

Neural Network Control Technique for Automatic Four Wheel Steered Highway Snowplow Robotic Vehicles

Seul Jung*, Ty Lasky**, and T. C. Hsia***

* Department of Mechatronics Engineering, Chungnam National University, Daejeon, Korea
(Tel : +82-42-821-6876; E-mail: jungs@cnu.ac.kr)

**Advanced Highway Maintenance and Construction Technology Center, University of California, Davis, USA
(Tel : +1-530-000-; E-mail: talasky@ucdavis.edu)

***Department of Electrical & Computer Engineering, University of California, Davis, USA
(Tel : +1-530-000-7440; E-mail: hsia@ece.ucdavis.edu)

Abstract: In this paper, a neural network technique for automatic steering control of a four wheel drive autonomous highway snowplow vehicle is presented. Controllers are designed by the LQR method based on the vehicle model. Then, neural network is used as an auxiliary controller to minimize lateral tracking error under the presence of load. Simulation studies of LQR control and neural network control are conducted for the vehicle model under a virtual snowplowing situation. Tracking performances are also compared for two and four wheeled steering vehicles.

Keywords: Neural network, LQR lateral control, highway snowplowing vehicle

1. INTRODUCTION

Recently, researches on automation of highway maintenance and construction vehicles have been quite actively conducted. Fusing robotic technology to utility vehicles yields a new concept of robotic vehicles such as a crack sealing vehicle, trash pick up vehicle, and an autonomous corn-handling vehicle[1,2]. Several vehicles have been already commercialized.

The advanced highway maintenance and construction technology (AHMCT) center at UC Davis has been known to be one of the leading research groups of conducting researches on this subject. One of their current projects is to automate and develop the snowplow vehicles. Snowplow vehicles are heavily used in mountain areas of having heavy snows in the winter. One typical snowplow vehicle is shown in Fig.1. To fully automate this kind of vehicle, automatic steering control should be developed along with GPS localization, GIS analysis, and sensor implementation.

The ultimate goal is to develop the snowplowing vehicle that can autonomously follow the guardrail of the highway without collision. The vehicle has to overcome the situation where the guardrail is hidden under the pile of snow. This requires accurate lateral control of the vehicle by maintaining a desired constant distance from the guardrail.

There have been many researches about lateral control of the vehicles. H_{∞} control and yaw rate feedback control methods for lateral positioning system have been proposed [3-5]. Experimental studies of robust lateral control of highway vehicles have also been conducted [6,7].

In our previous researches, neural network control technique has been applied for two wheel steering control of the bicycle model of the vehicle. Neural network with PD controllers has performed well to compensate for uncertainties. However, under the condition of heavy piles of snow, the front wheel steering may not be efficient. The rear steering control is also required for the better lateral control by helping the vehicle to push against snow. Many researches have been also considered the four wheeled drive vehicles. Fuzzy control methods have been used for steering four wheeled vehicles [8,9].



Fig. 1 CALTRAN's snowplow vehicle

In this paper, as an extension of our previous researches, the neural network control technique is applied to a four wheeled drive bicycle model of the vehicle. The LQR control method is used to obtain optimal controller gains. Even though the LQR controller can stabilize the system and works quite well, tracking performance under snowplowing load becomes very poor. A neural network controller is introduced to compensate for uncertainties. Neural network has been known as a powerful nonlinear controller and used intensively in the area of robot and motion control applications [9,10]. Recently, successful real time neural network applications have been achieved [11-14].

Simulation studies have been conducted and position tracking performances are compared with those of the LQR controller under the virtual plowing condition.

2. FOUR WHEEL DRIVE VEHICLE DYNAMICS

The vehicle is steered by both the front and the rear wheel. For simplicity, the vehicle dynamics is considered as a bicycle model as shown in Fig.2. The dynamic equations are derived based on the references [15,16].

