Tele-operated Control of Inverted Pendulum System for Intelligent Control Education

Jae Kook Ahn
Intelligent Systems and Emotional Engineering Laboratory
Department of Mechatronics Engineering
Chungnam National University,
Daejeon, Korea 305-764

and

Seul Jung
Intelligent Systems and Emotional Engineering Laboratory
Department of Mechatronics Engineering
Chungnam National University,
Daejeon, Korea 305-764

ABSTRACT

This paper presents tele-operated control of an inverted pendulum system for intelligent control education. Position commands for the pendulum to follow are given by a joystick through network. The pendulum is controlled by two methods; a PID control method and a neural network control method. Results are compared by experimental studies.

Keywords: Inverted pendulum, network based control, intelligent control education

1. INTRODUCTION

Recently, control education has become more important in control engineering field since the state-of-the art technologies demand more sophisticated control algorithms. It is true that proportional-derivative-integral (PID) controllers have been dominantly used in manufacturing industries. However, as technology develops rapidly, advanced control methods are demanded for regulating sophisticated and complicated control systems to satisfy high performance.

To educate advanced control methods to students effectively, the control theory and experimental demonstration should be provided. Providing experimental studies will compensate for the lack of simulation works for advanced control applications since real world has a lot of uncertain problems.

One of typical dynamical system for control education is the inverted pendulum system in Fig. 1. The inverted pendulum system has been dominantly used as a prototype system in the class. One particular reason of using inverted pendulum is to illustrate the concept and challenge of control. Also the balancing concept can be expanded to other application areas such as humanoid robot, two wheel balancing mobile robots, and other balancing systems.

The inverted pendulum system is controlled and tested by many control algorithms since its characteristics are quite challenging and nonlinear. One single input $u$ has to control two variables, angle $\theta$ and position $x$ to satisfy desired position tracking while balancing. Thus, suitable combinations between angle control and position control provide successful performance.

Recently, interests in intelligent control have been enormously increased as systems become more complex. A feasible candidate of intelligent tools is neural network whose structure is similar to the summation of a nonlinear adaptive filter. One advantage of using neural network is learning capability that system learns environment gradually by reinforcement.

Many articles on inverted pendulum system control have been presented in the literature[1-8]. Fuzzy logic is used as a controller to control the system[1,2]. Rotational inverted pendulum system have been presented[4]. Inverted pendulum system with two or three links have been presented[7, 8].

This paper aims two education goals, one is intelligent control and another is tele-operation control. Neural network is used as an intelligent tool for controlling the inverted pendulum system. Neural network learning algorithm is introduced and embedded on the hardware to achieve real time control. A joystick is used to command the desired position for the pendulum to follow. The pendulum is remotely controlled by the joystick to learn a time-delayed network-based control application. The pendulum follows the position command by the joystick while maintains balancing.
The inverted pendulum system is setup for experimental studies. Experimental works are demonstrated and results are compared with those of PID controllers.

![Fig. 1 Teleoperation of Inverted pendulum system](image1)

2. PID CONTROL

The PID control method is dominantly used. The angle error is defined as

\[ e_\theta = \theta_d - \theta \]  

where \( \theta_d \) is the desired angle value which is 0 and \( \theta \) is the actual angle. The error passes through the PID controller.

\[ u_\theta = k_{p \theta} e_\theta + k_{i \theta} \int e_\theta dt + k_{d \theta} \dot{e}_\theta \]  

where \( k_{p \theta}, k_{i \theta}, k_{d \theta} \) are controller gains.

The position error is defined by

\[ e_x = x_d - x \]  

where \( x_d \) is the desired position value and \( x \) is the actual position. The detailed PID controller output becomes

\[ u_x = k_{p x} e_x + k_{i x} \int e_x dt + k_{d x} \dot{e}_x \]  

where \( k_{p x}, k_{i x}, k_{d x} \) are controller gains.

The total control input is

\[ u = u_\theta + u_x \]  

Fig. 2 shows the PID control block diagram for the inverted pendulum system.

![Fig. 2 PID control structure](image2)

3. RBF NEURAL NETWORK

Neural network has a parallel processing structure which requires massive calculations. The radial basis function (RBF) network has a different structure from the multi-layered perceptron (MLP) network. The RBF network uses the Gaussian function as a nonlinear function, and there are no weights between the input layer and the hidden layer and the output layer is linear. However, the number of update weights is same as the one layered MLP network. The RBF structure is shown in Fig.3.

The Gaussian function used in the hidden layer is

\[ \psi_j(e) = \exp\left(-\frac{\|e - \mu_j\|^2}{2\sigma_j^2}\right) \]  

where \( e \) is the input vector \( e = [e_1 e_2 ... e_n]^T \) which is the error vector, \( \mu_j \) is the center value vector of the \( j \)th hidden unit, and \( \sigma_j \) is the width of the \( j \)th hidden unit. The forward \( k \)th output in the output layer can be calculated as a sum of outputs from the hidden layer.

\[ \phi_k = \sum_{j=1}^{M} \psi_j w_{jk} + b_k \]  

where \( \psi_j \) is \( j \)th output of the hidden layer in (6) and \( w_{jk} \) is the weight between the \( j \)th hidden unit and \( k \)th output, and \( b_k \) is the bias weight.

