Modeling and Posture Control of Lower Limb Prosthesis Using Neural Networks

Ju-Won Lee, Gun-Ki Lee, *Member, KIMICS*

Abstract—The prosthesis of current commercialized apparatus has considerable problems, requiring improvement. Especially, LLP(Lower Limb Prosthesis)-related problems have improved, but it cannot provide normal walking because, mainly, the gait control of the LLP does not fit with patient's gait manner. To solve this problem, HCI((Human Computer Interaction) that adapts and controls LLP postures according to patient's gait manner more effectively is studied in this research. The proposed control technique has 2 steps: 1) the multilayer neural network forecasts angles of gait of LLP by using the angle of normal side of lower limbs; and 2) the adaptive neural controller manages the postures of the LLP based on the predicted joint angles. According to the experiment data, the prediction error of hip angles was 0.32[deg.], and the predicted error of knee angles was 0.12[deg.] for the estimated posture angles for the LLP. The performance data was obtained by applying the reference inputs of the LLP controller while walking. Accordingly, the control performance of the hip prosthesis improved by 80% due to the control postures of the LLP using the reference input when comparing with LQR controller.

Index Terms—Lower Limb Prosthesis, Neural Networks, Gait Control.

I. INTRODUCTION

The number of patients whose one-side or all limbs or the lower half of body is paralyzed or upper/lower limbs are amputated has increased due to industrial or traffic accidents [1]-[3]. Current commercial assistant equipment or artificial legs to help the patients have been developed, but it cannot provide normal walking because the gait control of the LLP does not fit with patient's gait manner and it needs long-term training for normal gait. Furthermore, patients feel fatigue when they wear it long time. The main reason for these problems is that patients have difficulty in controlling the gait angle while they are walking with it [2]-[4]. Human gait has a gait cycle in which a lower limb returns from one motion to another while it draws a circular arc. Namely, the cycle starts from when one leg goes forward and the corresponding heel strike touches the ground while the body advances, to when the foot again goes forward and touches the ground. Each cycle can be broken up into two phases: the stance phase and the swing phase. The stance phase is also called as supporting phase of the body, and can be divided into the heel strike phase, the loading response phase, the mid-stance phase and the terminal stance phase. The swing phase begins from when the foot is off the ground and the body advances ahead, to when the foot touches the ground [2]-[5]. Therefore, a gait means the lower limb repeats one motion to another drawing a circular arc. This gait phase includes a simple genetic reflex function, learned activities as well as personal characteristics about both legs movement patterns, and it involves various factors such as the peripheral-central nervous system and heart/lung functions. Also, the gait angle of each joint depends on patient's gait habit, physical size etc [2]-[5]. It is hard to obtain the normal gait pattern of the patients whose lower limbs are amputated. This paper presents the method to control the gait appropriately for each patient's gait manner using the neural network that learns how to solve above-mentioned problems. To implement the system, the gait angle was determined by using the knee angle of the normal lower limb when the patient wore an artificial leg; with the extracted gait angle, the gait posture of an artificial leg from the neural network that learned kinematics of normal human gait was estimated; then, the gait similar to a normal person's was provided through the adaptive gait control using the neural network. In conclusion, the performance of the presented system is analyzed.

Manuscript received May 20, 2004.
Ju-Won Lee and Gun-Ki Lee are with the Depart. of Electronic Engineering and ERI, Gyeongsang National University, 900 Gajwa-dong, Jinju-si, Kyungnam-do, Jinju 660-701, Korea (e-mail: elecwave@bioee.gsnu.ac.kr).

