International Journal of Innovative Technology and Interdisciplinary Sciences
www.IJITIS.org
ISSN: 2613-7305
Volume 2, Issue 4, pp. 316-325, 2019
DOI: https://doi.org/10.15157/IJITIS.2019.2.4.316-325
Received November 1, 2019; Accepted November 20, 2019

# Advanced Method for Motion Control of a 3 DOFs Lower Limb Rehabilitation Robot

**Le Thi Hong Gam[a1], Dam Hai Quan[b2], Bui Trung Thanh[c3], Pham Van Bach Ngoc[d4]**

[1]Thai Nguyen University of Education, Thai Nguyen City, Vietnam

[2]Thai Nguyen College of Economics and Finance, Thai Nguyen City, Vietnam

[3]Hung Yen University of Technology and Education, Khoai Chau, Hung Yen, Vietnam

[4]VietNam Academy of Science and Technology, Ha Noi, Vietnam

[a]lehonggam@dhsptn.edu.vn, [b]damhaiquan@gmail.com, [c]buitrungthanh@utehy.edu.vn, [d]phamvanbachngoc@gmail.com

## ABSTRACT

This paper presents two motion control methods for a lower limb rehabilitation robot based on compensate gravity proportional-derivative and inverse dynamic proportional-derivative (PD) control algorithms. The Robot's mechanism is comprised of three active joints: hip joint, knee joint and ankle joint, which are driven by DC motors. Firstly, based on Robot's mechanism, a dynamic model of the Robot is built. Then, based on Robot's model, motion control systems for Robot are designed. Simulation results show good performances and workability of these proposed controllers. Finally, the calculation of the joint angle errors and toque of each controller is performed. The comparison of simulation results between proposed controllers and the adaptive fuzzy controller allows to choice suitable motion control methods for Robot that can meet the requirements of a 3 DOFs lower limb rehabilitation robot for post-stroke patient.

***Keywords:*** Rehabilitation robot, compensate gravity, motion control, inverse dynamic

## 1. INTRODUCTION

To assist post-stroke patients who suffer from motion after stroke and reduce the stress of physiotherapists, a 3 DOFs lower limb rehabilitation robot was designed and proposed in our previous paper [1]. This robot was designed for using in all stages of rehabilitation after stroke, including: early rehabilitation stage with continue repetitive passive exercises to maintain joint motion range and reduce muscle atrophy; intermediate training stage with active support exercises to encourage patients to strengthen their own advocacy efforts; advanced training stage with active and anti-active training mode helps to expand the range of joint motion and strengthen patient's muscles. In order to get satisfying control effect, various control methods have been used for Robot which is suitable to each stage of motion. This paper focuses on building advanced controllers for motion control of robot, which are used in the early rehabilitation stage of the post-stroke patient.
In the early rehabilitation stage, patients need to be supported in the passive mode with continue repetitive passive exercises. The position control strategy with the trajectory tracking control method is appropriate and required as shown in [2]. Various position control methods for lower limb rehabilitation robots have been proposed. Trajectory

generation methods have been desinged and implemented by Emken et al. in [3] with "teach-and-replay" technique for ARTHuR robot. In their research at [4], Vallery H et al. proposed the Complementary Limb Motion Estimation (CLME) method, which is implemented on the walking gait robot. LOPES or the "path control" method is introduced by Duschau Wicker et al. with "patient-cooperative" strategy that allows patients to automatically control their leg movements actively [5].

Once the desired motion pattern is determined, trajectory tracking control strategy to guide patient's limbs on the reference trajectory have been developed, including: i) a computed-torque controller with inverse dynamics model is used for the ankle rehabilitation robot ARBOT [6]; ii) an adaptive and robust learning control was developed by Renquan L. et al. to solve time-varying uncertainties in robotic model of a 4 DOFs lower limb rehabilitation [7]; iii) a sliding mode controller was used to designed a trajectory tracking controller in an orthosis robot [8] or a fuzzy logic controller, which work with a disturbance observer to compensate the no-linear characteristics of pneumatic muscle actuators in an ankle parallel rehabilitation robot [9]; iv) a fuzzy logic program was used in a lower limb rehabilitation robot system to create a fit in human-robot interaction [10]. Although various controllers have been used in rehabilitation robots, it is still difficult to achieve the expected results when applying model-based controllers with dynamic and uncertain systems.

