International Journal of Industrial Engineering and Management (IJIEM), Vol.1 No. 1, 2010, pp. 19 - 27
Available online at http://www.ftn.uns.ac.rs/ijiem
ISSN 2217-2661

PID Controller Design for A Nonlinear Motion Control Based on Modelling the Dynamics of Adept 550 Robot

Henry Zhang
Department of Mechanical Engineering Technology, Purdue University, Knoy Hall, Room 119, 401 North Grant Street, West Lafayette, IN 47907, US, hhzhang@purdue.edu

Received (21.05.2010); Revised (11.06.2010); Accepted (12.06.2010)

Abstract

The multiple link robot is a nonlinear system, in general whose characteristic polynomial is not unique, and whose control schemes depend on particular operating conditions. Among all kinds of controllers used for robot control, nearly 90% is PID controller in the industrial world, not only due to its simplicity and effectiveness, but also to its ability of coping with uncertainties and nonlinearities in the system. Since most of PID controllers are designed based on trial and error method for a black-box plant, for the precision motion control of a robot, it is more desirable to design the PID controller based on the mathematical model of the robot. This paper is dedicated to a PID controller design for a nonlinear motion control based on the mathematical modelling of the dynamics of Adept 550 Robot. The general relationships of the PID controller design on the robotic dynamics and the planned trajectory are derived. The analysis and simulations of its closed loop dynamics indicates its effectiveness in fast and accurate trajectory tracking. The results can be practically generalized to other cases of PID controller design for other robots in the industrial applications.

Key words: *nonlinear motion control, robot, PID controller*

1. INTRODUCTION

The multiple link robot is a nonlinear system, in general whose characteristic polynomial is not unique, and whose control schemes depend on particular operating conditions. Among all kinds of controllers used for robot control, nearly 90% is PID controller in the industrial world, not only due to its simplicity and effectiveness, but also to its universality and feasibility. PID controller can be designed according to many PID tuning methods, even if the mathematical model of the system is not available, or uncertainties and nonlinearities were found in the system.

PID controller has fairly easily understood physical meanings: the proportional control provides a restoring force that is proportional to the instantaneous error between the set point and the process variable at the moment, but it may end up with steady-state error due to friction, gravity or so; the integral control contributes a corrective force that is proportional to the sum of all past errors multiplied by their own time interval, so that the steady-state error can be eliminated, but it may cause overshoot of the system response; derivative control is introduced to address the overshoot issue by producing a counteractive force that is proportional to the rate of change of error.

With these understandings, often engineers who do not have a systematically training of control theory can design a PID controller for the industrial process whose transfer functions are unknown to them. That is why most of PID controllers are designed based on trial and error method for a black-box plant.

However, for the precision motion control of a robot, the trial and error approach is not an efficient way. In recent years, more and more researchers aim to design different types of PID controllers based on theoretical analysis to deal with nonlinearity or non-explicates for different applications, such as indoor blimp robot, gyro mirror line-of-sight system, two wheeled autonomous balancing robot, parallel manipulator, etc. [1, 2, 3, 4, 5, 6, 7, 8]. It is demonstrated that the theoretically designed PID controllers can be optimally tuned to fit the needs of a specific systems or a generic task, for better performance than that the trial-and-error learning designed PID controller has.

This paper is dedicated to explore a theoretically designed PID controller for a nonlinear motion control on the basis of the mathematical model of Adept 550 robot. As one of the most commonly used robots in the industrial production lines, the Adept 550 robot is a four-axis SCARA robot with three rotational joints and one translational joint (Fig. 1-a). Its shoulder is embodied with the rotational joint 1, its elbow with the rotational joint 2, and its wrist with the rotational joint 4. Its vertical motion is accomplished with the translational joint 3. Since it features a small motion envelope while its speeds and payloads are relatively high, Adept 550 robot can be found in mechanical assembly, material

handling, packaging, machine tending, screwdriving, and many other operations requiring fast and precise automation.

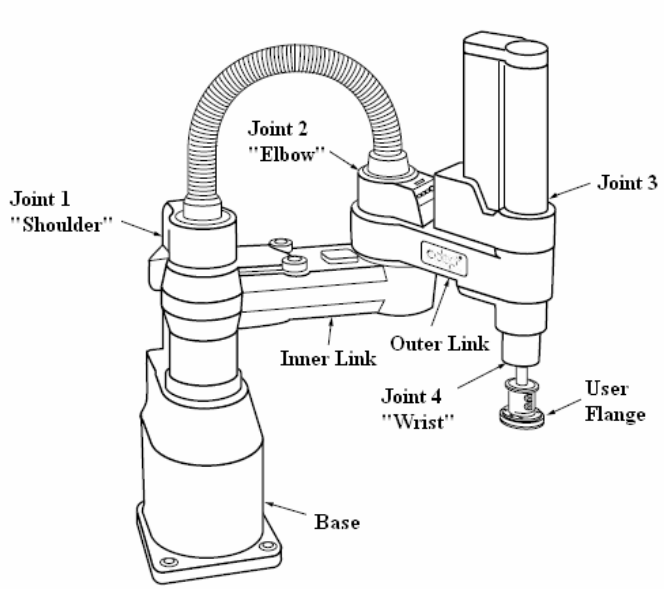


Figure 1. Adept 550 Robot with joint locations [1]

Needless to say, a complete study of PID controller design based on the mathematical model of Adept 550 is meaningful for its industrial applications. The geometric dimensions of Adept 550 are shown in Fig. 2 for modelling and simulations.

