Mechatronics 51 (2018) 85–96

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Trajectory tracking using online learning LQR with adaptive learning control of a leg-exoskeleton for disorder gait rehabilitation☆

Noppadol Ajjanaromvat*, Manukid Parnichkun

Department of Mechantronics, Asian Institute of Technology, P.O. Box 4, Klong Luang, Pathumthani 12120, Thailand

ARTICLE INFO

Keywords:
Leg-exoskeleton
Non-linear system
Iterative learning control
LQR on non-linear system
Adaptive controller

ABSTRACT

Precise trajectory tracking of gait pattern under varied load condition is necessary for rehabilitation using leg exoskeleton. In this paper, online iterative learning linear quadratic regulator (OILLQR) with adaptive iterative learning control is proposed to control trajectory tracking of a leg exoskeleton for rehabilitation developed at AIT. The algorithm determines an optimized weight matrix of the conventional LQR controller by using online learning method for every point of the trajectory and also the controller gain of each point. Iterative learning control gain is then optimized according to the optimized gain found from OILLQR algorithm which results in good tracking performance. The proposed control algorithm can significantly shorten the learning time of the system compared to the conventional PID and LQR controller with iterative learning control. Simulation and experimental results confirm short learning time of at least 2 times and good tracking performance of at least 40% reduction in tracking error for the proposed algorithm. An experiment on test subject with the system confirms good tracking performance which is suitable for gait disorder patient.

1. Introduction

For the past two decades a repetitive movement is having a big role in both industrial and medical treatment fields. Problems in most repetitive movement control algorithm are tracking problem, disturbance and uncertainties in a system. A rehabilitation system is one of an application in a medical treatment field which is used repetitive movement as a main movement scheme. Rehabilitation systems mostly have uncertainty and tracking problems because they are involved around human. Several lower limb (walking) rehabilitation systems have been introduced and developed. Development of walking rehabilitation systems requires three major activities; which are gait pattern design, mechanical design, and joints control. Several researches have been presented to improve performance of the gait pattern. Jezernik et al. [1] proposed several algorithms for gait pattern adaptation in a famous rehabilitation machine, called LOKOMAT. The first algorithm was based on inverse kinematics of the system in combination with the interaction between user and the machine during training session. The second algorithm was based on forward kinematics of the exoskeleton using the estimation data of the desired variation in acceleration of the gait pattern. Lastly, the third algorithm was based on gait pattern angular trajectories adaptation using impedance control. Scmidt et al. [2] presented algorithms to safely regain synchronous walking motion after perturbation on HapticWalker. The perturbation situations for recovery included stumbling, hitting an obstacle, and slipping.

Several mechanical designs for rehabilitation systems have been implemented. Veneman et al. [3] proposed a design of an actuator for exoskeleton. Several problems in the old type actuators were explained as heavy weight, complex mathematic model, and low torque which were not suitable for exoskeleton. They designed an actuator which consisted of a servo motor, a flexible Bowden cable transmission and a force sensor attached to an elastic part of the hardware. They showed that the new actuator performance was sufficient for using in lower limb exoskeleton with less complexity in mathematic model. Agrawal et al. [4] designed an exoskeleton mechanism which reduced the load from gravitational acceleration that acted on the hip and knee joints of human. The exoskeleton did not use any actuators, it was passive exoskeleton type. By using springs to absorb the forces that acted on the knee and the hip joints, actuators used to move the legs were not required. Jia-fan et al. [8], shows that the dynamic model of lower limb exoskeleton should be considered separately into two phases which are single support phase and double support phase then model the system accordingly. Czarnetzki et al. [9] developed a control algorithm for biped robot based on zero moment point for balancing a system. Kazerooni et al. [10] developed a control algorithm for BLEEX by using one degree of freedom model. The result shows a simplified controller

☆ This paper was recommended for publication by Associate Editor Prof. Dong Sun.

* Corresponding author.
E-mail address: st113052@ait.ac.th (N. Ajjanaromvat).

https://doi.org/10.1016/j.mechatronics.2018.03.003
Received 18 September 2017; Received in revised form 12 February 2018; Accepted 8 March 2018
Available online 22 March 2018
0957-4158/

which could stabilize the complex system. Beyl et al. [11] proposed a design of gait rehabilitationsystem. The mechanical structure of the exoskeleton is based on real human muscle by applying pleated pneumatic artificial muscles as a muscle. Torrealba et al. [16] proposed an actuator for assisting knee joint of a gait rehabilitation exoskeleton. The actuator was designed for knee joint exclusively by taking into consideration several important aspects such as gait pattern, kinematics, kinetics of natural human knee and robot. Actuator consisted of floating spring which allowed preload stiffness without taking more power from motor. As a result, a knee tissue like actuator could be attached to many knee rehabilitations. Several researches [21–23] focused on designing mechanical actuators which imitated human tissue applied for exoskeleton joints. Hyun et al. [26] proposed a design of lower limb exoskeleton for weight bearing assistance based on real human leg. Each joint used both active and passive springs for assisting human movement. Robot hip and knee joints were controlled by electric motor. Long et al. [30] developed a lower limb exoskeleton which only control knee joint using an active actuator. The other joints are passively controlled. Experiments are focused on comfortable of wearer and energy consumption of a system. Results show that a system could assist a wearer to walk follow the predefine trajectory.

Many researches applied different control algorithms to achieve their goals. Yang et al. [5] proposed a way to control an exoskeleton to follow human motion in real world. By using an adaptive-network-based fuzzy inference system to control each joint of the exoskeleton, the movement of the exoskeleton was close to the actual human walking gait. As a result, the fatigue caused to human during the training session was reduced. Fleischer and Hommel [6] used EMG signal from muscle to control their exoskeleton leg. Torque controller was used when the torque was obtained by using torque prediction algorithm acquired from EMG signal. Banala et al. [7] used the force sensor to improve assist gait training algorithm to provide more variable to adjust the gait training procedure. Longman [12] proposed an iterative learning control for controlling a repetitive type movement robot which reduce tracking error based on measurement state. An iterative learning control which could ignore uncertainties was also proposed by Bouakrif [13]. The algorithm adjusted control signal over a course of movement based on Lyapunov-like positive definite sequence. The result showed that the error monotonically decreased over a finite time interval. Majeed et al. [15] proposed a control algorithm for controlling a lower limb exoskeleton. The control algorithm was based on conventional PD controller for trajectory tracking. The control algorithm used force data to compensate disturbance which acted on the exoskeleton joint. A particle swarm optimization was used as a strategy to optimize the value from the force data. Simulation results showed significant improve in tracking performance when disturbances were presented. Oh et al. [17] designed a control framework for lower limb exoskeleton. The algorithm was designed by using several control algorithms for different phase of walking pattern. Experiment results showed smooth phase switching. Several researches [18–20] applied adaptive control algorithm for controlling lower limb exoskeleton for gait rehabilitation. Disturbances were presented in the system to test control algorithm performance for compensation of gait tracking error for better rehabilitation results. EMG signal was used in exoskeleton implementation [24,25]. An assistive control scheme was used to control exoskeleton by analyzing the gathered EMG signal from muscles. The data was used with control algorithm to control joint of exoskeleton. Zhang et al. [27] presented a control of an exoskeleton using neural network control. The proposed algorithm did not require system model in advance for controlling the system. A result showed better tracking performance compared to the convention PD controller. Long et al. [28] proposed a control algorithm which reduced tracking error from disturbance. The result of the proposed algorithm was better compared to the conventional PID controller. Li et al. [29] proposed an electric stimulation control for lower limb exoskeleton. The system is designed for paraplegic patient. An electric stimulation feedback provides info for the leg exoskeleton which aids the patient during the rehabilitation period. In this paper, an online learning LQR with adaptive iterative learning control algorithm is proposed to control a rehabilitation system which is a non-linear system. This control algorithm will address an uncertainties and disturbance problems of a robot while an adaptive ILC part will address a tracking error problem. Finally, experimental results and simulation results indicate an improvement of a repetitive control on leg exoskeleton for rehabilitation system over conventional control algorithm.

2. Human gait pattern

In rehabilitation a leg exoskeleton is expected to follow the desired trajectory of the walking pattern which is called gait pattern. There are two phases in a gait pattern which are swing phase and stance phase. A swing phase occurs during the non-ground contact period of the pattern, while stance phase occurs during ground contact. Normally, almost everyone has different walking pattern which results in different trajectory. A wrong walking pattern might result in a severe injure to the user that uses the rehabilitation system. Therefore, several walking patterns of healthy persons are captured and analyzed. The average walking pattern is then used for the rehabilitation system. Walking pattern consists of multiple points in the trajectory. Two most important parameters of the trajectory are position and speed of each joint at a time. By defining x_i as a position of point i in a trajectory, thus, dx_i/dt is the speed trajectory to be tracked. The rehabilitation system must be designed to track both position and speed of the average of healthy persons walking pattern as expressed in Eqs. (1) and (2).

$$x_i = \theta_i \tag{1}$$

$$\frac{dx_i}{dt} = \dot{\theta}_i \tag{2}$$

when

t is a sampling period.

