Open Access Journal

Journal of Sustainable Research in Engineering Vol. 5 (3) 2020, 138-148

Journal homepage: http://sri.jkuat.ac.ke/ojs/index.php/sri

ISSN (Online): 2409-1243

Gain tuning using neural network for contact force control of flexible arm

Minoru Sasaki*[1], Nobuto Honda[1], Waweru Njeri[1,2], Kojiro Matsushita[1] and Harrison Ngetha[2]

[1]*Gifu University (Department of Mechanical Engineering, 1-1 Yanagido, Gifu City, 1193, Japan).*

[2]*Dedan Kimathi University of Technology (Department of Electrical & Electronic Engineering, P.O Box 657-10100, Nyeri, Kenya).*

**Corresponding Author - Email: sasaki@gifu-u.ac.jp.*

Abstract This paper presents contact force control of a one link flexible arm consisting of a simple boundary feedback of bending moment at the base of the flexible arm. Gain tuning control system using neural network was developed and its control performance examined and compared with fixed gains by numerical simulation and experiment. In this work, feedback gain was tuned to correspond to the coupling coefficient of the neural network, and stabilized the learning by giving the initial value to the coupling coefficient of the neural network, thereby shortening the learning time. To adjust the gain value in real time, a sequential correction type technique(online learning) that repeats learning at every sampling was adopted as the learning scheme of the neural network. As a result, it was confirmed that by using the neural network, the value of the feedback gain was adaptively changed and the target contact force converged after 0.35 seconds. Comparing the performance with that obtained with fixed gain, it was found that neural network tuned controller took a shorter time to converge to the target value by 0.8 seconds confirming that the proposed controller is more effective for the contact force control of the flexible arm.

Keywords Flexible arm, contact force control, neural networks, gain tuning

1. Introduction

For heavy and highly rigid robotic arms used in factories and the like, it is common to increase the rigidity by increasing the wall thickness of the arm so as to obtain high accuracy in determining the position of the tip. In recent years, however, there has been an increasing demand for weight reduction of robot arms in order to realize high speed operation and energy saving. As the rigidity decreases however, the influence of elastic deformation due to the flexibility of the arm becomes large, so that position error and elastic vibration cannot be ignored [1]–[3]. Against such a background, a lot of research on control of flexible manipulators have been actively conducted.

In addition to the requirement mentioned above in regards to the need for lightweight robots, use of nursing care robots and welfare robots under human-contacting environments is on the rise. To work safely together with people, it is therefore necessary to control the contact force in addition to the positioning and vibration control [4]–[6].

In this paper, force control problem was examined for the constrained one-link flexible arm. In this conventional study, a proposal is made to extend the simple boundary feedback consisting of a bending moment at the root of the flexible arm and its time derivative against the force control problem of the constrained one-link flexible arm earlier developed in references [7], [8]. In this works, feedback control was used to obtain the exponential stability of the closed loop system. Researchers in [9]–[11] conducted several simulations to examine the effectiveness of feedback techniques involving tuning the feedback gains, but the method of adjusting the performance verification and feedback gain by the experiment of the proposed controller remained a problem.

Therefore, in this research, a one-link flexible arms was developed, first to experimentally verify the controller earlier proposed in [7], [8] and confirm its effectiveness. Next, a control system using a neural network controller for autotuning feedback gains was designed. It was experimentally verified, compared and examined the control performance of the neuro controller against

the fixed gain controller earlier proposed. Similar work in literature related to the control of flexible manipulator using neural networks includes [12]–[15].

The contribution of this paper is in the derivation of the dynamic model of a constrained one-link flexible manipulator, derivation of the solution of the model equation of motion in s domain and the development of a tuning algorithm using artificial neural network to tune PI gains of a one-link flexible arm for the control of contact force.

The rest of the paper is organized as follows: modeling of the flexible arm is presented in section 2, solution to the equation of motion in the frequency domain is presented in section 3, determination of feedback gains by trial and error is described in section 4, background theory of multilayer neural network is given is section 5, neural network controller design is presented in section 6, experimental results are presented and discussed in section 7 followed by conclusions drawn in section 8.

2. Modelling of One Link Flexible Arm

The model of one link Flexible Bernoulli-Euler Arm in this study is shown in Figure 1. Variable used in the discussion are as tabulated in Table 1

Table 1. List of variables

l	-	Length of the arm
E	-	Young's modulus
I	-	Secondary moment of area of the beam
J	-	Inertia of the motor, load and the gear system
θ_d	-	Desired joint angle
$\theta(t)$	-	Motor shaft angle
τ_m	-	Motor torque
ρ	-	Linear density of the arm
τ_a	-	Motor torque
$w(x,t)$	-	Transverse displacement at point x
EI	-	Flexure rigidity
$\lambda(t)$	-	Lagrange multiplier

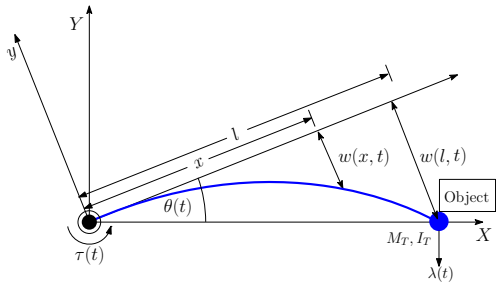

Fig. 1. Flexible Bernoulli-Euler Arm contact with an object

The base of Flexible Bernoulli-Euler Arm is equipped with a motor to rotate the arm, and rotation is controlled by the actuator. Also, the tip of the arm is in contact with an object. The equation of motion(derived in details in Appendix A) and the boundary condition of this model can be obtained as follows.

$$\begin{aligned} w(x,t) - x\theta(t) &= y(x,t) \\ \ddot{y}(x,t) + \frac{EI}{\rho} y''''(x,t) &= 0 \quad (1) \\ J\ddot{\theta}(t) = \tau_a(t) - EIw''(0,t) &= \tau(t) \\ y(0,t) = y(l,t) = y''(l,t) = l\theta(t) - w(l,t) &= 0 \\ EIw'''(l,t) &= \lambda(t) \end{aligned}$$