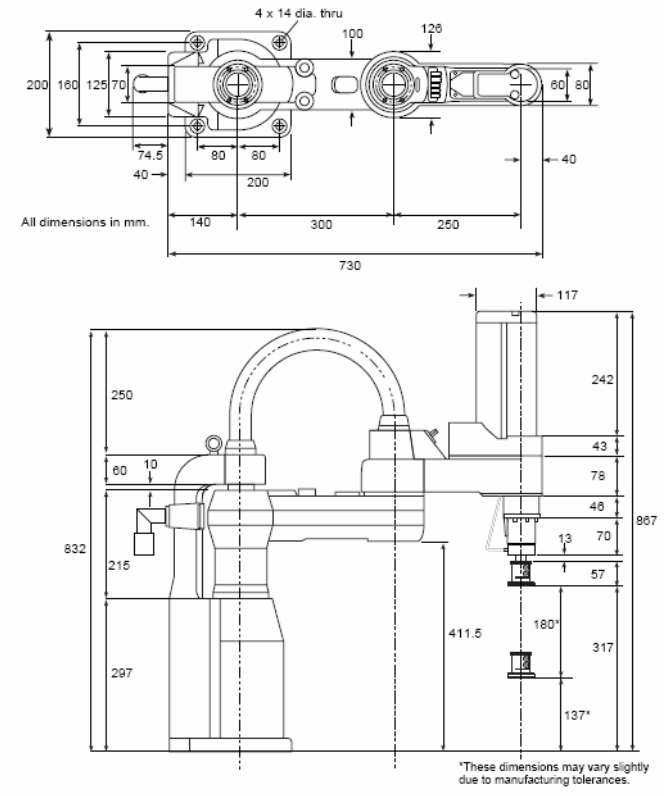


Figure 2. Adept 550 dimensions [1]

2. THE DYNAMICS OF ADEPT 550 ROBOT

2.1 The Link Positions and Inverse Kinematic Relationships

According to the Denavit – Hartenberg (D-H) coordinates, the Z-axis is in the direction of the joint axis, Adept 550 has the special case of parallel Z axes, connected with the rigid inner and outer links. The trajectory of the robot is determined by the motion of these two links, and at the wrist the rotational joint rotates about the Z axis to adjust the gripper angle, but not to change the trajectory. In order to focus on the trajectory study, without loss its generality, we assume the wrist's rotary angle is zero. Therefore, the coordinate systems to describe trajectory and link positions can be simply presented as follows (Fig. 3):

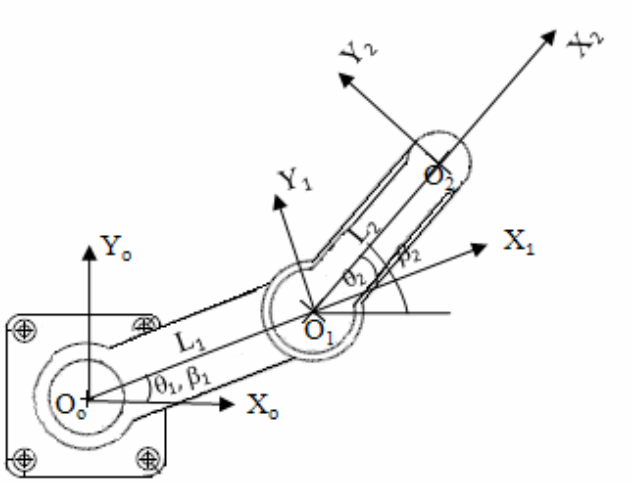


Figure 3. Top view of Adept 550 with joints rotated by angles

$$\beta_1, \theta_1, \beta_2, \theta_2$$

Table 1. lists the D-H parameters of the inner and outer links.

Table 1. D-H Parameters

Link	L_i	α_i	d_i	θ_i
Inner	L_1	0	0	θ_1
Outer	L_2	0	0	θ_2

For the inner link (i=1) and outer link (i=2), the coordinate transformation with rotation and translation components is described with the matrix:

$$A_i = R_{z,\theta_i} T_{z,d_i} T_{x,L_i} R_{x,\alpha_i} =$$

$$\begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & L_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & L_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The simplified transformation matrix of Adept 550 from the base to the gripper is as follows:

$$T_0^2 = A_1 A_2 =$$

$$\begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Noticing the relationships the angular position β_i $(i=1,2)$ of the motors and the angle θ_i $(i=1,2)$ about previous z from old x to new x: $\beta_1 = \theta_1$, $\beta_2 = \theta_1 + \theta_2$, the gripper's horizontal position (P_x, P_y) can be expressed as:

$$\begin{aligned} P_x &= L_1\cos\beta_1 + L_2\cos\beta_2 \\ P_y &= L_1\sin\beta_1 + L_2\sin\beta_2 \end{aligned} \quad (3)$$

From Equation (3), the motor angular positions (β_1, β_2) can be derived:

$$\begin{aligned} \beta_1 &= 2\tan^{-1}\left(\frac{P_y \pm \sqrt{P_x^2 + P_y^2 - R_1}}{P_x + R_1}\right) \\ \beta_2 &= 2\tan^{-1}\left(\frac{P_y \pm \sqrt{P_x^2 + P_y^2 - R_2}}{P_x + R_2}\right) \end{aligned} \quad (4)$$

where

$$\begin{aligned} R_1 &= \frac{P_x^2 + P_y^2 + L_1^2 - L_2^2}{2L_1} \\ R_2 &= \frac{P_x^2 + P_y^2 + L_2^2 - L_1^2}{2L_2} \end{aligned} \quad (5)$$

2.2 Forward and Backward Velocity and Acceleration Relationships

The forward velocity v can be found from Equation (3):

$$v = \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \end{pmatrix} = J_a \begin{pmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{pmatrix} \quad (6)$$

where J_a is Jacobian matrix

$$J_a = \begin{bmatrix} -L_1\sin\beta_1 & -L_2\sin\beta_2 \\ L_1\cos\beta_1 & L_2\cos\beta_2 \end{bmatrix} \quad (7)$$

And the backward velocity $(\dot{\beta}_1, \dot{\beta}_2)$ can be derived as follows:

$$\begin{pmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{pmatrix} = J^{-1} \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \end{pmatrix} \quad \text{or}$$

$$\begin{pmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{pmatrix} = \frac{1}{L_1 L_2 \sin(\beta_2 - \beta_1)} \begin{bmatrix} L_2\cos\beta_2 & L_2\sin\beta_2 \\ -L_1\cos\beta_1 & -L_1\sin\beta_1 \end{bmatrix} \begin{pmatrix} \dot{P}_x \\ \dot{P}_y \end{pmatrix} \quad (8)$$

