Learning Petri Network and Its Application to Non-linear System Control

Ohbayashi, Masanao
Department of Electrical and Electronic Systems Engineering

Hirasawa, Kotaro
Department of Electrical and Electronic Systems Engineering

Sakai, Shingo
Department of Energy Conversion Engineering, Interdisciplinary Graduate School of Engineering Sciences

Learning Petri Network and Its Application to Non-linear System Control

Masanao OHBAYASHI* , Kotaro HIRASAWA* , Singo SAKAI**

Abstract: According to the recent knowledge of brain science, it is suggested that there exists the functions distribution in the brain, which means that different neurons are activated depending on which sort of sensory information the brain receives. We have already developed a learning network with functions distribution which is called Learning Petri Network (LPN) and have also shown that this network could learn nonlinear and discontinuous mappings which Neural Network(NN) can not. In this paper, a more realistic application which has dynamic characteristics is studied. From simulation results of a nonlinear crane control system using LPN controller, it has been proved that the control performance of LPN controller is superior to that of NN controller.

**Keywords:** Neural network,Petri net,Learning petri network,Functions distribution

1. Introduction

The aim of this paper is to realize a new type of control system using advanced knowledges of the brain science. According to the recent knowledges of the brain science, there exist mathematical, musical and gymnastic knowledges in the brain and a specific part of the brain is activated corresponding to the specific knowledge. This phenomena is called functions distribution in the brain. It is also known that the brain can select the best action refering to the experience it mastered. This phenomena is called learning of the brain.

The proposed control methodology is a unique one in respect that functions distribution and learning capability the brain has are introduced to the control technology.

Petri network was chosen as a fundamental framework in order to realize functions distribution artificially. Petri network is composed of two kinds of nodes. One is the places where informations are stored, and the other one is the transitions which deal with transactions. Tokens in the petri network are transfered on a specific route instead of all network routes, as a result, a locally activated route of token transfer is established in the network. This establishment of the route depends on the inputs of network and parameter variables in the network. Therefore, by adjusting the parameter variables through learning an appropriately activated local route can be formed.

Using the concept of realizing functions distribu-
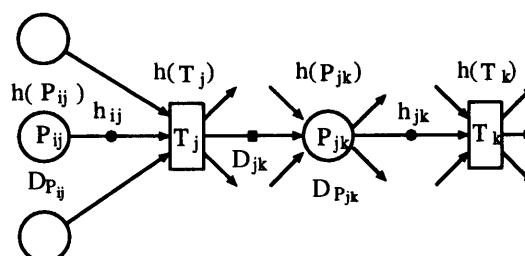
T_j : j transition
P_{ij} : place between T_i and T_j
D_{jk} : time delay from T_j to P_{jk}
$D_{P_{ij}}$: staying time of token at P_{ij}
h_{ij} : firing weight between P_{ij} and T_j
$h(P_{ij})$: firing signal of P_{ij}
$h(T_j)$: firing signal of T_j
$h°(T_j)$: target of firing signal of T_j

Fig.1   Basic structure of Learning Petri Network

tion artificially in the network, Learning Petri Network (LPN) has been proposed and applied to pattern recognition problems and nonlinear system idntification problems effectively[1)2)3)]

In this paper, it is studied whether LPN can be applied to the control of a nonlinear crane system or not, and it has been clarified that LPN has the excellent control performance compared to the neural network controller.

It is also stated how the architecture of LPN is improved to cope with the control of the nonlinear dynamic systems.

2. Fundamental Structure of Learning Petri Network

In **Fig.1**, fundamental structure of LPN is shown. The architecture of LPN is imitated by the architec-

* Department of Electrical and Electronic Systems Engineering
** Department of Energy Conversion Engineering, Interdisciplinary Graduate School of Engineering Sciences (At present;Kyushu Power Electric Corporation.)

ture of Petri network, which is not equipped with the ability of learning. Therefore, in order to make LPN have the learning capability and dynamic characteristics, the following 3 terms are added to Petri network

- Tokens are equipped with firing signals which are introduced in LPN.
- Firing weights are introduced on the input arcs of the transitions and learning can be realized by changing the firing weights.
- Recurrent connections between the nodes are available and time delays are introduced on the output arcs of the transitions.

Therefore, the dynamic behavior of LPN can be realized. Comparison between Petri network and LPN is shown in **Table 1** .