II. HIP PROSTHESIS MODELING

The hip prosthesis was modeled in this research in order to verify the controlling method for proper gait of a patient whose limbs below the hip were lost. The hip prosthesis is an artificial leg used after amputating all limbs below or including the hip. Artificial legs should give suitable appearance, convenience and maximum functions when the patient loses some or all lower limbs due to a congenital loss/disease or disaster. The artificial leg operates with inter-reaction between three joints (hip joint, knee joint and ankle joint) and feet; thus, kinetic modeling is required considering all [2]-[5]. In this study, however, an artificial leg with fixed ankle joint was modeled. Fig. 1 shows the modeling structure of the hip prosthesis. The hip joint in Fig. 1 was set as the origin of the motion ($x=0, y=0$); x_{tCG}, y_{tCG}, x_{sCG} and y_{sCG} (the positions of the center of gravity) considering L_{tCG} and L_{sCG} (the center of gravity of each link) are expressed as below:

$$x_{tCG} = L_{tCG} \sin\theta_h \quad (1)$$
$$y_{tCG} = -L_{tCG} \cos\theta_h \quad (2)$$
$$x_{sCG} = L_t \sin\theta_h + L_{sCG} \sin\theta_k \quad (3)$$
$$y_{sCG} = -L_t \cos\theta_h - L_{sCG} \cos\theta_k \quad (4)$$

$\dot{x}_{tCG}$, $\dot{y}_{tCG}$, $\dot{x}_{sCG}$ and $\dot{y}_{sCG}$ (the velocities of the center of gravity per link) can be obtained from differentiating Eq. (1) ~ (4) by t (time).

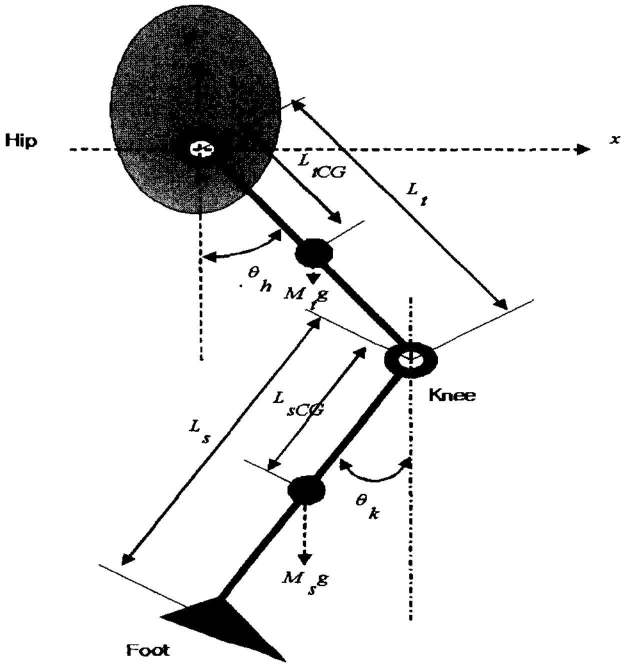

Fig. 1 Hip prosthesis model

$$\dot{x}_{tCG} = L_{tCG}\dot{\theta}_h \cos\theta_h \quad (5)$$
$$\dot{y}_{tCG} = L_{tCG}\,\dot{\theta}_h \sin\,\theta_h \quad (6)$$
$$\dot{x}_{sCG} = L_t\dot{\theta}_h \cos\theta_h + L_{sCG}\dot{\theta}_k \cos\theta_k \quad (7)$$
$$\dot{y}_{sCG} = L_t\dot{\theta}_h \sin\,\theta_h + L_{sCG}\,\dot{\theta}_k \sin\,\theta_k \quad (8)$$

Therefore, v_t and v_s (velocity vectors of the position of the center of gravity at the thigh and the shin) are:

$$\mathbf{v}_t = \begin{bmatrix} \dot{x}_{tCG} \\ \dot{y}_{tCG} \end{bmatrix} \quad (9)$$

$$\mathbf{v}_s = \begin{bmatrix} \dot{x}_{sCG} \\ \dot{y}_{sCG} \end{bmatrix} \quad (10)$$