To control the contact force of the tip of the arm, it is necessary to consider boundary control. Here, denoting the target contact force by λ_d and the target values of transverse position and joint angle are denoted as $y_d(x)$ and θ_d respectively, the following relational expression is obtained.

$$\begin{aligned} y_d(x) &= \frac{\lambda_d}{6EI} x(2l^2 - 3lx + x^2) \quad (2) \\ \theta_d &= -y_d'(0) = -\frac{l^2\lambda_d}{3EI} \end{aligned}$$

The objection of the controller is to satisfy the following conditions

$$\begin{cases} y(x,t) \to y_d(x) \\ \dot{y}(x,t) \to 0 \end{cases}$$

to which the following control law is proposed

$$\tau(t) = -\tilde{k}_1 EI[y''(0,t) - y_d''(0)] - \tilde{k}_2 EI\dot{y}''(0,t) \quad (3)$$

Where the feedback gains $\tilde{k}_1$ and $\tilde{k}_2$ are positive constants. In the control law in equation 3, the first item is the bending moment $EIy''(0,t)$ of the base of the arm, we desire that the bending moment will approach the target value $EIy_d''(0,t)$ in a manner that is asymptotic. The second item is a term for vibration suppression which act to increase the damping the control system [11](presented in details in Appendix B).

3. Solution to the equation of motion in s domain

This section present the solution of the fourth order linear differential equation of motion in equation 1 in Laplace transform domain. Taking the Laplace transform of the equation of motion, the boundary condition, and the control law, we have

$$\begin{aligned} y(x,s) &= w(x,s) - x\theta(s) \\ 0 &= s^2 y(x,s) + \frac{EI}{\rho} y''''(x,s) \quad (4) \\ Js^2\theta(s) &= \tau_a(s) - EIw''(x,s) = \tau(s) \\ y(0,s) &= y(l,s) = y''(l,s) = l\theta - w(l,s) = 0 \\ \lambda(s) &= EIw'''(l,s) \\ \tau(s) &= -\tilde{k}_1 EI[y''(0,s) - \frac{1}{s} y_d''(0)] - \tilde{k}_2 sEIy''(0,s) \quad (5) \end{aligned}$$

Solving for $y(x,s)$ from equation 4 yields

$$\begin{aligned} y(x,s) =& F_1(s)\exp\{-jx\frac{\left(s^2E^3I^3\rho\right)^{\frac{1}{4}}}{EI}\} \\ &+ F_2(s)\exp\{jx\frac{\left(s^2E^3I^3\rho\right)^{\frac{1}{4}}}{EI}\} \\ &+ F_3(s)\exp\{-x\frac{\left(s^2E^3I^3\rho\right)^{\frac{1}{4}}}{EI}\} \\ &+ F_4(s)\exp\{x\frac{\left(s^2E^3I^3\rho\right)^{\frac{1}{4}}}{EI}\} \end{aligned} \tag{6}$$

where $F_1(s)$, $F_2(s)$, $F_3(s)$ and $F_4(s)$ are unknowns and j is the complex operator for imaginary values. With

$$s^2\theta(s) = \frac{\tau(s)}{J} \tag{7}$$

Substituting 5 into equation 7 and taking $k_1 = \tilde{k}_1/J$, $k_2 = \tilde{k}_2/J$

$$s^2\theta(s) = -k_1EI[y''(0,s) - \frac{1}{s}y_d''(0)] - k_2sEIy''(0,s) \tag{8}$$

From 6

$$\begin{aligned} y''(0,s) =& -F_1(s)\frac{\sqrt{-s^2E^3I^3\rho}}{E^2I^2} + F_2(s)\frac{\sqrt{-s^2E^3I^3\rho}}{E^2I^2} \\ &- F_3(s)\frac{\sqrt{-s^2E^3I^3\rho}}{E^2I^2} + F_4(s)\frac{\sqrt{-s^2E^3I^3\rho}}{E^2I^2} \end{aligned} \tag{9}$$

From 2

$$y_d''(0) = \frac{\lambda_d l}{sEI} \tag{10}$$

Substituting equations 9 and 10 into the equation 8

$$\begin{aligned} s^2\theta(s) =& -\frac{1}{EIs}(-k_1F_1(s)\sqrt{-s^2E^3I^3\rho}s \\ &+ k_1F_2(s)\sqrt{-s^2E^3I^3\rho}s \\ &- k_1F_3(s)\sqrt{(-s^2E^3I^3\rho}s \\ &+ k_1F_4(s)\sqrt{-s^2E^3I^3\rho}s \\ &+ k_1F_1(s)\lambda_d lEI \\ &- k_2\sqrt{-s^2E^3I^3\rho} \\ &+ k_2s^2F_2(s)\sqrt{-s^2E^3I^3\rho} \\ &- k_2s^2F_3(s\sqrt{-s^2E^3I^3\rho}) \\ &+ k_2s^2F_4(s)\sqrt{-s^2E^3I^3\rho})) \end{aligned} \tag{11}$$

$$\begin{aligned} \theta(s) =& \frac{-s(k_1+sk_2)(-F_1(s)+F_2(s)-F_3(s)}{EIs^3} \\ &+ \frac{F_4(s))\sqrt{-s^2E^3I^3\rho} - k_1\lambda_d lEI}{EIs^3} \end{aligned} \tag{12}$$

