'94 KACC (1994. 10. 17 ~ 20)

신경회로망을 이용한 함수의 근사와 동적 시스템에의 응용

엄태덕, 홍선기, 김성우, 이주장
한국과학기술원 전기 및 전자공학과

Approximation of the Functional by Neural Network and Its Application to Dynamic Systems

Tae-Duck Ohm, Sun-Gi Hong, Sung-Woo Kim and Ju-Jang Lee
Department of Electrical Engineering, KAIST

Abstact - It is well known that the neural network can be used as an universal approximater for functions and functionals. But these theoretical results are just an existence theorem and do not lead to decide the suitable network structure. This doubtfulness whether a certain network can approximate a given function or not, brings about serious stability problems when it is used to identify a system. To overcome the stability problem, We suggest successive identification and control scheme with supervisory controller which always assures the identification process within a basin of attraction of one stable equilibrium point regardless of fittness of the network.

I. INTRODUCTION

The problems of approximating a function of several variables by neural network has been studied by many authors. Carroll and Dickinson [1] used the inverse Radon transformation. Cybenko used the functional analysis method [2], combining the Hahn-Banach theorem and Riesz representation theorem. Hornik *et all.* [3] applied the Stone-Weierstrass theorem. Tianping Chen [4] proved neural networks can be used to approximate functionals as well as functions. But these works are dedicated to found the theoretical basement of neural network application, so what kind of structure is best for a given function is remained unsolved. This mismatch between neural network output and actual output can make the state trajectory of the system diverge. So we suggest the heuristic method to confine the state trajectory in the bounded region or in the basin of attraction of stable equilibrium point.

The main concept can be split in two. First, the system identification is always performed with the state trajectory converging to the stable equilibrium point and start with the region identified under stability using supervisory control so the stability is always assured until task is completed. Second, identification on the domain of stable equilibrium point avoid nonuniform data sampling which occurs due to unstable equilibrium points.

A. Supervisory Control

The target system for which the discrete-type supervisory control can be applied must have a input as an additive term, that is,

$$x(k+n) = f(x(k), x(k+1), \cdots, x(k+n-1)) + bu \quad (1)$$

In such a system, if function f was bounded, $|f| < f^U$, by adding a supervisory control input u_s which takes an action when the Lyapunov function V of error dynamics is larger than V_M, small region is identified with state trajectory being confined in the assigned boundary. So to speak, supervisory control does the role of pushing when state trajectory shows a tendancy to exit out of the boundary.

When the supervisory control input is not activated, in other words, the current state is inside the pre-assigned region, we assume that the classical pd control is enough to roughly track the desired trajectory for getting uniformly distributed training samples. But most of cases we concerned does not have this property, so it takes much time to make the neural network finely approximate the nonlinear function.

Therefore, it's better to choose the supervisory controlled region as small as possible.

B. Identification and Control Using Multiple Neural Networks

After the small region is identified by the supervisory controller, the new procedure of successive identification and control starts. First, a stable equilibrium point is put to locate at the center of the identified region. Then, checking the relative stability by a certain criterion function, neural network is trained until the approximation error is smaller than the tolerance. This process assures stability cause the training samples is obtained from the state trajectory converging to the equilibrium point. The feasibility of this procedure is based on the fact that there always exists more or less stable region due to the generalization property of neural networks.

In early days of our research [5], the significant problem in combining supervisory control and succesive identification and control algorithm was that succesive identification and control algorithm is applied to the system which is different from (1). But when the order of the system is the first, two forms are same and if the first order supervisory controller is applied to indivisual state dynamics which has coupling with the other state variables, it becomes possible to make the respective state variable be bounded.

II. DESIGN OF THE SUPERVISORY CONTROLLER

The system equation is given by the equation of motion (1) where f is an unknown nonlinear function, b is a known positive constant and $u \in R$ is the input. Define the state vector $x(k)=[x(k),x(k+1),\cdots,x(k+n-1)]'$ and it is available by measurement. Our control objective is to design a neural network identifier and controller to track the given reference signal x_d and by using the pre-designed supervisory controller. The nonlinear funtion $f(x)$ is known and assumed to be bounded such that

$$|f(x)| \le f^U(x) \qquad (2)$$

Let the error vector $x_e = x - x_d$, then we can control the system with the following

$$u^* = \frac{1}{b}[-f(x)+x_d(k+n)-k'x_e] \qquad (3)$$

where the gain $k=[k_1,k_2,\cdots,k_n]'$ is chosen to make it possible that the polynomial $h(z)=z^n+k_nz^{(n-1)}+\cdots+k_1$ has eigen- values inside the unit circle. Applying (3) to (1) makes the system asymtotically stable, but this control input can not be implemented since $f(x)$ is not known. Our purpose is to design a neural network controller instead of using (3). Before designing the nerual network controller, we design a supervisory controller that is used to training the networks within the region of interest.