$v_t^T v_t$ and $v_s^T v_s$ can be expressed as Eq. (11) and (12):

$$\mathbf{v}_t^T\mathbf{v}_t = \begin{bmatrix} \dot{x}_{tCG} \\ \dot{y}_{tCG} \end{bmatrix}^T \begin{bmatrix} \dot{x}_{tCG} \\ \dot{y}_{tCG} \end{bmatrix} = L_{tCG}^2\dot{\theta}_h^{\,2} \quad (11)$$

$$\mathbf{v}_s^T\mathbf{v}_s = \begin{bmatrix} \dot{x}_{sCG} \\ \dot{y}_{sCG} \end{bmatrix}^T \begin{bmatrix} \dot{x}_{sCG} \\ \dot{y}_{sCG} \end{bmatrix} \quad (12)$$
$$= L_t^2\,\dot{\theta}_h^{\,2} + L_{sCG}^2\,\dot{\theta}_k^{\,2} + 2L_tL_{sCG}\dot{\theta}_h\dot{\theta}_k\cos(\theta_h-\theta_k)$$

The kinetic energy T_t and T_s and the potential energy U_t and U_s of each link are expressed as below. The kinetic energy (T_t) and the potential energy (U_t) of the thigh are Eq. (13) and Eq. (14); the kinetic energy (T_s) and the potential energy (U_s) of the shin are Eq. (15) and Eq. (16):

$$T_t = \frac{1}{2}M_t\left(L_{tCG}^2\dot{\theta}_h^{\,2} + I_t\dot{\theta}_h^2\right) \quad (13)$$

$$U_t = -M_t g L_{tCG}\cos\theta_h \quad (14)$$

$$T_s = \frac{1}{2}M_sL_t^2\dot{\theta}_h^{\,2} + L_{sCG}^2\,\dot{\theta}_k^{\,2} + 2L_tL_{sCG}\dot{\theta}_h\dot{\theta}_k\cos(\theta_h-\theta_k) + \frac{1}{2}I_s\dot{\theta}_k^2 \quad (15)$$

$$U_s = -M_s g\left(L_t\cos\theta_h + L_{sCG}\cos\theta_k\right) \quad (16)$$

The total kinetic energy (T) and the potential energy (U) can be derived from Eq. (13) to (16).

$$T = \frac{1}{2}\sum_{j=1}^{2}\left(M_jv_j^2 + I_jw_j^{\,2}\right) = T_t + T_s \quad (17)$$
$$= \frac{1}{2}M_tL_{tCG}^2\dot{\theta}_h^{\,2} + \frac{1}{2}M_s\begin{Bmatrix} L_t^2\dot{\theta}_h^{\,2} + L_{sCG}^2\dot{\theta}_h^{\,2} \\ +2L_tL_{sCG}\dot{\theta}_h\dot{\theta}_k\cos(\theta_h-\theta_k) \end{Bmatrix}$$
$$+\frac{1}{2}\left(I_t\dot{\theta}_h^{\,2} + I_s\dot{\theta}_k^{\,2}\right)$$

$$U = \sum_{j=1}^{n} M_jgy_j = (U_t + U_s) \quad (18)$$
$$= -\left\{M_tgL_{tCG}\cos\theta_h + M_sg\left(L_t\cos\theta_h + L_{sCG}\cos\theta_k\right)\right\}$$

Lagrangian L is obtained from Eq. (17) and (18); the kinetic equation of one-side artificial leg in 2-dimentsion plane is determined by using Lagrangian equation [6][7].

$$L = T - U \quad (19)$$
$$= \frac{1}{2}M_s\begin{Bmatrix} L_t^2\dot{\theta}_h^{\,2} \\ +2L_t\dot{\theta}_hL_{sCG}\dot{\theta}_k\cos(\theta_h-\theta_k) \\ +L_{sCG}^2\dot{\theta}_k^{\,2} \end{Bmatrix}$$
$$+M_tgL_{tCG}\cos\theta_h + M_sg\begin{pmatrix} L_t\cos\theta_h \\ +L_{sCG}\cos\theta_k \end{pmatrix}$$
$$+\frac{1}{2}M_tL_{tCG}^2\dot{\theta}_h^{\,2} + \frac{1}{2}I_t\dot{\theta}_h^{\,2} + \frac{1}{2}I_s\dot{\theta}_k^{\,2}$$

