Atlantis Highlights in Engineering, volume 5
International Conference on Precision Machining, Non-Traditional Machining and Intelligent Manufacturing (PNTIM 2019)

Comparison PID and PDF Control in 2 Degree of Freedom Robotic Manipulator

Qiang Mi
School of Mechatronic Engineering
Xi'an Technological University
Xi'an, China
E-mail: 1822418@chester.ac.uk

Feng Jia
School of Mechatronic Engineering
Xi'an Technological University
Xi'an, China
E-mail: jiafeng@xatu.edu.cn

Yu Bai
School of Mechatronic Engineering
Xi'an Technological University
Xi'an, China
E-mail: baiyv@xatu.edu.cn

Gerard Edwards
Department of Electronic & Electrical Engineering
University of Chester
Chester, United Kingdom
E-mail: gerard.edwards@chester.ac.uk

***Abstract*—This paper compare the differences between Proportional Integral Derivative (PID) and Pseudo Derivative Feedback (PDF) control algorithms in a two degree of freedom (DOF) planar robot manipulator. The PID and PDF control algorithms are compared in MATLAB which is a simple and practical tool for testing algorithms. An ODE45 function of MATLAB was used in order to solve differential equations. The solution of the differential equation contains the force acting on the actuator. Moreover, since the gain is unknown, a mathematical approach was introduced to carry out the gain. Simulation results showed that PDF control algorithm is shown to be superior to PID control algorithm by comparing the responses in MATLAB. Overshooting appeared in the PID algorithm disappears in the PDF algorithm.**

Keywords-PID; PDF; Control; Algorithms; MATLAB; Ode45

I. INTRODUCTION

Systems that require two independent coordinates to describe their motion are called two degree of freedom systems. Most robotic manipulators have 4 or 6 degree of freedom. The motion analysis of multi-degree of freedom requires the solution of partial differential equations, which is quite difficult. In fact, analytical solutions do not exist for many ordinary differential equations. The analysis of a 2 degree of freedom robotic manipulator on the other hand, requires the solution of a set of ordinary differential equations, which is relatively simple. Hence, for simplicity of analysis, multi-degree of freedom robotic manipulator are often approximated as two degree of freedom robotic manipulator.

In this paper, we propose to use MATLAB to estimate both PID and PDF control algorithm.

There is no need to use the production software and hardware during the design process thereby cutting down the cost in design. Moreover, the designer can also refine the system model iteratively and tune controller parameters while controller is running (on-the-fly). These features help to reduce the implementation time, which is important in any industry situations. And then gain experience on a simpler 2D system before tackling a 3D system.

This paper is organized as follows: in the second section, the methods used to estimate the algorithm of PID control and PDF control are presented. The third section is simulation results comparison. In the forth section are conclusions and future work are presented.

II. METHOD

A. Robotic Arm Dynamics Model

The dynamical analysis of the robot investigates a relation between the joint torques/forces applied by the actuators and the position, velocity and acceleration of the robot arm with respect to the time. Robot manipulators have complex non-linear dynamics that might make accurate and robust control difficult. Therefore, they are good examples to test performance of the controllers.

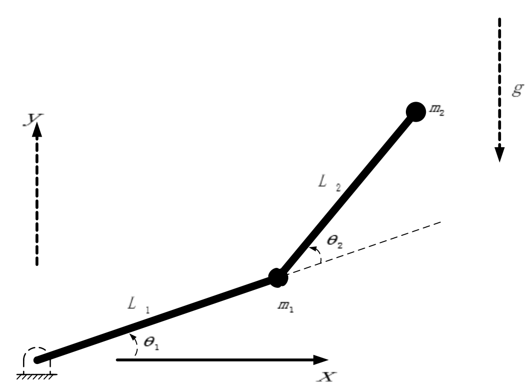


Figure 1. Model of a 2R open chain robot under gravity.

$$x_1 = L_1 \cdot \cos(\theta_1) \quad (1)$$

$$y_1 = L_1 \cdot \sin(\theta_1) \quad (2)$$

$$x_2 = L_1 \cdot \cos(\theta_1) + L_2 \cdot \cos(\theta_1 + \theta_2) \quad (3)$$

$$y_2 = L_1 \cdot \sin(\theta_1) + L_2 \cdot \sin(\theta_1 + \theta_2) \quad (4)$$