Substituting equations 6 and 12 into equation 4

$$\begin{aligned} w(x,s) =& F_1(s)\exp\{-j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &+ F_2(s)\exp\{-j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &+ F_3(s)\exp\{-j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &+ F_4(s)\exp\{-j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &- \frac{x(-s(k_1+sk_2)(F_1(s))\sqrt{-s^2E^3I^3\rho})}{EIs^3} \\ &- \frac{x(-s(k_1+sk_2)(F_1(s)) - k_1\lambda_d lEI)}{EIs^3} \\ &+ \frac{x(-s(k_1+sk_2)F_2(s)(\sqrt{-s^2E^3I^3\rho}))}{EIs^3} \\ &+ \frac{x(-s(k_1+sk_2)F_2(s)(-k_1\lambda_d lEI))}{EIs^3} \\ &- \frac{x(-s(k_1+sk_2)F_3(s)(\sqrt{-s^2E^3I^3\rho}))}{EIs^3} \\ &- \frac{x(-s(k_1+sk_2)F_3(s)(-k_1\lambda_d lEI))}{EIs^3} \\ &+ \frac{x(-s(k_1+sk_2)F_4(s)(\sqrt{-s^2E^3I^3\rho}))}{EIs^3} \\ &+ \frac{x(-s(k_1+sk_2)F_4(s)(-k_1\lambda_d lEI))}{EIs^3} \end{aligned} \tag{13}$$

$$\begin{aligned} w'''(l,s) =& \frac{1}{E^3I^3}[(-s^2E^3I^3\rho)^{\frac{3}{4}}IF_1(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &- F_2(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &- IF_3(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &+ F_4(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\}] \end{aligned} \tag{14}$$

Substituting equation 14 into equation 5

$$\begin{aligned} \lambda(s) =& \frac{1}{E^2I^2}[(-s^2E^3I^3\rho)^{\frac{3}{4}}(IF_1(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &- F_2(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &- IF_3(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\} \\ &+ F_4(s)\exp\{j(\frac{s^2E^3I^3\rho}{EI})^{\frac{1}{4}}x\})] \end{aligned} \tag{15}$$

By substituting the four unknowns $F_1(s)$, $F_2(s)$, $F_3(s)$, $F_4(s)$ from the simultaneous equations of the equation 5 into 13, we get the general solution of $w(x,s)$. Based on the above, the block diagram of the 1-link Flexible Arm system is shown in Figure 2.

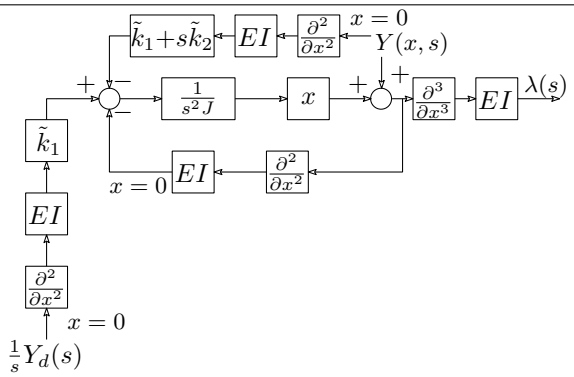

Fig. 2. Block diagram of 1 link flexible Bernoulli-Euler Arm

4. Determination of feedback gains $\tilde{k}_1$ and $\tilde{k}_2$ by simulation

Our arm comprised of a slender having cross-sectional breadth b and height h. Thus, in the simulation, the cross sectional area $A = bh$ (m^2), thus, the geometrical moment of inertia $I = \frac{bh^3}{12}$ (m^4), the linear density ρ (kgm^{-1}) is obtained as $\rho = \rho_a bh$. The feedback gain to be used is set to $\tilde{k}_1 = 1.2$ and $\tilde{k}_2 = 0.5$ obtained by trial and error. Hereafter, assuming that the target contact force $\lambda_d = 1N$, the results of the deflection distance at the arm tip of the model, the rotation angle of the motor, and the time response of the contact force exerted on the object by the arm tip are shows in Figure 3.

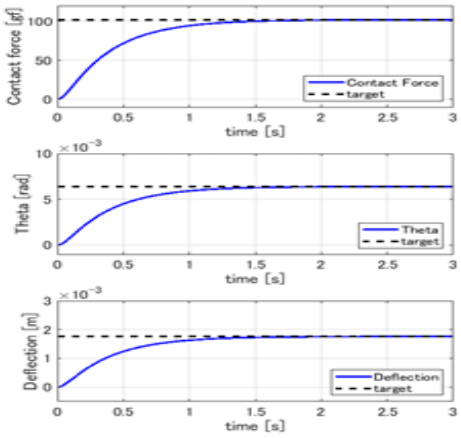

Fig. 3. Block diagram of 1 link flexible Bernoulli-Euler Arm

As can be seen from Figure 3, since the contact force at the tip of the arm converges to the target value around 2.3 seconds, the moment-PD control is also effective for this model. This result affirms the choice of the initial values of feedback gain used in this research as $\tilde{k}_1 = 1.2$ and $\tilde{k}_2 = 0.5$.

5. Multilayer Neural Networks

Neurons are the basic elements of the brain, and the artificial model of this is called a unit. The basic building block of the neural network is this unit and it plays an important role in explaining dynamics such as learning and adaptation. Figure 4 shows the model image of the unit. Its input/output relation is as follows.

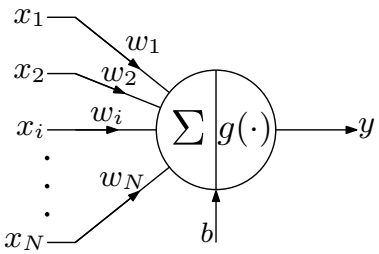

Fig. 4. Unit of neural network.