A walking gait pattern is analyzed by capturing images of a healthy person walking on a treadmill at a normal walking speed of 1 m/s or 3.6 km/h. Fig. 1 shows images of a healthy person walking on a treadmill at different time steps.

3. AIT leg-exoskeleton for rehabilitation

AIT leg-exoskeleton for rehabilitation is specially designed for gait disorder patients who cannot carry their own weights on their feet. Since the patient cannot carry their own weight, thus, over-ground gait training is required. According to the reports of [31–33], over-ground gait training is suitable for this type of patient as physical constraint on the patient weight is eliminated. Without this physical constraint, the

Fig. 1. Healthy person walking gait pattern analysis.

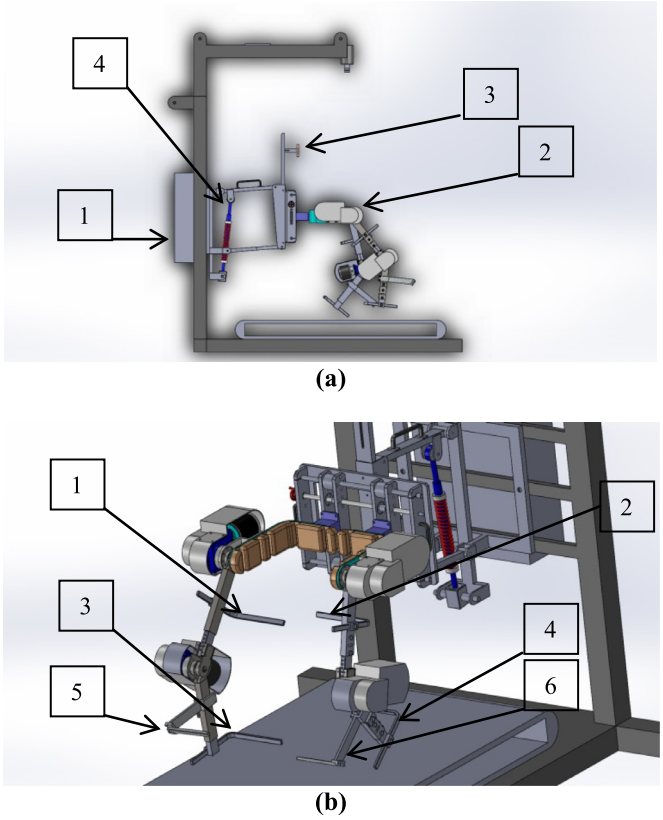

Fig. 2. (a) Side view (b) isometric view of the rehabilitation system.

patient can move their legs following the correct walking gait pattern, thus, this type of training can increase recovery speed of the patient. Therefore, a weight supporting system is required and equipped in the system. With full weight support, the patient does not carry his/her own weight so the system helps reduce load on their feet. The patient's feet are barely touched the ground while a gait pattern is executed according to healthy person gait pattern as described in Section 2. The patient has to wear the weight support suit before using the rehabilitation machine. The patient is carried by using the supporting weight as shown at position 1, the patient's hip is at position 2, the patient's back is supported at position 3, a supporting spring used to adjust according to the patient's height is at position 4 of Fig. 2(a) respectively. The patient's thighs are tied with thigh supports at positions 1 and 2, the patient's calves are tied with calf supports at positions 3 and 4 and the ankle joints are tied with springs attached at positions 5 and 6 of Fig. 2(b) respectively.

4. Dynamics model of AIT leg-exoskeleton

The leg-exoskeleton developed at AIT consists of two active joints which are hip and knee joints on each leg. Both legs are completely separated from each other by attaching at a rigid frame. As a result, two legs can be considered separately in any condition. The mechanical design of the leg-exoskeleton is based on real human legs so that a person can wear it during rehabilitation. The system is design to fully carry a patient during a training session. A weight support is equipped in the system. As a result, the patient's weight is not loaded on the leg-exoskeleton. The dynamic model is expressed in two different phases due to the human gait pattern. The dynamic model of human locomotion is considered separately for each leg.

4.1. Swing leg dynamic

Swing leg dynamic model is analyzed according to free body diagram as shown in Fig 3.where,

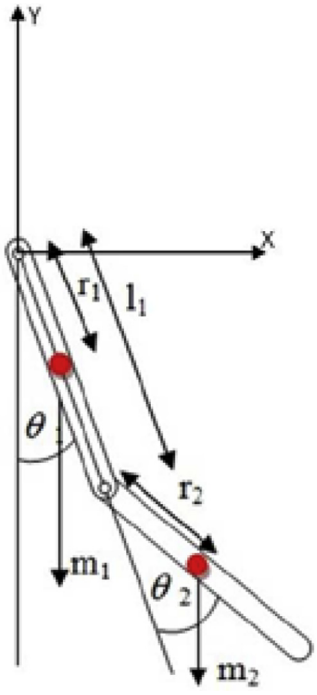

Fig. 3. Free body diagram of swing leg of AIT leg-exoskeleton for rehabilitation.

- θ_1 is the angle of the hip joint.
- m_1 is the mass of the upper leg.
- r_1 is the length between the hip joint and center of gravity of the upper leg.
- l_1 is the whole length of the upper leg.
- θ_2 is the angle of the knee joint.
- m_2 is the mass of the lower leg.
- r_2 is the length between the knee joint and center of gravity of lower leg.

4.2. Stance leg dynamic

Stance leg dynamic model is derived according to free body diagram as shown in Fig. 4. Stance phase occurs when a foot has ground contact as shown in a free body diagram in Fig. 4. During a stance phase, a linear movement in x direction is considered to be small and negligible. Therefore, only kinetic energy from rotational movement and potential energy are considered.where,

- θ_1 is the angle of the knee joint.
- m_1 is the mass of the upper leg.
- r_1 is the length between the knee joint and center of gravity of the upper leg.
- l_2 is the whole length of the lower leg.
- θ_2 is the angle of the ankle joint.
- m_2 is the mass of the lower leg.
- r_2 is the length between the ankle joint and center of gravity of the lower leg.

4.3. Mechanical model

Dynamics model of the leg-exoskeleton is derived by using Euler-Lagrange equation of motion. The Lagrangian, L, is obtained from the difference between total kinetic energy, T, and total potential energy, V. The equation is expressed in Eq. (3).

$$L = T - V \tag{3}$$

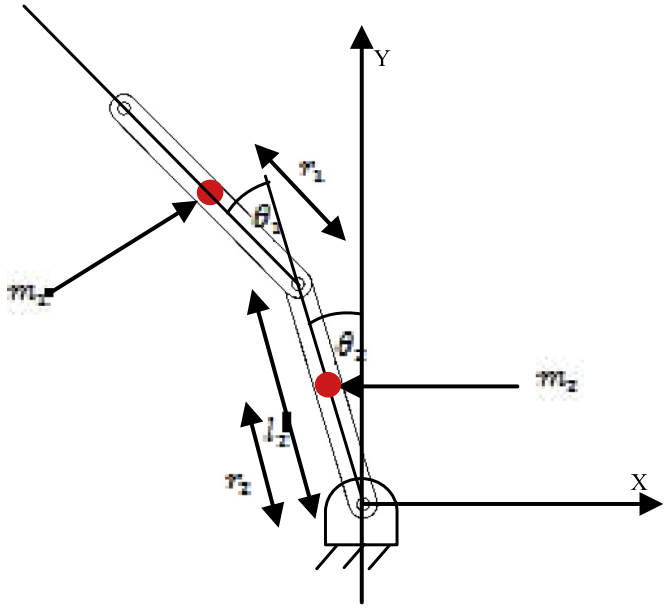

Fig. 4. Free body diagram of stance leg of AIT leg-exoskeleton for rehabilitation.

L of the swing leg of exoskeleton is shown in Appendix A. Euler-Lagrange equation of motion of the system is expressed in Eq. (4).

$$\tau_1 = \frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_1} - \frac{\partial L}{\partial \theta_1}$$
$$\tau_2 = \frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_2} - \frac{\partial L}{\partial \theta_2} \quad (4)$$

By applying Eqs. (3) and (4), dynamics equations of the leg-exoskeleton during swing leg are shown in Eq. (5) for the hip joint and in Eq. (6) for the knee joint.

$$\tau_1 = (I_1 + I_2 + m_2 l_1^2 + m_1 r_1^2 + 2m_2 l_1 r_2 \cos\theta_2 + m_2 r_2^2)\ddot{\theta}_1 + (I_2 + m_2 l_1 r_2 \cos\theta_2 + m_2 r_2^2)\ddot{\theta}_2 + (-2m_2 l_1 r_2 \sin\theta_2)\dot{\theta}_1\dot{\theta}_2 - (m_2 l_1 r_2 \sin\theta_2)\dot{\theta}^2{}_2 + gm_1 l_1 \sin\theta_1 + gm_1 r_1 \sin\theta_1 + gm_2 r_2 \sin(\theta_1 + \theta_2) \quad (5)$$

$$\tau_2 = (I_2 + m_2 l_1 r_2 \cos\theta_2 + m_2 r_2^2)\ddot{\theta}_1 + (I_2 + m_2 r_2^2)\ddot{\theta}_2 + (m_2 l_1 r_2 \sin\theta_2)\dot{\theta}_1{}^2 + gm_2 r_2 \sin(\theta_1 + \theta_2) \quad (6)$$

where,

- θ_1 is the angle of the hip joint.
- m_1 is the mass of the upper leg.
- r_1 is the length between the hip joint and center of gravity of the axis.
- l_1 is the whole length of the upper leg.
- θ_2 is the angle of the knee joint.
- m_2 is the mass of the lower leg.
- r_2 is the length between the knee joint and center of gravity of the axis.
- I_1 is the moment of inertia of the hip joint.
- τ_1 is the required torque to drive the hip joint.
- I_2 is the moment of inertia of the knee joint.
- g is the gravitational acceleration.
- τ_2 is the required torque to drive the knee joint.