3. Firing Rules of Transition

In this chapter, how the locally activated route is established will be explained. It is the most important point how to fire the transition in order to make an appropriately activated local route.

Firing signal $h(T_j, t)$ of T_j node at time t can be expressed by Eq.(1).

$$h(T_j, t) = f(\{h(P_{ij}, t)\}, \{\lambda_j\}) \qquad (1)$$

where
$h(P_{ij}, t)$: firing signal of P_{ij} place
at time t
λ_j : parameter variable related to
T_j node
$f(\{h(P_{ij}, t)\}, \{\lambda_j\})$: transition processing function

Table 1. Comparison between Petri Network and LPN

		Petri Net	**L.P.N.**
Learning		· does not have	· have
Delay		· firing delay exists	· time delay exists
firing	condition	· any transition can fire when there exist tokens satisfying certain conditions in the preceding places	· transition which has the maximum value of firing signal can fire
	transition	· tokens satisfying certain conditions are transfered from input places to output places	· only one token is taken away from input places and a token are transfered to output places
	signal	· there is not	· there is

Tokens which have the firing signal calculated by Eq.(1) are transfered in the network. Assuming the transition processing function to be sigmoid function, $h(T_j, t)$ can be calculated in the following way.

$$h(T_j, t) = f(\zeta_j) \qquad (2)$$

$$\zeta_j = \sum_{i \in I} h(P_{ij}, t) h_{ij} + \theta_j \qquad (3)$$

h_{ij} : firing weight between P_{ij} and T_j
(parameter variable)
θ_j : firing threshold of T_j (parameter variable)
$f(\zeta_j)$: sigmoid function
I : set of places whose output is connected to T_j
and from which a token has been transfered.
(hereafter, set and density are represented
by the same symbol)

The firing signals are calculated for the transitions which have at least one token in their input places. Which transition makes firing is determined by the value of firing signal. To be more specific, only the transition which has the maximum value of firing signal can fire at a time. When the transition, for example T_j, fires , only one token is taken away from each non-empty place connected to input side of T_j, and then a token is given the value of firing signal of $h(T_j, t)$ and transfered to each place connected from output side of T_j. The value of firing signal of P_{ij} ,that is, $h(P_{ij}, t)$ is defined as the firing signal of the taken away token.

According to the above procedure, tokens are tranfered in LPN depending on the value of firing signals of tokens in the input places, and the values of firing weights on the arcs of LPN.

4. Learning Algorithm

Learning Algorithm of LPN is based on the Back Propagation algorithm in neural networks, but it is a fundamental different point that the learning is carried out on the only path of the token transfers in LPN instead that all of the weights of neural networks are adjusted independently of the input informations. The following is a summary of learning algorithm[3][4][5].

$$h_{ij} \leftarrow h_{ij} - \gamma \left[\sum_{t' \in T} \left\{ \frac{\partial h(T_j, t')}{\partial h_{ij}} \delta(T_j, t') \right\} + \frac{\partial E}{\partial h_{ij}} \right] \qquad (4)$$

$$\delta(T_j, t) = \sum_{k \in K} \left[\frac{\partial h(T_k, t + D_{jk} + D_{P_{jk}})}{\partial h(T_j, t)} \right.$$
$$\left. \times \delta(T_k, t + D_{jk} + D_{P_{jk}}) \right] + \frac{\partial E}{\partial h(T_j, t)} \qquad (5)$$

h_{ij} : firing weight beteen P_{ij} and T_j

γ : learning coefficient

$h(P_{ij}, t)$: firing signal of P_{ij} at time t

$h(T_j, t)$: firing signal of T_j at time t

T : set of sampling time

D_{jk} : time delay from T_j to P_{jk}

$D_{P_{jk}}$: staying time of token at P_{jk}

E : criterion function

K : set of transition whose input is connected from T_j and to which a token has been transfered.

5. Nonlinear Crane System

In this chapter, the nonlinear crane system which was used to study the applicability of LPN will be explained. Dynamics of the nonlinear crane system can be expressed by the following differential equations(See **Fig.2**).