Units are regarded as multi-input single-output(MISO) elements based on the biological properties of neurons. A weight w_k representing the strength of the synapse connection is added to the connection through which the input signal x_k to the unit passes, and when the weighted sum $\sum w_k x_k$ exceeds the threshold θ, the neuron fires (outputs is 1). When it does not exceed it will not fire (output is 0). Weight with positive value excites the neuron to promote firing, whereas weight with negative value acts as inhibition to suppress firing. At this time, the output y of the neuron can be written as follows.

$$y = 1(h) \text{ where } h = \sum_{k=0}^{N} w_k x_k + \theta \tag{16}$$

Here, $1(h)$ represents a Heaviside function, 1 is returned when $h \geq 0$, and 0 is returned when $h < 0$. In the same respect, the neuron model can be generalized as follows. First, consider input of $x_0 = 1$ and replace the threshold θ with the weight w_0. Next, if the Heaviside function is an activation function $g(h)$, the expression 16 can be written as follows.

$$y = g(h), h = \sum_{k=1} w_k x_k \tag{17}$$

As a typical activation function, a sigmoid function is known as a smooth approximation of the Heaviside function. The value range is $0 < g(h) < 1$.

$$g(h) = \sigma(\beta h) = \frac{1}{1 + e^{-\beta h}} \tag{18}$$

Here, $\beta > 0$ is an inverse temperature parameter, and the larger the value of β is, the closer the function approximates the Heaviside function. In general, $\beta = 1$ is commonly used. The derivative of equation 18 can be written as $\beta g(h)(1 - g(h))$. The key strength of the sigmoid function is that whereas the Heaviside function is not differentiable at $h = 0$, the sigmoid function is differentiable for all values of h.

Furthermore, since the differential coefficient can be calculated only from the function value, it is very convenient in view of analysis and calculation and it is known that nonlinear continuous function can be realized, which

is easy to use in general. In this research we used sigmoid function.

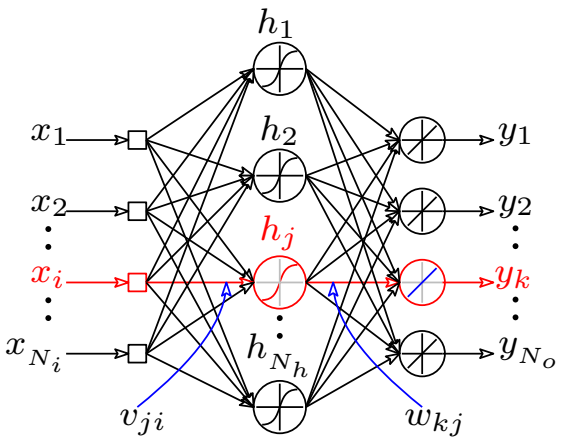

Fig. 5. Multilayer perceptron.

Next, perceptron as the one described above are configured to form a three-layer structure shown in Figure 5. Assuming that the exogenous input $x = [x_1, \cdots, x_k, \cdots, x_K]^T$, and the output from the hidden unit j is h_j, the relational expression is

$$h_j = g_1(z_j) \tag{19}$$

$$z_j = \sum_{k=1}^{N_h} v_{ji}x_k + \theta_j \tag{20}$$

$$= \sum_{k=0}^{N_h} v_{ji}x_k \quad \text{where } v_{j0} = \theta_j, x_0 = 1 \tag{21}$$

Where:

v_{ji}	:	interconnection weight of input layer unit i and hidden layer unit j.
z_j	:	Net input amount of hidden layer unit.
θ_j	:	Threshold value of hidden layer unit.
$g_1(\cdot)$	:	Activation function of the hidden neurons

This is basically unchanged from the formula describing the neuron model. Also, if the output of the unit j in the hidden layer is h_j, which form the input to the output layer. The summed input to the k_{th} neuron is expressed as

$$z_k = \sum_{k=1}^{N_o} w_{kj}z_j + \theta_k \tag{22}$$

$$= \sum_{k=0}^{N_o} w_{kj}z_j \quad \text{where } w_{k0} = \theta_k, x_0 = 1 \tag{23}$$

This summed input is squashed by the activation function of the output layer to yield the output of the network given by

$$y_k = g_2(z_k)$$

where $g_2(\cdot)$ is the activation function of the output layer of the network.

Generally, in learning of a multilayer perceptron, the weights and threshold values in the neural network are changed so that the output for the input x approximates the target value d corresponding to the input. In this article, we used error backpropagation(BP) as the basic learning method. BP method is a method applying the steepest descent method with multilayer perceptron weight learning as numerical optimization problem. Therefore, a gradient is necessary, and the activation function of the neural network must be differentiable. The specific calculation order will be explained using the multilayer perceptron shown in Figure 5.

When a certain pattern p is input, let the target output signal in output unit i be d_i. When E is the error function for all patterns, the error function for the specific pattern E_p is expressed by the following equation using the sum of squared error.

$$E_p = \frac{1}{2}\sum_{k=1}^{N_o}(d_k - y_k)^2 \tag{24}$$

$$E = \sum_{p=1}^{P} E_p \tag{25}$$

We find small change in interconnection weight Δw in the direction in which this error function E_p decreases. Consider the gradient of the error function E_p

$$\frac{\partial E_p}{\partial y_k} = y_k - d_k \tag{26}$$

$$\frac{\partial E_p}{\partial h_k} = \frac{\partial E_p}{\partial y_k}\frac{\partial y_k}{\partial h_k} \tag{27}$$

$$= (y_k - d_k)g_2'(z_i) \tag{28}$$

$$= \delta_k \tag{29}$$

Here, δ_k represents the error in the output unit k of the pattern. The slope for the weight w_{kj} from the hidden layer to the output layer is as follows.

$$\frac{\partial E_p}{\partial z_j} = \frac{\partial E_p}{\partial h_k}\frac{\partial h_k}{\partial w_{kj}} = \delta_k z_j \tag{30}$$

Similarly,

$$\frac{\partial E_p}{\partial z_j} = \sum_{k=1}^{I}\frac{\partial E_p}{\partial z_k}\frac{\partial z_k}{\partial z_j} \tag{31}$$

$$= \sum_{k=1}^{N_h}\delta_k w_{kj} \tag{32}$$

$$\frac{\partial E_p}{\partial z_j} = \frac{\partial E_p}{\partial z_j}\frac{\partial z_j}{\partial z_j} \tag{33}$$

$$= g_1'(z_j)\sum_{k=1}^{I}\delta_k w_{kj} \tag{34}$$

Here, δ_j represents the error in the hidden unit j of the pattern p. The gradient with respect to the weight w_{kj}

from the input layer to the hidden layer is as follows.

$$\frac{\partial E_p}{\partial w_{jk}} = \frac{\partial E_p}{\partial z_j} \quad (35)$$

$$\frac{\partial z_j}{\partial w_{jk}} = \delta_j x_k \quad (36)$$

By applying these to the steepest descent method, the updating expressions of the weights w_{ij} and w_{jk} are

$$\Delta w_{ij} = -\eta_t \frac{\partial E_p}{\partial w_{ij}} = -\eta_t \delta_j z_j \quad (37)$$

$$\Delta w_{jk} = -\eta_t \frac{\partial E_p}{\partial w_{jk}} = -\eta_t \delta_k x_k \quad (38)$$

Where η_t is a positive constant and is called a learning rate.