From now on, the supervisory controller is designed using Lyapunov theory to gaurantee the boundedness of the system trajectory. Suppose that the control u is the addition of the neural network based controller, u_c, which will be designed later, and the supervisory control, u_s, that is,

$$u = u_c + u_s \qquad (4)$$

Substituting (4) to (1), we have

$$x(k+n)=f(x(k))+b(u_c+u_s). \qquad (5)$$

By adding and substracting bu^* in (1), we obtain the equation of error:

$$x_e(k+n)=-k'x_e(k)+b(u_c+u_s-u^*), \qquad (6)$$

or equivalently

$$x_e(k+1)=\Lambda x_e(k)+b(u_c+u_s-u^*) \qquad (7)$$

where

$$\Lambda=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_n \end{bmatrix}, \qquad b=\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b \end{bmatrix}. \qquad (8)$$

Define $V=\frac{1}{2}x_e'Px_e$, where P is a symmetric positive definite matrix satisfying the following Lyapunov equation

$$\Lambda'P\Lambda - P = -Q, \qquad (9)$$

where $Q>0$. Then the derivative of V along the system trajectory becomes

$$\begin{aligned} \Delta V &= V(k+1)-V(k) \\ &= -\frac{1}{2}x_e'Qx_e + x_e'\Lambda'Pb(u_c+u_s-u^*) \\ &\le -\frac{1}{2}x_e'Qx_e \\ &\quad + |x_e'\Lambda'Pb|(|u^*|+|u_c|) \\ &\quad + x_e'\Lambda'Pbu_s \end{aligned} \qquad (10)$$

We build the supervisory control u_s as follows

$$u_s = -Isgn(x_e'\Lambda'Pb)[\,|u_c| + \frac{1}{b}(f^U + |x_d(k+n)| + |k'x_e|)] \quad (11)$$

where $I=1$ if $V>V_M$ and $I=0$ otherwise and V_M is a constant specified by the designer according to the training region of the neural networks. Substituting (11) and (3) into (10) for the case of $I=1$, we have

$$\begin{aligned}\Delta V &\le -\frac{1}{2}x_eQx_e + |x_e'\Lambda'Pb|\,\frac{1}{b}(|f|-f^U)\\ &\le -\frac{1}{2}x_eQx_e < 0\end{aligned} \quad (12)$$

Therefore using the supervisory control u_s of (11), we always have $V<V_M$. Further, since $P>0$. The boundedness of V implies the boundedness of x_e, which implies the boundedness of x.

Now, we explain the details of identification and control procedure then discuss about the sumulation study.

identification procedure : At first, the system is initiated with no control inputs. If the system is stable, then the system trajectory goes to one of its stable equilibrium points. In this case, there are no needs to use the supervisory controller. The neural networks learn and identify the nonlinear function during the iteration process. On the contrary, if the system is unstable, the system trajectory will be blown up. In the case when the system trajectory goes over the pre-determined region by V_M, that is $x(t)\in\Phi(x)^c$, where $\Phi(x)=\{x \mid \frac{1}{2}x_e'Px_e<V_M\}$, the supervisory controller is active to push the trajectory into the region. Therefore we can continuously obtain the training input/output samples in the region.

control procedure : The control strategy is to cancel out the nonlinear function in the system using the trained neural networks. The control input is generated by the following:

$$u_c = -\frac{1}{b}N_f(x) + u_{PD} = -\frac{1}{b}N_f(x) - k'x_e \quad (13)$$

where $u_{PD}=-k'x_e$ is a PD type controller. If the networks are sufficiently accurate to verify the cancellation, then the error is described by

$$x_e(k+n)+k_nx_e(k+n-1)+\cdots+k_1x_e\cong 0, \quad (14)$$

so that the system is controlled to be asymtotically stable.

III. SUCCESSIVE IDENTIFICATION AND CONTROL

The supervisory control is used to get the identification of small region under stability and set the base for extention to the outer space. In this chapter the concept of successive identification and control will be explained. In case of the first order system, the graphical analysis is adopted to show the feasibility of the control scheme.

