

Joint Control of a Robotic Manipulator in the Presence of Disturbances

Syed Irtiza Ali Shah, Muhammad Taha Shahid, Abdul Wahab, Majid Mehmood

Department of Mechanical & Aerospace Engineering
Air University, Islamabad, Pakistan

irtiza@mail.au.edu.pk, taha.satti2003@gmail.com, abdulwahabaa73@gmail.com, majid.mehmood@au.edu.pk

Abstract

Robotics as a field of study is still young but is developing very fast because of the progress in the sciences and technology, more so because robots are taking up more roles in industries where precision is important. This work is concerned with the control of the position of the end-effector in robotic manipulators with particular attention to the need for accurate dynamic modeling and the ramifications of poor modeling. Robotic manipulators provide substantial benefits in the operation requiring a high level of accuracy, which could be a problem for human arm in the long run especially in medical applications. In order to overcome these issues, various measures have been suggested to improve the precision of robotic arms. From these, PID controller was applied on the system, experimented with various inputs and the output was concluded. Although many of the past approaches that have been used work well, these techniques were developed for the use in repetitive processes and are not as useful when used in unpredictable situations. Our approach is to develop a model that works with high accuracy and is fully under the control of human signals at any time. There are certain drawbacks to the method, but with time improvements are being made, and the model provided here may be refined into a superior form. The advanced aspect of this approach is that it does not require any heavy computations, making it more practicable. Additionally, it enables us to do rid of the speed sensor that many sophisticated robotic manipulators need to control performance and job efficiency. The system will become stable as a result of the outcomes we obtain and has a finite error response to parabolic inputs. Stability enables the system to reach a steady state and maintain it for that specific input even after altering its settings.

Keywords- Position Control, Control of Robotic Manipulator, Stability analysis, Robotic systems, Robot dynamic modeling.

I. INTRODUCTION

A) Literature Review

Robotic manipulators carrying out monotonous jobs throughout the assembling process are a regular sight in modern times. One of their general responsibilities is the installation of electrical components, which needs extremely accurate positioning along the necessary path. The experiment made use of the SEIKO D-TRAN RT3200 four-link cylindrical configuration robot shown in Fig. 1.

To provide a smooth course, Joints T and Z were told to change the rotation and vertical location of the end-effector, or the tip of the screwdriver. The inverse kinematics form was utilized to determine the command for joints T and Z.

$$\begin{aligned}x &= -d_3 \sin(\varnothing) & (1) \\y &= d_3 \cos(\varnothing) & (2) \\z &= d_1 + d_2 & (3)\end{aligned}$$

where $\varnothing$ denotes the joint T's angle of rotation about the Z-axis. d_1 indicates a random location from the ground to the screwdriver's tip. To make things as simple as possible, the value is set to zero, ensuring that the change in the Z-axis is caused solely by joint Z movement. The Z and Y axes are where positions d_2 and d_3 are located [1].

Another crucial area of research is robot force control in flexible situations. The real-time force control approach combines impedance-based control with force tracking capabilities and a

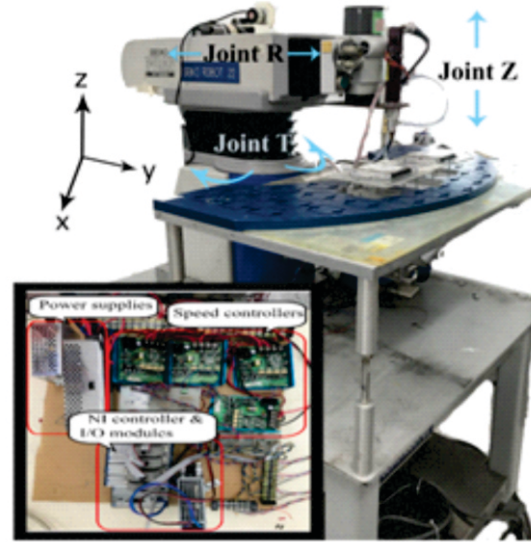


Fig. 1 Seiko D-tran Rt3200 [1]

hybrid force/position control algorithm [2]. The experimental setup consists of a non-rigid contact surface, a 2 DOF planar manipulator, a 6-axis force/torque sensor, and control hardware/software. (shown in Fig. 2).

