A NEURALLY–BASED AUTO–LOCK–ON TARGET–TRACKING SIGHT CONTROLLER FOR AIRBORNE FIRE CONTROL SYSTEM

X. G. Gao, D. Q. Chen and P. S. Zhu
Dept. of Electronic Engineering
Northwestern Polytechnical University
Xi'an, Shaanxi Province, 710072
People's Republic of China

Abstract

The design of controller in a auto–lock–on target–tracking sight control system by neural network is discussed throughout the paper. Based on expert human exemplary data, a back–propagation neural network (BPNN) is trained to learn the natural intelligence characteristics of the operator in the manual control process, and the trained network then acts as a neurocontroller by mimicking the tracking control action of the expert. It is found that the neurocontroller can successfully track fast maneuvering target, and the neurocontrol approach can provide a new and viable solution to the design of auto–lock on target–tracking sight control system.

1. Introduction

Auto–lock–on target–tracking control system can be found in many applications, and a key work to bulid up such system is the design of the controller. Classical methods available for designing controller have a common weakness, i.e., most, if not all, of them are based on a model given a priori, which is used to discribe the structural representation of the controlled process. On the other hand, human control action exhibits appealling performance. An expert human operator, for example, can cleverly analyze target maneuvers and accurately track the target. There remains the challenge to understand the human cognitive control action in executing the exciting yet straightforward task of target–tracking and use the characteristics of human control action as a guidance in the design of target–tracking controller.

The present study develops a neurocontroller for target–tracking sight control system, which can be used in the simulation study of airborne fire control system. The neurally based controller for target tracking sight control system uses back propagation neural network (BPNN) to learn the natural intelligence characteristics of an expert human operator tracking control action. The trained BPNN then replace the operator by mimicking the action.

The problem we consider here is a class of continuous one–dimensional compensatory tracking manual control task as shown diagramtically in figure 1, where the operator's problem is to make $y(t)$, the output of the controlled process, correspond as closely as possible to $r(t)$, the displayed ideal or reference, i.e., the operator observes and acts upon the error $e(t)$, equal to $r(t)-y(t)$, and drives the controlled process by means of stick or other hand or foot control to null or compensate for error.

The goal for studying the neurocontroller is to use neurocomputing techniques to identify eye–hand coordination of an expert target–tracking human operator, capture his cognition in predicting target maneurve, and furthermore, provide a class of auto–lock–on target–tracking sight control system for the simulation study of airborne fire control system.

The tasks for designing the neurocontroller involve:

- selecting typical data representing human expert tracking control action,
- selecting suitable topologies for the BPNN,
- training the BPNN to provide a neurocontroller for target–tracking sight control system,

2. Background on Back–Propagation Neural Network

Artificial neural networks [1] [2] [3] have been studied for many years in the hope of achieving human-like performance, and back-propagation neural network [4] [5] is one of the widely used neural networks. BPNN has the topology as shown in figure 2. The network consists of input layer L_0, output layer L_N and several hidden layers $L_1, \cdots, L_{N-1}$. In kth layer, there are n_k nodes. Let x_j^k denote the jth node in the kth layer, and w_{ij}^k deonte the adjustable connection between the node x_j^k and the node x_i^{k-1}, then each node output, or state, is computed as

$$x_j^k = (1 - Exp(-a_j^k)) / (1 + Exp(-a_j^k)) \qquad (1)$$

$$a_j^k = \sum_{i=1}^{n_{k-1}} W_{ij}^k x_i^{k-1} \qquad (2)$$

Back–propagation neural network belongs to the class of mapping neural network architectures and therefore the information processing function that it carries out is the approximation of a bounded mapping or function $f: A \subset R^n \rightarrow R^m$, from a compact subset A of n–dimensional Euclidean space to a bounded subset $f[A]$ of m–dimensional Euclidean space, by means of training on examples (x_1, y_1), $\cdots$, (x_m, y_m), $\cdots$ of the mapping, where $y_k = f(x_k)$. Recent theoretical works [6] [7] have rigorously proved that, even with only one hidden layer, the neural network can uniformly approximate any continuous function. The theoretical basis for modelling operator characteristics in a manual control system by the network therefore is sound. Furthermore, according to the works performed by Billings et al., [8] [9] the network can be viewed as just another class of functional representation for non–linear dynamic system.

3. Design of Neurocontroller

For the reasons mentioned above, training a BPNN to provide neurocontroller, i.e., modelling the natural intelligence characteristics of an expert human tracking action, is a problem of identification of non–linear dynamic system, and therefore the design of neurocontroller consists of the three phases as follows:

Selection of expert human exemplary data Because the human's performance as a controller depends upon a great number of variables, properties of the particular situation he finds himself in, the expert human exemplary data used to train neural network should be "rich" enough to represent the human performance in accordance with the particular task.