The torque (τ_h) of the hip joint angle and the torque (τ_k) of the knee joint angle can be derived using Lagrangian equation:

$$\tau_h = \left(M_tL_{tCG}^2 + M_sL_t^2 + I_t\right)\ddot{\theta}_h + M_sL_tL_{sCG}\ddot{\theta}_k\cos(\theta_h-\theta_k)$$
$$+M_sL_tL_{sCG}\dot{\theta}_k^{\,2}\sin(\theta_h-\theta_k) + g\left(M_tL_{tCG} + M_sL_t\right)\sin\theta_h \quad (20)$$

$$\tau_k = M_sL_tL_{sCG}\ddot{\theta}_h\ \cos(\theta_h-\theta_k) + \left(M_sL_{sCG}^2 + I_s\right)\ddot{\theta}_k$$
$$-M_sL_{sCG}\dot{\theta}_h^{\,2}L_t\sin(\theta_h-\theta_k) + M_sgL_{sCG}\sin\theta_k \quad (21)$$

Eq. (20) and (21) does not consider the ground reaction force occurs in human walking. If the reaction force per the vertical/horizontal position of the foot on the ground is considered, the kinetic equation of the dynamic modeling of lower limb for human gait can be expressed as Eq. (22) and (23):

$$\begin{aligned}\tau_h =& \left(M_t L_{tCG}^2 + M_s L_t^2 + I_t\right)\ddot{\theta}_h + M_s L_t L_{sCG}\ddot{\theta}_k \cos(\theta_h - \theta_k) \\ &+ M_s L_t L_{sCG}\dot{\theta}_k^2 \sin(\theta_h - \theta_k) + g\left(M_t L_{tCG} + M_s L_t\right)\sin\theta_h \\ &- X_G L_t \sin\theta_h + Y_g L_t \cos\theta_h\end{aligned} \tag{22}$$

$$\begin{aligned}\tau_k =& M_s L_t L_{sCG}\ddot{\theta}_h \cos(\theta_h - \theta_k) + \left(M_s L_{sCG}^2 + I_s\right)\ddot{\theta}_k \\ &- M_s L_t L_{sCG}\dot{\theta}_h^2 \sin(\theta_h - \theta_k) + M_s g L_{sCG}\sin\theta_k \\ &- X_g L_s \sin\theta_k + Y_g L_s \cos\theta_k\end{aligned} \tag{23}$$

The physical parameters and the initial conditions of all status were set up as 0. The Physical parameters for simulation are as shown Table 1.

Table 1. Physical parameters of an artificial leg

Parameters	Units	Hip prosthesis
L_t	[m]	0.42
L_{sCG}	[m]	0.18
L_s	[m]	0.51
L_{tCG}	[m]	0.24
M_t	[kg]	8.1
M_s	[kg]	4.5
I_t	Kgm^2	0.06
I_s	kgm^2	0.11

III. PROPOSED CONTROL SYSTEM

Human gait depends on walking speed, walking posture, the type of road etc. Accordingly, this research assumes two conditions to predict patient's gait angle. First, the characteristic of the gait of the amputated leg is same as those of normal lower limb; second, the angle of knee increases if the knee joint bends and decreases if extends. In this study, the control structure and technique are suggested as shown Fig. 2. The proposed method is to realize the same gait posture of the patients whose one-side lower limb is lost as normal person. The characteristics of proposed method is that first, the neural network learned a normal person's gait manner, which was similar to the patient; second, the hip joint angel and the posture angles of the artificial leg for walking were estimated based on the knee angle of normal one-side lower limb; third, the predicted angle was set up as the target value to control the posture adaptively.

A. Gait angle prediction and Results of Simulation

In this study, the MNN(multilayer neural networks) to predict the gait angles of LLP were used in this method as shown in Fig. 3. In Fig. 3, the input of neural network to estimate the angles of LLP used a knee angle signal that was obtained from the knee joint angle of the normal lower limb. The knee angle sensor to measure angles of the lower limb while walking used a tilt sensor (DAS's TILT SA1 [8]) which has ±60° ranges as shown Table 2. The outputs of MNN to predict the angles of LLP are a estimated hip angle, $\hat{\theta}_h^R$ and a estimated knee angle $\hat{\theta}_k^R$.