In the RBF network, updated weights are \( \mu_j, \sigma_j, w_{jk}, \) and \( b_k \).
4. NEURAL NETWORK CONTROL METHOD

The PID controller works for balancing with ease, but has the difficulty of tracking desired position at the same time. The PID control suffers from outer disturbance as well. Thus, adding a neural network improves the robustness to outer disturbance and compensates for uncertainties to improve position tracking performance.

Fig. 4 shows the one of neural control method called the reference compensation technology (RCT) that compensates at the input trajectory level[3,6,9]. The reference input \( r \) is modified by the neural network output \( \Phi \) to generate modified the error \( \varepsilon \). The error \( \varepsilon \) is different from the output error \( e = r - y \).

\[
\varepsilon = r - y + \phi_N \quad (8)
\]

where \( r \) is the desired input, and \( \phi_N \) is from the neural network output. The detailed PID controller output becomes

\[
u_{th} = k_{\phi t}(e_t + \phi_t) + k_{e}(e + \phi) + k_{\int}(\int e dt + \phi) \quad (9)
\]
\[
u_{in} = k_{\phi}(e + \phi_N) + k_{e}(e + \phi) + k_{\int}(\int e dt + \phi) \quad (10)
\]

where \( \phi_i \) is the \( i \)th neural network output.

The total control input is

\[
u = u_{th} + u_{in} = v + \Phi_{\phi} + \Phi_e \quad (11)
\]

where,

\[
v = u_{th} + u_{in}, \quad u_{th} = k_{\phi t}(e_t + \phi_t) + k_{e}(e + \phi) + k_{\int}(\int e dt + \phi), \quad \Phi_{\phi} = k_{\phi}(e + \phi_N) + k_{e}(e + \phi) + k_{\int}(\int e dt + \phi). \quad \Phi_e = k_{\phi t}(e_t + \phi_t) + k_{e}(e + \phi) + k_{\int}(\int e dt + \phi) .
\]

The training signal \( v \) is

\[
v = u - (\Phi_{\phi} + \Phi_e) \quad (12)
\]

To minimize the error \( v \), the objective function is defined as

\[
E = \frac{1}{2} v^2 \quad (13)
\]

The back-propagation algorithm requires differentiating (10) with respect to the weight vector \( w \) as

\[
\frac{\partial E}{\partial w} = \frac{\partial E}{\partial v} \frac{\partial v}{\partial w} = v - v_{th} = -v_{th} \frac{\partial \Phi_{\phi}}{\partial w} - \frac{\partial \Phi_e}{\partial w} \quad (14)
\]

Then weights are updated as

\[
w(t+1) = w(t) - \eta \frac{\partial E}{\partial w} \quad (15)
\]

where \( \eta \) is the learning rate.

The detailed neural network control structure for the inverted pendulum system is shown in Fig. 5.
5. SIMULATION

Simulation studies on neural network control for the inverted pendulum system are performed. The position command is to move the cart from 0m to 0.5 m. Fig. 6 shows the simulation results. The balancing angle and the cart position are well regulated.

Fig. 6. Simulation results

6. EXPERIMENTS

Experimental setup

The inverted pendulum system is built for the experiment as shown in Fig. 7. The system setup consists of inverted pendulum actuated by a dc motor, a computer, a joystick, and control hardware. The pendulum moves on the guided rail.

Fig. 7. Experimental setup

Control hardware block diagram is shown in Fig. 8. Main processors are a DSP and an FPGA. The joystick command is received at the computer and the computer communicates with the DSP through a serial port. The DSP calculates neural network learning algorithm along with PID control algorithm in online fashion and the FPGA processor interfaces with the actuators. The FPGA processor collects two encoder data, counts them, and generates a pulse-width-modulation (PWM) signal to the motor driver.

Fig. 8. Control hardware block diagram

Experimental results

Position tracking control experiments of the inverted pendulum system are conducted. A user commands the desired position by moving the joystick. Then the pendulum follows the desired position while balancing. Figure 9 shows the experimental results. Fig. 9 (a) shows the desired trajectory commanded by the joystick. Since
the command from the joystick is rough, it is filtered to be a smooth trajectory command.

Command starts after around 21 seconds. Fig. 9 (b) and (c) show the actual tracking results. In Fig. 9 (b), the inverted pendulum initially maintains balancing without the joystick command, and then it follows the joystick commands around 21 seconds. Fig. 9 (c) shows the enlarged figure of the joystick commanded part. Although there are small tracking errors, the pendulum successfully follows the desired trajectory while balancing.

7. CONCLUSIONS

The inverted pendulum system is built and controlled by the neural controller. Cart position is controlled to follow the trajectory generated by the joystick remotely while the pendulum angle is maintained balancing. The neural network control method along with the PID control method is employed for advanced intelligent control education. The neural controller performs well for tele-operated environment. However, a time delay issue in tele-operated control has not been considered although it is a most important problem in network based control. Therefore a time delay effect on the control performance of tele-operation tasks will be further investigated in the future.

ACKNOWLEDGEMENT

This research was financially supported by Korea Research Foundation (KRF) through the basic research program (R01-2008-000-10992-0).
REFERENCES