Also, the relationships of the forward acceleration $(\ddot{P}_x, \ddot{P}_y)$ can be determined:

$$\begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \end{pmatrix} = J_a \begin{pmatrix} \ddot{\beta}_1 \\ \ddot{\beta}_2 \end{pmatrix} + J_v \begin{pmatrix} \dot{\beta}_1^2 \\ \dot{\beta}_2^2 \end{pmatrix} \quad (9)$$

where

$$J_v = \begin{bmatrix} -L_1\cos\beta_1 & -L_2\cos\beta_2 \\ -L_1\sin\beta_1 & -L_2\sin\beta_2 \end{bmatrix} \quad (10)$$

The relationships of the backward acceleration $(\ddot{\beta}_1, \ddot{\beta}_2)$ are as follows:

$$\begin{pmatrix} \ddot{\beta}_1 \\ \ddot{\beta}_2 \end{pmatrix} = J_a^{-1} \begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \end{pmatrix} - J_a^{-1} J_v \begin{pmatrix} \dot{\beta}_1^2 \\ \dot{\beta}_2^2 \end{pmatrix} = \quad (11)$$

$$= \frac{1}{L_1 L_2 \sin(\beta_2 - \beta_1)} *$$

$$\left\{ \begin{bmatrix} L_2\cos\beta_2 & L_2\sin\beta_2 \\ -L_1\cos\beta_1 & -L_1\sin\beta_1 \end{bmatrix} \begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \end{pmatrix} + \begin{bmatrix} L_1 L_2\cos(\beta_2 - \beta_1) & L_2^2 \\ -L_1^2 & -L_1 L_2\cos(\beta_2 - \beta_1) \end{bmatrix} \begin{pmatrix} \dot{\beta}_1^2 \\ \dot{\beta}_2^2 \end{pmatrix} \right\}$$

2.3 Modelling and Simulation of the Dynamics of Adept 550 Robot

For the simplified structure of the robot as shown in Fig. 4, the inner link and outer link are represented as rigid bars with the central mass m_i $(i=1,2)$ at their centers, and the gripper and load are symbolized as a mass m_3. Out of several commonly used methods for modeling the robot dynamics, Lagrange method is employed to develop the dynamic model of Adept 550 robot. The Lagrange function L is the difference of the

total kinematic energy T and the total potential energy V, i.e.,

$$L = T - V \tag{12}$$

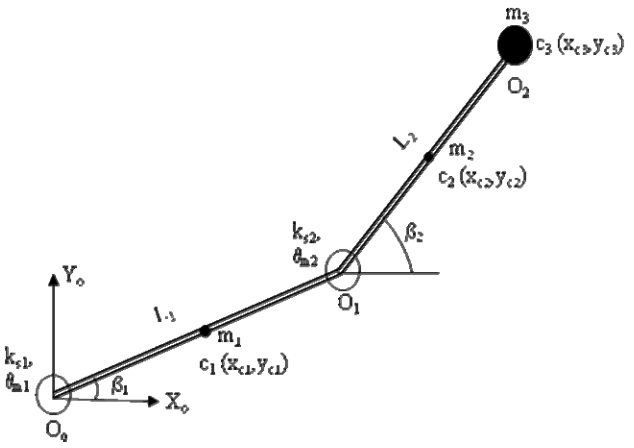


Figure 4. Simplified structure of Adept 550 robot

For the robotic structure shown in Fig. 4, the total kinematic energy T is expressed as:

$$T = T_{prismatic} + T_{rotation} \tag{13}$$

where

$$T_{prismatic} = \frac{1}{2} m_1 v_{c1}^2 + \frac{1}{2} m_2 v_{c2}^2 + \frac{1}{2} m_3 v_{c3}^2 =$$

$$= \frac{1}{2} m_1 \left(\frac{1}{4} L_1^2 \dot{\beta}_1^2 \right) + \frac{1}{2} m_2 \left(L_1^2 \dot{\beta}_1^2 + L_1 L_2 \dot{\beta}_1 \dot{\beta}_2 \cos(\beta_2 - \beta_1) + \frac{1}{4} L_2^2 \dot{\beta}_2^2 \right)$$
$$+ \frac{1}{2} m_3 \left(L_1^2 \dot{\beta}_1^2 + 2 L_1 L_2 \dot{\beta}_1 \dot{\beta}_2 \cos(\beta_2 - \beta_1) + L_2^2 \dot{\beta}_2^2 \right) \tag{14}$$

$$T_{rotation} = \frac{1}{2} \left(\frac{1}{3} m_1 L_1^2 \right) \dot{\beta}_1^2 + \frac{1}{2} \left(\frac{1}{3} m_2 L_2^2 \right) \dot{\beta}_1^2 \tag{15}$$

And the total potential energy V is given by:

$$V = V_{gravity} + V_{spring} \tag{16}$$

where

$$V_{gravity} = \left(\frac{1}{2} m_1 g L_1 + m_2 g L_1 + m_3 g L_1 \right) \sin \beta_1 +$$
$$+ \left(\frac{1}{2} m_2 g L_2 + m_3 g L_2 \right) \sin \beta_2 \tag{17}$$

$$V_{spring} = \frac{1}{2} k_{s1} \left(r \theta_{m1} - \beta_1 \right)^2 + \frac{1}{2} k_{s2} \left(r \theta_{m2} - \beta_2 \right)^2 \tag{18}$$

Thus the dynamics of Adept 550 robot described by Lagrange method is as follows:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\beta}_1} \right) - \frac{\partial T}{\partial \beta_1} + \frac{\partial V}{\partial \beta_1} &= \tau_1 \\ \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\beta}_2} \right) - \frac{\partial T}{\partial \beta_2} + \frac{\partial V}{\partial \beta_2} &= \tau_2 \end{aligned} \tag{19}$$