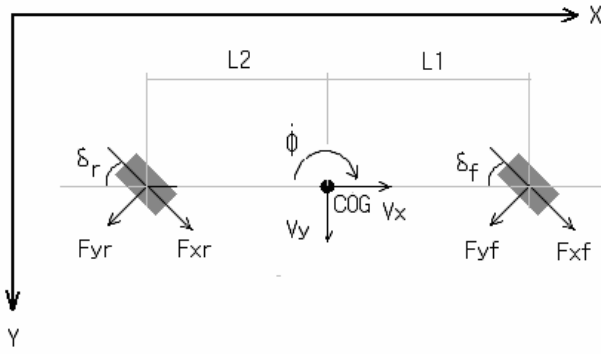


Fig.2. Bicycle model of vehicle

$$m(\dot{V}_x - V_y \dot{\phi}) = F_{xf} \cos \delta_f + F_{xr} \cos \delta_r - F_{yf} \sin \delta_f + F_{yr} \sin \delta_r, \quad (1)$$

$$m(\dot{V}_y + V_x \dot{\phi}) = F_{yr} \cos \delta_r + F_{yf} \cos \delta_f + F_{xf} \sin \delta_f - F_{xr} \sin \delta_r, \quad (1)$$

$$I_z \ddot{\phi} = L_1 F_{yf} \cos \delta_f - L_2 F_{yr} \cos \delta_r + L_1 F_{xf} \sin \delta_f + L_2 F_{xr} \cos \delta_r,$$

where m is total mass of the vehicle, ϕ is a yaw angle, δ_f is a steering angle of the front wheel, δ_r is a steering angle of the rear wheel and

L_1 is a distance from the center of the front wheel to COG,

L_2 is a distance from the center of the rear wheel to COG,

V_x is a velocity of the vehicle along x direction

V_y is a velocity of the vehicle along y direction

F_{xf} is a longitudinal force of the front wheel along x direction

F_{xr} is a longitudinal force of the rear wheel along x direction

F_{yf} is a lateral force of the front wheel along y direction

F_{yr} is a lateral force of the rear wheel along y direction

F_{xf} is a longitudinal force of the front wheel along x direction

I_z is a yaw moment of inertia along z direction.

Configuration of slip angles are shown in Fig.2.

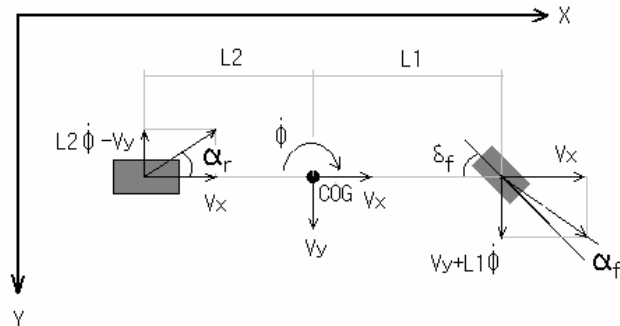


Fig. 3 Slip angle model

Slip angles are defined by

$$\alpha_f = \delta_f - \frac{L_1 \dot{\phi} + V_y}{V_x}, \quad \alpha_r = \delta_r - \frac{V_y - L_2 \dot{\phi}}{V_x}, \quad (2)$$

where α_f is a slip angle of the front wheel and α_r is a slip angle of the rear wheel.

Lateral forces are defined by assuming that the left and the right tires are same as

$$F_{yf} = 2C_{af} \alpha_f, \quad F_{yr} = 2C_{ar} \alpha_r \quad (3)$$

where C_{af} and C_{ar} are cornering stiffness.

Under the assumption that the vehicle is moving at a constant velocity in longitudinal direction and the steered angles are small, then $\cos \delta_f \approx 1, \cos \delta_r \approx 1, \sin \delta_f \approx \delta_f, \sin \delta_r \approx \delta_r$.

Steering angles are considered as a control input and a yaw angle ϕ and a lateral position y are outputs to be controlled.