6. Design of control system using neural network

For the purpose of shortening the convergence time from that observed using the feedback gain determined by trial and error as an initial value, a control system that tunes gain adaptively was designed. In this research, advantage of the numerous strengths on the artificial neural network was taken, primarily, that it can be used to approximate any nonlinear function. Block diagram of the arm with the neuro-controller is as shown in Figure 6 below.

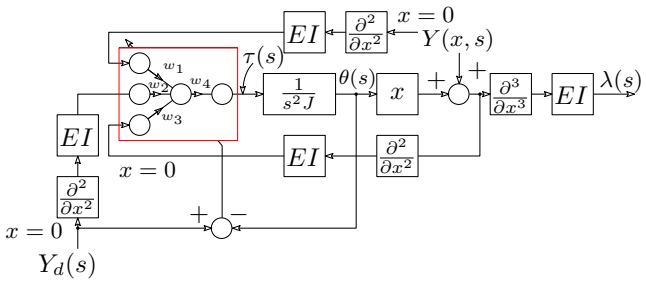

Fig. 6. One link Flexible arm using a neural network

The neural network to be used is a three-layer hierarchical type having three units in the input layer, one hidden neuron and one neuron in the output layer corresponding to the driving torque as shown in Figure 7.

Also, the input is the target value and the present value of the bending moment of the base of the arm and the present value of deflection, and the sigmoid function is used for the output function of the hidden layer and the output layer. As the learning method, the error backpropagation method is used and the steepest descent method is used for the updating the coupling coefficient between the units in an online version. Figure 8 shows the block diagram of the control controller.

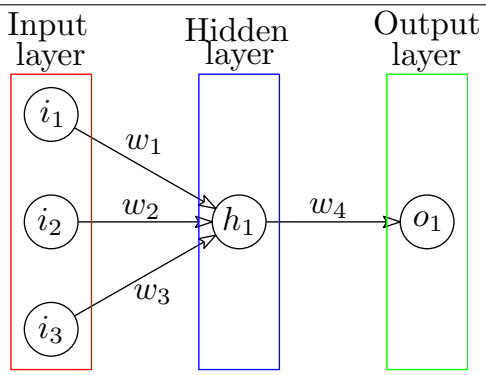

Fig. 7. Three-layered neural network

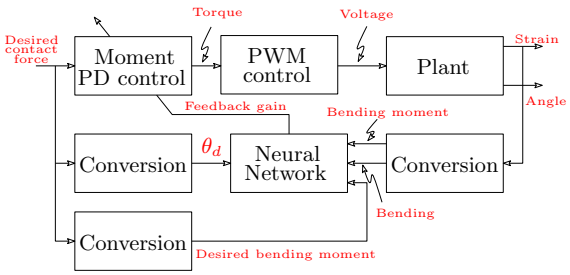

Fig. 8. Neural network control system

7. Design experimental setup

Figure 9 shows the overall view of the 1-link flexible arm to be controlled. The material of the arm is aluminum (3003), the sectional shape is rectangular, the distance from the root is l = 0.275 m. A brushed DC motor equipped with a metal gearbox with a gear ratio of 50:1 and equipped with an integrated type orthogonal encoder is attached at the root of the arm to provide the driving torque. The resolution of the encoder is 64 counts per rotation of the motor shaft (corresponding to 3200 counts per shaft rotation of gearbox output). In order to measure the elastic deformation in the direction of rotation of the motor, the strain gauge is attached in a 2-gauge method. The parameters used in this study are shown in Table 2.

Table 2. System parameters

Parameter	Symbol	Dimension
Height	h	0.02 m
Breadth	b	0.00447 m
Length	l	1.05 m
Density	ρ_a	7874 kgm-3
Mass moment of inertia	I_ρ	2.79×10^{-4}kgm
Youngs modulus	E	2.06×10^{-11} Pa

Figure 10 shows a conceptual diagram of the system configuration used in this study.

A control system consist of Laboratory Virtual Instrumentation Engineering Workbench(LabVIEW) manufactured by National Instrument installed on a PC running Microsoft Windows 7 operating system. LabVIEW

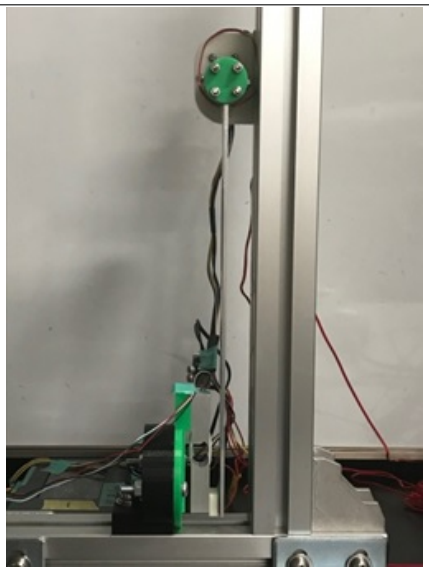

Fig. 9. Link flexible arm.