With the mechanical design of AIT leg-exoskeleton, two legs can move separately from each other. Therefore, dynamic model of both legs can be simplified by considering each leg individually. With a weight support system, the load from upper body weight is assumed negligible. By applying Eqs. (3) and (4), dynamic equations for stance leg are shown in Eqs. (7) and (8). Lagrangian, L, of the stance leg of the leg-exoskeleton is shown in Appendix A.

$$\tau_1 = I_1\ddot{\theta}_1 + I_1\dot{\theta}_2 + m_1 g r_1 \sin(\theta_1 + \theta_2) \quad (7)$$

$$\tau_2 = I_1\ddot{\theta}_2 + I_1\dot{\theta}_1 + m_1 g r_1 \sin(\theta_1 + \theta_2) + m_2 g l_2 \sin(\theta_2) + m_2 g r_2 \sin(\theta_2) \quad (8)$$

where,

- θ_1 is the angle of the knee joint.
- m_1 is the mass of the upper leg.
- r_1 is the length between the knee joint and center of gravity of the upper leg.
- l_2 is the whole length of the lower leg.
- θ_2 is the angle of the ankle joint.
- m_2 is the mass of the lower leg.
- r_2 is the length between the ankle joint and center of gravity of the lower leg.
- I_1 is the moment of inertia of the upper leg.
- τ_1 is the required torque to drive the knee joint.
- I_2 is the moment of inertia of the knee joint.
- g is the gravitational acceleration.
- τ_2 is the required torque to drive the ankle joint.

However, with the AIT leg-exoskeleton design, a system itself does not contact the ground. Therefore, real stance phase does not exist. Furthermore, the patient does not have ground contact either with the weight supporting system. As a result, only swing phase only is considered in controller design. At any time within the gait pattern, the movement of the system is considered as swing leg phase.

5. Control algorithm

Controller of leg-exoskeleton for rehabilitation has to handle several uncertainties such as patient walk behavior, patient weight and physical parameters of the system. It is difficult to determine optimal gain of controller of the system. Therefore, a learning based control algorithm is appropriate to solve this problem. Online iterative learning linear quadratic regulator (OILLQR) with adaptive iterative learning control is proposed to control the leg-exoskeleton.

5.1. Online iterative learning linear quadratic regulator (OILLQR) for non-linear system

Since the leg-exoskeleton is a non-linear system, a linear controller cannot be applied directly. In order to apply LQR to the non-linear leg-exoskeleton system, the system dynamics model is linearized at several reference points. The gains at several reference points are then adjusted online using iterative learning method. Let's define

$$e_1 = \theta_1 - \theta_{1d} \quad (9)$$

$$e_2 = \theta_2 - \theta_{2d} \quad (10)$$

Substitute Eqs. (9) and (10) into Eqs. (5) and (6) then determine the derivatives respect to e_1, e_2, τ_1, τ_2 thus,

$$\frac{\partial F}{\partial e_1} = M^{-1}\begin{pmatrix} A_{11} \\ A_{12} \end{pmatrix} = A_1$$

$$\frac{\partial F}{\partial e_2} = M^{-1}\begin{pmatrix} A_{21} \\ A_{22} \end{pmatrix} = A_2$$

$$\frac{\partial F}{\partial \tau_1} = M^{-1}\begin{pmatrix} A_{31} \\ A_{32} \end{pmatrix} = A_3$$

$$\frac{\partial F}{\partial \tau_2} = M^{-1}\begin{pmatrix} A_{41} \\ A_{42} \end{pmatrix} = A_4$$

when

$F = L$
$A_{11} = gm_1 l_1 \cos(e_1 + \theta_{1d}) + gm_1 r_1 \cos(e_1 + \theta_{1d})$
$+ gm_2 r_2 \cos(e_1 + \theta_{1d} + \theta_{2d})$
$A_{12} = gm_2 r_2 \cos(e_1 + \theta_{1d} + \theta_{2d})$
$A_{21} = gm_2 r_2 \cos(e_2 + \theta_{1d} + \theta_{2d})$
$A_{22} = gm_2 r_2 \cos(e_2 + \theta_{1d} + \theta_{2d})$
$A_{31} = 1$
$A_{32} = 0$
$A_{41} = 0$
$A_{42} = 1$

A new state space model of the system can be expressed as Eq. (11).

$$\dot{e} = A_{new} e + B_{new} u \tag{11}$$

when

$$A_{new} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ A_1 & A_2 & 0 & 0 \\ & & 0 & 0 \end{bmatrix}$$

$$B_{new} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ A_3 & A_4 \end{bmatrix}$$

By substitution of the trajectory parameters into the system state space model during the movement, linearized models of the system at the reference points are obtained. In the conventional LQR, state and control weight matrices are defined in order to obtain the desired performance. Defining of all the weight matrices for all the reference point of the trajectory in order to obtain the desired performances at all the reference points requires a lot of computation and effort. In this paper, an online weight adjustment using iterative learning method is proposed to solve the problem. The cost function of LQR is expressed in Eq. (12).

$$J = e^T Q_i e + \sum_{i=0}^{n-1} (e_i^T Q_i e_i + u_i^T R_i u_i) \tag{12}$$

At different reference points of the trajectory, Q and R weight matrices are different. The cost functions must be optimized in order to achieve the desired performance at all the reference points. By applying an online learning method according to Eq. (13) and (14), the weight matrices at all reference points of the trajectory are finally obtained.

$$Q_{inew} = Q_{iold} + \rho[\alpha_i] \tag{13}$$

$$R_{inew} = R_{iold} + \rho[\beta_i] \tag{14}$$

when

$0 < \rho \leq 1$

Eqs. (13) and (14) represent both state, Q, and control, R, weight matrices for the reference point i of the trajectory. ρ is learning gain for both matrices. $\alpha$and β are the measured state for the reference point i of the trajectory. In one sampling period, t, α_i is obtained from Eq. (15). β_i can be set to zero when R is fixed at unity.

$$\alpha_i = \begin{bmatrix} \alpha_{i1} & 0 & 0 & 0 \\ 0 & \alpha_{i2} & 0 & 0 \\ 0 & 0 & \alpha_{i3} & 0 \\ 0 & 0 & 0 & \alpha_{i4} \end{bmatrix} \tag{15}$$

when

$\alpha_{i1} = \sum_{n=1}^{t} \frac{e_i}{n}$
$\alpha_{i2} = \dot{e}_i$ when $(max\ e_i - min\ e_i) > \gamma$
$\alpha_{i2} = 0$ when $0 \leq (max\ e_i - min\ e_i) \leq \gamma$
$\alpha_{i3} = \sum_{n=1}^{t} \frac{\dot{e}_i}{n}$
$\alpha_{i4} = \ddot{e}_i$ when $(\max \dot{e}_i - min\ \dot{e}_i) > \delta$
$\alpha_{i4} = 0$ when $0 \leq (max\ \dot{e}_i - min\ \dot{e}_i) \leq \delta$

The optimal state weight matrix Q will be obtained online according to the leg-exoskeleton movement under disturbance during the learning period. Using this technique, the system will attenuate the influences of disturbances and uncertainties even when different load is provided. After the state weight matrix Q and the control weight matrix R are obtained, the controller gain can be obtained directly by solving the algebraic Riccati equation expressed in Eq. (16).

$$A^T_{inew} S + S A_{inew} - (S B_{inew} + N) R^{-1}_{inew} (B^T_{inew} S + N^T) + Q_{inew} = 0 \tag{16}$$

where,

A_{inew} is the state matrix at trajectory i.
S is covariance matrix.
B_{inew} is input matrix at trajectory i.
R_{inew} is control weight matrix for trajectory i.
Q_{inew} is state weight matrix for trajectory i.