Substituting Equations (13) – (18) into Equation (19) results in:

$$\tau_1 = \tag{20}$$

$$\left(\frac{7}{12} m_1 + m_2 + m_3 \right) L_1^2 \ddot{\beta}_1 + \left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \cos(\beta_2 - \beta_1) \ddot{\beta}_2 -$$
$$\left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \sin(\beta_2 - \beta_1) \dot{\beta}_2^2 + \left(\frac{1}{2} m_1 + m_2 + m_3 \right) g L_1 \cos \beta_1 - k_{s1} \left(r \theta_{m1} - \beta_1 \right)$$

$$\tau_2 = \tag{20}$$

$$\left(\frac{7}{12} m_2 + m_3 \right) L_2^2 \ddot{\beta}_2 + \left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \cos(\beta_2 - \beta_1) \ddot{\beta}_1 +$$
$$\left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \sin(\beta_2 - \beta_1) \dot{\beta}_1^2 + \left(\frac{1}{2} m_2 + m_3 \right) g L_2 \cos \beta_2 - k_{s2} \left(r \theta_{m2} - \beta_2 \right)$$

Denoting $\beta = \begin{pmatrix} \beta_1 & \beta_2 \end{pmatrix}^T$, and considering damping torques $\tau_{damping}$ of the inner and outer links, Equation (20) can be rewritten as:

$$D(\beta) \ddot{\beta} + H(\beta, \dot{\beta}) \dot{\beta} + G(\beta) = \tau + \tau_{damping} \tag{21}$$

where

$$D(\beta) = \begin{bmatrix} \left(\frac{7}{12} m_1 + m_2 + m_3 \right) L_1^2 & \left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \cos(\beta_2 - \beta_1) \\ \left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \cos(\beta_2 - \beta_1) & \left(\frac{7}{12} m_2 + m_3 \right) L_2^2 \end{bmatrix}$$

$$H(\beta, \dot{\beta}) = \begin{bmatrix} 0 & -\left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \sin(\beta_2 - \beta_1) \dot{\beta}_2 \\ \left(\frac{1}{2} m_2 + m_3 \right) L_1 L_2 \sin(\beta_2 - \beta_1) \dot{\beta}_1 & 0 \end{bmatrix}$$

$$G(\beta) = \begin{pmatrix} \left(\frac{1}{2} m_1 + m_2 + m_3 \right) g L_1 \cos \beta_1 - k_{s1} \left(r \theta_{m1} - \beta_1 \right) \\ \left(\frac{1}{2} m_2 + m_3 \right) g L_2 \cos \beta_2 - k_{s2} \left(r \theta_{m2} - \beta_2 \right) \end{pmatrix}$$

$$\tau = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix}$$

$$\tau_{damping} = \begin{pmatrix} \tau_{damping1} \\ \tau_{damping2} \end{pmatrix} = \begin{bmatrix} -C_1 & 0 \\ 0 & -C_2 \end{bmatrix} \begin{pmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{pmatrix} = C \begin{pmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{pmatrix}$$

where matrix C is composed of the damping coefficients.

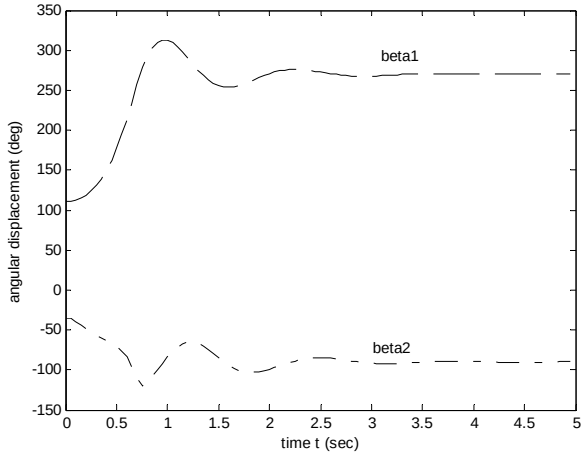


Figure 5. Dynamic responses: angular displacements

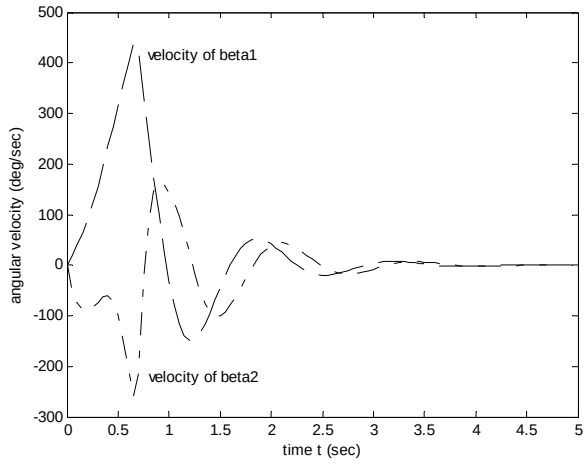


Figure 6. Dynamic responses: angular velocities

It can be observed that the dynamic model of Adept 550 robot is a highly nonlinear system. The simulation of the robot dynamics will be performed with Adams-Bashforth recursive scheme, which has been applied to general open-chain type manipulators and proven to be much more efficient in computational load than the nonrecursive methods. This is especially the case when the system is highly nonlinear.

Fig. 5 and 6 display the dynamic responses of angular displacements and velocities with the initial conditions at a given gripper position. The simulation illustrates the dynamics modeling of Adept 550 developed by Lagrange method, and demonstrates the stability of the robot motion. It has been observed that the heavier the load m_3 is, the longer it takes the system to be settled; and when the load is heavier than a certain weight, the gripper system becomes unstable.