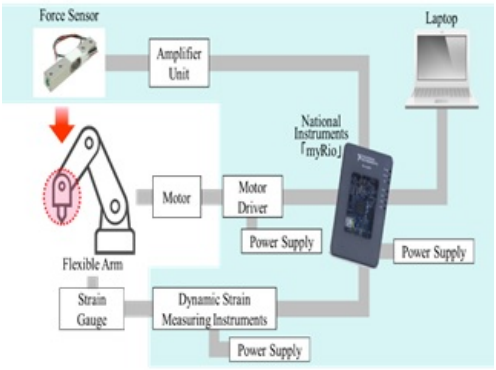

Fig. 10. Experiment system.

makes it easier to express complex logic on a diagram by using a graphical programming approach to visualize every aspect of an application. Designing a distributed test, measurement and control system can be performed efficiently. For the controller, NI myRIO (manufactured by National Instrument Corporation) embedded hardware device for education was used. For the force sensor, 2 kg of load cell single point (beam type) manufactured by Sensor and Control was employed together with amplifying circuit. The arrangement was such that the object made a vertical contact with the tip of the arm. Experiments were conducted with a dynamic strain measuring instrument (DPM 713 B made by Kyowa Denki) with the low pass filter set at 100 Hz, the measurement range set at 500$\mu\Omega$, and the configuration value set at 500$\mu\varepsilon$/2V. Initial values of interconnection weights of the neural network were set as; w_1 = 1.7, w_2 = 1.2, w_3 = -1.0, and a random number is used for w_4. Learning was performed at every sampling time of 2milliseconds, and the experiments were performed with the target value contact force λ_d set to $1N$. Figure 11 shows temporal changes of the feedback gains (k_1, k_2), Figure 12 shows the distortion of the base of the arm, deflection of the arm tip, the rotation angle of the motor, and the contact force.

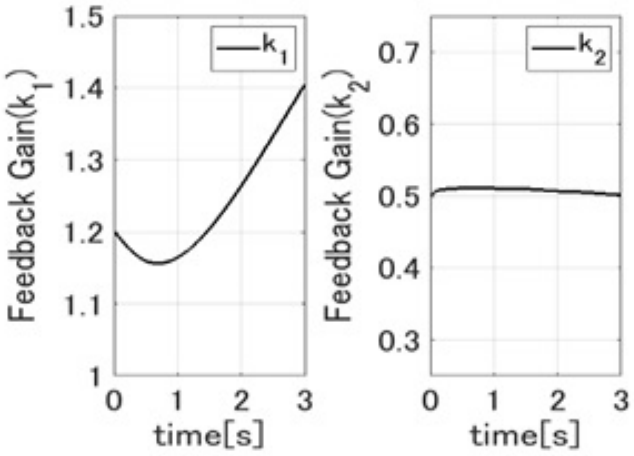

Fig. 11. Time response of feedback gain.

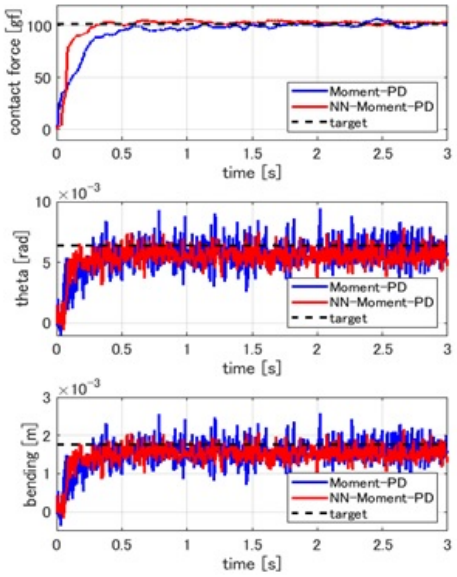

Fig. 12. Transient responses; blue line: Moment-PD; red line: NN-Moment-PD

From Figure 12, the blue line show the performance when moment-PD is fed back via fixed gains whereas the red line show the performance of the NN-moment-PD control. To make the comparison easier, median filter is employed for smoothing. From this result, it can be seen that the time taken by the neuro-controller to convergence is 0.8 seconds shorter than the time taken by the fixed moment-PD control. In addition, it can be seen that the control system to which the NN-moment-PD control is applied stably converges to the target value.

From the above, it is confirmed that by designing the control system using the neural network as the gain adjustment method, it is possible to adaptively change the value of the gain and to quickly converge to the target value of the contact force. These results confirms the effectiveness of this study in terms of enhancing the speed of convergence.

8. Conclusion

In this research focusing on force control of a constrained one link flexible arm, a one-link flexible arm was developed and a simple boundary feedback controller (Moment-PD control) consisting of bending moment at the base of the arm proposed. A control system using a neural network was designed as a gain adjustment method. Comparison and examination of control performance by numerical simulation and actual machine were performed. First, regarding moment-PD control, general solutions were derived from the model by formulating a theoretical expression from the model, and the value of the feedback gain to be used in this research were determined using numerical simulation by trial and error. Thereafter, using those values on an experimental laboratory flexible arm, it was confirmed that the target contact force converged about 1.15 seconds. From this, the effectiveness of the moment-PD control proposed was confirmed. Next, control system applying the neural network as the gain adjustment method was designed to enhance the performance obtained with fixed gains. Weight updating scheme adopted in this work was the online error backpropagation in which learning is repeated for each sampling for learning timing of the neural network. Analyzing the experimental results, it was confirmed that by using the control system employing the neural network, the value of the feedback gain is adaptively changed, and the target contact force converges around 0.35 seconds. Relative to the control law having fixed gains, Neuro-controller succeeded in enhancing the time it takes for convergence to the target value by 0.8 seconds confirming the effectiveness of the controller proposed in the actual machine. The learning rate employed in this is fixed, as a future prospect, a proposal to enhance learning by the introduction of adaptive learning rate is hereby made. This way, the learning rate will be set to higher values when the error is high and to smaller values when the error decrease to lower values. The learning will gradually decay as the network learns and approach the target values. Also, plans are in place to compare the performance of this scheme with other popular machine learning schemes like support vector machine and deep learning.