Let $\omega = \dot{\phi}$ and combining equations (1), (2), and (3) yields the MIMO system that has the following state space representation of the vehicle dynamics as

$$\begin{bmatrix} \dot{V}_y \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\frac{a_2}{m} & -(V_x + \frac{a_1}{m}) \\ -\frac{a_1}{I_z} & -\frac{a_3}{I_z} \end{bmatrix} \begin{bmatrix} V_y \\ \omega \end{bmatrix} + \begin{bmatrix} \frac{2C_{af}}{m} & \frac{2C_{ar}}{m} \\ \frac{2L_1 C_{af}}{I_z} & -\frac{2L_2 C_{ar}}{I_z} \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (4)$$

where

$$a_1 = \frac{2L_1 C_{af} - 2L_2 C_{ar}}{V_x}, \quad a_2 = \frac{2C_{af} + 2C_{ar}}{V_x}$$

$$a_3 = \frac{2L_1^2 C_{af} + 2L_2^2 C_{ar}}{V_x}$$

where L_1 and L_2 are the length indicated in Fig.2

We see from [17] that the difference from the front steering model is the addition of an input column of δ_r in (4).

3. LQR CONTROLLER FOR LATERAL POSITION

The LQR control block diagram for lateral position tracking is shown in Fig. 4. The goal of the LQR control method is to minimize the following objective function

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (5)$$

where Q, R are the weighting matrices. Those Q, R values are selected based on trial and error processes. The ultimate goal is to control the lateral position y .

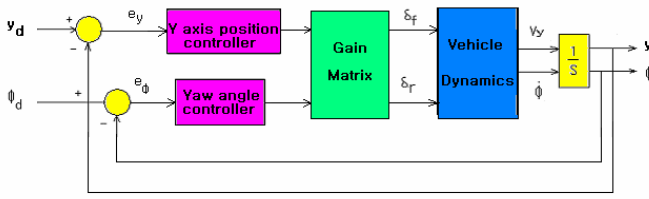


Fig. 4. Lateral control block diagram

The lateral position error is formed as

$$e_y = y_d - y. \quad (6)$$

The yaw angle error is formed as

$$e_\phi = \phi_d - \phi. \quad (7)$$

Those errors are multiplied by controller gains found by the LQR control method. The controller inputs for the front steering u_f and the rear steering u_r are defined as

$$\begin{aligned} u_y &= k_{yf} e_y + k_{\phi r} e_\phi, \\ u_r &= k_{yr} e_y + k_{\phi f} e_\phi, \end{aligned} \quad (8)$$

where $k_{yf}, k_{\phi r}, k_{yr}, k_{\phi f}$ are controller gains obtained by the LQR algorithm. For simplicity, we have the relationship $\delta = u$ by not considering dynamics of a steering mechanism.

4. NEURAL NETWORK CONTROL

4.1 Neural network control structure

In this paper, we are implementing an on-line learning algorithm for the neural network. The idea is that a neural network compensates for uncertainties by minimizing the errors. The objective function is minimized in on-line fashion by adding compensating signals from the neural network to the LQR controller output. Control inputs are defined by adding neural network outputs u_N to (8) as

$$\begin{aligned} \delta_f &= u_y + u_{Nf}, \\ \delta_r &= u_\phi + u_{Nr}, \end{aligned} \quad (9)$$

where u_{Nf}, u_{Nr} are neural network outputs.

Fig. 5 shows the neural network control block diagram. Neural network outputs are added to the LQR controller outputs. At each sampling time, weights of the neural network are updated to generate new compensating signals. The key issue here is how to implement on-line learning and control. To achieve on-line control, certain fast computing power will be required.