So, Kinetic Energy could be formed as

$$KE = \frac{1}{2}(M_1 + M_2)L_1^2\dot{\theta}_1^2 + \frac{1}{2}M_2L_2^2(\dot{\theta}_1^2 + \dot{\theta}_2^2) + M_2L_2^2\dot{\theta}_1\dot{\theta}_2 + M_2L_1L_2\cos(\theta_2)(\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_1^2) \quad (5)$$

And Potential Energy is

$$PE = M_1gL_1\sin(\theta_1) + M_2g(L_1\sin(\theta_1) + L_2\sin(\theta_1 + \theta_2)) \quad (6)$$

So, by Lagrange Dynamics, we form the Lagrangian

$$L = KE - PE \quad (7)$$

So, forming the dynamics equations to be

$$f_{\theta_{1,2}} = \frac{d}{dt}\left[\frac{\partial L}{\partial \dot{\theta}_{1,2}}\right] - \frac{\partial L}{\partial \theta_{1,2}} \quad (8)$$

So, the dynamic equations after simplifications become

$$f_{\theta_1} = ((M_1 + M_2)L_1^2 + M_2L_2^2 + 2M_2L_1L_2\cos(\theta_2))\ddot{\theta}_1 + (M_2L_2^2 + M_2L_1L_2\cos(\theta_2))\ddot{\theta}_2 - M_2L_1L_2\sin(\theta_2)(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) + (M_1 + M_2)gL_1\cos(\theta_1) + M_2gL_2\cos(\theta_1 + \theta_2) \quad (9)$$

$$f_{\theta_2} = (M_2L_2^2 + M_2L_1L_2\cos(\theta_2))\ddot{\theta}_1 + M_2L_2^2\ddot{\theta}_2 + M_2L_1L_2\sin(\theta_2)\dot{\theta}_1^2 + M_2gL_2\cos(\theta_1 + \theta_2) \quad (10)$$

To compare performance of PID and PDF control algorithm, the 2 DOF robot shown in Fig. 1 was selected as an example problem. The dynamic equations of the serial robot are usually represented by the following coupled non-linear differential equations:

$$B(q)\ddot{q} + C(\dot{q}, q) + g(q) = F \quad (11)$$

Where $B(q)$ is the inertia matrix, $C(\dot{q}, q)$ is the Coriolis/centripetal matrix, $g(q)$ is the gravity vector, and F is the control input torque. The joint variable q is an n-vector containing the joint angles for revolute joints. The dynamic equation of the 2 DOF planar robot can be computed by:

$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad (12)$$

$$B(q) = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \quad (13)$$

$$B_{11} = (M_1 + M_2)L_1^2 + M_2L_2^2 + 2M_2L_1L_2\cos(\theta_2) \quad (14)$$

$$B_{12} = M_2L_2^2 + M_2L_1L_2\cos(\theta_2) \quad (15)$$

$$B_{21} = M_2L_2^2 + M_2L_1L_2\cos(\theta_2) \quad (16)$$

$$B_{22} = M_2L_2^2 \quad (17)$$

$$C(\dot{q}, q) = \begin{bmatrix} -M_2L_1L_2\sin(\theta_2)(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) \\ M_2L_1L_2\sin(\theta_2)\dot{\theta}_1^2 \end{bmatrix} \quad (18)$$

$$g(q) = \begin{bmatrix} (M_1 + M_2)gL_1\cos(\theta_1) + M_2gL_2\cos(\theta_1 + \theta_2) \\ M_2gL_2\cos(\theta_1 + \theta_2) \end{bmatrix} \quad (19)$$

$$F = \begin{bmatrix} f_{\theta_1} \\ f_{\theta_2} \end{bmatrix} \quad (20)$$

where M_i is link mass, L_i is link length, g is the gravity and θ_i , $\dot{\theta}_i$ and $\ddot{\theta}_i$, respectively are the joint positions, velocities and accelerations. Here we have:

$$M_1 = M_2 = 1kg \;,\; L_1 = L_2 = 1m$$

B. Control Design

1) PID Design

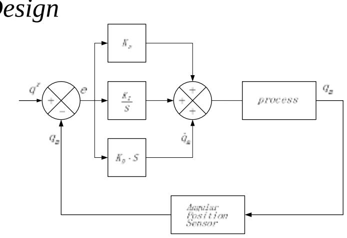


Figure 2. Block diagram of PID control system (s-domain).