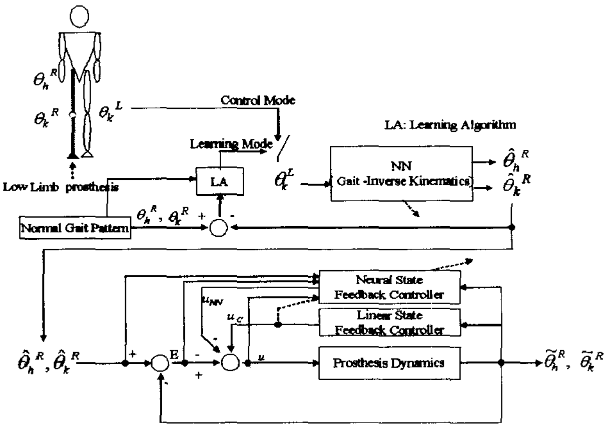

Fig. 2 Proposed structure to control the posture of the prosthesis

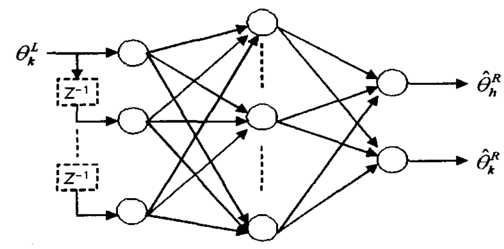

Fig. 3 Structure of the gait angle predictor based on the estimated knee angle and the multilayer neural network

Table 2. Specifications of the tilt sensor.

Measuring range	±60°
Resolution	<0.1 degrees
Non-Linearity	<1% FS
Response time	<0.5% at ±60°tilt
Power supply	5Vdc
Current consumption	<1mA
Sensitivity	appox 30mV/°
Zero offset at 5V	2.5±0.2Volt
Output impedance	10kOhm

This proposes the identification method of the dynamics for patient's walking pattern. Namely, to estimate the hip joint and knee joint angles of the artificial leg of the patient whose one-side leg is lost, this method obtains physically similar person's the gait angel and uses the data for learning the gait angle estimator. The learning algorithm used EBPA(Error Back Propagation Algorithm) of the multi-layer. Below is the neural network learning method [9]-[11], and the output O_i, O_j and O_k of neurons in the input layer, hidden layer and output layer are:

$$O_i = x_i \tag{24}$$

$$O_j = \lambda\, f\left(net_j\right), \qquad net_j = \sum_{i=1}^{I} w_{ji} O_i \tag{25}$$

$$O_k = \lambda\, f\left(\sum_{k=1}^{K} net_k\right), \qquad net_k = \sum_{j=1}^{J} w_{kj} O_j \tag{26}$$

where f is the activate function; net_j and net_k are sums of the multiplication of the previous neuron output and the current layer weight; λ is the slope of f. To learn the neural network, the error from the desired value vector of the neural network, $d = [\theta_h^R \;\; \theta_k^R]$, is obtained as shown in Eq. (27).

$$E = \frac{1}{2}\begin{bmatrix} \left(\theta_h^R - \hat{\theta}_h^R\right)^2 \\ \left(\theta_k^R - \hat{\theta}_k^R\right)^2 \end{bmatrix} \tag{27}$$

The purpose of the learning is to adjust the weight to minimize E. For this, the weight should changes to the negative gradient direction. Therefore, the weight variation can be obtained by partially differentiating the direction vector of the weight for the error. The weight variation in each layer is:

$$\Delta w_{kj} = -\eta \frac{\partial E}{\partial w_{kj}}, \qquad \eta > 0 \tag{28}$$

where η is a learning constant. To minimize the error, the weight in the hidden layer should change to the negative gradient direction.