- Subsystem of crane stand moving

$$\ddot{x} = -\frac{mg}{M}\theta - \frac{D+G}{M}\dot{x} + \frac{G}{M}u_d \tag{6}$$

$$\ddot{\theta} = -\frac{M+m}{lM}g\theta - \frac{D+G}{lM}\dot{x} + \frac{G}{lM}u_d \tag{7}$$

- Subsystem of load rolling up

$$\ddot{l} = -\frac{C+G_m}{m}\dot{l} + \frac{G_m}{m}u_m \tag{8}$$

where

x : distance of crane stand moving

θ : angle of swinging rope

l : length of load rolling up

M : mass of stand

m : mass of load

D, C : coefficients of friction

g : *gravity*

G, G_m : transfer coefficients from torque to driving force

u_d, u_m : control variables for crane stand moving and load rolling up subsystem respectively

Putting $h(T_1, t) = x(t), h(T_2, t) = \dot{x}(t), h(T_3, t) = \theta(t)$, $h(T_4, t) = \dot{\theta}(t), h(T_5, t) = l(t), h(T_6, t) = \dot{l}(t)$, Eq.(6) $\sim$ (8) can be written in the following discrete form.

- Subsystem of crane stand moving

$$h(T_1, t) = h(T_1, t-1) + \triangle T h(T_2, t-1) \tag{9}$$

$$h(T_2, t) = \left(1 - \frac{D+G}{M}\triangle T\right)h(T_2, t-1)$$
$$- \frac{mg}{M}\triangle T h(T_3, t-1) + \frac{G}{M}\triangle T u_d(t) \tag{10}$$

$$h(T_3, t) = h(T_3, t-1) + \triangle T h(T_4, t-1) \tag{11}$$

$$h(T_4, t) = -\frac{D+G}{M}\triangle T \frac{h(T_2, t-1)}{h(T_5, t-1)}$$
$$- \frac{M+m}{M}g\triangle T \frac{h(T_3, t-1)}{h(T_5, t-1)}$$
$$+ h(T_4, t-1) + \frac{G}{M}\triangle T \frac{u_d(t)}{h(T_5, t-1)} \tag{12}$$

- Subsystem of load rolling up

$$h(T_5, t) = h(T_5, t-1) + \triangle T h(T_6, t-1) \tag{13}$$

$$h(T_6, t) = \left(1 - \frac{C+G_m}{m}\triangle T\right)h(T_6, t-1)$$
$$+ \frac{G_m}{m}\triangle T u_m(t) \tag{14}$$

where

$\triangle T$: sampling time

In **Fig.3**, the nonlinear crane system which is described by Eq.(9) $\sim$ (14) and its controllers are shown in the framework of Universal Learing Network[5].

There exist two controllers. One is to control the moving of a crane stand and to cotrol the angle of a swinging rope, the other is to control the rolling up of a load. Each controller is a three layered neural network whose hidden layer is made up of 10 nodes.

6. Design of a Nonlinear Crane System Controller

In this chapter, LPN and neural network controller for a nonlinear crane system are designed. In order to conrtol nonlinear systems in a stable way, the value of the control variables should not change rapidly. But it
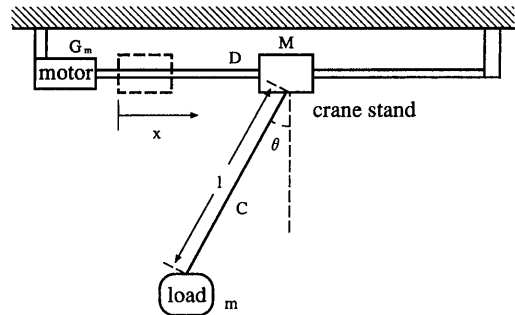

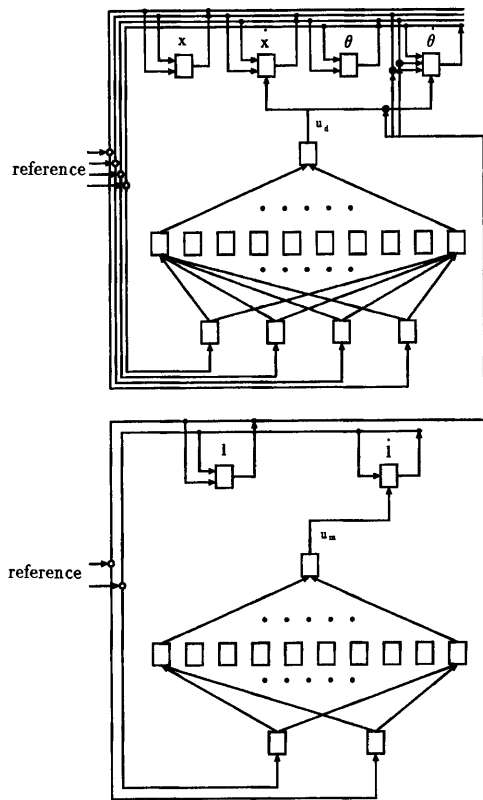**Fig.2**   Structure of nonlinear crane system

Fig.3 Network of crane control system

may happen that the output value of the LPN controller changes abruptly at the time of route changes. Particularly at the beginning of the learning, abrupt changes of the LPN controller output may make the system unstable, though such phenomena do not occur after enough learning.

