMA 560 arm,” in Robotics and Automation. Proceedings. 1986 IEEE International Conference on, 1986, vol. 3, pp. 510–518.

[6] D. Receanu, “Modeling and Simulation of the Nonlinear Computed Torque Control in Simulink / MATLAB for an Industrial Robot,” vol. 10, no. 2, pp. 95–106, 2013.

[7] A. K. J. and R. B. M. hatem al-dois, “dynamic analysis of serial robot manipulators,” IJAMS, vol. 15, no. 2, 2015.

[8] J. Kim, K. Chang, B. Schwarz, A. S. Lee, S. A. Gadsden, and M. Al-Shabi, “Dynamic Model and Motion Control of a Robotic Manipulator.,” JRNAL, vol. 4, no. 2, pp. 138–141, 2017.

[9] M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot modeling and control, vol. 3. Wiley New York, 2006.

[10] “Hassan Mohammad Alwan and Zaid Hikmat Rashid, 2018. Kinematic Analysis and Simulation of Three Link (Open Chain) Robot Manipulator with Six DOF. Journal of Engineering and Applied Sciences, 13: 1829-1834.” .

[11] R. N. Jazar, Theory of applied robotics: kinematics, dynamics, and control. Springer Science & Business Media, 2010.

[12] L. Sciavicco and B. Siciliano, Modelling and control of robot manipulators. Springer Science & Business Media, 2012.

[13] B. Siciliano and O. Khatib, Springer handbook of robotics. Springer, 2016.

[14] R. M. Murray, A mathematical introduction to robotic manipulation. CRC press, 2017.

[15] C. T. Lewis, Frank L.; Dawson, Darren M.; Abdallah, Manipulator Control Theory and Practice. 2004.

السيطرة على حركة روبوت ذراع مناور ذي ثلاثة روابط (النوع المفتوح) مع معصم كروي

حسن محمد علوان *   زيد حكمت رشيد**

**قسم الهندسة الميكانيكية / الجامعة التكنولوجيا / بغداد/ العراق*

***قسم التقنيات الميكانيكية / المعهد التقني المسيب/ جامعة الفرات الاوسط التقنية / بابل / العراق*

البريد الالكتروني: 20071@uotechnology.edu.iq

لبريد الالكتروني: me.21329@uotechnology.edu.iq

الخلاصة

الاذرع المناورة هي نظام متعدد المدخلات و المخرجات ذو دينامية غير خطية معقدة يحتاج الى مسيطر متقدم من اجل تعقب مسار محدد. في هذا البحث، الكينماتك الامامي و العكسي تم حسابه بالاعتماد على طريقة دانفينت – هارتنبيرغ من اجل تحويل احداثيات النهايه المؤثرة لمسار مخطط من الفضاء الكارتيزي الى فضاء المفاصل و العكس صحيح حيث تم استخدام الاستكما□ المكعب لاجزاء المسار لضمان الاستمرارية في السرعة و التعجيل. كذلك الموديل الديناميكي المشتق يعتمد على طريقة الطاقة اويلر – لاكرانج لتتضمن تاثير عزوم الاحتكاك و الاضطراب بجانب تأثير الزخم و الكوريوليس. و تم تطبيق نوعين من المسيطر: المسيطر ذي العزم المحسوب غير الخطي و المسيطر الابسط له نسبي مشتق زائد الجاذبية لتقليل اخطاء التعقب للمسار الى الصفر حيث كانت قيم Kpو Kv هي ٩٠٠ و ٦٠ و معد□ الجذر التربيعي للاخطاء لمدخل الخطوة نوع المسيطر الاو□ (2.5e-14 و 4.4e-14 و 5.0e-14 و -4.7e-14 و -3.9e-14 و-4.6e-14) درجة و للمسيطر الثاني (-1.77e-5 و -1.22e-6 و-4.28e-6 و -8.97e-6 و -1.32e-5 و1.05e-5) درجة للمفاصل من واحد الى سته تعاقبياً. أظهرت النتائج ان المسيطر الاو□ اكثر دقة و لكن يتطلب مدخلات اكثر و عمليات حسابية اكثر بينما المسيطر الثاني يؤدي بقيم تعقب مقبولة للخطأ لا سيَما للتطبيقات التي تهتم بالموقع.