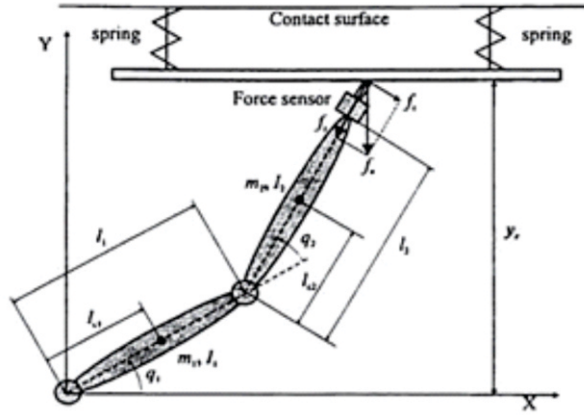


Fig. 2 Top-view of the robotic setup schematic diagram [3]

The pace of technological development is accelerating as time goes on, and along with it, the body of knowledge is growing exponentially. It is possible to build a more straightforward impedance control system in quasi-static conditions, however doing so requires assuming that the velocities described in the joint space and the Cartesian space, respectively, were equal to $\dot{x}=0$ and $\dot{q} \approx 0$. The resulting simplified impedance control law is as follows:

$$\tau = J(q)^T (B_d \dot{e} + K_d e) + g(q) \quad (4)$$

where $e = x_d - x$, $\dot{e} = \dot{x}_d - \dot{x}$ are, respectively, the Cartesian position and velocity errors [3].

The method for using a dynamic controller on a manipulator without direct drive joints is examined in the next article. A torque-to-position conversion method for servo-driven robotic manipulators that was previously described was integrated with an adaptive sliding mode controller that was step-back based for dynamic trajectory tracking control [4]. For greater tracking performance and finite-time stability, the suggested controller employed a finite-time non-singular sliding surface [5]. In order to estimate the model nonlinearities, avoid the structurally complex control rule, and obtain partial model independence of the controller, a time-delayed control philosophy was used to calculate the soft nonlinearities of the manipulator.

The controller was partially model-free because the soft nonlinearities of the manipulator were estimated using the time-delay estimation method. This will make it easier to use the controller with more degrees of freedom manipulators with sophisticated dynamics. [6].

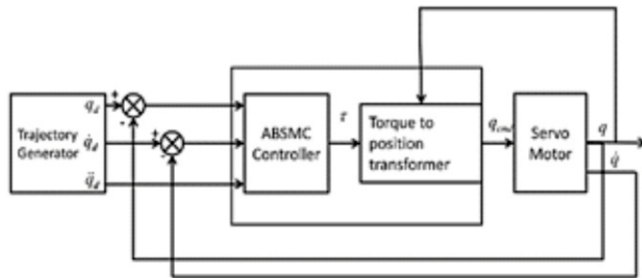


Fig. 3 The proposed dynamical control's block diagram [6]

The robotic manipulator may be controlled independently through a joint, which greatly facilitates the design of the control system. This method treats each joint actuator as a straightforward inertial system with an added perturbing moment that represents all un-modeled dynamics [7]. An easy-to-implement PD control system is made possible by the quick calculation and rejection of the flaw using a simple technique [8]. The modeling of an n-linked robotic manipulator is characterized by using a highly non-linear second-order differential equation [10];

$$D(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) + F(\dot{\theta}) = \tau \quad (5)$$

And, for the DC motor, a 2nd - order linear differential equation is used [1];

$$J_i \ddot{\theta}_{mi} + B_i \dot{\theta}_{mi} = K_i u_i - r_i \tau_i \quad (6)$$

J_i = total inertia for the motor and gear
 B_i = corresponding effective damping coefficient
 $\dot{\theta}_{mi}$ = motor shaft angular velocity
 $\ddot{\theta}_{mi}$ = motor shaft angular acceleration
 K_i = gain

This study presents a method for designing independent joint controllers. Each of the manipulator joint actuators modeled [9].

TABLE I. PUMA 560 JOINT DRIVE PARAMETERS [9]

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
J	1.14	4.71	0.827	0.20	0.179	0.193
K	23.125	23.125	23.125	3.564	3.564	3.564
r	0.01597	0.00931	0.01884	0.01428	0.01391	0.01303

Table I shows the joint drive specifications for the PUMA 560. The proposed control law's stability is examined. The recommended controller is tested on a PUMA 560 microcomputer-controlled arm. It is demonstrated that the control technique is easy to use, is practical, and adaptable to an existing industrial manipulator.