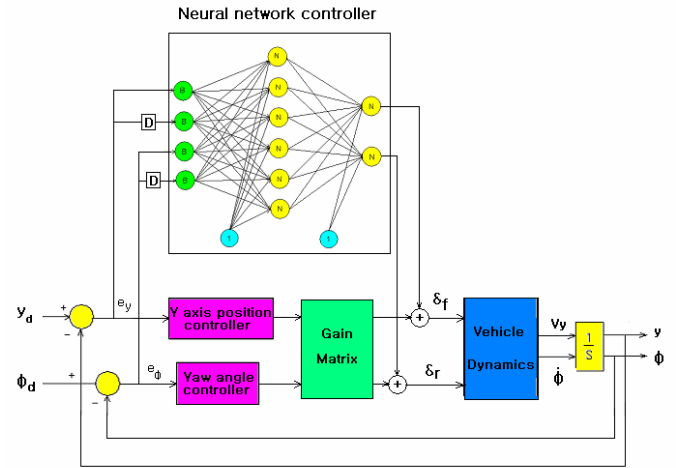


Fig.5. Neural network control block diagram

4.2 Neural network learning

For neural network learning, we have used a general feed-forward structure that has an input layer, a hidden layer, and an output layer shown in Fig. 6. For the control application, 4 input buffers, 6 hidden units and 2 output units are used. Increasing the number of inputs and hidden units did not help to improve the performance in our simulations.

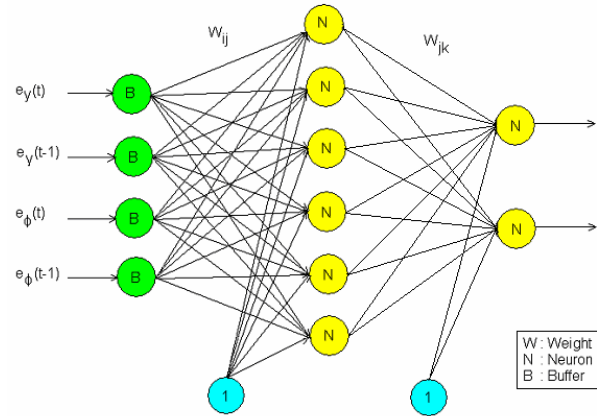


Fig. 6. Neural network structure

For a nonlinear function at a hidden layer and an output layer we have used the hyperbolic tangent function as

$$f(x) = \frac{1 - \exp(-x)}{1 + \exp(-x)}. \quad (10)$$

Here, neural network learning algorithm is derived. Since we are doing on-line learning and control, selecting the training signal is very important. Typically, the bicycle model of the vehicle is a squared MIMO system that two control inputs has to satisfy two different outputs simultaneously.

If $f(y, \dot{y}, \ddot{y}, \phi, \dot{\phi}, \ddot{\phi}, \delta)$ is the system dynamics, equation (4), (8) and (9) can be represented as follows:

$$k_{yf}e_y + k_{\phi f}e_\phi + k_{yf}e_y + k_{\phi r}e_\phi = f(y, \dot{y}, \ddot{y}, \phi, \dot{\phi}, \ddot{\phi}, \delta) - u_N \quad (11)$$

where $u = [u_{Nf} u_{Nr}]^T$ and $\delta = [\delta_f \delta_r]^T$.

So here we define the training signal of neural network as the sum of outputs of PD controllers.

$$v = k_{yf}e_y + k_{\phi f}e_\phi + k_{yr}e_y + k_{\phi r}e_\phi \quad (12)$$

The objective function is defined as

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

Differentiating (13) with respect to weights, we have

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial v} \frac{\partial v}{\partial w} = v \frac{\partial v}{\partial w} = -v \frac{\partial u_N}{\partial w} \quad (14)$$

The update equation in the back propagation algorithm is

$$\Delta w(t) = \eta \frac{\partial u_N}{\partial w} v + \Delta w(t-1), \quad (15)$$

$$w(t+1) = w(t) + \Delta w(t). \quad (16)$$

5. SIMULATION STUDIES

5.1 Vehicle model

The following parameters of the vehicle are used for simulation studies.