2.4 **Sensitivities to Static Torques in Open Loop Control**

In the open loop control of Adept 550 robot, the sensitivities of angular displacements to static torques indicate how well the gripper can hold its position. These sensitivities of angular displacements to static torques are as follows:

$$\frac{\partial \beta_1}{\partial \tau_1} = -\frac{1}{\left(\frac{1}{2}m_1 + m_2 + m_3\right) g L_1 \sin \beta_1}$$
$$\frac{\partial \beta_2}{\partial \tau_2} = -\frac{1}{\left(\frac{1}{2}m_2 + m_3\right) g L_2 \sin \beta_2} \qquad (23)$$

Variations of static torques τ in the open loop will produce errors in angular displacements β. A closed loop position control is desirable to address this issue (Fig.7).

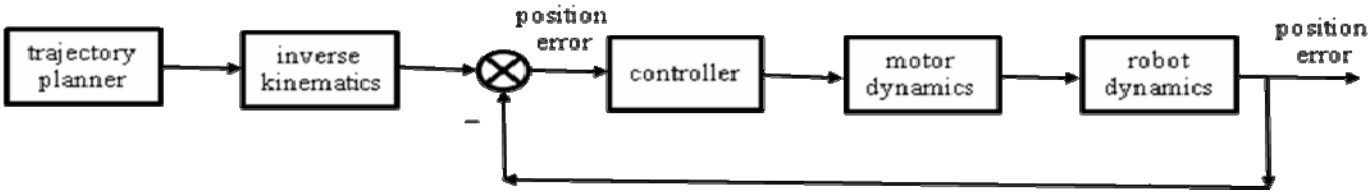


Figure 7. Robot control diagram

3. ROBOT CONTROL AND ITS SIMULATIONS

3.1 **Robot Dynamics with PID Controller**

It is assumed that the motors driving the inner and outer links are the same type of motors. Considering the motor parameters, dynamics of the two links is described as:

$$\sum_{j=1}^{n} d_{jk}(\beta)\ddot{\beta}_j + \sum_{i,j=1}^{n} h_{ijk}(\beta)\dot{\beta}_i\dot{\beta}_j + g_k(\beta) = \tau_k - C_k\dot{\beta}_k,$$
$$J_{m,k}\ddot{\theta}_{mk} + \left(B_{m,k} + K_{b,k}\frac{K_{m,k}}{R_k}\right)\dot{\theta}_{mk} = \frac{K_{m,k}}{R_k}V_k - \tau_{m,k} \qquad (24)$$
$$k = 1,2$$

Since $\beta_k = r\theta_{m,k}$, $\tau_{m,k} = r\tau_k$, where r is the gear ratio, the two dynamic equations of robot link and its driving motor expressed in Equation (24) can be combined into a single one:

$$J_{eff,k}\ddot{\theta}_{mk} + B_{eff,k}\dot{\theta}_{mk} = KV_k - C_k\theta_{mk} - rd_k, \qquad k = 1,2 \quad (25)$$

The PID control law for the controller and motor drive at the inner or outer link is as follows:

$$V_k = K_{P,k}\left(\theta_k^d - \theta_{mk}\right) + K_{D,k}\left(\dot{\theta}_k^d - \dot{\theta}_{mk}\right) + \\ + K_{I,k}\int\left(\theta_k^d - \theta_{mk}\right)dt, \qquad k = 1,2 \quad (26)$$

The block diagram based on the transfer functions of Equations (25) and (26) for the robot control is shown in Fig. 8.

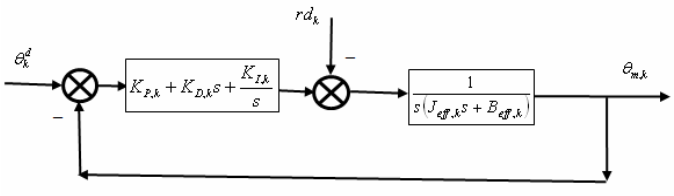


Figure 8. PID controlled robotic motions

The transfer function of the closed loop system is:

$$\theta_{m,k} = \frac{K_{P,k}s + K_{D,k}s^2 + K_{I,k}}{J_{eff,k}s^3 + \left(B_{eff,k} + K_{D,k}\right)s^2 + K_{P,k}s + K_{I,k}}\theta_k^d \\ - \frac{rs}{J_{eff,k}s^3 + \left(B_{eff,k} + K_{D,k}\right)s^2 + K_{P,k}s + K_{I,k}}d_k \quad (27)$$

where the nonlinear terms are treated as disturbances to the system:

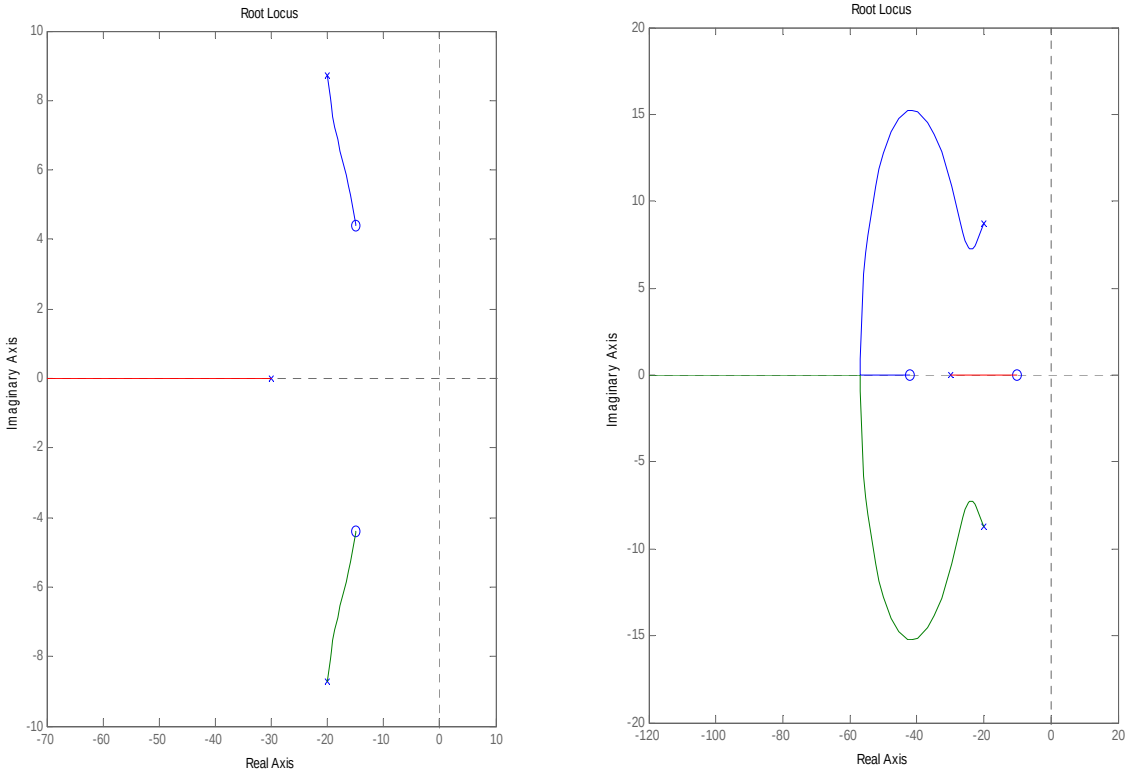
$$d_1 = \left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1)\ddot{\beta}_2 - \left(\frac{1}{2}m_2 + m_3\right)L_1L_2\sin(\beta_2 - \beta_1)\dot{\beta}_2^2 \\ + \left(\frac{1}{2}m_1 + m_2 + m_3\right)gL_1\cos\beta_1 \quad (28)$$

$$d_2 = \left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1)\ddot{\beta}_1 - \left(\frac{1}{2}m_2 + m_3\right)L_1L_2\sin(\beta_2 - \beta_1)\dot{\beta}_1^2 \\ + \left(\frac{1}{2}m_2 + m_3\right)gL_2\cos\beta_2$$

Placing the three poles $-p_k$, $-\varsigma_k\omega_{n,k} \pm j\omega_{n,k}\sqrt{1-\varsigma_k^2}$ of the characteristic polynomial onto the stable half-plane, and designing $-\varsigma_k\omega_{n,k} \pm j\omega_{n,k}\sqrt{1-\varsigma_k^2}$ as the dominant poles, and $-p_k$ to manipulate the root locus yield the following relationships on the PID control parameters:

$$K_{P,k} = \left(\omega_{n,k}^2 + 2\varsigma_k\omega_{n,k}p_k\right)J_{eff,k} \\ K_{I,k} = \omega_{n,k}^2 p_k J_{eff,k} \quad (29) \\ K_{D,k} = \left(2\varsigma_k\omega_{n,k} + p_k\right)J_{eff,k} - B_{eff,k} \\ k = 1,2$$

Its root locus analysis for the inner link and outer link of the tuned PID controllers is demonstrated in Fig. 9.



a) b)

Figure 9. Root locus analysis for inner link (a) and outer link (b)

Noticing $\beta_k = r\theta_{m,k}$, $\beta_k^d = r\theta_k^d$, the dynamics of the inner or outer link can be described as

$$J_{eff,k}\ddot{\beta}_k + \left(B_{eff,k} + K_{D,k} + C_k r\right)\ddot{\beta}_k + K_{P,k}\dot{\beta}_k + K_{I,k}\beta_k = = K_{D,k}\ddot{\beta}_k^d + K_{P,k}\dot{\beta}_k^d + K_{I,k}\beta_k^d - r^2\dot{d}_k \quad (30)$$
$$k = 1,2$$

Equation (30) can be rewritten in matrix form:

$$D_{PID}(\beta)\ddot{\beta}_k + H_{PID}(\beta)\ddot{\beta}_k + G_{PID}(\beta)\dot{\beta}_k + + K_{PID}(\beta)\beta_k + C_{PID}(\beta) = \left(0_{2\times 1}\right) \quad (31)$$

where

$$D_{PID}(\beta) = \begin{bmatrix} J_{eff,1} & r^2\left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1) \\ r^2\left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1) & J_{eff,2} \end{bmatrix}$$

$$H_{PID}(\beta) = \begin{bmatrix} B_{eff,1} + K_{D,1} + C_1 r & -\left(\frac{1}{2}m_2 + m_3\right)L_1L_2\sin(\beta_2 - \beta_1)\left(3\dot{\beta}_2 - \dot{\beta}_1\right)r^2 \\ -\left(\frac{1}{2}m_2 + m_3\right)L_1L_2\sin(\beta_2 - \beta_1)\left(\dot{\beta}_2 - 3\dot{\beta}_1\right)r^2 & B_{eff,2} + K_{D,2} + C_2 r \end{bmatrix}$$

$$G_{PID}(\beta) = \begin{bmatrix} K_{P,1} - \left(\frac{1}{2}m_1 + m_2 + m_3\right)gL_1\sin\beta_1 r^2 & -\left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1)\dot{\beta}_2(\dot{\beta}_2 - \dot{\beta}_1)r^2 \\ \left(\frac{1}{2}m_2 + m_3\right)L_1L_2\cos(\beta_2 - \beta_1)\dot{\beta}_1(\dot{\beta}_2 - \dot{\beta}_1)r^2 & K_{P,2} - \left(\frac{1}{2}m_2 + m_3\right)gL_2\sin\beta_2 r^2 \end{bmatrix}$$

$$K_{PID}(\beta) = \begin{bmatrix} K_{I,1} & 0 \\ 0 & K_{I,2} \end{bmatrix}$$

$$C_{PID}(\beta) = \begin{pmatrix} -K_{D,1}\ddot{\beta}_1^d - K_{P,1}\dot{\beta}_1^d - K_{I,1}\dot{\beta}_1^d \\ -K_{D,2}\ddot{\beta}_2^d - K_{P,2}\dot{\beta}_2^d - K_{I,2}\dot{\beta}_2^d \end{pmatrix} \quad (32)$$