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}}, \qquad \eta > 0 \tag{29}$$

Therefore, the weights variations are:

$$w_{ji} = w_{ji} + \Delta w_{ji} \tag{30.a}$$

$$w_{kj} = w_{kj} + \Delta w_{kj} \tag{30.b}$$

The initial weight ranges –0.5 to 0.5. Namely, the gait posture angle of each joint of the artificial leg was estimated using the knee joint angle estimated during walking with forward operation of the neural network, based on the weight data of the neural network which learns the normal gait data. To verify the performance of the gait angle estimator of the artificial leg during walking, Table 3's parameters were tested. Fig. 4 and Table 4 and 5 show the result. According to Table 4 and 5, the estimated error average of each joint is 0.22, giving 97.5% of the accuracy of the estimated posture angle, which can be expected that the estimator is available for controlling the motion of the artificial leg for walking. In this study, the estimated angle was set as the reference signal of the posture controller.

Table 3. Parameters of neural network to estimate the gait angles

Input Neurons	100
Hidden Neurons	20
Output Neurons	2
Activation function	bipolar sigmoid
Learning rate	0.1
Iterations	200

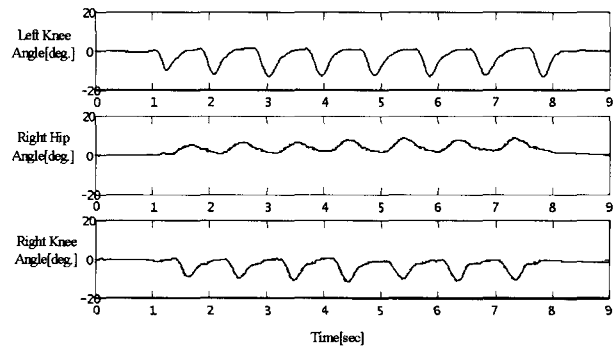

Fig. 4 Estimated angles of LLP during walking

Table 4. Errors of the estimated angles using the MLNN

simulation Item	Absolute Mean Error	
	Hip	Knee
gait	0.32	0.12

IV. GAIT POSTURE CONTROL

In this study, the multilayer neural network was used to control each joint angle of the artificial leg. The controlling algorithm of this controller is the feed-forward learning controller suggested by Kawato [11], known as on-line learning controller. Traditional feedback controller is placed in parallel in this neural network controller, and the neural network repeats the required period while the feed-forward error is feedback through the neural network, learning and controlling the system on line. Namely, the feed-forward error, which is the output of the feed-forward controller, is feedback to learn the neural network in plenty of learning cycles until the error converges. During converging, the neural network learns the inverse of the plant to control it, as shown in Fig. 2. In Fig. 2, the overall control outputs u_H, u_K are:

$$u_H = u_H^C + u_H^{NN}, \quad u_K = u_K^c + u_K^{NN} \tag{31}$$

where u^C and u^{NN} are the outputs of the linear controller and the neural network controller respectively.

A. Design of the LQR controller

To design the feedback controller of state variables, the controllability and observability should be checked [12]. If the control u, which allows the initial state $x(0)$ of the system, which is matrix (A,B), to move to other desired place $x(t)$, exists, it is controllable. If Eq. (32) is the state equation of the modeled artificial system, Eq. (33) is applied to investigate the controllability.

$$\dot{\mathbf{x}} = \mathbf{A}x + \mathbf{B}u, \ \mathbf{y} = \mathbf{C}x \tag{32}$$

$$rank\left[B\ AB\ A^2B \ldots A^{n-1}B\right] = n \tag{33}$$

The controllability of the system can be determined by checking the given algebraic conditions of the matrix. The necessary and sufficient condition of that all roots of an equation can be placed in desired positions in s-plane is that the system should be observed and controllable. Therefore, if an output has any factor affects on each state variable, the system is observable. The necessary and sufficient condition of that the system is observable is that the initial state *x(0)* can be determined from the given control input *u(t)* and the output *y(t)* at limited time *T*. Namely, if Q_R matrix is not zero, the system is observable. Observability can be obtained from Eq. (34):

$$\mathbf{Q_R} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{34}$$

In this study, the ranks of the modeled artificial hip prosthesis systems is 4; Q_R is not 0, so the systems are controllable and observable. In this study, the state feedback controller was designed as the LQR controller. Since the regulator is time-independent feedback controller, the plant is maintained within the desired deviation from the reference state, by using containable amount of contains. Because all state measurement is valid for the artificial models in this research, all states can be feedback. For optimal control, the state feedback gain matrix *K* is obtained from Eq. (35) to minimize the performance index [12].