Table 1. Parameters of the vehicle

M(kg)	Iz(Kgmsec ²)	Cf, Cr(Kg/rad)	L1(m)	L2(m)
2612.7	810.2	4082.4	1.45	1.94

Here we assume the constant longitudinal velocity at $V_x = 10.973\text{km/sec}$.

Then the state space representation of (4) becomes

$$\begin{bmatrix} \dot{V}_y \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -0.625 & -9.5030 \\ 0.4884 & -19.3453 \end{bmatrix} \begin{bmatrix} V_y \\ \omega \end{bmatrix} + \begin{bmatrix} 3.125 & 3.125 \\ 14.6212 & -19.5051 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix}$$

Controller gains are selected by the LQR method for the simulation studies. We have found the controller gains as

$$K = \begin{bmatrix} -0.0238 & 0.0194 \\ 0.0849 & -0.0492 \end{bmatrix}$$

The vehicle is required to follow the desired lateral trajectory. A load is assumed to be present. The desired lateral position is 3m.

5.2 LQR control method

Fig. 7 shows the tracking performance of LQR controllers.

Different tracking responses of selecting gains as $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $R = \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix}$ are plotted. We see that the

large value r gives the good tracking performance in the aspect of overshoot and settling time before 15 seconds. But when load is present after 15 seconds, tracking performance became the worst. We also altered the value of Q matrix, but $Q=I$ is found to be the best. We found that the value of Q matrix is more sensitive than that of R. Small change in Q results in bad tracking performance. Since this is a coupled system, gain changes affect both tracking performances of the position and the yaw angle. It is clearly shown that offset tracking errors occur when load appears as shown in Fig. 7.

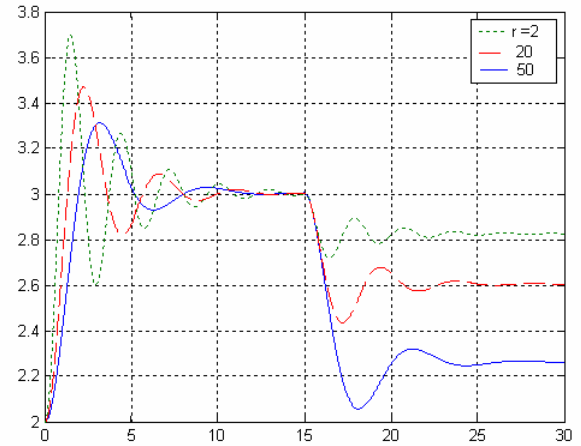


Fig. 7. Lateral position tracking by LQR controllers

Next is the comparison result of lateral control between two wheel drive and four wheel drive vehicles. We see here from Fig 8. that both cases show tracking errors when load is present. However, a less lateral position tracking error has been observed for a four wheel drive steering vehicle even though a two wheel drive vehicle shows no overshoot at a transient response. This explains that the four wheel drive vehicle system pushes more to the lateral direction to achieve the fast settling time when load is present. We also observed that a two wheel drive vehicle become unable to track desired trajectory with other LQR controller gains.

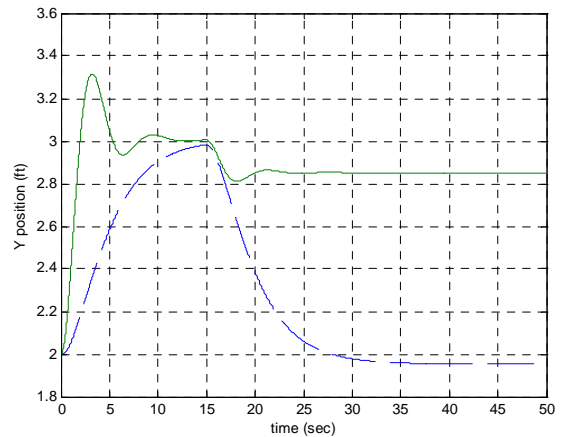


Fig.8. Lateral position tracking of two(dash line) and four(solid line) wheel drive vehicles