We focuses on the system which can be expressed as the following nonlinear difference equation:

$$x(k+1)=f[x(k)]+u(k) \quad (15)$$

where $x(k)$ and $u(k)$ are the state and control input at time k, $u(k),x(k)\in R^n$, and $f\,R^n\rightarrow R^n$ is an unknown mapping. If we know the function in some domain we can cancel out the nonlinearity and make a point asymptotically stable. Such a input $u(k)$ is $-f[x(k)]+Ax(k)+(I-A)x^*$. Here matrix A has eigenvalues inside the unit circle. So, if the system is partially known(region D_0), a stable equilibrium point can be made at the center of it. Then it is reasonable to think basin of attraction of this point is stretching outside due to the generalization property. Here the word generalization seems a little ambiguous. But it becomes clear if graphical analysis is used for the first order case.

The concept of succesive identification and control was first proposed by Kumpati S. Narendra and Asriel U. Levin [7]. It was the new approach to obtain the uniform sampling and the stability in application of neural network for identification but there are several problems. First, a theorem(proof in [8]) below is used to assure that the identification is performed in the basin of attraction of a stable equilibrium point. But that is a purely mathematical theorm so in practical applications it is difficult to find input bound u that makes the state trajectory inside the basin of attraction. Also, they use the error between real function and approximated neural network as another neural network initiation condition. For this reason, though the paper contributes to the uniform sampling, it failed to find a method to keep stability.

Theorem: *Let p be the unique asymptotically stable equilibrium point of f on the interval I, then starting at any point* $x(0) \in I$ $\exists \varepsilon$ *s.t if* $\| u(k) \| < \varepsilon$ *then* $\forall k$ $x(k) \in I$.

To solve these problems we suggest a new neural network initiation condition. The validaty of this criterion is showed using graphical analysis. In Fig. 1 (a) below if the initial condition is min(L1,L2) apart from the equilibrium point they converge to the equilibrium point because in this region it is always satisfied that $| x(k+1) - x^* | < | x(k) - x^* |$. Let's assume that we have a function N_f which approximates $f(x(k))$ with the error less than δ in the region $| x - x^* | < D$. Then, we can make a equilibrium point at the center of the known region by subtracting N_f from f and adding x^*. The function $f - N_f + x^*$ has the value near x^* in the region $| x - x^* | < D$ and does not vary fastly outside this region by the generalization property of the neural networks(Fig. 1. (b)). This makes us get some stable region and if we identify the region where $| x(k+1) - x^* | < D$, we can satisfy $| x(k) - x^* | < \delta$ in 2-step(Fig. 1. (c)). After finishing the 1st identification we use another neural network and repeat this procedure to reach the goal.

We explain the procedure with the aid of the graphical analysis in the case of the 1st order system, but this algorithm can be applied for the n-th order system which can be expressed as (15).

SIMULATION STUDY:

The simulation of successive identification and control algorithm with supervisory control is perform for the function $f[x(k)] = \cos[x(k)]\sqrt[3]{4x(k)}$ which needs 6 neural networks with Narendra's approach. In our approach, the region obtained by the supervisory control is [-5,-3] and the whole region [-5, 5] is identified with 3 neural networks.(Fig. 2)

IV. CONCLUSION

In this paper, the successive identification and control algorithm with supervisory control is developed to contribute to the uniform sampling and the stability. But the most serious problem of this algorithm is that the class of system on focus is limited to (15). The extension of this algorithm to the system where input and state have the different dimensions is now proceeding.

References

[1] S. M. Carroll and B. W. Dickinson, "Construction of neural nets using Radon transform," in *Proc. IJCNN I*, 1989, pp. 607-611

[2] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Contr., Signals, Syst.*, vol. 2, pp. 303-314, 1989.

[3] K. Hornik, M. Stichcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359-366, 1989.

[4] Tianping Chen and Hong Chen, "Approximations of continuous functionals by neural networks with application to dynamic systems," IEEE Trans. Neural Networks, vol. 4, no. 6, pp. 910-918, 1993.

[5] Taeduck Ohm, Sun-Gi Hong, Sung-Woo Kim and Ju-Jang Lee, "Stable identification and control using multiple neural networks and supervisory control," *제어계측연구회 춘계학술대회 논문집*, 1994.

[6] Kumpati S. Narendra and Asriel U. Levin, "Regulation of Nonlinear Dynamical Systems Using Multiple Neural Networks", *procedings of ACC*, 1992.