$$J = \int_0^\infty (\mathbf{x'Qx} + \mathbf{u'Ru})\, dt \tag{35}$$

$$\mathbf{u} = \begin{bmatrix} u_h \\ u_k \end{bmatrix} = \begin{bmatrix} k_1(\theta_h - x_1) - (k_2x_2 + k_3x_3 + k_4x_4) \\ k_2(\theta_k - x_2) - (k_1x_1 + k_3x_3 + k_4x_4) \end{bmatrix} \tag{36}$$

Q-matrix and *R* can be defined as below; the gain matrix *k* is obtained to design the controller.

$$Q = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \end{bmatrix}, \ R = \begin{bmatrix} 1\,0 \\ 0\,1 \end{bmatrix} \tag{37}$$

$$\mathbf{K} = \begin{bmatrix} 3.0940 & -0.0720 & 2.7288 & 0.6198 \\ 0.0607 & 3.5701 & 0.7098 & 1.7839 \end{bmatrix}$$

B. Neural Controller

In neural networks controller, the learning of the feed-forward error-learning controller minimizes the error function *E* given from Eq. (38). The desired output vector $d(n) = [\hat{\theta}_k, \hat{\theta}_h]$; the output of the artificial system is $y(n) = [\tilde{\theta}_k, \tilde{\theta}_h]$.

$$E = \frac{1}{2}\begin{bmatrix} \left(\hat{\theta}_h^R - \tilde{\theta}_h^R\right)^2 \\ \left(\hat{\theta}_k^R - \tilde{\theta}_k^R\right)^2 \end{bmatrix} \tag{38}$$

If the activate function for the output layer of this neural network controller has a linear function *net=f(net)*, the output layer and the hidden layer are calculated using the gradient descent learning algorithm, as Eq. (39) and (40) respectively:

$$\Delta w_{kj}(n+1) = -\eta \frac{\partial E}{\partial w_{kj}(n)}, \quad \eta > 0 \tag{39}$$

$$\Delta w_{ji}(n+1) = -\eta \frac{\partial E}{\partial w_{ji}(n)}, \quad \eta > 0 \tag{40}$$

The neural networks controller is to learn Jacobian of hip prosthesis system, $dy_k/du_{H,K}$.

C. Experiment and Result

To verify the performance of the suggested controlling method, the posture controller with the structure in Fig. 2 was designed and simulated. The structure of the neural network for controlling the posture of the hip prosthesis is time-delaying neural network; 18 input neurons, 8 hidden layers and 2 output layers consist in. The activate function is sigmoid. The input vector of the controlling neural network is the state vector of hip joint angle ($\theta_h(n)$), its angular velocity ($\dot{\theta}_h$), knee joint angle ($\theta_k(n)$), its angular velocity ($\dot{\theta}_k$) of the artificial leg; each first degree delaying is $\theta_h(n-1)$, $\theta_k(n-1)$, $\dot{\theta}_h(n-1)$ and $\dot{\theta}_k(n-1)$; the control errors are $e(n)$ and $e(n-1)$; the hip joint angles, which are the estimation outputs of the neural network, are $\hat{\theta}_h(n)$ and $\hat{\theta}_h(n-1)$; the knee joint angles are $\hat{\theta}_k(n)$ and $\hat{\theta}_k(n-1)$; the outputs of the state feed-forward controller are $u_k^C(n)$, $u_h^C(n)$, $u_h^C(n-1)$ and $u_k^C(n-1)$; overall control outputs $u_h(n)$ and $u_k(n)$ are set up as inputs. This controller has same control rules as artificial thigh controller; the posture is adaptively controlled through the learning convergence of the neural network controller while the LQR controller controls linearly. LQR controller's gain derives from Eq. (49). Fig. 5 and Table 5 show the control result when the neural network is learned 12 times. According to the data, the neural network controller provided improvement after learning, depending on patient's gait status.

