## A NOUVELLE MOTION STATE-FEEDBACK CONTROL SCHEME FOR RIGID ROBOTIC MANIPULATORS

Ahmad Manasra, <u>135037@ppu.edu.ps</u> Department of Mechanical Engineering, Palestine Polytechnic University, Hebron, Palestine

**Islam Abukhalaf,** <u>135417@ppu.edu.ps</u> Department of Mechanical Engineering, Palestine Polytechnic University, Hebron, Palestine

**Mahmoud Sharabati**, <u>135277@ppu.edu.ps</u> Department of Mechanical Engineering, Palestine Polytechnic University, Hebron, Palestine

Karim Tahboub, <u>tahboub@ppu.edu</u> Department of Mechanical Engineering, Palestine Polytechnic University, Hebron, Palestine

## 1. ABSTRACT

Although the problem of controlling the motion of rigid robotic manipulators is a well-studied classical problem, a nouvelle method is presented in this article and demonstrated to have its own merits. This paper proposes a simple linear integral control scheme without gravity compensation. A simulation study is performed to compare the performance of different existing methods with the proposed scheme. For this case, the famous PUMA-560 manipulator arm is selected and a motion benchmark is considered. MATLAB/SIMULINK simulation results show that the performance of the proposed method is comparable to that of the inverse-dynamics method with the advantage of not having to compute the model online.

Keywords: Robot motion control, state feedback with integral control, nonlinear control.

# 2. INTRODUCTION

The motion control problem is one of the major control issues for robotic manipulators, where the robot has to track a specific trajectory or to move the end-effector from one point to another [1]. The challenge in the motion control problem is that robotic manipulators have usually highly nonlinear dynamics. The designed controller based on the robot's dynamic model has to be efficient enough to achieve the desired performance and robust against model identification errors.

Most contributions in this field are model based [1]-[5], and some contributions are model-free controllers, see [6] where the author proposed a simple fuzzy controller with integral action with the absence of modeling parameters. Many nonlinear control algorithms have been established for robotic manipulators motion control like inverse dynamic control, and PD with gravity compensation control. These controllers work very well when all dynamic and physical parameters are known. However, their performance degrades when the robot manipulator model has variation in dynamic parameters [1], [2]. Some of the recent proposed schemes are designed with saturation constraints to reduce the actuator failure due to excessive input torque to these

actuators [3]-[5].

This paper proposes a simple linear integral control scheme without gravity compensation to track a reference trajectory with uncertainty. Uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, dynamic parameter variations) [8]. The control effort is distributed over cascade and feedback compensation where the integral (I) part is inserted in forward path whereas the proportional-derivative (PD) part is located in the feedback path to constitute a nouvelle PID control scheme, see Figure 3. The integral part is responsible for robust tracking and disturbance rejection of step-like inputs whereas the PD part is similar to state feedback for stabilizing and improving the system performance. The proposed scheme organized as a state feedback with integral control for each link independently. Unlike the PD control with gravity compensation and inverse dynamic control, the proposed scheme does not need an online computation for the dynamic model, which leads to a reduction in the computation load with a comparable performance with these controllers.

To implement the adopted controller in simulation, the famous PUMA-560 manipulator arm is selected and a motion benchmark is considered. The strategy in this paper is to carry out the simulation for the inverse dynamics control, PD control with gravity compensation and the proposed control scheme to compare the performance between the proposed scheme and the two other schemes.

This paper is organized as follows. In section 3, the dynamics of rigid robotic manipulators is presented. Detail of motion control methods are presented in section 4. In section 5, the simulation results are presented and finally in section 6, the discussion and conclusion are presented.

### 3. Dynamics of rigid robotics manipulators:

Rigid robotics manipulator's dynamic model describes the relationship between the applied joint torques/forces and the motion of the robot [1]. Lagrange formulation is a convenient method to describe the dynamic model for such a complex system. For a robot arm of n-degree of freedom (DOF), let the ( $n \times 1$ ) vector q indicates the joint variables of that robot arm. The general equation of motion of the robot manipulators is given as:

$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau(q) \tag{1}$$

where B(q) is the  $(n \times n)$  mass matrix,  $C(q, \dot{q})\dot{q}$  is the  $(n \times 1)$  Coriolos and centrifugal force vector, g(q) is the  $(n \times 1)$  gravity force vector and  $\tau(q)$  is the  $(n \times 1)$  generalized force vector [1]. Noting that all these matrices and vectors are configuration dependent (function of joint positions), this mathematical dependency emphasizes the physical fact that the robot arm has a varying inertia based on its posture [7].