[7] R. L. Devaney, "An Introduction to Chaotic Dynamical Systems" *Addison Welsley*, Second Edition, 1989

[8] D. Psaltis, A. Sideris, and A. A. Yamamura,

"A multilayered Neural Network controller," *IEEE Control system. Magazine*, pp. April 1988

[9] Li-Xin Wang, "Stable Adaptive Fuzzy Control of Nonlinear Systems", *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, 1993

[10] K.S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, no. 1,1990

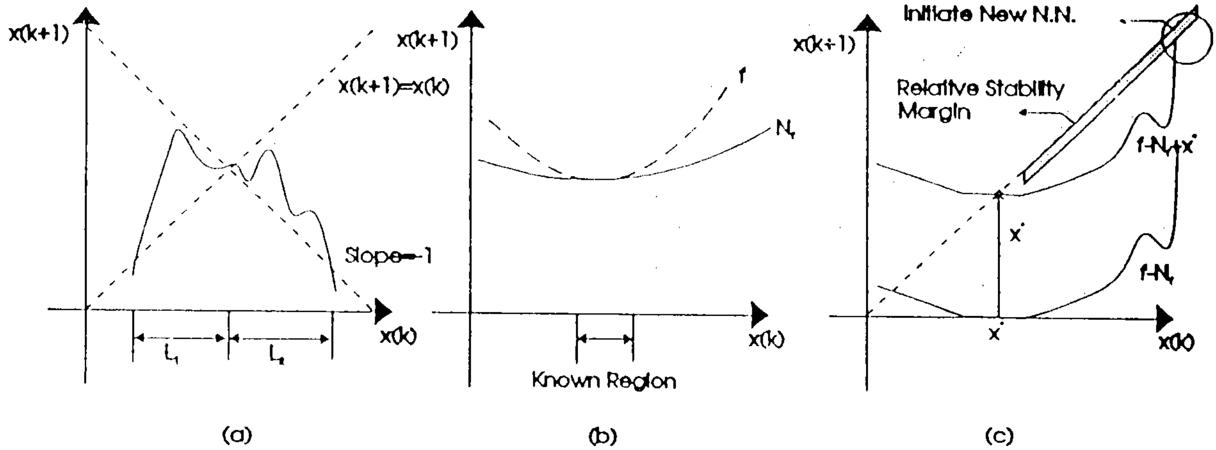

Fig. 1: Graphical Analysis

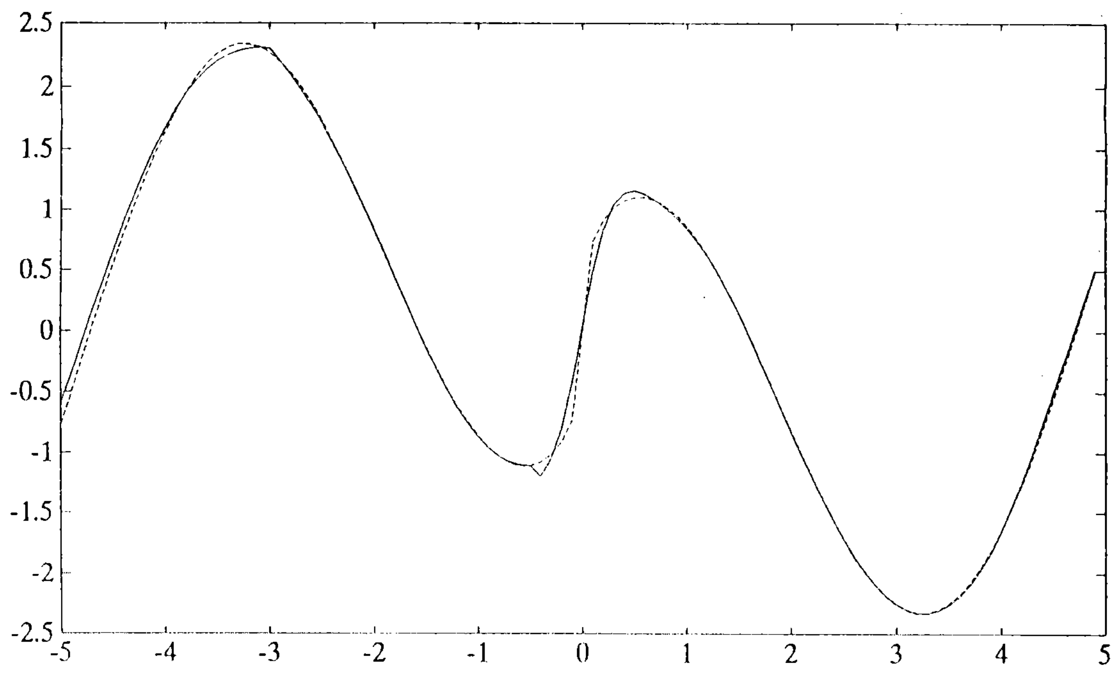

Fig. 2: $f[x(k)] = \cos[x(k)]\sqrt[3]{4x(k)}$ approximation by 3 NN

(----: desired ——: approximated)

Note : discontinuity near the joint