In this paper, we propose two advanced motion control methods for the 3 DOFs lower limb rehabilitation robot based on compensate gravity proportional-derivative (CGPD) and inverse dynamic proportional-derivative (IDPD) control algorithms. This paper is organized as follows: Section 2 presents advanced methods for motion control of the Robot. Simulation results and discussion are presented in Section 3. Finally, Section 4 concludes this paper.

## 2. ADVANCED MOTHODS FOR MOTION CONTROL OF THE ROBOT

### 2.1 Robot's Mechanism and Dynamic Model

In our previous research [1], we have investigated the mathematical model of the 3 DOFs lower limb rehabilitation robot. The mechanical structure of the robot provides supports for rehabilitation training with three main joints of lower limb, including hip, knee and ankle joints. Movement of each joint is driven by a DC motor. The mechanism of Robot is shown in Fig. 1.

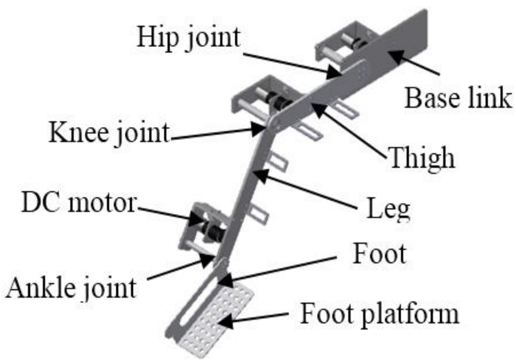



**Fig. 1.** Mechanism of Robot

The dynamic model of the Robot is written in matrix form as [1]:

$$M(q)\ddot{q} + V(q,\dot{q}) + G(q) = \tau \qquad (1)$$

Where:

+ M = $\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{11} \end{bmatrix}$ is the manipulator inertia matrix,

+ $V(q,\dot{q}) = [V_1 \quad V_2 \quad V_2]^T$ is the velocity coupling vector,

+ $G(q) = [G_1 \quad G_2 \quad G_2]^T$ is the gravitational vector,

+ $\tau = [\tau_1 \quad \tau_2 \quad \tau_2]^T$ is the vector of generalized forces,

+ $q = [\theta_1 \quad \theta_2 \quad \theta_3]^T$ is vector of generalized Lagrange coordinates.

## 2.2 Compensate gravity proportional-derivative control algorithm

Block diagram of the proposed compensate gravity proportional-derivative **(**CGPD) controller is shown in Fig.2.

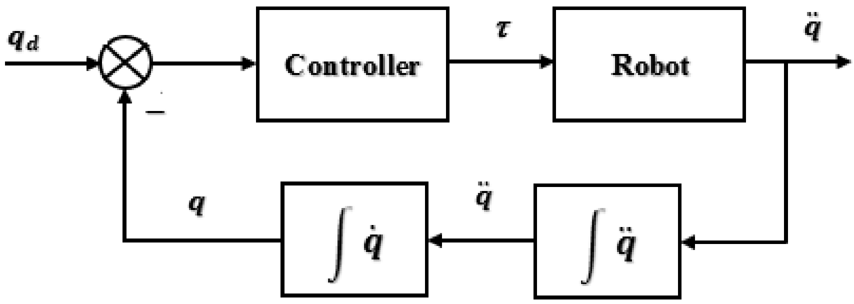



**Fig.2.** Block diagram of the CGPD controller

Based on Liapunov's stability direct method as shown in [11], we build a control law with Lyapunov function is selected below:

$$V_L = \frac{1}{2}[e^T K_p e + \dot{q}^T M(q)\dot{q}] \qquad (2)$$

Where:

$[e^T, \dot{q}^T]^T$ are state variables, $V_L$ is total energy of system, $\frac{1}{2}e^T K_p e$ is potential energy accumulation with $K_p$ gain, $\frac{1}{2}\dot{q}^T M(q)\dot{q}$ is robot kinetic energy.