Articulated robot arm's equation of motion is highly nonlinear; this become obvious by

observing the derived dynamic model for the PUMA-560 in [9], where the authors identified the inertial parameters to obtain an accurate model.

$$\ddot{q} = M^{-1}(q)(\tau - \tau'),$$
 (2)

where,

$$\tau'(q,\dot{q}) = C(q,\dot{q})\dot{q} + g(q)$$

Solving the direct dynamic problem is essential for robot simulation and testing the designed control algorithm [1]. Joint accelerations can be computed from equation (2), joint velocities and joint positions can be computed by integrating the system of nonlinear differential equations in equation (2) [1].

For designing the proposed controller, the mass matrix can be decoupled into constant terms and configuration-dependent terms, this decoupling is shown below [1].

$$B(q) = \bar{B} + \Delta B(q), \tag{3}$$

where,  $\overline{B}$  is a diagonal matrix whose elements represent the resulting average inertia at each joint [1]. Each one of these average inertias will be controlled to have a desired dynamic performance and minimum steady state error.

### 4. Control methods

PD control with gravity compensation and invers dynamics control are presented to be used in the simulation case to compare their results with the proposed control method. These control methods are widely used and investigated in the literature and thus form a valid reference for comparison.

#### PD control with gravity compensation (PDGC)

Let the  $(n \times 1)$  vector  $q_d$  indicates the desired reference trajectory for the motion controller. Choose the control law  $u_{PDGC}$ :

$$u_{PDGC} = g(q) + K_P(q_d - q) - K_D \dot{q}$$
<sup>(4)</sup>



Closed-loop stability is insured by selecting  $K_P$  and  $K_D$  to be positive definite matrices. A Lyapunov global asymptotic stability proof is detailed in [1]. The control action is a combination of a nonlinear compensation of gravity terms and a linear proportional-derivative (PD) action as shown in Figure 1. An on-line computation of the gravity terms, g(q), is required. If the compensation is imperfect, this will lead to imperfection in the motion control [1].

#### **Inverse Dynamic Control (IDC)**

The idea of the inverse dynamics control is to take advantage of the known dynamic model of the rigid manipulator in performing not an approximate linearization but an exact linearization of the system dynamics by compensating for all nonlinearity of the system  $n(q, \dot{q})$  [1].



Figure 2: Block scheme of inverse dynamics control (IDC) [1]

Let the  $(n \times 1)$  vectors  $\ddot{q}_d$ ,  $\dot{q}_d$ , and  $q_d$  indicate the desired trajectory that n - DOF rigid manipulator has to track. Choose the control law  $u_{IDC}$  [1].

$$u_{IDC} = B(q)y + n(q, \dot{q}), \tag{5}$$

where

$$y = \ddot{q}_d + K_d(\dot{q}_d - \dot{q}) + K_P(q_d - q)$$

The global asymptotic stability is insured for that control law in [1]. Two feedback loops are represented; an inner loop based on the manipulator dynamic model, and an outer loop operating on the tracking error, as shown in Figure 2. The function of the inner loop is to obtain a linear and decoupled input/output relationship. Whereas the outer loop is required to stabilize the overall system. To guarantee global asymptotic stability,  $K_p$  and  $K_d$  must be positive definite [1].

#### State Feedback with Integral control (I-PD)

The linear state-feedback control is to place the poles of the linear part of the system in the s-

Figure 1: Block scheme of joint space (PDGC) [1]

plane as desired to achieve the desired response characteristics. The integral action with output feedback is to achieve zero steady state error and to reject external disturbances and be robust against parameter changes [9]. The proposed scheme of the motion control for n - DOF rigid manipulator, is to design an I-PD controller for each link independently. The nonlinearity of the robot in addition to the external disturbances and unmodeled dynamics will be considered as disturbances acting on each link controller.

The control law is given by  $u_{IPD}$  [9]:

$$u_{IPD} = -Kx + K_i x_n, \tag{5}$$

where, x are the states of the system, and  $x_n$  is the additional state from the integral action. K are the state feedback gains,  $K_i$  is the integral gain [9].