By solving Eq. (16) online, the covariance matrix, S, is obtained. The result from Eq. (16) is used to determine the optimal controller gain K_i for each trajectory i according to Eq. (17). In the conventional LQR method, the obtained gain is used for the whole period which results in non-optimal performance at every point in the trajectory. However, in this research, the gain is determined online at each point i in the trajectory for optimal performance at every point in the trajectory. The gain for the controller is determined from

$$K_i = {R_{inew}}^{-1} (B^T_{inew} S + N^T) \tag{17}$$

By solving Eq. (17), the control signal for OILLQR controller becomes

$$u_{iOILLQR} = -K_i x_i \tag{18}$$

when,

x_i is the system state at trajectory i.

The number of sets of gain for the OILLQR is the same as the number of reference points in the trajectory. If there are i reference points in the desired trajectory, there will be i sets Q of and R matrices and also i sets of the optimal gain for the desired movement in the trajectory. The control block diagram for OILLQR is shown in Fig. 5.

5.2. Iterative learning control based on OILLQR

By applying OILLQR to the leg-exoskeleton system, the tracking

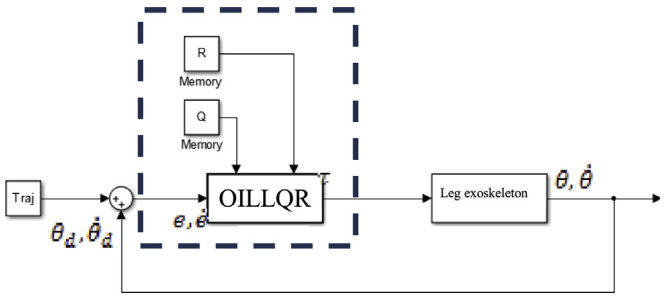

Fig. 5. Block diagram of OILLQR.

performance is better however small tracking error still exist in the system. To get rid of such error, an iterative learning control is added on top of OILLQR. Iterative learning control gain is normally optimized by several types of algorithm such as contraction mapping, partial isometric and quadratic cost ILC Panomruttanarug [14]. In this paper, an optimal gain based iterative learning control based on OILLQR controller is proposed. From the previous section, the optimal state weight matrix Q and the optimal control weight matrix R at each reference point are obtained. Quadratic cost iterative learning control gain is normally obtained by optimizing the cost function expressed in Eq. (19).

$$J = e^T Q e + u^T R u \tag{19}$$

By selecting state weight matrix Q and control weight matrix R, the gain can be calculated using Eq. (20). However, to optimize every reference point of the desired trajectory of the leg-exoskeleton, the entire weight matrices must be found in advance.

$$L = (P^T Q P + R)^{-1} P^T Q \tag{20}$$

when

$$P = \begin{bmatrix} CB & 0 & 0 & 0 \\ CAB & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & 0 \\ CA^{n-1}B & \cdots & CAB & CB \end{bmatrix}$$

C is the system output matrix
Q is the state weight matrix
R is the control weight matrix

However with iterative learning control based OILLQR, the gain doesn't have to be obtained in advance. Instead, it is obtained online during the movement period. As a result, the optimized learning gain of iterative learning control of each reference point of the trajectory is continuously adjusted over time.

$$L_i = ({P^T}_i Q_i P + R_i)^{-1} {P^T}_i Q_i \tag{21}$$

Eq. (21) shows the learning gain for reference point i of the trajectory. According to Eq. (21), P matrix also changes for each reference point i at the cycle number t. As a result, P matrix can be written as Eq. (22).

$$P_i = \begin{bmatrix} {CA_i}^{t-n}B & 0 & 0 & 0 \\ {CA_i}^{t-n+1}B & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & 0 \\ {CA_i}^{t-1}B & \cdots & {CA_i}^{t-n+1}B & {CA_i}^{t-n}B \end{bmatrix} \tag{22}$$

when

n is the current maximum number of cycle of the system

According to Eq. (22), a learning gain has equal number as the number of reference points in the trajectory. For example, if there are 15 reference points in a trajectory, P matrix will consist of 15 sets which will result in 15 sets of the optimal learning gain. The combination of OILLQR and adaptive gain learning ILC has good performance for parameter uncertainties and disturbance rejection. Finally, the finely adjusted control signal is applied according to Eq. (23).

$$u_{it} = u_{it-1} + L_{it} e_{it} \tag{23}$$

when

t is cycle number
I is reference point number of the trajectory

The final control signal is expressed in Eq. (24),

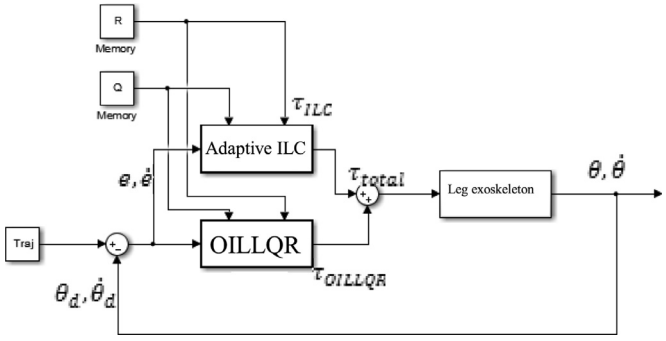

Fig. 6. Block diagram of OILLQR with adaptive gain learning ILC.

$$u_i = u_{iOILLQR} + u_{iILC} \tag{24}$$

The control block diagram of the leg-exoskeleton controlled by OILLQR with adaptive gain learning ILC is shown in Fig. 6.

6. Implementation of AIT leg-exoskeleton

In this section, the design of AIT leg-exoskeleton for rehabilitation is explained. The implementation of the system consists of two parts which are software designed and a hardware implementation.

6.1. Hardware design and implementation

AIT leg-exoskeleton is designed to carry a person with the weight up to 80 kg. The design from SolidWorks program is shown in Fig. 7.

In the design, each leg of the exoskeleton is attached to the waist frame connected with the machine base as shown in Fig. 7. Both legs are not contacted the ground and driven separately. Therefore, each leg has no effect to the other. As a result, a dynamic model for each leg can be individually considered. In the design of the system, a spring at position 1 of Fig. 7 is used to adjust the exoskeleton height according to the patient's posture and height. Each joint is driven by a DC motor. The motor is connected to a worm gear which generates sufficient torque to drive each joint. The speed and position of each joint are measured by an encoder attached at the end of the joint. A counter balance weight is used to support the weight of the patient and adjustable for supporting of 0–100% of the patient's weight. All parts of the leg-exoskeleton are machined, manufactured and assembled. Both legs are designed symmetrically as shown in Fig. 8a and b. Hip supporting part is shown at position 1 of Fig. 8a. The width of hip supporting part is adjustable according to the patient's hip size. It is made of aluminum. Position 2 of Fig. 8a is a hip joint which is a rotation joint. Motor is connected with worm gear at position 3 of Fig 8a. A stainless steel adjustable thigh link is shown at position 4. A thigh link is adjusted according to the patient's height and posture. Position 5 of Fig. 8a is a thigh support which is used

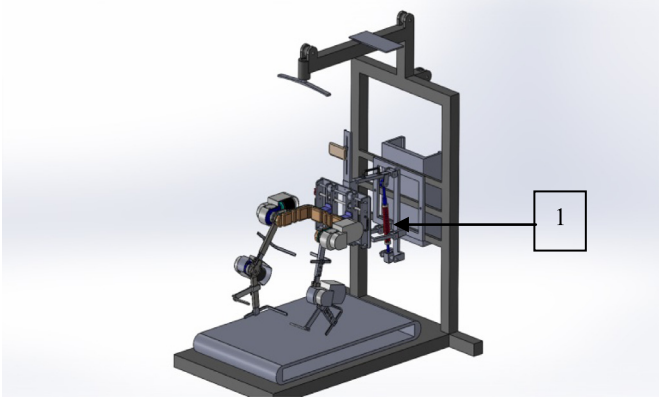

Fig. 7. AIT Leg-exoskeleton design from SolidWorks program.