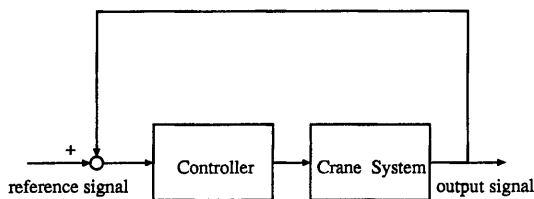In order to solve this problem, LPN has been improved in the following two points.

Fig.4 Structure of control system with error signal
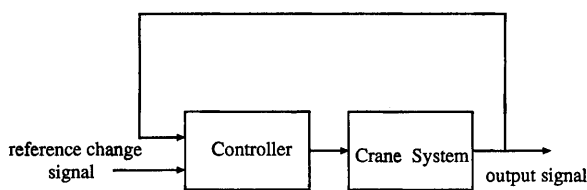

Fig.5 Structure of control system without error signal

As mentioned in chapter 3, firing rule was such that only the transition which has the maximum value of firing signal can fire at a time. One of the improvement is to change the firing rule so as to fire n transitions which have the large value of firing signal. According to the one transition firing rule, the value of the control variable may change abruptly, but adopting the n transition firing rule can make the system work stably.

The other improvement is to filter the out of the controller in the following ,

$$h(T_j, t) \leftarrow \alpha * h'(T_j, t) + (1.0 - \alpha) * h(T_j, t - 1) \qquad (15)$$

where

$h(T_j, t)$: controller output after filtering
$h'(T_j, t)$: controller output before filtering
α : filter constant

Filtering the output of the controller makes the possibility of unnecessary change of the route decrease. As Eq.(15) is a kind of computation which can be realized by the recurrent network with time delays, filter constant α can also be adjusted by learning.

Hereafter, two control systems are studied to investigate the effectiveness of the LPN controller compared with the neural network(NN) controlller. One is the controller system where error signal of the crane output and its reference value is inserted to the controller (Control system with error signal) , the other one is the system where reference change signal instead of error signal is inserted to the controller (Control system without error signal) (See **Fig.4,Fig.5** respectively).

Control system with error signal is a so called feedback control system which calculates the control variables for crane stand moving and load rolling up subsystem u_d, u_m from error signals of $x(t)$ and x_{ref}, $l(t)$ and l_{ref} and from $\dot{x}(t)$, $\dot{l}(t)$, $\theta(t)$, $\dot{\theta}(t)$. On the other hand, control system without error signal is especially designed to show the usefullness of the LPN controller and the inputs of the controller are the system outputs $x(t)$, $\dot{x}(t)$, $\theta(t)$, $\dot{\theta}(t)$, $l(t)$, $\dot{l}(t)$ and reference change signal. Reference change signal is a step function given at the time when reference signal x_{ref} and l_{ref} is altered.

Both control systems are the same in a sense that the inputs to the controller change abruptly by the change of reference signals. But, because of the excellent capability of the negative feedback system, selecting on appropriately activated local route in the controller may not improve the control performance of the system in the control system with error signal. On the contrary, an appropriately activated local route of the LPN controller in the control system without error signal is expected to improve the performance because functions distribution

in the LPN controller work effectively. In the next chapter, simulations for a nonlinear crane control system are carried out to study the effectiveness of selecting an appropriate local route in the LPN controller compared with NN controller.

7. Simulations of a Nonlinear Crane Control System

The aim of control is to move the crane stand and to roll up the load in order to reach the target concurrently. In the subsystem of crane stand moving, in the first half of the control time the crane stand should be moved from $-0.5m$ to $0.0m$ and in the latter half of the control time the crane stand should be moved from $0.0m$ to $0.5m$. In the subsystem of load rolling up, in the first half of the control time the load should be rolled up from $1.0m$ to $0.5m$ and in the latter half of the control time, the load should be rolled down from $0.5m$ to $1.0m$.