9. Acknowledgement

This work is partially supported by Grants-in-aid for Promotion of Regional Industry-University-Government Collaboration from Cabinet Office, Japan.

References

[1] Sasaki, M., Nagaya, K., Endo, T., Matsushita, K., and Ito, S., "End Point Force Control of a Flexible Timoshenko Arm," *Journal of Computer and Communications*, vol. 3, pp. 106–112, 2015.

[2] Minoru Sasaki, Takeshi Ueda, Yoshihiro Inoue, and Wayne J. Book, "Passivity-Based Control of Rotational and Translational Timoshenko Arms," *Advances in Acoustics and Vibration*, vol. 6, 2012.

[3] Minoru Sasaki, Takashi Kuwabara, Waweru Njeri, Kojiro Matsushita, and Harrison Ngetha, "Comparison of Simulation of Contact Force Control of Flexible Manipulator Using Bernoulli-Euler theory and Timoshenko theory," *Journal of Applied Sciences, Engineering and Technology for Development*, vol. 3, pp. 11–23, July 2018.

[4] T. Endo, M. Sasaki, F. Matsuno, and Y. Jia, "Contact-force control of a flexible timoshenko arm in rigid/soft environment," *IEEE Transactions on Automatic Control*, vol. 62, pp. 2546–2553, May 2017.

[5] Yoshifumi Morita, Fumitoshi Matsuno, Yukihiro Kobayashi, Motohisa Ikeda, Hiroyuki Ukai, and Hisashi Kando, "Dynamics based force control of a one-link flexible arm considering bending and torsional deformation," *Transactions of the Society of Instrument and Control Engineers*, vol. 38, 01 2002.

[6] L.-Y. Liu and H.-C. Lin, "Tip-contact force control of a single-link flexible arm using feedback and parallel compensation approach," *Robotica*, vol. 31, no. 5, p. 825835, 2013.

[7] T. Endo, F. Matsuno, and H. Kawasaki, "Force control and exponential stabilisation of one-link flexible arm," *International Journal of Control*, vol. 87, no. 9, pp. 1794–1807, 2014.

[8] T. Endo, M. Sasaki, and F. Matsuno, "Contact-force control of a flexible timoshenko arm," *IEEE Transactions on Automatic Control*, vol. 62, pp. 1004–1009, Feb 2017.

[9] M. Sasaki, R. Nakamura, W. Njeri, K. Matsushita, and S. Ito, "Adaptive gain tuning feedback control of a flexible manipulator," in *2017 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017)*, Atlantis Press, 2017.

[10] K. Takahashi and I. Yamada, "Neural-network based learning control of flexible mechanism with application to a single-link flexible arm," *Journal of Dynamic Systems Measurement and Control*, vol. 116, pp. 792–795, 12 1994.

[11] Z.-H. Luo, "Direct strain feedback control of flexible robot arms: new theoretical and experimental results," *IEEE Transactions on Automatic Control*, vol. 38, pp. 1610–1622, Nov 1993.

[12] M. Kawafuki, M. Sasaki, and F. Fujisawa, "Feedback-error-learning neural network for trajectory control of a flexible micro-actuator," in *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, p. 73, June 1997.

[13] M. Kawafuku and M. Sasaki, "Neural network for trajectory control of a flexible micro actuator," *Transactions of the Japan Society of Mechanical Engineers Series C*, vol. 64, no. 626, pp. 3813–3819, 1998.

[14] M. Kawafuku, M. Sasaki, and K. Takahashi, "Comparison of feedback controllers for feedback-error-learning neural network control system with application to a flexible micro-actuator," *JSME International Journal Series C*, vol. 43, no. 1, pp. 149–156, 2000.

[15] Waweru Njeri, Minoru Sasaki, and Kojiro Matsushita, "Gain tuning for high-speed vibration control of a multilink flexible manipulator using artificial neural network," *ASME Journal of Vibration and Acoustics*, vol. 141, pp. 1–11, Aug. 2019.

Appendix A
Derivation of the equation of motion

Considering a uniform single link flexible arm which is rotated by a control motor in the horizontal plane as shown in Figure 13. Let $w(x,t)$ denote the arm deflection at position $0 < x < l$ along the link and at time t and let θ denote the rotation angle of the motor shaft. Denoting by ρ and EI the mass density per unit length and the bending rigidity of the flexible arm respectively. Denote by J the moment of inertia of the total load attached to the motor shaft of radius l_0.

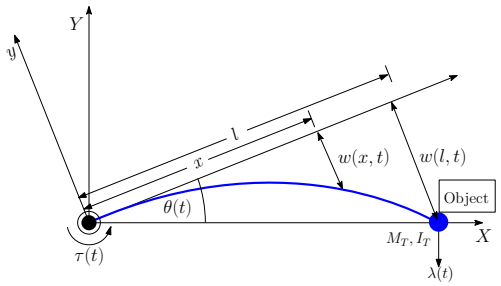

Fig. 13. Single link flexible arm

At the tip, we introduce a constraint

$$\phi(w(l,t),\theta(t)) = w(l,t) - l\theta(t) = 0 \tag{39}$$

Kinetic and potential energies of the manipulator are expressed as

$$T = \frac{1}{2}J\dot{\theta}(t)^2 + \frac{1}{2}\int_0^l \rho(x\dot{\theta}(t) - \dot{w}(x,t))^2 dx \tag{40}$$

$$U = \frac{1}{2}\int_0^l EIw''(x,t)^2 dx \tag{41}$$

and virtual work

$$\delta W = \tau\delta\theta(t)dt \tag{42}$$

Using Hamilton's principle

$$\int_{t_1}^{t_2}(\delta T - \delta U + \delta W + \lambda\delta\phi)dt = 0 \tag{43}$$