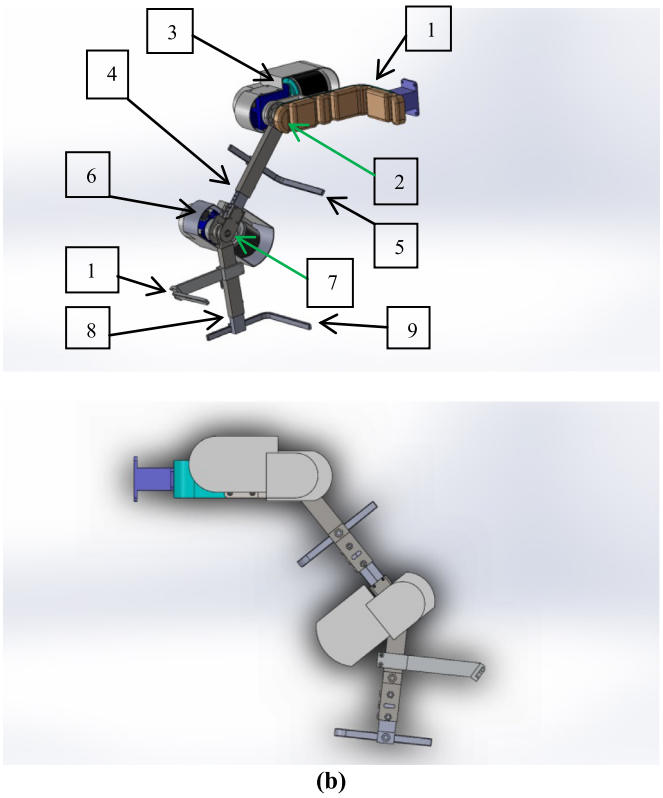

Fig. 8. (a) Isometric view of right leg design (b) side view of right leg design.

to tie the patient's thigh to the system. A thigh support is made from steel and attached with soft fabric and Velcro tape for attaching the patient's thigh with the system leg. Motor with worm gear of knee joint is attached at position 6 of Fig. 8a. Position 7 is a knee joint. The length of stainless steel knee support link is also adjustable according to the patient's height and posture as shown at position 8 of Fig. 8a. The patient's calf is attached to a steel calf supporting part as shown in Fig. 8a at position 9. A Velcro tape and fabric are also used at the calf supporting part. Springs are attached at position 10 of Fig 8a. The springs are used for passively carrying the patient's feet. A side view of the designed leg-exoskeleton is also shown in Fig. 8b.

Fig 9 shows a photo of the fully assembling AIT leg-exoskeleton. The leg-exoskeleton parameters are obtained from direct measurement or from Solidworks program and shown in Table 1.

6.2. Electrical implementation

Arduino DUE model is used as the main microprocessor of the system as shown in Fig. 10a. The controller uses 32-bit ARM Cortex-M3 with up to 84 MHz clock frequency which is sufficient for processing the proposed control algorithm. The controller provides up to 54 digital I/O and 12 analog inputs which are also sufficient for the hardware requirement of the leg-exoskeleton. Each motor is driven by a DC motor driver having an idle current up to 60 A at 24 V. The motor driver receives pulse width modulation as a command signal. Position and speed of the motor are controlled by using the proposed control algorithm. Photo of the electrical control box is shown in Fig. 10b. Each motor position is measured by an encoder at each joint of the leg. RTOS is used as the operating system for the system. A sampling period of 10 ms is used for the sensor reading. By separating into several sub programs, each link could be controlled separately. The main program synchronizes all the joints. Sub programs are used to control each joint by receiving the commands from the main program.

Fig. 9. Photo of assembled leg-exoskeleton.

Table 1
Leg-exoskeleton parameters.

Parameter	Value	unit
I_1	2.3085	kg m^2
I_2	0.0685	kg m^2
m_1	10.2199	kg
m_2	2.3470	kg
l_1	0.4156	m
r_1	0.2982	m
r_2	0.1211	m

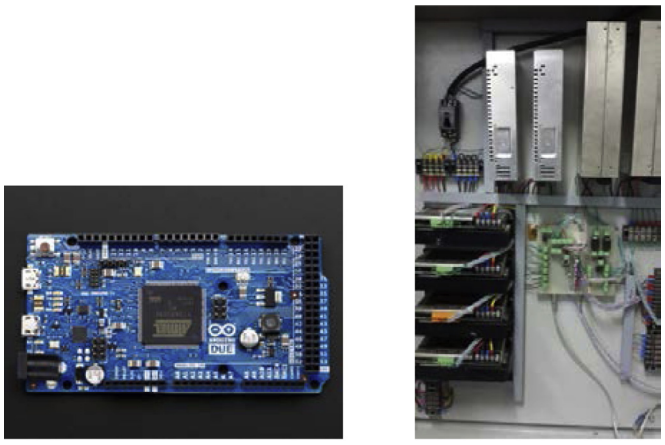

Fig. 10. (a) Arduino DUE model (b) photo of electrical control box.

7. Simulation and experimental results

Simulations and experiments are conducted and described in this section. Simulations are conducted on MATLAB program while the

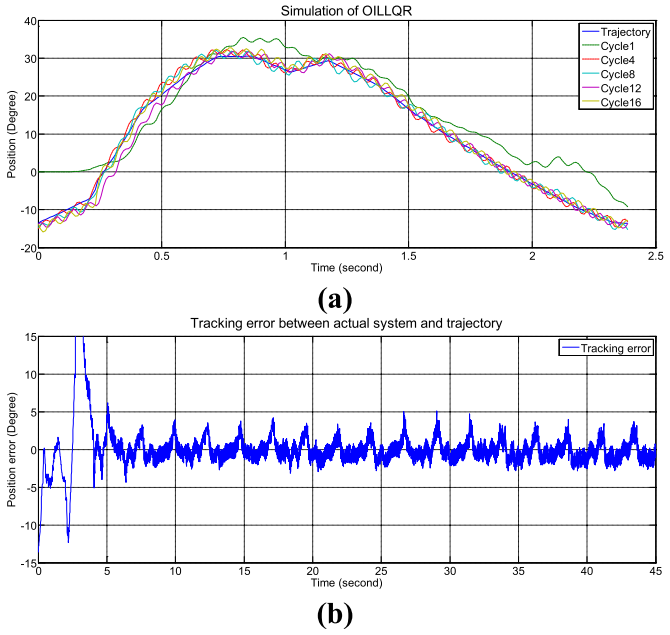

Fig. 11. (a) Simulation result of trajectory tracking using OILLQR on the leg-exoskeleton (b) tracking error obtained from OILLQR.

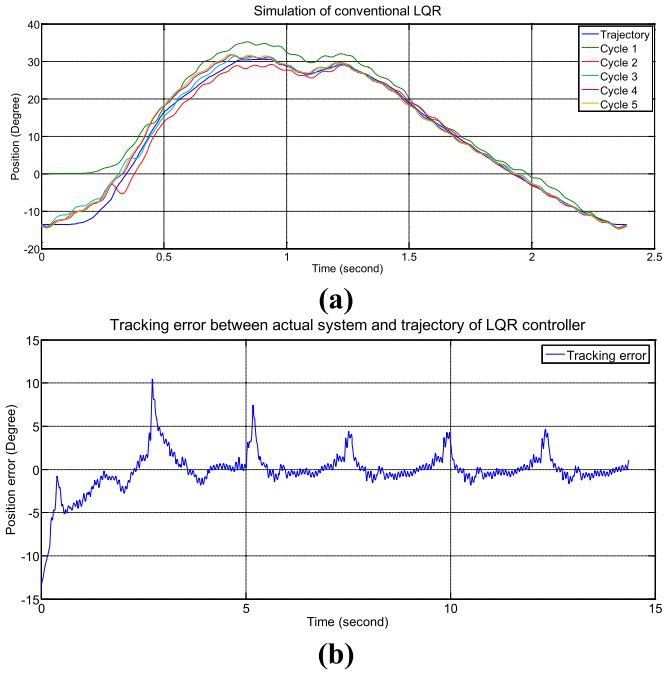

Fig. 12. (a) Simulation result of trajectory tracking using LQR on the leg-exoskeleton (b) tracking error of LQR.

experiments are conducted on the real leg-exoskeleton.

7.1. Simulations of trajectory tracking control using OILLQR conventional LQR and PD controller

The system is controlled by using OILLQR, LQR and PD controller at the walking speed of 1.2 m/s. The learning gain of OILLQR is selected at 0.01.

The simulation results of trajectory tracking of the conventional LQR and PD controller are almost the same since the beginning of the simulation until the end of the simulation period while trajectory tracking of OILLQR controller is worse than LQR and PD controller at the beginning. However, as the simulation goes on, OILLQR performs far better than LQR and PD controller. At the steady state, OILLQR shows only $\pm$ 3° tracking error which is better than both LQR and PD controller as shown in tracking error results in Figs. 11b, 12b and 13b. OILLQR performs better and better because the weights are adjusted continuously, the gain from OILLQR is finally better than the gain from LQR and PD controller. The trajectories tracking for the proposed OILLQR, PD controller and LQR controller for several cycles are shown in Figs. 11a, 12a and 13a respectively.

7.2. Simulations of trajectory tracking control using OILLQR with adaptive gain ILC and PD controller with ILC

Tracking performance of the proposed OILLQR with adaptive gain ILC and PD controller with conventional ILC are evaluated at the walking speed of 1.2 m/s. The gain of PD controller and learning gain of ILC are obtained using conventional method.

According to simulation results, OILLQR with adaptive gain ILC shows better tracking performance than PD with ILC controller. Tracking error for OILLQR with adaptive gain ILC reduces to 2° or about 50% reduction compared to PD controller with ILC. PD controller with ILC performs far worse than the proposed OILLQR with adaptive gain ILC for both learning rate and tracking error criteria as shown in Fig. 14a and b for proposed control algorithm and Fig. 15a and b for conventional control algorithm. Also, the settling time for the system to reach steady state of OILLQR with adaptive gain ILC is 4 times faster than PD with ILC controller. Tracking errors at higher number of cycle also significantly reduce in OILLQR with adaptive gain ILC.