The criterion function which can realize the above mentioned dynamics of a nonlinear crane control system is represented as follows.

$$E = E_x + E_l \tag{16}$$
$$E_x = \frac{1}{2}\sum_s \Big[Q_{11}(x_{ref} - h(T_1, s))^2$$
$$+Q_{12}(h(T_2, s))^2 + Q_{13}(h(T_3, s))^2$$
$$+Q_{14}(h(T_4, s))^2 + R_1(u_d(s))^2\Big] \tag{17}$$
$$E_l = \frac{1}{2}\sum_s \Big[Q_{21}(l_{ref} - h(T_5, s))^2 + Q_{22}(h(T_6, s))^2$$
$$+R_2(u_m(s))^2\Big] \tag{18}$$

E_x : criterion function of crane stand moving
E_l : criterion function of load rolling up
Q, R : weighting coefficients
x_{ref} : reference value of $x(t)$
l_{ref} : reference value of $l\ (t)$

Using the criterion function E, the control performance of the nonlinear crane control system by the LPN controller is compared with that of the neural network controller.

Simulation conditions are shown **Table 2**. As mentioned above, the object of the simulations is to investigate whether an appropriate route can be selected in the network according to the change of the environment(change of reference value) as well as to study the effectiveness of the LPN controller compared with the neural network controller.

Table 2. Simulation conditions

mass of crane stand　M	40.0 [kg]
mass of load　m	2.0 [kg]
coefficient of friction of crane stand moving　D	300.0 [kg/sec]
transfer coefficient from torque to driving force　G_m	700.0 [N/v]
sampling time	0.02(s)
number of sampling	2000
number of learning	50000
range of random number for initial value of firing weights	± 0.03
range of random number for initial value of firing thresholds	± 0.005
learning coefficient of firing weights	0.00005
learning coefficient of firing thresholds	0.00000001
weighting coefficient of criterion function　Q_{11}	1.00
Q_{12}	1.00
Q_{13}	1.00
Q_{14}	1.00
R_1	0.01
Q_{21}	1.00
Q_{22}	1.00
R_2	0.01
Controller of crane stand moving	
number of nodes in the first layer	4
number of nodes in the hidden layer	10
number of nodes in the output layer	1
function of nodes in the first layer	Sigmoid
function of nodes in the hidden layer	Sigmoid
function of nodes in the output layer	Sigma
Controller of load rolling up	
number of nodes in the first layer	2
number of nodes in the hidden layer	10
number of nodes in the output layer	1
function of nodes in the first layer	Sigmoid
function of nodes in the hidden layer	Sigmoid
function of nodes in the output layer	Sigma

7.1 Simulation Results of Control System with Error Signal

Setting up the number of firing transitions to 5,3,1, simulations were carried out three times changing the initial value of parameter variables.

In Table 3, the values of the criterion functions E, E_x, E_l obtained by the LPN controller and neural network controller are shown.

In **Fig.6** and**Fig.7**, dynamics of the crane stand moving and load rolling up are also shown, where the number of the firing transitions in the hidden layer is three. From **Fig.6** and **Fig.7**, it is seen that there is no difference between the dynamics of the system obtained by the LPN controller and that obtained by the neural network controller. And it is also seen that the number of the firing transitions has little influence on the dynamics of the system. This is because the capability of the feedback control exceeds the locally activated appropriate routes in the LPN controller in the control system of error signal.

Learning curves of the filter constant α of the crane stand moving and load rolling up are shown in **Fig.8** in

case that the number of firing transitionins is three. In **Fig.9**, results of the route formation at various times in the LPN controller of crane stand moving are also shown.

From **Fig.8** and **Fig.9**, it is clarified that α converges to the appropriate constant after learning and different routes are constituted at different control times, which means the creation of functions distribution depending on the change of the dynamics of the system.