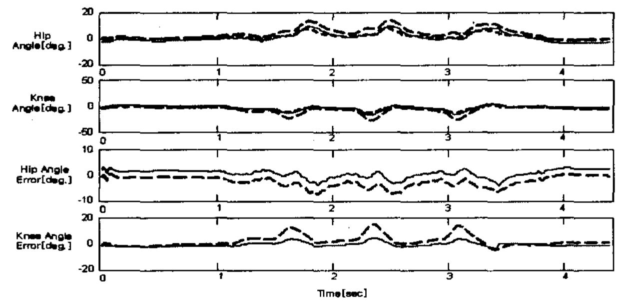

Fig. 5 Result of the adaptive posture control of knee and hip joints while walking ; NN(-), LQR(--), Estimated angle (···).

Table 5. Error of the posture control using the neural controller

simulation Item	Absolute Mean Error			
	LQR		NN	
	Hip	Knee	Hip	Knee
gait	2.212	3.114	1.312	1.105

V. CONCLUSION

In this study, the posture control technique was proposed using the learning function of the artificial neural network in order to recover patient's walking pattern to the normal person's. The proposed technique predicts the gait postures of an artificial leg from the neural network that learns the normal peoples gait dynamics based on the estimated current gait angles using the knee angle of the normal lower limb. According to the experimental result, the absolute average error of the estimated angles of LLP was 0.22[deg.], showing excellent estimation performance. The simulation verifies that the posture angles during walking can be used as inputs of the controller. This technique controls the posture of the artificial leg using the estimated gait angles, providing more 80% controllability than LQR controller, and is expected to control the artificial leg more effectively.

REFERENCES

[1] Park, Sang Hee, Status and Prospect of Prosthesis Controller Using EMG signal, Electricity Research Institute Journal 34(9) : 553-561, 1985.

[2] Kim, Kyong Hoon, The development and evaluation of Korean artificial limb, Ministry of Heal & Welfare, Research Report, 1998.

[3] Myer Kutz, Bioemedical Engineering & Design, HcGraw-Hill Standard Handbooks, New York, 2003.

[4] Nihat Ozkaya, Margareta Nordin, Fundamentals of Biomechanics, Springer-Verlag. 1999.

[5] Esquenanzi A, Keenan MA, Gait Analysis In Rehabilition Medicine, Principles and Practice, Delisa JA, Gans BM (Eds.), JB Lippincott Co., 1993.

[6] V.D. Kalanovic, D. Popovic, N.T. Skaug, Feedback Error Learing Neural Network for Trans-Femoral Prosthesis, IEEE, Trans. on Rehabilitation Engineering, 8 (1), 71-80, 2000.

[7] John J. Craig, Introduction to Robotics Mechanics and Control, Addison-Wesley, 1989.

[8] DAS, Tilt SA1 Sensor Data Sheet, Digital Advanced Sensor Technology Co., 2001.

[9] Jacek M. Zurada, Introduction to Artificial Neural System, West Publishing Company, 1992.

[10] Lin Lee, Neural Fuzzy System, Printice Hall, 1996.

[11] Sigeru Omatu, Neuro-Control and Its Applications, Springer, 1996.

[12] Katsuhiko Ogata, Discrete-Time Control Systems, Prentice-Hall, 1994.

Ju-Won Lee

Received the B.S. degree in electronic engineering of Jinju National University in 1997, M.S degree in electronic communi-cation engineering of Korea Mari-time National University in 1999 and Ph.D. degree in electronic engineering of Gyeongsang National University, South of Korea in 2003. Currently, he is studying for robot control using electromyogram in Biomedical Electronic Eng. Lab. of Gyeongsang National University. He is interested in biomedical engineering, artificial intelligence, HCI and related areas.

Gun-Ki Lee

Received his M.S degree in electrical engineering in 1978, and Ph.D. degree in biomedical engine-ering in 1990 all from Yonsei University. He is presently professor of the Department of electronic engineering and president of Engineering Research Institute (ERI), Gyeongsang national university. He is interested in biomedical signal processing, telemedicine and ubiquitous networks.