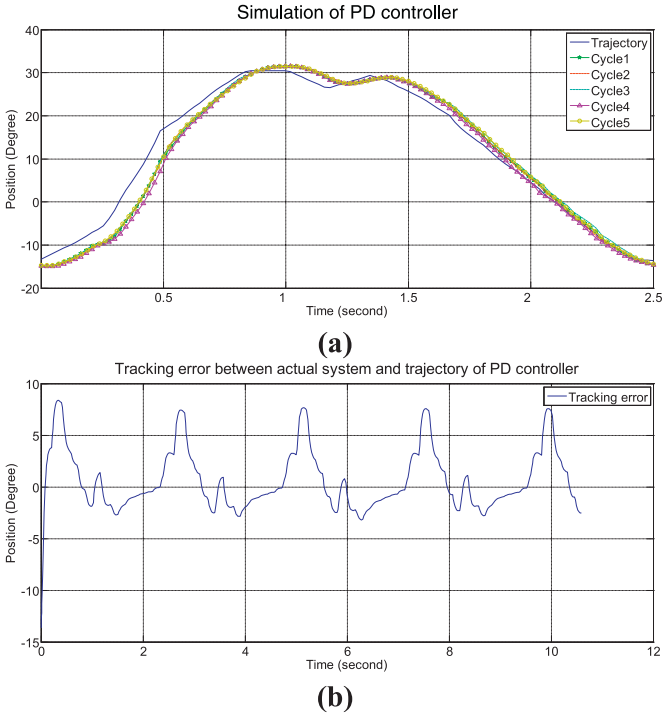

Fig. 13. (a) Simulation result of trajectory tracking using PD controller on the leg-exoskeleton (b) tracking error obtained from PD controller.

7.3. Experiments of trajectory tracking control using OILLQR on the real AIT leg-exoskeleton

OILLQR is implemented on the real AIT leg-exoskeleton. The experiments are conducted at the walking speed of 0.9 m/s using walking gait pattern of the average healthy person as explained in Section 2. The leg-exoskeleton system is run for about 20 cycles to observe the performance of OILLQR on the leg exoskeleton.

Experimental result shows that tracking performance of the first

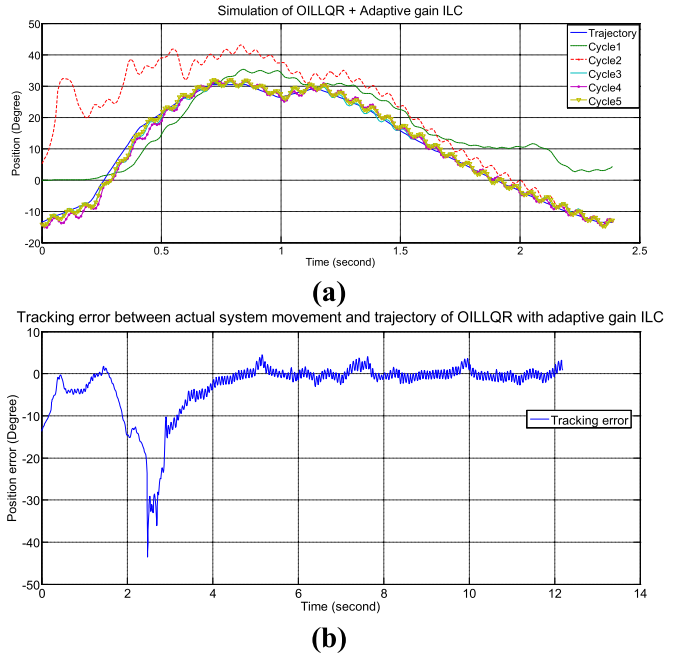

Fig. 14. Simulation result of (a) trajectory tracking using OILLQR with adaptive gain ILC (b) tracking error of OILLQR with adaptive gain ILC.

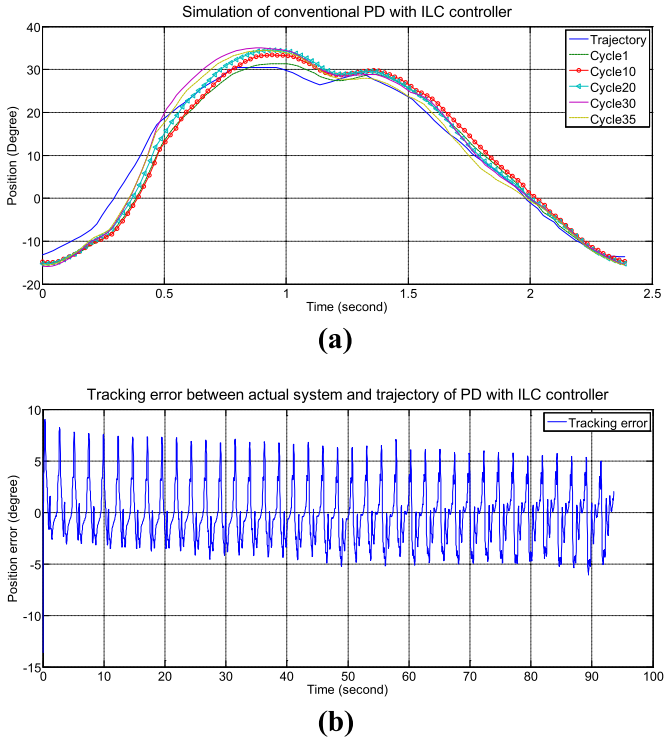

Fig. 15. Simulation result of (a) trajectory tracking using conventional PD with ILC controller (b) tracking error of conventional PD with ILC controller.

cycle of movement is the worst compared to the following cycles. However, as the experiment goes on, better gain is obtained which results in better tracking performance. Finally, the leg-exoskeleton can track the desired trajectory but with some error as shown in Fig. 16a andb.

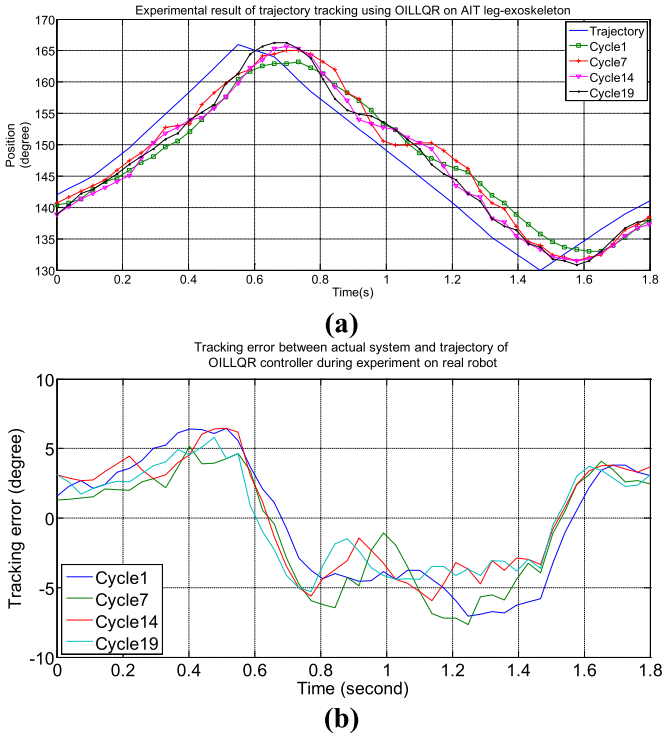

Fig. 16. Experimental result of (a) trajectory tracking using OILLQR on AIT leg-exoskeleton (b) tracking error of OILLQR on AIT leg-exoskeleton.

7.4. Experiments of trajectory tracking control using OILLQR with adaptive gain ILC and conventional PD with ILC controller on the real AIT leg-exoskeleton

Both OILLQR with adaptive gain ILC and PD with the conventional ILC are implemented on the real AIT leg-exoskeleton. The experiments are conducted at the walking speed of 0.9 m/s using walking gait pattern of the average healthy person. Gains for PD controller are obtained offline while the optimal learning gain for ILC is obtained using contraction mapping method [14].

Experimental result shows almost the same tracking performance as obtained from OILLQR at the beginning of the experiment. However, after a few cycles, OILLQR with adaptive gain ILC performs better than OILLQR. The tracking error is reduced significantly. Learning time is also shorter. At only after 7 cycles, tracking error from OILLQR with adaptive gain ILC is much less compared to OILLQR. From the results in Figs. 17 and 18, OILLQR with adaptive gain ILC has better tracking performance than PD with ILC. At cycle 19, the tracking error of OILLQR with adaptive gain ILC is obviously low. On the other hand, at cycle 23, PD controller with ILC still has 2 times higher tracking error than the proposed control algorithm. As a result, with better tracking performance and faster learning rate, the proposed controller is far more superior for rehabilitation application since the patient's movement can follow the healthy person's gait pattern.