Because $K_p$ is positive symmetric matrix, $V_L > 0$ with $e, \dot{q} \neq 0$. The first order derivative of $V_L$ is calculated:

$$\dot{V}_L = \frac{1}{2}\dot{e}^T K_p e + \frac{1}{2}e^T K_p \dot{e} + \frac{1}{2}\ddot{q}^T M(q)\dot{q} + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \frac{1}{2}\dot{q}^T M(q)\ddot{q} \qquad (3)$$

Since $q_d$ is constant so $\dot{e} = -\dot{q}$

M(q) is positive symmetric matrix:

$$\dot{e}^T K_p e = e^T K_p \dot{e} \text{ và } \ddot{q}^T M(q)\dot{q} = \dot{q}^T M(q)\ddot{q} \qquad (4)$$

Using these constraints, Eq. (3) can be written as follow:

$$\dot{V}_L = -\dot{q}^T K_p e + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \dot{q}^T M(q)\ddot{q} \qquad (5)$$

Eq. (1) is rewritten as below:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \qquad (6)$$

With:

$C(q,\dot{q})\dot{q} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{11} \end{bmatrix}$ is Centrifugal and Coriolis matrices

From Eq. (6) we obtain the inertial matrix:

$$M(q)\ddot{q} = \tau - C(q,\dot{q})\dot{q} - G(q) \tag{7}$$

Substituting Eq. (6) into Eq. (5), and by adding or removing $q^T K_d \dot{q}$ (with $K_D$ is positive symmetric gain matrix) we obtain:

$$\dot{V}_L = -\dot{q}^T K_p e + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \dot{q}^T[\tau - C(q,\dot{q})\dot{q} - G(q)] + q^T K_d \dot{q} - q^T K_d \dot{q}$$

$$= \frac{1}{2}\dot{q}^T[\dot{M}(q) - 2C(q,\dot{q})]\dot{q} + \dot{q}^T[\tau - G(q) - K_p e + K_d \dot{q}] - q^T K_d \dot{q} \tag{8}$$

Note that the properties of dynamics function [12]:

$$M(q) - 2C(q,\dot{q}) = 0 \tag{9}$$

Chose $\tau = G(q) + K_p e - K_d \dot{q}$ then:

$$\dot{V}_L = -\dot{q}^T K_p \dot{q} < 0 \tag{10}$$

With condition (4) and (10), system will be absolute stable around stability point with position joint error, $e = 0$, and $q = q_d$.

The CGPD controller is chosen below:

$$\tau_{đk} = G(q) + K_p e - K_d \dot{q} \tag{11}$$

With two parts:

+ $G(q)$ - gravity compensation part
+ $K_p e - K_d \dot{q}$ Proportional - derivative (PD) terms

**2.3 Inverse dynamic proportional-derivative control algorithm**

Because the robot is a nonlinear system, we select a control law which can eliminate the nonlinear components of the dynamic model and separate kinematic characteristic of links. Then, we will get a linear system and can be easily to design a controller based on the classical method of linear system to ensure the accurate motion as required. Block diagram of the proposed Inverse dynamic proportional-derivative **(**IDPD) controller is shown in Fig.3.

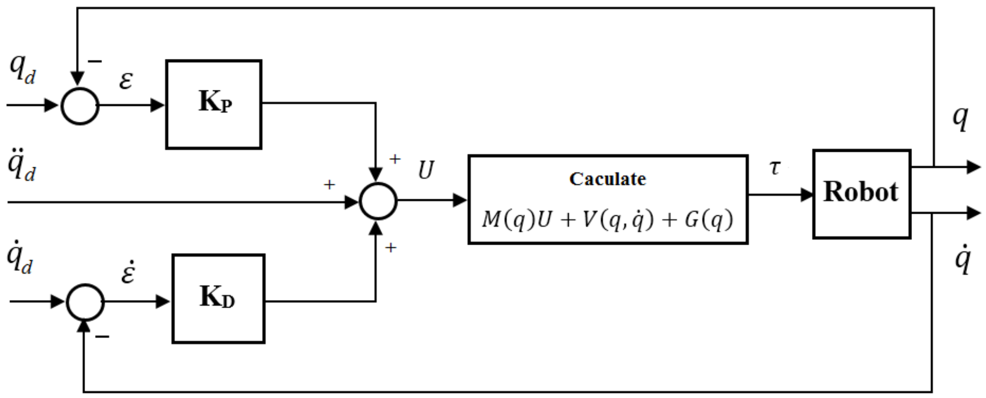



**Fig.3.** Block diagram of the IDPD controller

Base on robot dynamic model, the control law is chosen below:

$$\tau = M(q)U + V(q,\dot{q}) + G(q) \tag{12}$$

Where:

$U \epsilon R^n$ is control signal vector

Equated Eq. (1) and Eq. (12), because M(q) is a positive symmetric matrix, we obtain second order linear differential equation:

$$\ddot{q} = U \tag{13}$$

Eq. (13) is a second-order linear differential equation, which depends on each joint, so we can design PD controllers for each joint. The PD control law is proposed as below:

$$U = \ddot{q}_d + K_p e + K_d \dot{e} \tag{14}$$

With: $\ddot{q}_d = \frac{d}{dt}\dot{q}_d$ ; $e = q_d - q, \dot{e} = \dot{q}_d - \dot{q}$ are joint acceleration, angle error, velocity error, respectively; $K_p \in R^{nxm}, K_d \in R^{nxm}$ are positive diagonal matrices.
Substituting Eqs. (13) into (14), we obtain:

$$\ddot{q} = \ddot{q}_d + K_p e + K_d \dot{e} \quad \rightarrow \quad \ddot{e} + K_d \dot{e} + K_p e = 0 \tag{15}$$

Where: $\ddot{e} = \ddot{q}_d - \ddot{q}$ is joint acceleration error.
With $i^{th}$ joint:

$$\ddot{e}_i + K_{di}\dot{e}_i + K_{pi}e_i = 0 \tag{16}$$

Characteristic equation in Laplace form can be written as:

$$s^2 + K_{di}s + K_{pi} = 0 \tag{17}$$

$K_{pi}$, $K_{di}$ are calculated follow stability criteria.

As shown in Fig.3 $q \in R^n, \dot{q} \in R^n$ are actual angle joint and actual angle velocity vector, respectively; $q_d \in R^n, \dot{q}_d \in R^n$ are desired angle joint and desired angle velocity vector, respectively; $\tau \in R^n$ is moment of actuator joint; $U \in R^n$ is control signal vector.

## 3. SIMULATION RESULTS AND DISCUSION

### 3.1 Simulation results of the compensate gravity PD cotroller

Equated the Robot's dynamic equation (1) with the equation of the CGPD controller (10), we have:

$$M(q)\ddot{q} + V(q,\dot{q}) + G(q) = G(q) + K_p(q_d - q) - K_d\dot{q}$$
$$=> M(q)\ddot{q} + V(q,\dot{q}) = K_p(q_d - q) - K_d\dot{q}$$
$$=> \ddot{q} = M^{-1}(q)[\,K_p(q_d - q) - K_d\dot{q} - V(q,\dot{q})] \tag{18}$$

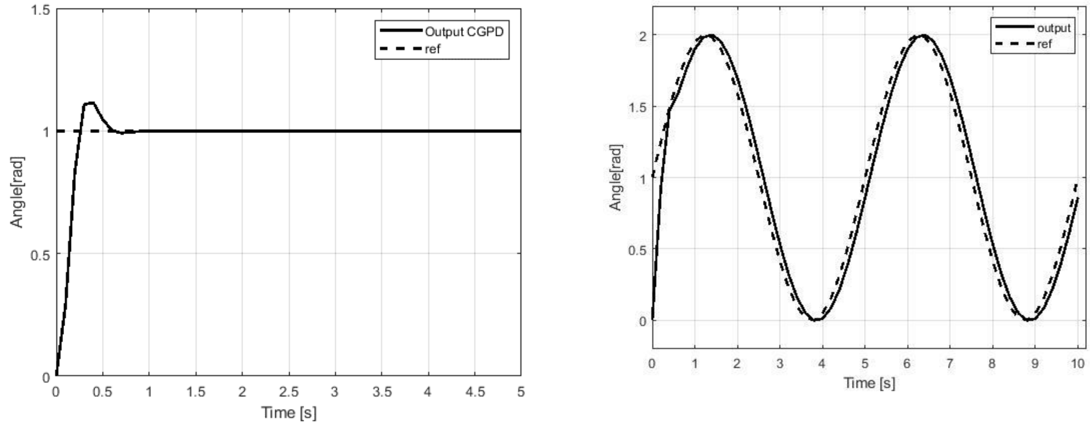
From Eqs.(18), a simulation model is built in Simulink of MATLAB with system parameters are shown in Table 1.