Table 3. Criterion function of crane control system with error signal

controller		E	E_x	E_l
NN		34.41	17.10	17.30
		38.52	17.08	21.43
		33.83	17.10	16.72
LPN	5	35.95	16.14	19.80
		35.38	13.58	21.80
		34.73	13.54	21.19
	3	38.41	16.31	22.10
		34.91	12.88	22.03
		36.43	14.56	21.87
	1	38.05	15.96	22.09
		34.82	12.83	21.99
		34.45	12.85	21.60

Fig.7 Dynamics of load rolling up (with error signal)

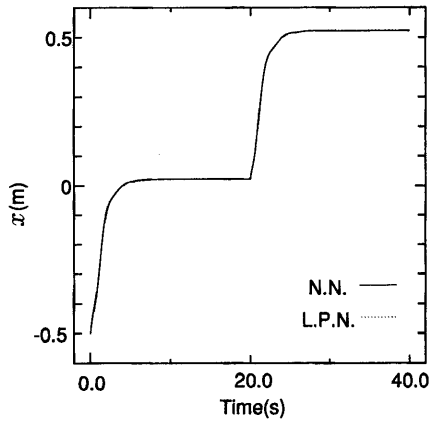**Fig.6** Dynamics of crane stand moving (with error signal)

Fig.8 Learning curves of parameter α (with error signal)

7.2 Simulation Results of Control System without Error Signal

As the case of control system with error singnal, setting up the number of firing od transitions to 5, 3, 1, simulations were carried out 3 times changing the initial value of parameter variables. In Table 4, the value of the criterion function E, E_x, E_l obtained by the LPN controller and neural network controller are shown.
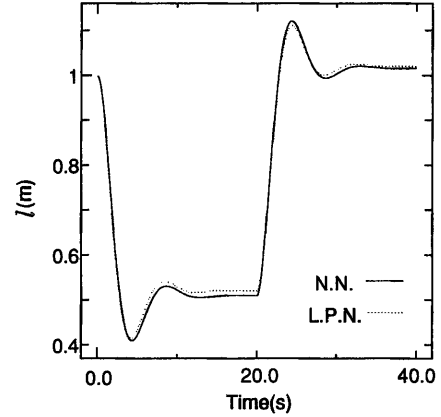
In **Fig.11** and **Fig.12**, dynamics of the crane stand moving and load rolling up with 3 firing transitions in the hidden layer are also shown.

It is also shown in **Fig.13** and **Fig.14** that learning curves of the filter constant α under the condition that the three transitions can fire, and results of the route establishment in the LPN controller at various times. Although α should take the value from 0.0 to 1.0, it has been found that α converged to the value which is more than 1.0. It is not valid for α to take the value more than 1.0 in an usual sense of exponential filtering. Nevertheless the results in **Fig.13** is admissible in the sense of finding optimal α in order to optimize the criterion function in the recurrent network. It is also clarified
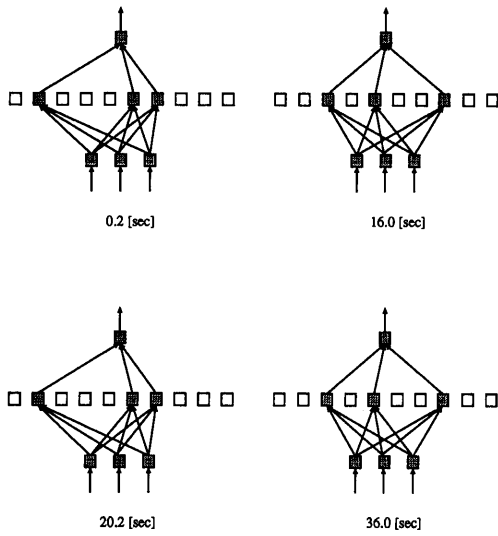
Fig.9 Route Formations at various times
(with error signal)

Fig.10 Changing of $h(T_j, t)$ caused by
alterating reference change signal

Table 4. Criterion function of crane control system
without error signal

controller		E	E_x	E_l
NN		49.76	16.78	32.97
		51.60	16.93	34.66
		45.57	18.27	27.29
LPN	5	34.50	14.33	20.16
		32.86	15.15	17.70
		34.79	16.31	18.48
	3	33.85	16.69	17.15
		33.48	15.83	17.65
		33.38	14.61	18.77
	1	38.08	15.04	23.04
		39.87	15.94	23.93
		41.54	16.99	24.54

that different routes are constructed at different control times depending on the behavior of the dynamics of the system in the LPN controller.

From the results mentioned above, it is clear that in the simulation of control system without error signal, better control performance can be achieved by the LPN controller than the neural network controller.

As shown in **Fig.10**, change of the output of the neural network controller caused by alteration of the reference change signal is small. On the contrary, as an appropriate route can be established in the LPN controller by changing the reference change signal, the LPN controller can realize better performance than the neural network controller.