7.5. Experiments with test subject on AIT leg-exoskeleton using OILLQR with adaptive gain ILC and conventional PD with ILC

In the experiment, walking speed is set at 0.9 m/s. The data of 40 walking cycles are recorded. An average of walking pattern of healthy person which contains 50 points is used in the experiment. The weight of the tested patient is about 80 kg. Full weight support is used during the experiment. Tracking performance during training is determined and analyzed. Movement with tracking error which is less than 3° from

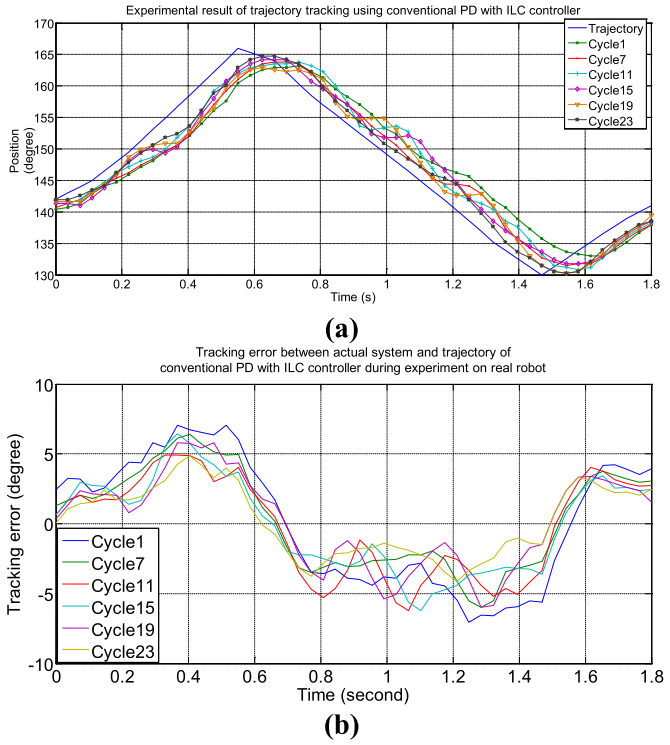

Fig. 17. Experimental result of (a) trajectory tracking using conventional PD with ILC controller on AIT leg-exoskeleton (b) tracking error of conventional PD with ILC controller on AIT leg-exoskeleton.

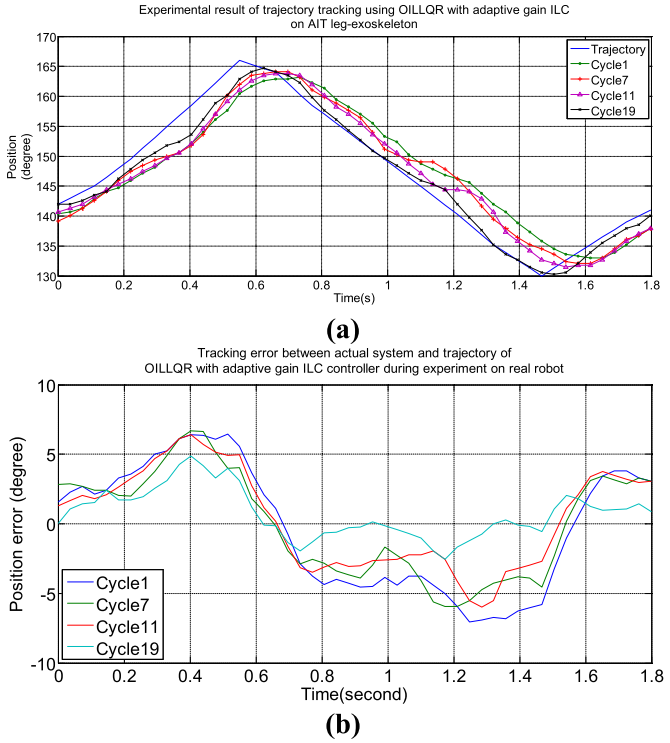

Fig. 18. Experimental result of (a) trajectory tracking using OILLQR with adaptive gain ILC on AIT leg-exoskeleton (b) tracking error of OILLQR with adaptive gain ILC on AIT-leg-exoskeleton.

the target trajectory is considered as the correct movement. Both OILLQR with adaptive gain ILC and PD with ILC are implemented to the system.

According to the experimental results in Tables 2 and 3, in the first cycle, the tested patient follows only 24% of the desired trajectory for the proposed OILLQR with adaptive gain ILC and only 26% of the desired trajectory for the conventional PD with ILC respectively. However, after gain adjustment by OILLQR with adaptive gain ILC, the tracking performance rises to 44% only after 7 cycles, while PD with ILC still tracks only 36%. After 40 cycles, the movement of the tested patient follows 86% of the desired trajectory by OILLQR with adaptive gain ILC while PD with ILC still follows only 64% of the desired trajectory. With full weight support, the patient can move freely without physical constraint. Therefore, the patient can move their legs following the correct walking gait pattern. As a result, the patient's walking pattern is adjusted faster to the correct gait pattern. In conclusion, the proposed OILLQR with adaptive gain ILC is better in both learning rate and tracking performance than PD with ILC.

Fig. 19. Tested patient wearing weight support suit on leg-exoskeleton for rehabilitation during experiment.

Table 2
Experimental result of AIT leg-exoskeleton for rehabilitation on the tested patient using OILLQR with adaptive gain ILC.

Cycle number	Correct movement point (out of 50)	Tracking performance (%)
1	12	24
7	22	44
16	24	48
24	25	50
30	32	64
40	43	86

Table 3
Experimental result of AIT leg-exoskeleton for rehabilitation on the tested patient using PD with ILC.

Cycle number	Correct movement point (out of 50)	Tracking performance (%)
1	13	26
7	18	36
16	21	42
24	24	48
30	31	62
40	32	64

8. Conclusion

Online iterative learning linear quadratic regulator with adaptive gain iterative learning control (OILLQR with adaptive gain ILC) was proposed in this research to control trajectory tracking of walking pattern of AIT leg-exoskeleton. The leg-exoskeleton was designed, manufactured, assembled and implemented for rehabilitation purpose. The leg-exoskeleton dynamics model was obtained using Euler-Lagrange equation of motion. A non-linear learning based controller, OILLQR, was firstly used in order to control trajectory tracking of the system. An adaptive gain ILC was added to OILLQR to improve the tracking performance of OILLQR. Experimental and simulation results showed better tracking performance compared to the conventional control methods such as PD and LQR. Learning gain had influence on the learning rate for both OILLQR and adaptive gain ILC since adaptive gain ILC optimized the gain according to the weight matrices obtained by OILLQR during the movement period. The simulation and experimental results showed the potential of using OILLQR with adaptive gain ILC for many other machines which have repetitive movements. The experiment on real patient also confirmed that the proposed OILLQR with adaptive gain ILC has better performance than the conventional PD with ILC. The results showed that the patient could track the correct gait pattern. The weight support system assisted the patient to follow the correct gait pattern of the rehabilitation machine without physical constraint which resulted in better recovery rate.

Appendix A

Lagrangian of the leg-exoskeleton is the difference between the total kinetics energy and of the total potential energy of the leg-exoskeleton. The equation is derived as followings.

$$\dot{x}_1 = r_1\dot{\theta}_1\cos\theta_1$$
$$\dot{y}_1 = r_1\dot{\theta}_1\sin\theta_1$$

$$x_2 = l_1\sin\theta_1 + r_2\sin(\theta_1+\theta_2)$$
$$y_2 = -(l_1\cos\theta_1 + r_2\cos(\theta_1+\theta_2))$$

$$\dot{x}_2 = l_1\dot{\theta}_1\cos\theta_1 + r_2(\dot{\theta}_1+\dot{\theta}_2)\cos(\theta_1+\theta_2)$$
$$\dot{y}_2 = l_1\dot{\theta}_1\sin\theta_1 + r_2(\dot{\theta}_1+\dot{\theta}_2)\sin(\theta_1+\theta_2)$$

Kinetic energy of swing leg,

$$T = \frac{1}{2}m_1{v_1}^2 + \frac{1}{2}I_1{\dot{\theta}_1}^2 + \frac{1}{2}m_2{v_2}^2 + \frac{1}{2}I_2(\dot{\theta}_1+\dot{\theta}_2)^2$$

$$T = \tfrac{1}{2}m_1(\dot{x}_1+\dot{y}_1)^2 + \tfrac{1}{2}I_1{\dot{\theta}_1}^2$$
$$+ \tfrac{1}{2}m_2(\dot{x}_2+\dot{y}_2)^2 + \tfrac{1}{2}I_2(\dot{\theta}_1+\dot{\theta}_2)^2$$

Potential energy of swing leg,

$$P_1 = -m_1gr_1\cos\theta_1$$

$$P_2 = -m_1gl_1\cos(\theta_1) - m_2gr_2\cos(\theta_1+\theta_2)$$

$$V = P_1 + P_2$$

$$V = -m_1gr_1\cos\theta_1 - m_1gl_1\cos(\theta_1) - m_2gr_2\cos(\theta_1+\theta_2)$$