Table 1. Parameters of system

| **Parameter** | **i = 1** | **i = 2** | **i = 3** |
|---|---|---|---|
| $m_i$ (kg) | 3.2 | 3.1 | 0.25 |
| $L_i$ (m) | 0.4 | 0.48 | 0.33 |
| $a_i$ (m) | 0.2 | 0.24 | 0.165 |
| $F_s$ (Nm) | 0.9 | 0.9 | 0.9 |
| $F_c$(Nm) | 0.8 | 0.8 | 0.8 |
| $b_i$(Nm/rad/s) | 1.2 | 1.2 | 1.2 |
| $K_m$(Nm/A) | 0.08 | 0.08 | 0.08 |
| $K_e$(V/rad/s) | 0.287 | 0.287 | 0.287 |
| $L_m$ (H) | 0.0026 | 0.0026 | 0.0026 |
| $R_m$ (Ω) | 1.2 | 1.2 | 1.2 |
| $I_m$ (kg m$^2$) | 0.000025 | 0.000025 | 0.000025 |
| $K_g$ | 19 | 19 | 19 |

Parameters of controller are selected as following:

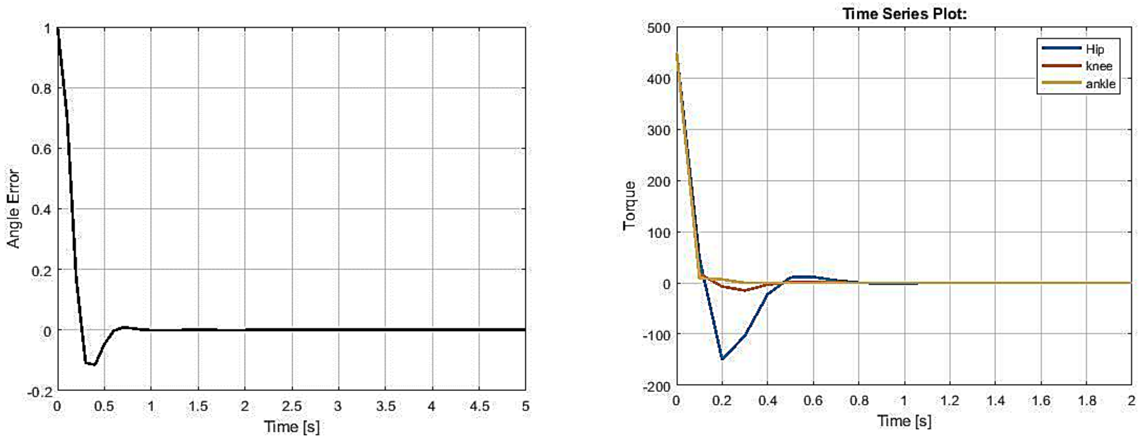
$K_p$=[450 0 0;0 450 0;0 0 450],
$K_D$=[50 0 0;0 50 0;0 0 50]
and q_design =[1.5,1,1.2]




a. Hip angle with reference angle is equal to 1 (rad)

b. Hip angle with reference angle is sinusoidal function

**Fig. 4**. Simulation results of the CGPD controller




**a.** Angle error of hip joint with the CGPD controller

**b.** Toque of hip, knee and ankle joints with the CGPD controller

**Fig. 5**. Angle error and torque of joints with the CGPD controller

The simulation of hip, knee and ankle joints are implemented. Because the results for three DOFs are quite similar, we present the simulation results of hip angle, which simulated with two different reference angles as shown in Fig.4. Angle error and toque of hip, knee and ankle joints are shown in Fig.5.

These results show that the CGPD controller in the joint space has been designed to be stabilize with the following performances:

i) Transient time is small. because the frictions of the mechanical system, the inertia of the actuator and disturbances were not taken into account during the simulation. In fact, it is certain that transient time will be larger.

ii) Overshoot is about 15%.

iii) Steady-state error: To evaluate steady-state error of the system, the Integral of the Square of the Error (ISE) is used as below [11]:

$$J_{ISE} = \int_0^{+\infty} e^2(t)dt \tag{19}$$

The designed CDPD controller has $J_{ISE} = 1.06$.