Selection of neural network topology A major work to design neurocontroller is to find a suitable topology for the BPNN, i.e., determine the number of layers in the network and the number of nodes in each layer. Let J be the training performance measure defined by

$$J = 1/2 \sum_{i=1}^{m} (y_i - \hat{y}_i) \tag{3}$$

where m is the total number of samples, y_i is expected network output (i.e., the exemplary data), and $\hat{y}_i$ is the actual network output, then an optimal topology for the neurocontroller is the one which, for given exemplary data, minimizes J.

Training of neurocontroller In order to train neurocontroller, we should select a suitable initial topology for the controller, and then use the back–propagation algorithm to adjust (train) the weights in the network. Theoretically, any method of non–linear optimization can be used to train the weights in the network, but back–propagation algorithm may be a simple and effective one.

The back–propagation learning algorithm which searches for a minimum of J is summaried below in Box 1, and the structural representation of modelling a human manual control action by a BPNN is shown diagramtically in figure 3.

Back–Propagation Algorithm

Step 1: Initialize w_{ij}^k, $k = 1, \cdots, N$, $j = 1, \cdots, n_k$, $i = 1, \cdots, n_{k-1}$, to small random values in the range $[-1,1]$

Stpe 2: For each data pair (x_h, y_h), $h = 1, \cdots, m$, do the following

a. Calculate the network output $\hat{y}_k$

b. Compute the errors of the output layer as
$$\delta_j^N = (y_h - \hat{y}_h)(1. - (\hat{y}_h)^2)$$
where $j = 1, \cdots, n_N$, and the errors at the internal hidden layers as
$$\delta_j^k = x_j^k (1 - (x_j^k)^2) \sum_{l=1}^{n_{k+1}} \delta_l^{k+1} w_{jl}^{k+1}$$
where $k = N-1, \cdots, 1$, $j = 1, \cdots, n_k$,

c. Adjust the weights by
$$\Delta w_{ij}^k = \eta \delta_j^k x_i^{k-1}$$
where $\eta > 0$, $k = N, \cdots, 1$, $j = 1, \cdots, n_k$, $i = 1, \cdots, n_{k-1}$

Step 3: Repeat step 2 until the error value δ_j^k, for each $j = 1, \cdots, n_k$, and each $k = 1, \cdots, N$, is either sufficiently low or zero.

Box 1

By comparing the tracking performance measure J over different network topologies, we can determine the best one that emulates the human tracking control action.

4. Simulation Results

In our simulation study, the expert human exemplary data used to train back–propagation neural network is given in figure 4, which represent the typical response of the expert to a series of step singals. The model of the controlled process, represented by Laplace transform, is $1/s$ (some examples of such control situations are automobile heading control by steering wheel for small angle deflection, aircraft pitch angle control by elevator as well as attitude control of vehicles with argumented damping).

On the basis of the data, numerous trainings were carried out using the back–propagation algorithm and different neural network topologies. From the design and trainging procedure, we have found that three–input and single hidden layer with 50 hidden nodes topology provides one of the satisfactory training performance measures, and the training result is shown in figure 5. Figure 6 shows a typical tracking action carried out by the trained back–propagation neural network, it depicts the trail of target as it was chased by the reticle of the neurocontol sight. The simulation result has shown that the basic moves of human tracking action have been learned and mimicked by the neurocontroller. Also, further simulations have shown that the neurocontroller can consistently and successfully track fast maneuvering target, some of which may be not easily tracked by human manual operation.

5. Summary

As demonstrated by the above results, back–propagation neural network is a new tool for approximating functions on the basis of examples, and it offers a new approach of information processing because of its adaptivity, ability to learn as well as massive parallelism. As a result, the neurocontrol approach provides a viable alternative exciting solution to the design of auto–luck–on target–tracking sight control system. In order to extend our approaches to more complicated tracking process, further research will be focused on the relation between the approximation ability of BPNN and the dynamics of expert human manual control action.

References

【1】 Anderson, J. A., and Rosenfeld, E., Neurocompuing: Foundations of Research, MIT Press, 1988.

【2】 Lippmann, R. P., An Introduction to Computing with Neural Nets, IEEE ASSP Magazine, April 1987.

【3】 DARPA NEURAL NETWORK STUDY, AFCEA International Press, 1988.

【4】 Rumelhart, D. E., and McClelland, J. L., Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol.Ⅰ, MIT Press, 1987.