Lagrangian of swing leg becomes,

$$L_{swing} = \begin{pmatrix}\tfrac{1}{2}m_1(\dot{x}_1+\dot{y}_1)^2 + \tfrac{1}{2}I_1{\dot{\theta}_1}^2 + \tfrac{1}{2}m_2(\dot{x}_2+\dot{y}_2)^2\\ +\tfrac{1}{2}I_2(\dot{\theta}_1+\dot{\theta}_2)^2\end{pmatrix}$$
$$-\left(-m_1gr_1\cos\theta_1 - m_1gl_1\cos(\theta_1) - m_2gr_2\cos(\theta_1+\theta_2)\right)$$

Kinetic energy of stance leg,

$$T = \frac{1}{2}I_1(\dot{\theta}_1+\dot{\theta}_2)^2 + \frac{1}{2}I_2{\dot{\theta}_2}^2$$

Potential energy of stance leg,

$$P_1 = -m_1gr_1cos(\theta_1+\theta_2) - m_2gl_2cos\theta_2$$

$$P_2 = -m_2gr_2cos\theta_2$$

$$V = P_1 + P_2$$

$$V = -m_1gr_1cos(\theta_1+\theta_2) - m_2gl_2cos\theta_2 - m_2gr_2cos\theta_2$$

Lagrangian of stance leg becomes,

$$L_{stance} = \tfrac{1}{2}I_1(\dot{\theta}_1+\dot{\theta}_1) + \tfrac{1}{2}I_2\dot{\theta}_2^2$$
$$-\left(-m_1gr_1cos(\theta_1+\theta_2) - m_2gl_2cos\theta_2 - m_2gr_2cos\theta_2\right)$$

References

[1] Jezernik S, Colombo G, Morari M. Automatic gait-pattern adaptation with a 4-DOF robotic orthosis. IEEE Trans Robot Autom 2004;20(June (3)).
[2] Schmidt H, Bernhardt FPR, Kruger J. Synthesis of perturbations for gait rehabilitation robots. Proceedings of the 9th international conference on rehabilitation robotics, June 28–July 1. 2005.
[3] Veneman JF, Ekkelenkamp R, Kruidhof R, van der Helm FCT, Vander Kooij H. Design of a series elastic-and Bowdencable-based actuation system for use as torque-actuator in exoskeleton-type training. IEEE 9th international conference on rehabilitation robotics, June 28–July 1. 2005.
[4] Agrawal SK, Banala SK, Fattah A, Sangwan V, Krishnamoorthy V, Scholz JP, Hsu WL. Assessment of motion of a swing legand gait rehabilitation with a gravity balancing exoskeleton. IEEE Trans Neural Syst Rehabilit Eng 2007;15(September (3)).
[5] Y. Can-Jun, N. Bin, C. Ying. Adaptive neuro-fuzzy control based development of a wearable exoskeleton leg for human walking power augmentation. International conference on advanced intelligent mechatronics monterey. 2005.
[6] Fleischer C, Hommel G. Calibration of an EMG-based body model with six muscles to control a leg exoskeleton 2007. IEEE international conference on robotics and automation. 2007.
[7] Banala SK, Kim SH, Agrawal SK, Scholz JP. Robot assisted gait training with active leg exoskeleton (ALEX). IEEE Trans Neural Syst Rehabilit Eng 2009;17(February (1)).
[8] Zhang J-f, Dong Y-m, Yang C-j, Geng Y, Chen Y, Yang Y. 5-Link model based gait trajectory adaption control strategies of the gait rehabilitation exoskeleton for post-stroke patients. J Mechatron 2010;20(April (3)):368–76.
[9] Czarnetzki S, Kerner S, Urbann O. Observer-based dynamic walking control for biped robots. Robot Auton Syst 2009;57(31 July (8)):839–45. Humanoid Soccer Robots.
[10] Kazerooni H, Racine J-L, Huang L, Steger R. On the control of the Berkeley lower extremity exoskeleton (BLEEX). Proceedings of the 2005 IEEE international conference on robotics and automation. 2005.
[11] Beyl P, Van Damme M, Van Ham R, Versluys R, Vanderborght B, Lefeber D. An exoskeleton for gait rehabilitation: prototype design and control principle. 2008 IEEE international conference on robotics and automation pasadena. 2008.
[12] Longman RW. Iterative learning control and repetitive control for engineering practice. Int J Control 2010;73(10):930–54 Special Issue on Iterative Learning Control.
[13] Bouakrif F. Iterative learning control with forgetting factor for robot manipulator system. Int J Robot Autom 2011;26:206–3407.
[14] Phan MQ, Longman RW, Panomruttanarug B, Lee SC. Robustification of iterative learning control and repetitive control by averaging. Int J Control 2013;86(5):855–68.
[15] Majeed APPA, Taha Z, Abidin AFZ, Zakaria MA, Khairuddin IM, Razman MAM, Mohamed Z. The control of a lower limb exoskeleton for gait rehabilitation a hybrid active force control approach. Proc Comput Sci 2017;105:183–90.
[16] Torrealba RR, Udelman SB, Fonseca-Rojas ED. Design of variable impedance actuator for knee joint of a portable human gait rehabilitation exoskeleton. Mech Mach Theory 2017;116(October):248–61.
[17] Oh S, Baek E, Song S-k, Mohammed S, Jeon D, Kong K. A generalized control framework of assistive controllers and its application to lower limb exoskeletons. Robot Auton Syst 2015;73(November):68–77.
[18] Wu J, Gao J, Song R, Li R, Li Y, Jiang L. The design and control of a 3DOF lower limb rehabilitation robot. Mechatronics 2016;33(February):13–22.
[19] Akdogan E, Adli MA. The design and control of therapeutic exercise robot for lower limb rehabilitation: Physiotherabot. Mechatronics 2011;21(April (3)):509–22.
[20] Belkadi A, Oulhadj H, Touati Y, Khan SafdarA, Daachi B. On the robust PID adaptive controller for exoskeleton: a partical swarm optimization based approach. Appl Soft Comput. 2017;60(November):87–100.
[21] Liu L, Leonhardt S, Misgeld BJE. Design and control of mechanical rotary variable impedance actuator. Mechatronics 2016;39(November):226–36.
[22] Oliver-Salazar MA, Szwedowicz-Wasik D, Blanco-Ortega A, Aguilar-Acevedo F, Ruiz-Gonzalez R. Characterization of pneumatic muscles and their use for the position control of a mechatronic finger. Mechatronics 2017;42(April):25–40.
[23] Yeh T-J, Wu M-J, Lu T-J, Wu F-K, Huang C-R. Control of McKibben pneumatic muscles for a power-assist lower-limb orthosis. Mechatronics 2010;20(September (6)):686–97.
[24] Hamaya M, Matsubara T, Noda T, Teramae T, Morimoto J. Learning assistive strategies for exoskeleton robots from user-robot physical interaction. Pattern Recognit Lett 2017;99(November (1)):67–76.
[25] Ganesan Y, Gobee S, Durairajah V. Development of an upper limb exoskeleton for rehabilitation with feedback from EMG and IMU sensor. Proc Comput Sci 2015;76:53–9.
[26] Hyun DJ, Park H, Ha T, Park S, Jung K. Biomechanical design of an agile, electricity-powered lower-limb exoskeleton for weight-bearing assistance. Robot Auton Syst 2017;95(September):181–95.
[27] Zhang X, Wang H, Tian Y, Peyrodie L, Wang X. Model-free based neural network control with time-delay estimation for lower extremity exoskeleton. Neurocomputing 2018;272(January (10)):178–88.
[28] Long Y, Du Z, Cong L, Wang W, Zhang Z, Dong W. Active disturbance rejection control based human gait tracking for lower extremity rehabilitation exoskeleton. ISA Trans 2017;67(March):389–97.
[29] Li M, Yuan Z, Wang X, Hasegawa Y. Electric stimulation and cooperative control for paraplegic patient wearing an exoskeleton. Robot Auton Syst 2017;98(December):204–12.
[30] Long Y, Du Z, Chen C, Wang W, He L, Mao X, Xu G, Zhao G, Li X, Dong W. Development and analysis of an electrically actuated lower extremity asistive exoskeleton. J Bionic Eng 2017;14(April (2)):272–83.
[31] Alias NA, Saiful Huq M, Ibrahim BSKK, Omar R. The efficacy of state of the art overground gait rehabilitation robotics: a bird's eye view. procedia computer science. IEEE international symposium on robotics and intelligent sensors, IRIS 2016. 2016.
[32] Mun K-R, Lim SB, Guo Z, Yu H. Biomechanical effects of body weight support with a novel robotic walker for over-ground gait rehabilitation. Med Biol Eng Comput 2016:1–12.
[33] Swinnen E, Baeyens J-P, Hens G, Knaepen K, Beckwee D, Michielsen M, Clijsen R, Kerckhofs E. Body weight support during robot-assisted walking: influence on the trunk and pelvis kinematics. NeuroRehabilitation 2015;36(1):81–91.