To investigate the affection of $K_p$, $K_d$ to controller performance, we try:

i) Keep Kp, when increasing Kd in range from 15 to 20, then the oovershoot and ttransient time decreasing and vice versa.

ii) Keep Kd, when increasing Kp in range from 100 to 180, then the ttransient time decreasing and vice versa.
Fig.5 show that with sinusoidal desired trajectory, after the transient time is about 0.6s, the actual joint angle tracks the trajectory.

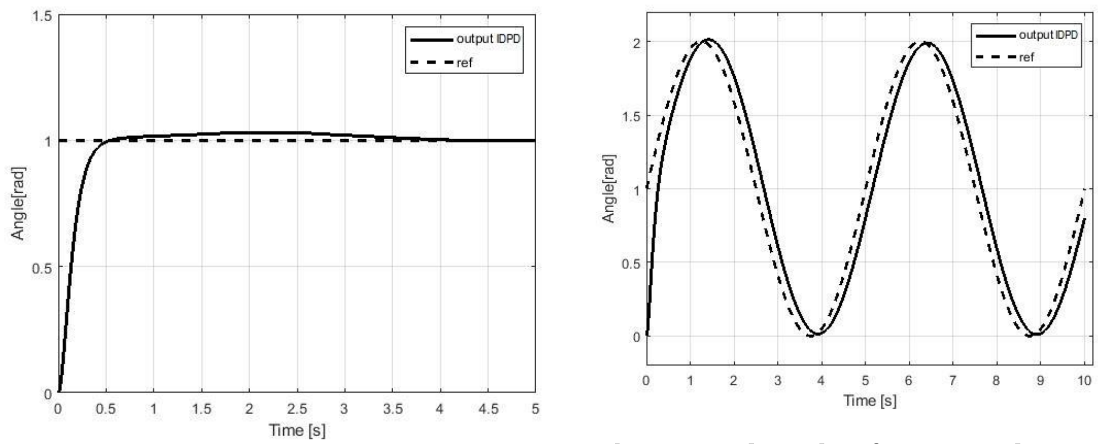
### 3.2 Simulation results of the inverse dynamic PD controller

Based on controller proposed in section 2.3, the simulation is performed in the MATLAB/Simulink software. Model parameters are summarized in Table 1. Parameters of controller are chosen as follows:

$$K_p = [156.25 \;\; 0 \;\; 0;0 \;\; 156.25 \;\; 0;0 \;\; 0 \;\; 156.25]$$

$$K_d = [25 \;\; 0 \;\; 0;0 \;\; 25 \;\; 0;0 \;\; 0 \;\; 25]$$

Simulation results of hip joint angle with two desired angle are shown in Fig.6. Error of joint angle and actuated joint torque are shown in Fig. 7.

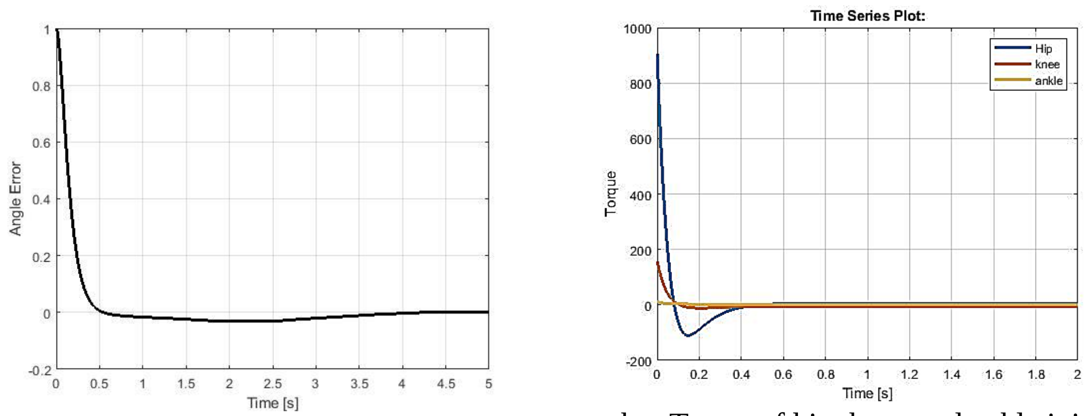



a. Hip angle with reference angle is equal to 1 (rad)

b. Hip angle with reference angle is sinusoidal function

**Fig.6**. Simulation results of the IDPD controller

Simulation results of inverse dynamic controller shows that overall parameters are satisfactory.




a. Angle error of hip joint with the IDPD controller

b. Toque of hip, knee and ankle joints with the IDPD controller

**Fig. 7**. Angle error and torque of joints with the IDPD controller

To compare the performance of two proposed controllers, the Integral of the Square of the Error (ISE) is used. The results of calculation on both controllers show ISE value of CGPD controller ($J_{ISE}$=1.06) is less than the value of IDPD controller ($J_{ISE}$=10.12).