5.3 Neural network control method

1. Snowplowing load is 138N.

To minimize those tracking errors, the same experiment has been conducted by adding a neural network controller. For the neural network parameters, $\eta = 0.000055, \alpha = 0.9$ are used and those values are optimized by trial and errors. Fig. 9 shows the lateral position tracking results for the LQR control method and the neural network control method. We clearly see by comparison that a tracking error is minimized by the neural network controller. Fixed LQR controller cannot minimize the tracking error when load is present after 15 seconds.

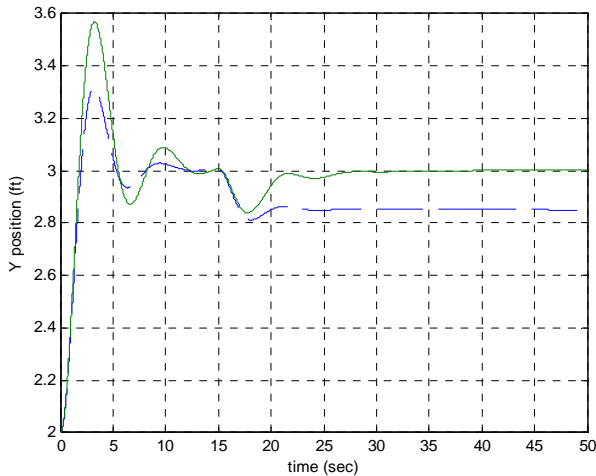


Fig. 9. Lateral position tracking of LQR(dotted line) and Neural network control(solid line).

2. Snowplowing load is 691N.

Now the load has been increased after 15 seconds. We see from Fig. 10 that the neural network controller outperforms in tracking performance while the LQR controller cannot recover from deviation. Even though load is increased 5 times, a position tracking error is compensated fast by the neural network controller.

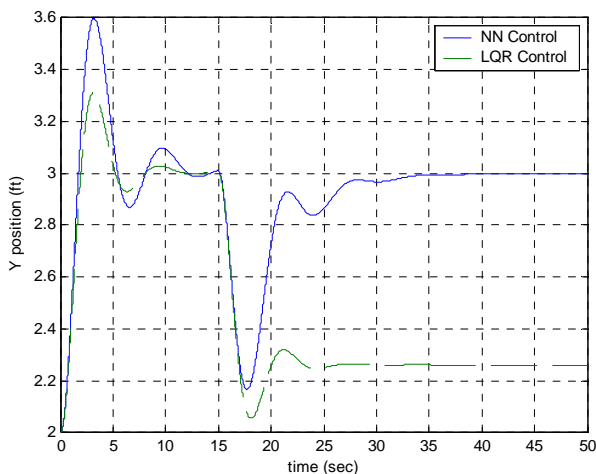


Fig. 10. Lateral position tracking.

6. CONCLUSIONS

This paper presented lateral tracking control simulation studies of the autonomous four wheel drive vehicle. The simplified bicycle model has been used for simulation studies.

LQR controllers are designed for the optimal gains. In addition, neural network controllers are introduced to minimize tracking errors due to the present load.

Simulation studies show that the neural network controller outperforms over the LQR controller under present load conditions. We observed that the two wheeled drive vehicle often went unstable when load is present. The four wheeled drive vehicle performs better than two wheel drive vehicle specially when load is present. It turns out that the neural network controller is robust to variation of load.

ACKNOWLEDGMENTS

The author Seul Jung has been a sabbatical leave at University of California, Davis for 6 months from March to August of 2004. Dr. Seul Jung would like to thank both Chungnam National University and the University of California, Davis for their sincere help.