General structure of PID controller for any input would be

$$f = K_p e + K_d \dot{e} + K_i \int e dt \qquad (21)$$

So, in our case,

$$f_1 = K_{p1} e(\theta_1) - K_{d1}\dot{\theta}_1 + K_{i1}\int e(\theta_1) dt \qquad (22)$$

$$f_2 = K_{p2} e(\theta_2) - K_{d2}\dot{\theta}_2 + K_{i2}\int e(\theta_2) dt \qquad (23)$$

2) Solution

In order to apply all controls of Proportional-Derivative-Integral actions, a ‘dummy’ state is added for each angle to resemble the integration inside the computer.

$$x_1 = \int e(\theta_1) dt \Rightarrow \dot{x}_1 = e(\theta_1) \qquad (24)$$

$$x_2 = \int e(\theta_2) dt \Rightarrow \dot{x}_2 = e(\theta_2) \qquad (25)$$

So, the complete system equations are

$$\begin{cases} \dot{x}_1 = \theta_{1f} - \theta_1 \\ \dot{x}_2 = \theta_{2f} - \theta_2 \\ \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = B(q)^{-1} \ -C(\dot{q},q) - g(q) \ + \hat{F} \end{cases} \qquad (26)$$

By trial & error, the 2 controllers’ parameters were tuned to have the best performance. The best values for the parameters was found to be

$$K_{p1} = 15,\ K_{d1} = 7,\ K_{i1} = 10.$$

$$K_{p2} = 15,\ K_{d2} = 10,\ K_{i2} = 10.$$

3) PDF Design

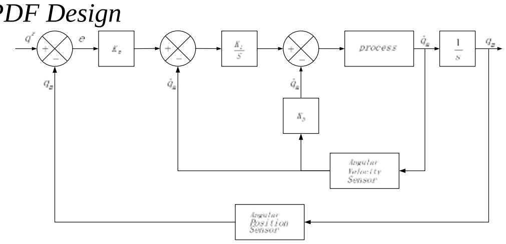


Figure 3. Block diagram of PDF control system (s-domain).

General structure of PDF controller for any input would be

$$f = K_i \int (K_p e + \dot{e}) dt + K_d \dot{e} \qquad (27)$$

So, in our case,

$$f_1 = K_{i1} \int (K_{p1} e(\theta_1) + \dot{e}(\theta_1)) dt + K_{d1}\dot{e}(\theta_1) \qquad (28)$$

$$f_2 = K_{i2} \int (K_{p2} e(\theta_2) + \dot{e}(\theta_2)) dt + K_{d2}\dot{e}(\theta_2) \qquad (29)$$

4) Solution

$$x_1 = \int (K_{p1} e(\theta_1) + \dot{e}(\theta_1)) dt$$

$$\Rightarrow \dot{x}_1 = K_{p1} e(\theta_1) + \dot{e}(\theta_1) \qquad (30)$$

$$x_2 = \int (K_{p2} e(\theta_2) + \dot{e}(\theta_2)) dt$$

$$\Rightarrow \dot{x}_2 = K_{p2} e(\theta_2) + \dot{e}(\theta_2) \qquad (31)$$

So, the complete system equations are

$$\begin{cases} \dot{x}_1 = K_{p1}(\theta_{1f} - \theta_1) - \dot{\theta}_1 \\ \dot{x}_2 = K_{p2}(\theta_{2f} - \theta_2) - \dot{\theta}_2 \\ \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = B(q)^{-1} \ -C(\dot{q},q) - g(q) \ + \hat{F} \end{cases} \qquad (32)$$

By trial & error, the 2 controllers’ parameters were tuned to have the best performance. The best values for the parameters was found to be

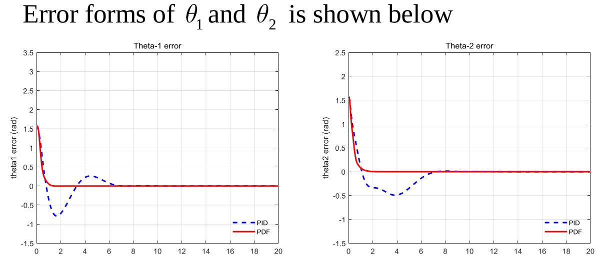
$$K_{p1} = 2.5,\ K_{d1} = 20,\ K_{i1} = 150.$$

$$K_{p2} = 2.5,\ K_{d2} = 20,\ K_{i2} = 350.$$

III. RESULTS

A. *States Results of PID and PDF Control*

Error forms of θ_1 and θ_2 is shown below



(a) (b)

Figure 4. (a) Error wave of theta1 (b) Error wave of theta2.