The extended system in state space representation is given by [9]:

$$\begin{bmatrix} \dot{x} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} (A - BK) & BK_i \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$
(6)

Where r is the reference trajectory, see Figure 3. The gains K and  $K_i$  are to be designed based on the desired second order response characteristics. The characteristic equation of the system is given as: [9].

$$\Delta = \det(SI - Aext),\tag{7}$$

where,

$$Aext = \begin{bmatrix} (A - BK) & BK_i \\ -C & 0 \end{bmatrix}$$



Figure 3. The proposed control scheme

The state space representation for Link *i* is given as:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{\bar{M}_{ii}} \end{bmatrix} u \tag{8}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \tag{9}$$

where  $\bar{B}_{ii}$  is extracted from matrix  $\bar{B}$  in equation (3).

Although there is still no stability proof for this control scheme, it is assumed that it will stabilize the system and drive it to track the desired trajectory. This is demonstrated by the simulation results.

## 5. Experimental simulation results

Simulations are carried out, that the PUMA-560 robot arm has to track a point to point trajectory for the first three links of the arm. It is desired that the percentage overshoot (%OS) does not exceed 4.6% with 0.5 second settling time (T<sub>s</sub>) with an acceptable steady state error. Accordingly, the damping ratio ( $\zeta$ ) for each link is 0.7 and the natural frequency ( $\omega_n$ ) is 11.4 rad/second for each link. The selected reference trajectory used for all control schemes guarantees that all links move simultaneously to make sure that the nonlinear couplings between links take place.

The choice of  $K_P$  and  $K_D$  for the PDGC is given by the following matrices:

$$K_{P} = \begin{bmatrix} \overline{M}_{11}\omega_{n1}^{2} & 0 & 0 \\ 0 & \overline{M}_{22}\omega_{n2}^{2} & 0 \\ 0 & 0 & \overline{M}_{33}\omega_{n3}^{2} \end{bmatrix} = \begin{bmatrix} 431.02 & 0 & 0 \\ 0 & 837.87 & 0 \\ 0 & 0 & 151.5 \end{bmatrix},$$

$$K_{D} = \begin{bmatrix} 2\overline{M}_{11}\zeta_{1}\omega_{n1} & 0 & 0 \\ 0 & 2\overline{M}_{22}\zeta_{2}\omega_{n2} & 0 \\ 0 & 0 & 2\overline{M}_{33}\zeta_{3}\omega_{n3} \end{bmatrix} = \begin{bmatrix} 52.8 & 0 & 0 \\ 0 & 102.64 & 0 \\ 0 & 0 & 18.56 \end{bmatrix},$$

The choice of  $K_p$  and  $K_d$  for the IDC is given by the following matrices:

$$K_{p} = \begin{bmatrix} \omega_{n1}^{2} & 0 & 0 \\ 0 & \omega_{n2}^{2} & 0 \\ 0 & 0 & \omega_{n3}^{2} \end{bmatrix} = \begin{bmatrix} 130.6 & 0 & 0 \\ 0 & 130.6 & 0 \\ 0 & 0 & 130.6 \end{bmatrix},$$
$$K_{d} = \begin{bmatrix} 2\zeta_{1}\omega_{n1} & 0 & 0 \\ 0 & 2\zeta_{2}\omega_{n2} & 0 \\ 0 & 0 & 2\zeta_{3}\omega_{n3} \end{bmatrix} = \begin{bmatrix} 16 & 0 & 0 \\ 0 & 16 & 0 \\ 0 & 0 & 16 \end{bmatrix},$$

The gains for each I-PD controller is shown in Table 1:

Table 1. I-PDs gains

	<i>K</i> <sub>1</sub>	<i>K</i> <sub>2</sub>	K <sub>i</sub>
First joint I-PD	$9.5 \times 10^{3}$	$0.452 \times 10^{3}$	$1.0053 \times 10^5$
Second joint I-PD	$1.84 \times 10^{4}$	$0.088 \times 10^{4}$	$1.954 \times 10^{5}$

11111 JOINT 1-FD 2.539 X 10 0.159 X 10 5.555 X 10	Third joint I-PD	3.339 × 10 <sup>3</sup>	$0.159 \times 10^{3}$	$3.533 \times 10^{4}$
---	------------------	-------------------------	-----------------------	-----------------------