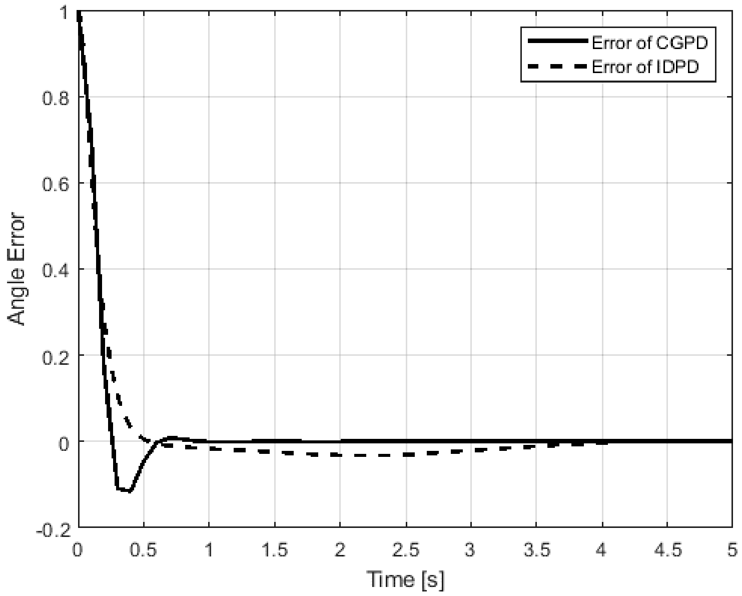



**Fig. 8.** Comparing angle error between two controllers

The angle joint error of both controller in Fig.8 shows that the angle joint error of the CGPD controller is larger than IDPD controller in transient state, but much less than the value of the IDPD controller in steady state. Table 2 shows comparisons of the performance between two controllers.

Table 2. Compare the performance of two controllers

| **Specifications** | **CGPD controller** | **IDPD controller** |
|---|---|---|
| Transient time | 0.6 s | 0.5s |
| Overshoot | 15% | ≈ 0 |
| Steady-state error ($J_{ISE)}$ | 1.06 | 10.12 |

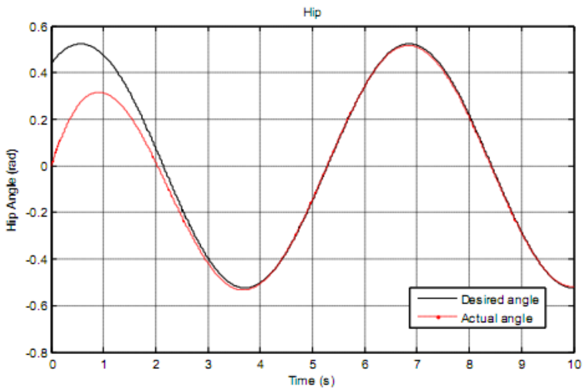



**Fig. 9.** Hip angle with sinusoidal input of the adaptive fuzzy controller [13]

In addition, to evaluate the effectiveness of the proposed controller, the comparison of the trajectory tracking of propose controllers with the adaptive fuzzy controller which is studied in [13] was conducted. Fig. 9 is hip angle with sinusoidal input of the adaptive fuzzy controller. Fig.4b, Fig.6b and Fig.9 show the trajectory tracking error of the CGPD controller, the IDPD controller and the trajectory tracking of the adaptive fuzzy controller are similar but transient time of the advanced methods presented are better (about 2s with the adaptive fuzzy controller and 0.5s with propose controllers).

## 4. CONCLUSION

This paper presented advanced methods for motion control a 3 DOFs lower limb rehabilitation robot used for improving lower limb motions exercises which are used in the early rehabilitation stage of the post-stroke patient. In the early rehabilitation stage, patients need to be supported in the passive mode with continue repetitive passive exercises, so the position control strategy with the trajectory tracking control method is appropriate.