II. METHODOLOGY AND MODELING

This study is a continuation of a prior paper [5]. They worked on the design and analysis of a robotic arm's control system as well as its modelling. The disruptions are a result of the manipulator's incorrect dynamic modeling. In the previous work, the dynamic modeling of the manipulator was completed. We followed up on those findings by designing and analyzing a control system.

A. Approach

The gear reduction ratio of the gearbox at the motors is high for the proposed robotic hand system. Normal operating speeds and accelerations are restricted. Furthermore, because the robotic hand often grasps or manipulates the object in a narrow region, the motion range of each joint is rather short. Due to these characteristics, each joint may be modelled using a linear model, and the parts of the non-linear and coupling terms of robotic hand dynamics can be considered as disturbances. The transfer function of the system is illustrated below;

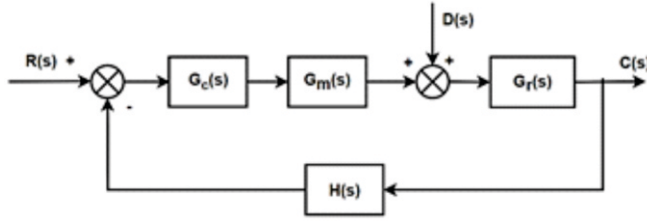


Fig. 4 Block Diagram of the system

We began by modeling it analytically. After modeling, we would test the combined Transfer Function of the robotic joint and the DC motor against different inputs to determine and assess the output against each input. The shortcoming of this strategy is that the values that will make our system stable are assumed.

B) Modeling

Assume that the joint's transfer function is $G(s)$, which contains the DC motor gain $G_m(s)$ and the dynamics of the link $G_r(s)$. The plant transfer role therefore becomes;

$$G(s) = G_m(s)G_r(s) \quad (7)$$

Fig. 5 shows a coupled to an inertial load via a DC motor. The voltages applied to the motor's field and armature sides are indicated by the V_f and V_a . The resistances and inductances of the motor's field and armature sides are denoted by the letters R_f , L_f , R_a , and L_a .

The currents in the motor's field i_f and armature side i_a determine how much torque the motor can generate.

$$T_m = K_i i_a \quad (8)$$

The voltage/current relation for the motor's field side is

$$V_f = V_R + V_L \quad (9)$$

$$V_f = R_f i_f + (L_f di_f/dt) \quad (10)$$

Taking Laplace transforms of both sides of this equation allows one to determine the transfer function from the voltage to the output current.

$$G_m(s) = \frac{(1/L_f)}{s + (R_f/L_f)} \quad (11)$$

Now, for the dynamics of the link;

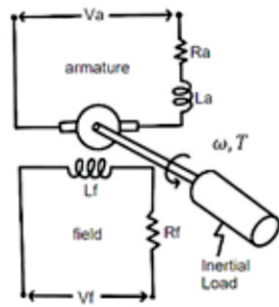


Fig. 5 Schematic diagram of a DC motor

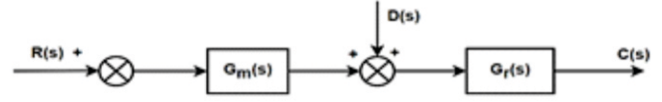


Fig. 6 Block diagram of the System

Using governing equations,

$$\sum T = J\alpha \quad (12)$$

$$K_T \tau(t) - D \frac{d\theta(t)}{dt} = \frac{J d^2\theta(t)}{dt^2} \quad (13)$$

Taking Laplace and re-writing as the transfer function,

$$G_r(s) = \frac{K_T/J}{s^2 + Ds/J} \quad (14)$$

The following is an open-loop block diagram of such a system with disturbances added;

The similar closed loop transfer function $G(s)$ is;

$$G(s) = \frac{K_T/L_f J}{s^3 + s^2(R_f L_f + D J)/L_f J + s R_f/L_f J} \quad (15)$$

By assumptions of some values, we get the plant reduced transfer function as:

$$G(s) = \frac{12}{s^2 + 3s + 12} \quad (16)$$

The time domain block diagram is represented below:

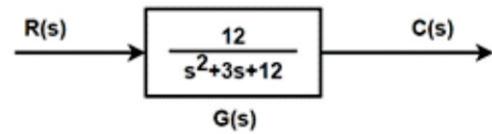


Fig. 7 Open-loop transfer function block diagram

Considering unity feedback ($H(s)=1$), the system can be reduced as;

$$T(s) = \frac{12}{s^2 + 3s + 24} \quad (17)$$

III. IMPLEMENTATION

For the modeling, designing, and characteristics of our system, we have used the following software