The time response for the trajectory for the first three links is shown in Figures 4, 5 and 6 respectively. IDC, PDGC and I-PD have almost the same steady state error. In the middle of the trajectory the error of each controller is shown in Table 2:

Table 2. Maximum error during the middle of the trajectory

		<u> </u>	
	IDC (rad)	PDGC (rad)	I-PD (rad)
First joint	0	0.0268	0.0389
Second joint	0	0.0662	0.0468
Third joint	0	0.0825	0.0593







Torques for the first three links is shown in Figures 7, 8 and 9 respectively.



The consumed energy by each joint actuator during the specified trajectory is computed through the computation of the mechanical work exerted during that trajectory, see Table 3. The mechanical work is given by:

$$W = \int_{t_0}^{t_f} \tau \, . \, \omega \, dt$$

Table 5. The consumed energy by each joint actuator for each control scheme	Table 3. The consumed	lenergy	by each	joint actuator	for each	control	scheme
---	-----------------------	---------	---------	----------------	----------	---------	--------

	IDC (J)	PDGC (J)	I-PD (J)
First joint actuator	0.69	0.96	0.97
Second joint actuator	23.97	24.16	24.04
Third joint actuator	4.30	4.80	4.78

### 6. Conclusion

The proposed control scheme; which is designed to n state feedback with integral control for each link independently for rigid robotics manipulators, shows that it can be compared to inverse dynamic control, and PD control with gravity compensation. The results of the simulation for the PUMA-560 robotics manipulator with the control schemes shows the robust tracking for the proposed scheme. The error during the specified trajectory in the proposed scheme is acceptable in comparison with the other control schemes, moreover the energy consumption in the I-PD control scheme lies between those of PDGC and IDC control schemes.

The proposed scheme reduces the complexity of the existing robotic motion control schemes; by

using a simple linear control scheme without the need to an online computation for the dynamic model. It is responsible for robust tracking and disturbance rejection so that the performance is comparable to the other existing control schemes.

Finally, the proposed scheme needs to be validated by implementing it in a real-world robotic manipulator to ensure that the proposed scheme can be as efficient as the simulation results have shown.

# References

[1] Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010). Robotics: Modelling, Planning and Control. Springer London. Retrieved from <u>https://books.google.ps/books?id=jPCAFmE-logC</u>.

[2] Siciliano, B., & Khatib, O. (Eds.). (2016). Springer handbook of robotics. Springer.

[3] Su, Y., & Swevers, J. (2013, February). A new asymptotic tracking approach for robot manipulators with actuator saturation. In Mechatronics (ICM), 2013 IEEE International Conference on (pp. 652-657). IEEE.

[4] Su, Y., Muller, P. C., & Zheng, C. (2010). Global asymptotic saturated PID control for robot manipulators. IEEE Transactions on Control Systems Technology, 18(6), 1280-1288.

[5] Flacco, F., De Luca, A., & Khatib, O. (2015). Control of redundant robots under hard joint constraints: Saturation in the null space. IEEE Transactions on Robotics, 31(3), 637-654.

[6] Zheng, C., Su, Y., & Mercorelli, P. (2017, July). A simple fuzzy controller for robot manipulators with bounded inputs. In Advanced Intelligent Mechatronics (AIM), 2017 IEEE International Conference on (pp. 1737-1742). IEEE.

[7] F. Fahimi, Autonomous robots: modeling, path planning, and control. Springer Science & Business Media, 2008.

[8] Piltan, F., Yarmahmoudi, M. H., Shamsodini, M., Mazlomian, E., & Hosainpour, A. (2012). PUMA-560 robot manipulator position computed torque control methods using Matlab/Simulink and their integration into graduate nonlinear control and Matlab courses. International Journal of Robotics and Automation, 3(3), 167-191.

[9] Armstrong, B., Khatib, O., & Burdick, J. (1986, April). The explicit dynamic model and inertial parameters of the PUMA 560 arm. In Robotics and Automation. Proceedings. 1986 IEEE International Conference on (Vol. 3, pp. 510-518). IEEE.

[10] Nise, N. S. (2007). CONTROL SYSTEMS ENGINEERING, (With CD). John Wiley & Sons.