REFERENCES

- [1] T. A. Lasky and B. Ravani, "Sensor based path Planning and Motion Control for a Robotic System for Roadway Crack Tracking", pp. 609-622, IEEE Trans. On Control Systems Technology, col. 8, No. 4, 2000
- [2] Hyun Taek Cho, Poong woo Jeon, and Seul Jung, "Implementation and control of a roadway crack tracking mobile robot with force regulation", IEEE Conf. On Robotics and Automations, pp. 2444-2449, 2004
- [3] R. T. O'Brien, P. A. Iglesias, and T. J. Urban, "Vehicle lateral control for automated highway systems", IEEE Trans. On Control Systems Technology, pp. 266-273, 1996
- [4] S. Y. Yang, S. T. Park, J. H. Jeong, B. R. Park, "Development of intelligent automated driving control system (lateral control)", KORUS'99, pp. 334-337, 1999
- [5] S. J. Hong, J. Y. Choi, Y. I. Jeong, K. Y. Jeong, M. H. Lee, K. T. Park, K. S. Yoon, and N. S. Hur, "Lateral control of autonomous vehicle by yaw rate feedback", IEEE Symposium on Industrial Electronics, pp. 1472-1476, 2001
- [6] R. H. Byrne, C. T. Abdallah, and P. Dorato, "Experimental results in robust lateral control of highway vehicles", IEEE Control Systems Magazine, pp. 70-76, 1997
- [7] Longitudinal and lateral control of heavy duty trucks for automated vehicle following in mixed traffic: Experimental results from the CHAUFFEUR project", IEEE Conf. On Control Applications, pp. 1348-1352, 1999
- [8] O. Pages, A. E. Hajjaji, R. Ordenez, " H_{∞} Tracking for Fuzzy Systems with an Application to Four Steering of Vehicles", IEEE Conf. On , pp. 4364-4369, 2003
- [9] A. B. Will, M. C. Teixeira, and S. H. Zak, "Four Wheel Steering Control System Design Using Fuzzy Models", IEEE Conf. On Control Applications, pp. 73-78, 1997
- [10] L. Cai, A. B. Rad, and K. Y. Cai, "A robust fuzzy PD controller for automatic steering control of autonomous vehicles", IEEE Conf. On Fuzzy Systems, pp. 549-554, 2003
- [11] S. Jung and T. C. Hsia, "Neural Network Inverse Control Techniques for PD Controlled Robot Manipulator", ROBOTICA, pp. 305-314, vol. 19, No 3, 2000
- [12] H. Miyamoto, K. Kawato, T. Setoyama, and R. Suzuki,

- “Feedback error learning neural network for trajectory control for of a robotic manipulator”, *Neural Networks*, vol. 1, pp. 251-265, 1988
- [13] Sungsu Kim and Seul Jung, Hardware Implementation of a real time neural network controller with a DSP and an FPGA, *IEEE Conf. On Robotics and Automations*, pp. 4639-4644, 2003
- [14] S. Jung and S. B. Yim “Experimental studies of neural network control technique for nonlinear systems”, *ICASE*, pp. 918-926, vol. 7, no. 11, 2001
- [15] H. T. Cho and S. Jung, “Balancing and Position control of an Inverted pendulum on an X-Y Plane using Decentralized neural networks”, *Proceedings of the 2003 International Conference on Advanced Intelligent Mechatronics*, pp. 181- 186, 2003.
- [16] H. T. Cho and S. Jung, “Neural network position tracking control of an inverted pendulum by an x-y table robot”, *International Conference on Intelligent Robots and Systems*, pp. 1210-1215, 2003
- [17] J. Y. Wong, “Theory of Ground Vehicles”, John Wiley & Sons, 1978
- [18] S. H. Yu and J. J. Moskwa, “A Global Approach to Vehicle Control: Coordination of Four Wheel Steering and Wheel Torques, *Journal of Dynamic Systems, Measurement, and Control*, Vol. 116, pp. 659-667, 1994