Based on dynamic model of the 3 DOFs lower limb rehabilitation has been studied, two advanced control methods for motion control of this robot was designed and simulated. The CGPD controller was built with two parts, G(q) - gravity compensation part and $(K_p e - K_d \dot{q})$ - PD part, which is used to compensate the gravity of robot and the patient's leg and control Robot to perform tasks. The IDPD controller was built with robot inverse dynamic model and could eliminate the nonlinear components of the Robot and separate kinematic characteristic of links. It can also control Robot to perform the moving in the desired trajectory task.

Simulation results show that both controllers meet the performance of a motion control system. The comparison of the performance between two proposed controllers with the adaptive fuzzy controller shows that the performance of two proposed controllers are better. The advanced methods presented in this study will serve as framework for the design and development of patient-oriented rehabilitation exercises. In the future study, we plan to present some experiment test results to compare with the simulation results. Further studies will also focus on the integration of the force control in the control algorithms to encourage patients to strengthen their own advocacy efforts during training process with active and anti-active exercises in later stages of the rehabilitation of post-stroke patients.

## REFERENCES

[1] Le Thi Hong Gam, Dam Hai Quan, Bui Trung Thanh, Pham Van Bach Ngoc. Mathematical Modeling and Simulation of 3 DOFs Lower Limb Rehabilitation Robot for Post-Stroke Patient. Journal of Science and Technology – ISSN 2354-0575, Vol. 21, 2019, p. 8-14.

[2] Wei Meng, Quan Liu, Zude Zhou, Qingsong Ai, Bo Seng, Sengquan Xie. Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation. Mechatronics 2015, Vol. 31, p. 132-145.

[3] Jeremy L. Emken, Susan J. Harkema, Janell A. Beres-Jones, Christie K. Ferreira, and David J. Reinkensmeyer. Feasibility of Manual Teach-and-Replay and Continuous Impedance Shaping for Robotic Loco motor Training Following Spinal Cord Injury. IEEE Transactions on Biomedical Engineering, Vol. 5, No.1, p. 322-334.

[4] Vallery H, van Asseldonk EHF, Buss M, van der Kooij H. Reference trajectory generation for rehabilitation robots: complementary limb motion estimation. IEEE Transaction on Neural Systems Rehabilitation Engineering; Vol.17, No.1, p. 23–30.

[5] Duschau-Wicke A, von Zitzewitz J, Caprez A, Luenenburger L, Riener R. Path control: a method for patient-cooperative robot-aided gait rehabilitation. IEEE Transaction on Neural Systems Rehabilitation Engineering; Vol.18, p. 38–48.

[6] Saglia JA, Tsagarakis NG, Dai JS, Caldwell DG. Control strategies for patient assisted training using the ankle rehabilitation robot (ARBOT). IEEE/ASME Trans Mechatron 2012, p.1–10

[7] Renquan L, Zhijun L, Chun-Yi S, Anke X. Development and learning control of a human limb with a rehabilitation exoskeleton. IEEE Trans Ind Electron 2014; Vol. 61, p. 3776–85.

[8] Hussain S, Xie SQ, Jamwal PK. Control of a robotic orthosis for gait rehabilitation. Robot Autonom Syst 2013; Vol. 61(9), p. 911–19.

[9] Prashant K. Jamwal, Sheng Q. Xie, Shahid Hussain, and John G. Parsons. An Adaptive Wearable Parallel Robot for the Treatment of Ankle Injuries. IEEE/ASME Transactions on Mechatronics, 2014, Vol. 19(1), p. 64–75.

[10] Martin P, Emami MR. Aneuro-fuzzy approach to real-time trajectory generation for robotic rehabilitation. Robot Autonom Syst 2014; Vol. 62(4), p. 568–78.

[11] Nguyen Manh Tien. Industrial Robot Control. Science and Engineering Publishing House, 2007.

[12] Lorezo Sciavicco, Bruno Siciliano. Modling and control of Robot Manipulators. MacGraw Hill Companies Inc. 1996.

[13] Trung Hai Do, Duc Tan Vu. An Intelligent Control for Lower Limb Exoskeleton for Rehabilitation. SSRG International Journal of Electrical and Electronics Engineering (SSRG-IJEEE) – Volume 4 Issue 8 – August 2017, p. 13-19.