- MATLAB
- SIMULINK

Using MATLAB, for step input

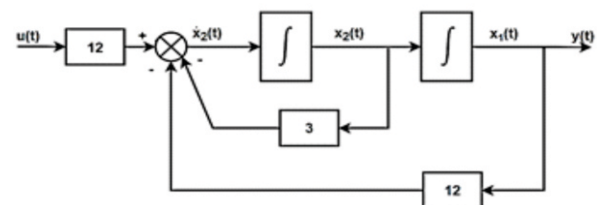


Fig. 8 State space block diagram of the system

RiseTime: 0.4678
 SettlingTime: 2.5142
 SettlingMin: 0.9009
 SettlingMax: 1.2031
 Overshoot: 20.3127
 Undershoot: 0
 Peak: 1.2031
 PeakTime: 1.0745

Fig. 9 MATLAB results

To obtain the system characteristics for various inputs, including step, ramp, and parabolic inputs, SIMULINK was then employed. It provided us with the system and graph features of these inputs, and we later compared them to get our conclusions. Additionally, the root locus was plotted using MATLAB. That revealed the stable areas of the system. Finally, we tuned our system using MATLAB once more using the necessary controller. Figure 13 displays a flowchart of the research project.

IV. RESULTS

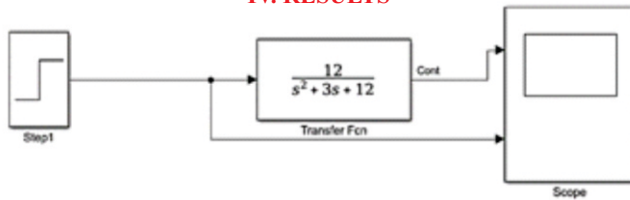


Fig. 10 Block schematic for Step input in SIMULINK

The system was put to the test using various inputs. The benefits and drawbacks of each response were charted and discussed.

1) Step Input:

The system's response is very stable and follows the step input very precisely yet,

- It has an overshoot of about 21%
- Some initial oscillations and a long settling time.

But it has no steady state error.

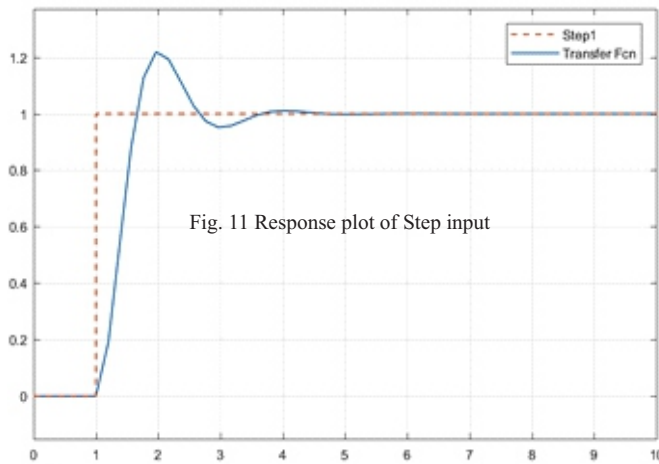
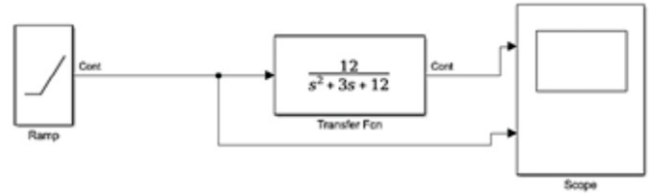


Fig. 11 Response plot of Step input

Fig. 12 Block schematic for Ramp input in SIMULINK



2) Ramp Input:

The system's response is very stable and follows the ramp input very precisely yet, it has some minimal oscillations in the beginning and is lacking a little bit behind the input command till infinity. Hence has some steady state error in it.