3.2 **Trajectory Planning**

The PID controller can be designed to perform a trajectory tracking from the initial position (x_o, y_o) to the final position (x_f, y_f) at time t_f. Without loss of generality, the motion in x axis is assumed with following constraints: $x_o > x_f$, and constant acceleration $-a_x$ in time interval $[0, t_b]$, constant speed $-v_x$ in time interval $\lfloor t_b, t_f - t_b \rfloor$, and constant acceleration a_x in time interval $\lfloor t_f - t_b, t_f \rfloor$, where for a given constant speed v_x,

$$t_b = \frac{x_o - x_f + v_x t_f}{v_x}$$
$$a_x = \frac{v_x}{t_b} \quad (33)$$

The trajectory component P_x can be expressed as follows:

$$P_x = \begin{cases} x_o + \frac{1}{2}a_x t^2, & 0 \le t < t_b \\ \frac{x_f + x_o - v_x t_f}{2} + v_x t, & t_b \le t < t_f - t_b \\ x_f - \frac{1}{2}a_x t_f^2 + a_x t_f t - \frac{1}{2}a_x t^2, & t_f - t_b \le t \le t_b \end{cases} \quad (34)$$

Fig. 10 displays the simulated trajectory described in Equations (33) and (34) for a given set of parameters. And the trajectory component P_y can also be expressed in the similar way. Then the desired angular displacement β^d, velocity $\dot{\beta}^d$ and acceleration $\ddot{\beta}^d$ can be determined:

$$\begin{pmatrix} \beta_1^d \\ \beta_2^d \end{pmatrix} = \begin{pmatrix} 2\tan^{-1}\left(\frac{P_y \pm \sqrt{P_x^2 + P_y^2 - R_1}}{P_x + R_1}\right) \\ 2\tan^{-1}\left(\frac{P_y \pm \sqrt{P_x^2 + P_y^2 - R_2}}{P_x + R_2}\right) \end{pmatrix}$$
$$\begin{pmatrix} \dot{\beta}_1^d \\ \dot{\beta}_2^d \end{pmatrix} = J_a(\beta)^{-1}\begin{pmatrix} \dot{P}_x \\ \dot{P}_y \end{pmatrix} \quad (35)$$
$$\begin{pmatrix} \ddot{\beta}_1^d \\ \ddot{\beta}_2^d \end{pmatrix} = J_a(\beta)^{-1}\begin{pmatrix} \ddot{P}_x \\ \ddot{P}_y \end{pmatrix} - J_a(\beta)^{-1}J_v(\beta)\begin{pmatrix} \left(\dot{\beta}_1^d\right)^2 \\ \left(\dot{\beta}_2^d\right)^2 \end{pmatrix}$$

3.3 **Simulations of the Closed Loop Controlled Robot Dynamics**

Denoting $X = \left[\ddot{\beta}_1 \quad \ddot{\beta}_2 \quad \dot{\beta}_1 \quad \dot{\beta}_2 \quad \beta_1 \quad \beta_2\right]^T$, Equation (31) can be rewritten as

$$\dot{X} = \begin{bmatrix} 0_{2\times 2} & I_{2\times 2} & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} & I_{2\times 2} \\ -D_{PID}^{-1}K_{PID} & -D_{PID}^{-1}G_{PID} & -D_{PID}^{-1}H_{PID} \end{bmatrix}X + \begin{pmatrix} 0_{2\times 1} \\ 0_{2\times 1} \\ -D_{PID}^{-1}C_{PID} \end{pmatrix} \quad (36)$$

Per Equations (35), the trajectory related matrices and vector given by Equation (32) can be determined. Then the closed loop controlled dynamics of Adept 550 robot expressed in Equation (30) or (31) during the planned trajectory can numerically simulated.

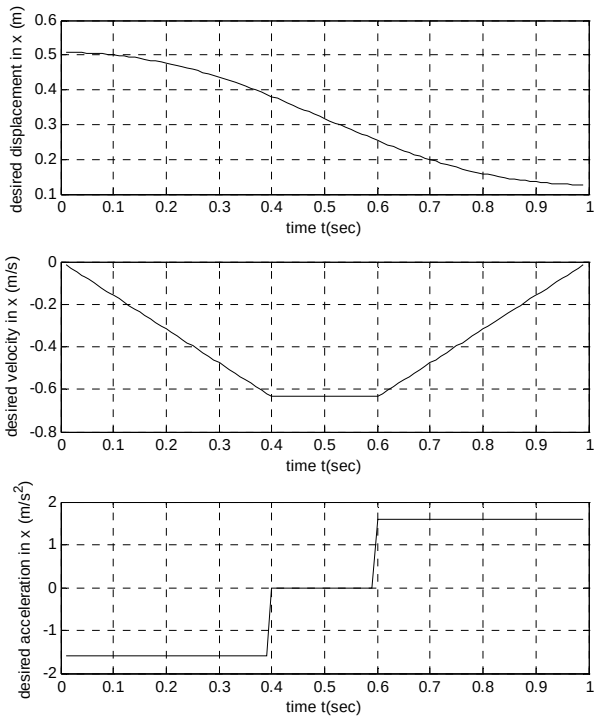


Figure 10. Desired trajectory planning in x axis

In order to verify the effectiveness of the PID controller based on dynamics of Adept 550 robot, the following case is given for simulations: assume the PID controller is with a critical damp for both inner link and outer link, the gripper moves from the initial position $(P_x = 508\ mm, P_y = 254\ mm)$ to the final position $(P_x = 127\ mm, P_y = 254\ mm)$ within $t_f = 1\,\text{sec}$. Fig. 11 shows the simulation results of Equation (36) recursively integrated by Adams – Bashforth method.