It is also seen that the larger the number of firing transitions is, the better the control performance is. This is because firing a number of transitions can acquire better nonlinear characteristics of the controller than firing just only one transition.

But, firing a good many number of transitions degrades the control performance as it approches to the neural network controller.

Threrfore, it is important to decide the appropriate number of firing trasitions in the hidden layer of the LPN controller so as to make the functions distribution work effectively.
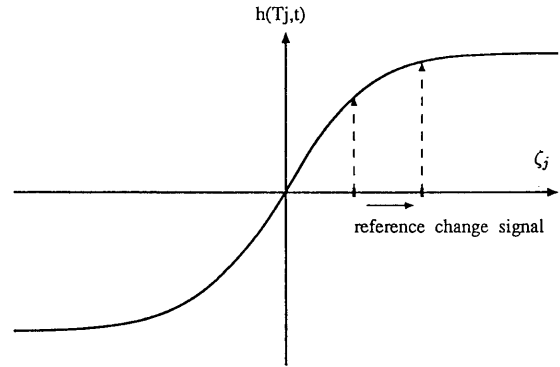

Fig.11 Dynamics of crane stand moving
(without error signal)

the nonlinear systems depending on the changes of the environments.

The control system without error signal is designed self-willedly in order to show the usefullness of the LPN controller.

In a near future, it should be studied that the LPN controller shows its effectiveness mentioned in the privious chapter even when it is applied to the usual feedback control systems which are more complicated and large scale than the one studied in this paper.
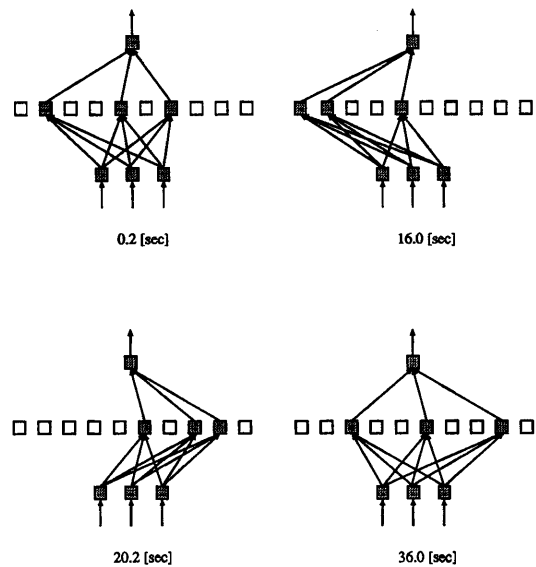

Fig.12   Dynamics of load rolling up
(without error signal)

0.2 [sec]   16.0 [sec]

20.2 [sec]   36.0 [sec]

Fig.14   Route Formations at various times
(without error signal)

Fig.13   Learning curves of parameter α
(without error signal)

8.   Conclusion

In this paper, a new control methodology for nonlinear system has been proposed, which has the structure of functions distribution and the capability of learning. From simulations, it has been clarified that the LPN controller can achieve the desirable control characteristics of the system even when the neural network controller can not control the system preferably.

This is a first step to realize a self-organizing system that can reorganize the structure of the controller of

References

1) K.Hirasawa, M.Ohbayashi, J.Murata et al, " A new Modelling Method of Functional Distribution of Brain by Modified Petri Net", Tran. of the IEE of Japan, Vol.115-C , No.5, pp719-727, 1995

2) K.Hirasawa, S.Oka, M.Ohbayashi, S.Sakai, J.Murata, "Learning Network based on Petri Network with Functional Distributed Capability", Tran. of SICE, Vol.32, No.2, pp241-250,1996

3) K.Hirasawa, S.Oka, S.Sakai, M.Ohbayashi, J.Murata, "Learning Petri Network with Route Control", IEEE International Conference on Systems, Man and Cybernetics, pp.2706-2711, 1995

4) K.Hirasawa, M.Ohbayashi, F.Fujita and M.Koga, "Universal Learning Network", Tran. of the IEE of Japan, Vol.116-C, No.7, 1996

5) K.Hirasawa, M.Ohbayashi and J.Murata, "Universal Learning Network and Computation of its Higher Order Derivative",IEEE International Conference on Neural Networks, Vol.3, pp.1273-1277,1995