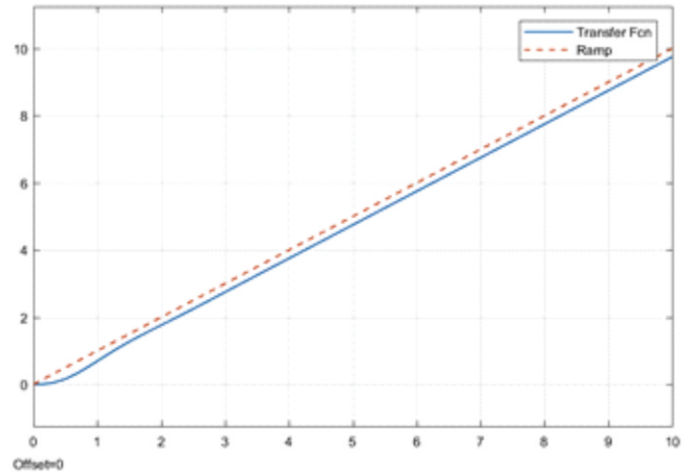


Fig. 13 Response plot of Ramp input

3) Parabolic Input:

The system's response is very stable and follows the parabolic input very precisely yet, it has some minimal oscillations in the beginning and is lacking a little bit behind the input command till infinity. Hence has some steady state error in it.

A) Stability of the System

From all the above plots, we concluded that our system is stable and only a few alterations are required (which will be done by implementing either of the P, PD or PID controller) to make it suitable for our application.

According to a block diagram, the open loop transfer function with P controller gain K and the unity feedback system is:

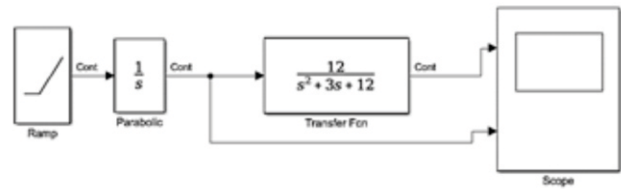


Fig. 14 Block schematic for Parabolic input in SIMULINK

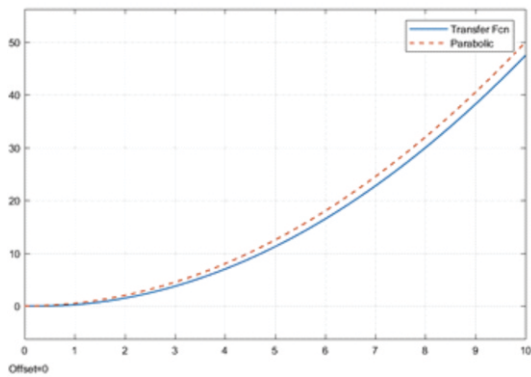


Fig. 15 Response plot of Parabolic input

The following system can be reduced to get equivalent transfer function $T(s)$.

$$T(s) = \frac{12K}{s^2 + 3s + (12 + 12K)} \quad (18)$$

Using the Routh Hurwitz technique, we can identify the system's stability as well as the range of P-controller gain K that the system is stable for.

Hurwitz Table;

$$\begin{array}{ccc} s^2 & 1 & 12+12K \\ s^1 & 3 & 0 \\ s^0 & -\left| \begin{array}{cc} 1 & 12+12 \\ 3 & 0 \end{array} \right| & 0 \end{array}$$

$$\begin{array}{c} 3 \\ 12+12K \end{array}$$

There shouldn't be any indication changes in the first column for a stable system. So, the term $12+12K$ must be greater than zero,

$$12+12K>0 \quad (19)$$

The limit for the gain is $K > -1$.

B. Steady State Error of the System

The steady state error for the system against multiple inputs is be tested and the results are discussed below;

$$G(s) = \frac{12}{s^2 + 3s + 12} \quad (20)$$

1) *Step Input:*

Using the simplified form of final value theorem; (21)

$$e_{step}(\infty) = 0.5$$

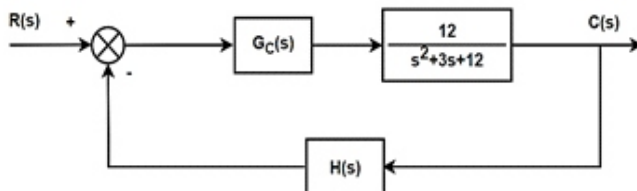


Fig. 16 Block diagram of closed-loop transfer function

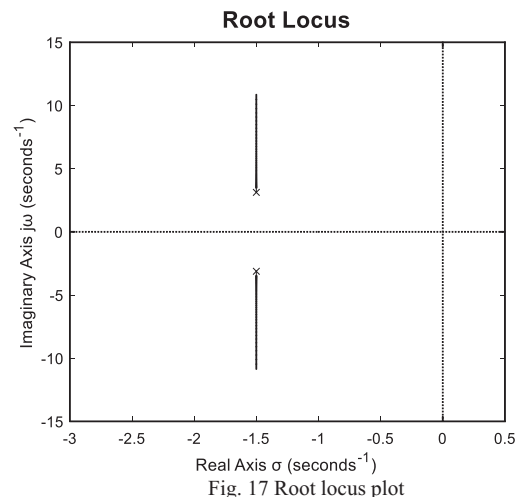


Fig. 17 Root locus plot

2) *Ramp Input:*

Using the simplified form of final value theorem; (22)