Comments: You can see from above waveforms that:

- Overshoot in PID controller is larger than in PDF controller.
- The overshoot disappears entirely in PDF controller.
- Settling time in PDF controller is obviously shorter than PID controller

B. Torques Results of PID and PDF control

The waveforms of joints torques are

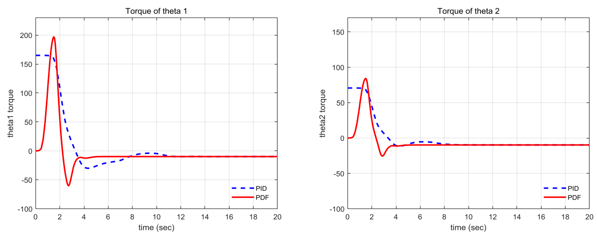


Figure 5. (a) Actual torques on joint 1. (b) Actual torques on joint 2.

Comments: from above plots,

- θ_1 and θ_2 – joint encounter high starting torque in relatively small time in PID controller.
- Overall acceptable performance as relatively energy spent is fine.
- Acceptable starting torque in PDF controller.
- Settling time in PDF control algorithm is obviously shorter than PID control algorithm

Fig. 6 and Fig. 7 show for comparison the performances of a PID controller and a PDF controller in other two situations.

C. Case 1

θ_1 from 0 to $\frac{\pi}{2}$, θ_2 from 0 to $\frac{\pi}{2}$.

A common way to test how well a controller works is to specify a nonzero initial error $\theta_e(0)$ and see how quickly, and how completely, the controller reduces the initial error. A good controller is characterized by

- little or no steady-state error.
- little or no overshoot.
- a short 2% settling time.

The waveform of case 1 is shown in Fig. 4 and Fig. 5. The performance is shown in Tab.1 and Tab.2.

TABLE I. SIMULATION RESULTS COMPARISON OF θ_1

Controller	PID	PDF
Rise time (s)	0.5366	0.5753
%2 settling time (s)	6.3971	1.0984
Overshoot (%)	49.9970	0.0850

TABLE II. SIMULATION RESULTS COMPARISON OF θ_2

Controller	PID	PDF
Rise time (s)	0.8248	0.6441
%2 settling time (s)	6.7731	1.3293
Overshoot (%)	31.2844	5.5305e-06

D. Case 2

θ_1 from 0 to π, θ_2 from $\frac{\pi}{2}$ to $-\frac{\pi}{2}$.

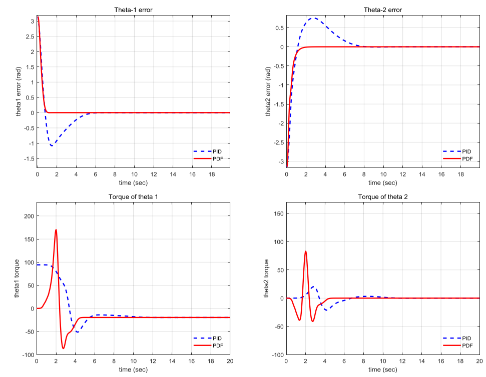


Figure 6. Error wave and actual torques about case 2.

The simulated results were interpreted as shown in Table III and Table IV. The values of the response parameters such as rise time, settling time and percentage overshoot are tabulated. The simulation results show that the system that used PDF controller have faster response than the system that used PID controller, which was expected. The most important part to be observed is the rise time and the settling time, the values of rising time and settling time for the system using the PDF controller is less compared to the system that use the PID controller. Conclusively, the PDF controller reduced rise time, decreased the overshoot and the settling time.

TABLE III. SIMULATION RESULTS COMPARISON OF θ_1

Controller	PID	PDF
Rise time (s)	0.5384	0.5232
%2 settling time (s)	4.9904	0.8730
Overshoot (%)	34.4527	0.0292

TABLE IV. SIMULATION RESULTS COMPARISON OF θ_2

Controller	PID	PDF
Rise time (s)	0.7835	0.6932
%2 settling time (s)	6.8704	1.3609
Overshoot (%)	24.0683	4.0169e-06

E. Case 3

θ_1 from $\frac{\pi}{2}$ to $\frac{3\pi}{2}$, θ_2 from $\frac{\pi}{2}$ to $-\frac{\pi}{2}$.

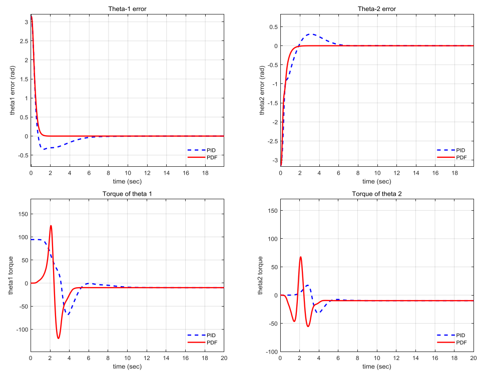


Figure 7. Error wave and actual torques about case 3.