The first term of equation 43 can be expressed as

$$\int_{t_1}^{t_2}\delta T dt = \int_{t_1}^{t_2} J\dot{\theta}\delta\dot{\theta}dt + \int_{t_1}^{t_2}\int_0^l \rho(x\dot{\theta}(t) - \dot{w}(x,t))(x\delta\dot{w}(x,t) - \delta\dot{w}(x,t))dxdt \tag{44}$$

integrating the first term by parts,

$$\int_{t_1}^{t_2}\delta T dt = -\int_{t_1}^{t_2} J\ddot{\theta}\delta\theta dt - \rho\int_0^l\int_{t_1}^{t_2}(x\ddot{\theta}(t) - \ddot{w}(x,t))(x\delta\theta(t) - \delta w(x,t))dtdx \tag{45}$$

$$\tag{46}$$

For the second term

$$-\int_{t_1}^{t_2}\delta U dt = -\int_{t_1}^{t_2}\int_0^l EIw''(x,t)\delta w''(x,t)dxdy \tag{47}$$

integrating by parts yields

$$-\int_{t_1}^{t_2}\delta U dt = -\int_{t_1}^{t_2} EIw''(x,t)\delta w'(x,t)\Big|_0^l dt + \int_{t_1}^{t_2} EIw'''(x,t)\delta w(x,t)\Big|_0^l dt - \int_{t_1}^{t_2}\int_0^l EIw''''(x,t)\delta w(x,t)dxdt$$

For the variation of the virtual work

$$\int_{t_1}^{t_2}\delta W = \int_{t_1}^{t_2}\tau\delta\theta \tag{48}$$

Finally, the variation of the constraints

$$\int_{t_1}^{t_2} \lambda\delta\phi(w(l,t),\theta(t))dt \quad = \quad \int_{t_1}^{t_2} \lambda\delta w(l,t) - \lambda l\delta\theta(t)dt \tag{49}$$

Adding all the energy terms and gathering the like terms together:

- $\delta\theta(t)$:

$$-\int_{t_1}^{t_2} \left[J\ddot{\theta}(t)dt + \tau - \rho\int_0^l x(x\ddot{\theta}(t) - \ddot{w}(x,t))dx - \lambda l\right] dt = 0 \tag{50}$$

- $\delta w(x,t)$:

$$\rho\int_{t_1}^{t_2}\int_0^l \left[(x\ddot{\theta}(t) - \ddot{w}(x,t)) - EIw''''(x,t)\right] dxdt = 0 \tag{51}$$

- $\delta w(l,t)$:

$$\int_{t_1}^{t_2} \left[EIw'''(l,t) + \lambda\right] dt = 0 \tag{52}$$

- $\delta w'(l,t)$:

$$-\int_{t_1}^{t_2} EIw''(l,t)dt = 0 \tag{53}$$

For equations 50,51, 52 and 53, the we have the following integro-differential equations of motion and boundary conditions.

$$J\ddot{\theta}(t)dt + \rho\int_0^l x(x\ddot{\theta}(t) - \ddot{w}(x,t))dx \quad = \quad \tau(t) - \lambda(t)l \tag{54}$$

$$\rho(x\ddot{\theta}(t) - \ddot{w}(x,t)) - EIw''''(x,t) \quad = \quad 0 \tag{55}$$

$$EIw'''(l,t) \quad = \quad -\lambda(t) \tag{56}$$

$$EIw''(l,t) \quad = \quad 0 \tag{57}$$

$$w(0,t) = w'(0,t) \quad = \quad 0 \tag{58}$$

From equation 55

$$x\rho(x\ddot{\theta}(t) - \ddot{w}(x,t)) \quad = \quad xEIw''''(x,t)$$

integrating by parts, equation 54 can be rewritten as

$$J\ddot{\theta}(t) + EIw''(0,t) = \tau(t) \tag{59}$$

Appendix B

How the second term increases system damping

In this section, the importance of the second term of the proposed control law in the mitigation if link vibration by increasing system damping will be presented. Introducing the inner product in a Hilbert space $\mathcal{H}$ derived as

$$\langle u, v\rangle = \rho\int_0^l \bar{u}(x)v(x)dx \quad \forall u, v \in \mathcal{H}$$

where $\bar{u}(x)$ denote the complex conjugate of the variable $u(x) \in \mathcal{H}$. Further, introducing an operator A as

$$\begin{cases} D(A) = \{u(x)|u(x)'''' \in \mathcal{H}, u(0) = u'(0) = u''(0) = u'''(0) = 0\} \\ Au(x) = \frac{EI}{\rho}u(x)'''', \forall u(x) \in D(A) \end{cases}$$

using the above operator, the equation of motion in 55 can be rewritten as an abstract differential equation on $\mathcal{H}$ as

$$\ddot{w}(t) + D\dot{w}(t) + Aw(t) \quad = \quad x\theta(t)$$

where the term $D\dot{w}(t)$ is introduced to describe natural damping of the link. Further introducing another operator $\Pi_\star$ defined as

$$\Pi_\star u = xu''(0) \quad x \in (0,l) \quad \forall u \in D(A)$$

From equation 59

$$\begin{aligned} \tau(t) &= J\ddot{\theta}(t) + EIw''(0,t) \text{ from which} \\ \ddot{\theta}(t) &= \frac{\tau}{J} - \frac{EI}{J}w''(0,t) \end{aligned}$$

substituting this into the equation of motion yields

$$\rho(\ddot{w}(x,t)) + DEI\dot{w}''''(x,t) + EIw''''(x,t) + x\frac{EI}{J}w''(t,0) = x\frac{\tau(t)}{J}$$

Letting the motor torque $\tau(t) = -k_2\dot{y}''(t,0)$, the closed loop system equation can be expresses in Hilbert space by utilizing the operators defined before as

$$\ddot{y}(t) + \underbrace{(D + \frac{k_2}{J}\Pi_\star)}_{\text{Damping}}\dot{y}(t) + \underbrace{(A + \frac{EI}{J}\Pi_\star)}_{\text{Stiffness}}y(t) = 0$$

From this expression, it can be seen that the third term increases the damping term by a value equal to $\frac{k_2}{J}\Pi_\star$ which act to suppress vibration of the arm.