$$e_{ramp}(\infty) = \infty$$

3) Parabolic Input

Using the simplified form of final value theorem; (23)

$$e_{para}(\infty) = \infty$$

Since the system was a Type 0 system, as expected, for ramp and parabolic input the steady state error is infinite. And for the step input, the error is 1.

Finally plotting the root locus using MATLAB. Root Locus is a powerful method of analysis and design for stability and transient response.

From the plot, we conclude that root locus starts at the poles of the system and ends at infinity. Note: This is the plot of the system itself with no controller.

We must install a controller to enhance our system's characteristics and time response. Using the PID Tuner command, the controller is added to the MATLAB program.

The factors that need to be improved are;

- Transient Response
- Steady-State Error
- Settling Time

A PID controller was designed to improve these factors using MATLAB

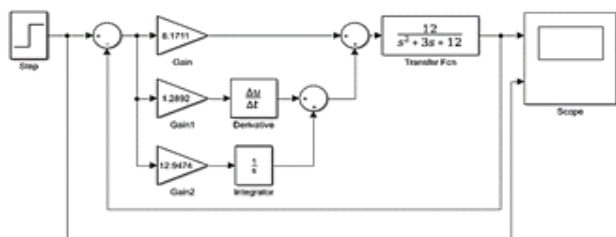


Fig. 18 SIMULINK block diagram

This gave us the following plot;

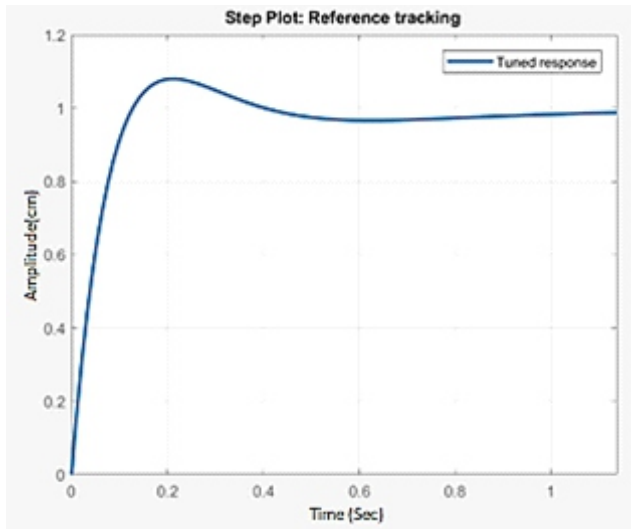


Fig. 19 Tuned response plot of the system

The tuned response has no steady-state error along with the desired settling time and %overshoot.

The controller has significantly improved the percentage overshoot and settling time of the system. This work's main drawback is that the transfer function's initial values were chosen on the assumption that they would provide us with a stable transfer function right away. Thus, the final results show that the PID controller enhanced the system's performance regarding the settling time and overshoot, yet there is still potential for enhancement, specifically with regard to ramp and parabolic inputs. Further studies will be directed towards improving the basic control methods by applying higher level control concepts like adaptive control or model based predictive control in a more complex and uncertain world.

CONCLUSION

This paper examined various approaches to managing the joint angular position of robotic manipulators to achieve the required precision. It covered the importance of accurate dynamic modeling and the impacts of inaccurate modeling. Robotic manipulators were highlighted as particularly useful in tasks requiring high accuracy, as illustrated in medical operations where the human arm is inefficient due to fatigue and imprecision. In this study, we established a robotic link of a manipulator and described an angular control system. After

Performance and Robustness	
	Tuned
Rise time	0.0905 seconds
Settling time	0.938 seconds
Overshoot	7.97 %
Peak	1.08
Gain margin	Inf dB @ NaN rad/s
Phase margin	76.5 deg @ 17.6 rad/s
Closed-loop stability	Stable