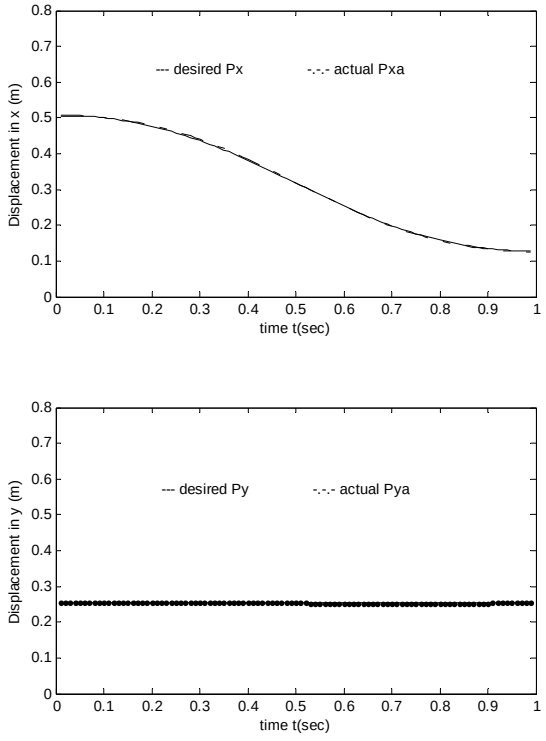


Figure 11. Simulation of PID controlled trajectory tracking

It can be observed that the trajectory tracking process is fast and stable. Its evaluation is given in the appendix.

5. CONCLUSION

The PID controller design for a nonlinear motion control based on the mathematical modelling of the dynamics of Adept 550 Robot is explored.

The general relationships of the PID controller design on the robotic dynamics and the planned trajectory are derived. The analysis and simulations of its closed loop dynamics indicates its effectiveness in fast and accurate trajectory tracking.

The procedure and analysis of this research can be practically generalized to other cases of PID controller design for other robots in the industrial applications. It is meaningful for optimizing the commonly used PID controller without trial-and-error testing, and it is especially important for precision operations.

Acknowledgement

The author sincerely thanks Ford Motor Co. for sponsoring this research as a part of Ford Stewardship funding. The helpful discussion with Professor Celal Batur at The University of Akron is also greatly appreciated.

6. REFERENCES

[1] Tan, K.K., Lee, T.H., and Khoh, C.J. (2000) "*PID-Augmented Adaptive Control of A Gyro Mirror Los System*", Asian Journal of Control, Vol. 4, No. 2, June 200, pp. 240-245

[2] Ouyang, P. R., Zhang, W. J., and F. X. Wu (2002) "*Nonlinear IPD Control for Trajectory Tracking with Consideration of the Design for Control Methodology*", Proceedings of the 2002 IEEE International Conference on Robotics *8* Automation, Washington, DC May 2002, pp. 4126-4131

[3] Kawamura, H., Kadota, H., Yamamoto, M., Takaya, T. and Ohuchi, A. (2005) "*Motion Design for Indoor Blimp Robot with PID Controller*", Journal of Robotics and Mechatronics, Vol. 17, No. 5, pp. 500-508

[4] Su, Y.X., Sun, D. and Duan, B.Y. (2005) "*Design of an enhanced nonlinear PID controller*", Mechatronics, Vol. 15, pp. 1005–1024,

[5] Chee, O.Y., Shukri, M. and Abidin, Z. (2006), "*Design and Development of Two Wheeled Autonomous Balancing Robot*", IEEE Student Conference on Research and Development, Shah Alam, Selangor, Malysia, June 27-28, pp. 169-172

[6] Ramos, J.M., Vargas, E., Gorrostieta, E., Romero, R.J. and Pedraza, J.C. (2006) „*Pneumatic Cylinder Control PID for Manipulators Robot*", The 2006 Conference on Dynamics, Instrumentation and Control. August 13-16, Querétaro, Qro.

[7] Takaya, T., Kawamura, H., Minagawa, Y., Yamamoto, M. and Ohuchi A. (2006) "*PID landing orbit motion controller for an indoor blimp robot*", Artificial Life and Robotics, Volume 10, Number 2, November, pp. 177-184

[8] Su, W. (2007) "*A Model Reference-Based Adaptive PID Controller for Robot Motion Control of Not Explicitly Known Systems*", International Journal of Intelligent Control and Systems, Vol. 12, No. 3, September, pp. 237-244

[9] Adept 550 Robot Instruction Handbook, www1.adept.com/main/ke/data/Robot/Adept550/550IHB.PDF

[10] Hollerbach, J.M. (1980) "*A recursive Lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity*", Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol. 102, No. 2, pp. 69-76

Appendix: Values of system parameters and evaluations

The values of the system parameters of the robot inner and outer links are listed as bellows for numerical simulation:
The transfer function for the inner link:

$$G_{InnerLink} = \frac{\left(s^2 + 29.86s + 242.10\right)}{(s+30)(s^2 + 40s + 476)}$$

The transfer function for the outer link:

$$G_{OuterLink} = \frac{(s+42.06)(s+10.04)}{(s+30)(s^2 + 40s + 476)}$$

The PID controller for the inner link:

$K_{P,1}$ = 9.003 Nm, $K_{I,1}$ = 72.997 Nm/s, $K_{D,1}$ = 0.302 Nms
The PID controller for the outer link:
$K_{P,2}$ = 3.937 Nm, $K_{I,2}$ = 31.917 Nm/s, $K_{D,2}$ = 0.076 Nms

The position error is defined as

$$e = P_{trajectory} - P_{actual} = \begin{pmatrix} P_x \\ P_y \end{pmatrix} - \begin{pmatrix} P_{xa} \\ P_{ya} \end{pmatrix}$$

It should satisfy the following requirement at the final position:

$$e(t_f) \le 2.5mm$$

The actual error at the final position is $e(t_f) = 1.94mm$.
The cost function

$$J = \frac{1}{2N}\sum_{k=1}^{N} e^T(k) \times e(k)$$

It should satisfy the following requirement over the entire trajectory:

$$J \le 2.5 \quad mm^2$$

The actual cost function is $J = 2.38\ mm^2$.