TABLE V. SIMULATION RESULTS COMPARISON OF θ_1

Controller	PID	PDF
Rise time (s)	0.4637	0.5715
%2 settling time (s)	5.3200	1.0408
Overshoot (%)	11.0434	0.0105

TABLE VI. SIMULATION RESULTS COMPARISON OF θ_2

Controller	PID	PDF
Rise time (s)	1.2370	0.7355
%2 settling time (s)	5.4967	1.3970
Overshoot (%)	9.7152	1.6308e-05

IV. CONCLUSIONS AND FUTURE WORK

A. Conclusions

Using simulation experiments, we have compared our PID control algorithm and PDF control algorithm. Many points could be concluded:

- 2-DOF robotic manipulator under the PDF control has good dynamic performance.
- Under the PDF control, 2-DOF robotic manipulator has superior anti-overshoot performance.
- In the PDF control system, high torque of PID controller in start-up is effectively suppressed.
- The PDF control algorithm exhibits a very fast transient response with accurate feedback.
- In same situation, PID control algorithm cannot make a fast transient response with accurate feedback.

B. Future work

Through this work, comparing PID and PDF control algorithm was presented using MATLAB for estimating the error and torque in every joint. However, the model is too simple. In the future, I need to add friction and station into this model. And then transform this model from 2 DOF to 6 DOF and from 2D to 3D. Use 3D model to simulate the robot arm with 6 DOF.

V. ACKNOWLEDGMENT

This research is funded by the Xi'an Technological University President Foundation [Grant No. XAGDXJJ17004] as well as Research Foundation of Education Bureau of Shaanxi Province, China [Grant No. 18JS043].

VI. REFERENCES

[1] Alfaro, V., O., A. & R., V., 2009. Robust tuning of Two-Degree-of-Freedom (2-DoF) PI/PID based cascade control systems. Journal of process control, 19(10), pp. 1658-1670.

[2] Bingül, Z. & Karahan, O., 2011. A Fuzzy Logic Controller tuned with PSO for 2 DOF robot trajectory control. Expert Systems with Applications, 38(1), pp. 1017-1031.

[3] Lochan, K. & Roy, B., 2015. Control of Two-link 2-DOF Robot Manipulator Using Fuzzy Logic Techniques: A Review. In: Das K., Deep K., Pant M., Bansal J., Nagar A. (eds) Proceedings of Fourth International Conference on Soft Computing for Problem Solving. Advances in Intelligent Syste. s.l., Springer, New Delhi.

[4] Mituhiko, A. & Hidefumi, T., 2003. Two-Degree-of-Freedom PID Controllers. International Journal of Control, Automation, and Systems, 1(4), pp. 401-411.

[5] Phelan, R. M., 1971. PSEUDO-DERIVATIVE-FEEDBACK (PDF) CONTROL. LAWRENCE RADIATION LABORATORY University of California LIVERMORE, 9 April, p. 54.

[6] Yukitomo, M., Shigemasa, T., Baba, Y. & Kojima, F., 2004. A two degrees of freedom PID control system, its features and applications. Melbourne, Victoria, Australia, Australia, IEEE.

[7] Samer S. Saab, Rayana H. Jaafar. (2019) A proportional-derivative-double derivative controller for robot manipulators. International Journal of Control 0:0, pages 1-13.

[8] Aleksandr Andreev & Olga Peregudova (2019) Trajectory tracking controlfor robot manipulators using only position measurements, International Journal of Control, 92:7,1490-1496, DOI: 10.1080/00207179.2017.1397755

[9] Rajendiran, S. & Lakshmi, P. J Mech Sci Technol (2016) 30: 4565. https://doi.org/10.1007/s12206-016-0927-6

[10] Hamza Kamişli, Bülent Özkan, Metin U. Salamci, "Design of a cascaded control system for an electromechanically-actuated launcher to reduce the thrust effect", Carpathian Control Conference (ICCC) 2016 17th International, pp. 308-313, 2016.

[11] Dokumaci, K., Aydemir, M.T. & Salamci, M.U. 2014, "Modeling, PID control and simulation of a rocket launcher system", IEEE, , pp. 1283.

[12] Zhang, J., Liu, Z. & Pei, R. 2002, "Two-degree-of-freedom PID control with fuzzy logic compensation", IEEE, , pp. 1498.