Fig. 20 Tuned system characteristics

the modeling, we obtained the transfer function of the system and computed its characteristic values, which confirmed the system's consistency. The system was then tested against various inputs, and the output was graphed and analyzed. Stability margin was assessed before designing the controller, and with the help of the Routh table, the controller gain limits were determined. The root locus was plotted for the transient response and settling time. To address these issues, a PID controller was introduced. With varied the values of K_p , K_i , and K_d , the responses were observed until the results were acceptably stable. Despite a few weaknesses in the process, it was expected that ongoing refinements would continue to increase the model's efficiency. This technique proved useful as it did not involve complex computations and did not require a speed sensor, which is necessary in other advanced robotic manipulators. The results showed that the system stabilized and responded to parabolic inputs with a certain amount of error, maintaining a steady state for a given input even when system parameters were changed. Future work in this area will involve improving the system's ability to operate in a variety of dynamic settings and shortening response times to increase the overall speed and accuracy of the system. Moreover, it is also possible to continue the research of non-linear controllers to enhance the steady-state errors and stability of the system at different operating conditions.

REFERENCE

- [1] B. Panomruttanarug, "Position control of robotic manipulator using repetitive control based on inverse frequency response design," *International Journal of Control, Automation and Systems*, vol. 18, pp. 2830-2841, 2020.
- [2] J. A. Bagnell, F. Cavalcanti, L. Cui, T. Galluzzo, M. Hebert, M. Kazemi, et al., "An integrated system for autonomous robotics manipulation," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 2955-2962.
- [3] L. F. Baptista and J. M. S. da Costa, "Force and position control of robotic manipulators: An experimental approach," *IFAC Proceedings Volumes*, vol. 33, pp. 27-32, 2000.
- [4] J. A. N. Cocota, H. S. Fujita, and I. J. da Silva, "A low-cost robot manipulator for education," in *2012 Technologies Applied to Electronics Teaching (TAE)*, 2012, pp. 164-169.
- [5] T. Umeno, T. Kaneko, and Y. Hori, "Robust servosystem design with two degrees of freedom and its application to novel motion control of robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 40, pp. 473-485, 1993.
- [6] N. Adhikary and C. Mahanta, "Sliding mode control of position commanded robot manipulators," *Control Engineering Practice*, vol. 81, pp. 183-198, 2018.
- [7] R. Lozano, A. Valera, P. Albertos, S. Arimoto, and T. Nakayama, "PD control of robot manipulators with joint flexibility, actuators dynamics and friction," *Automatica*, vol. 35, pp. 1697-1700, 1999.
- [8] S. Kannan, S. Bezzaoucha, S. Quintanar-Guzman, M. A. Olivares-Mendez, and H. Voos, "Adaptive control of robotic arm with hysteretic joint," in *Proceedings of the 4th International Conference on Control, Mechatronics and Automation*, 2016, pp. 46-50.
- [9] T. S. Hsia, T. A. Lasky, and Z. Guo, "Robust independent joint controller design for industrial robot manipulators," *IEEE transactions on industrial electronics*, vol. 38, pp. 21-25, 1991.
- [10] Sira-Ramirez, H., et al. (1992). "Dynamical feedback control of robotic manipulators with joint flexibility." *IEEE Transactions on Systems, Man, and Cybernetics* 22(4): 736-747.
- [11] Puljiz, D., et al. (2019). "Referencing between a Head-Mounted Device and Robotic Manipulators." *Robotics (cs.RO)*.

- | | |
|--|---|
| <p>[12] Yahagi, S. and I. Kajiware (2021). "Direct tuning of gain-scheduled controller for electro-pneumatic clutch position control." <i>Advances in Mechanical Engineering</i> 13(8).</p> <p>[13] Guo, Q., et al. (2015). "Robust $H(\infty)$ positional control of 2-DOF robotic arm driven by electro-hydraulic servo system." <i>ISA Trans</i> 59: 55-64.</p> <p>[14] Butler, J. and M. Tomizuka (1992). "A Suboptimal Reference Generation Technique for Robotic Manipulators Following Specified</p> | <p>Paths." <i>Journal of Dynamic Systems, Measurement, and Control</i> 114(3): 524-527.</p> <p>[15] Zhao, D., et al. (2009). "A new terminal sliding mode control for robotic manipulators." <i>International Journal of Control</i> 82(10): 1804-1813.</p> |
|--|---|
-
-