【5】 Hecht–Nielsen, R., Theory of the Backpropa–gation Neural Network, IJCNN, 1989.

【6】 Cybenko, G., Approximations by superpositions of a sigmoidal function, Mathematics of Control, signals and Systems, 2, 1989.

【7】 Funahashi, K., On the approximate realization of continuous mappings by neural networks, Neural Networks, Vol. 2, 1989.

【8】 Leontaritis, I. J. and Billings, S. A., Input–output parametric models for non–linear systems, Part I: deterministic non–linear systems, INT.J.CONTROL, Vol. 41, 1985.

【9】 Chen, S., Billings, S. A., and Grant, P. M., Non-linear system identification using neural networks, INT.J.CONTROL, Vol. 51. No.6, 1990.

【10】 Sheridan, T. B., Ferrell, W. R., Man–Machine Systems, MIT Press paperback edition, 1981.

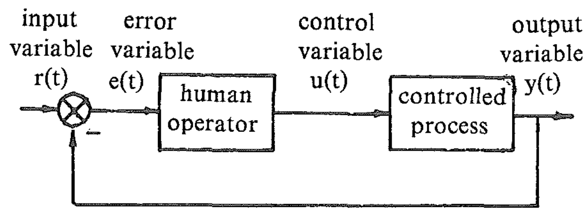

Fig.1 Compensatory control

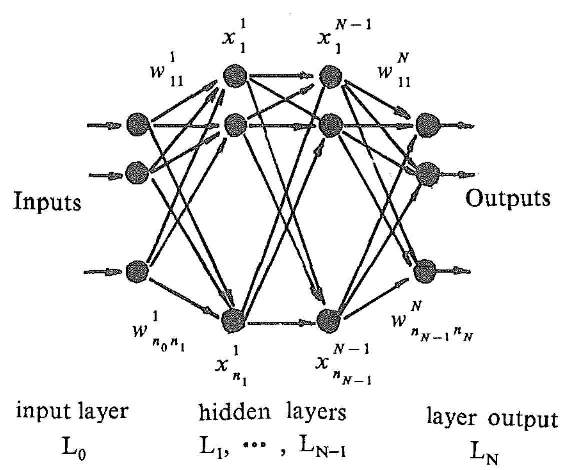

Fig.2 The structural representation of a BPNN

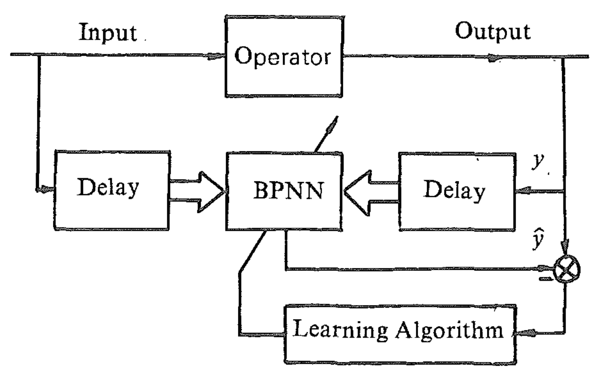

Fig.3 The structural representation of modelling a human manual control action by a BPNN

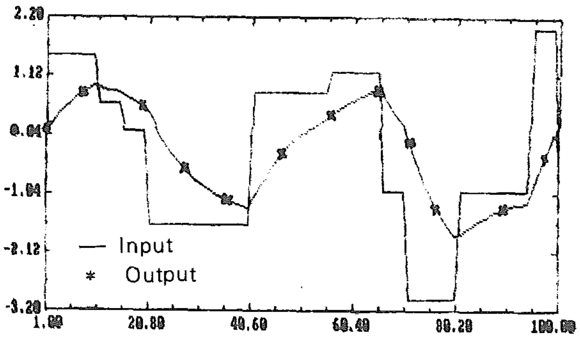

Fig.4 Expert huamn exemplary data

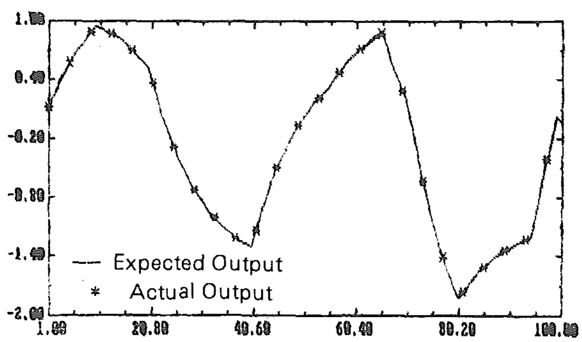

Fig.5 The training performance of the BPNN with 3–input, single hidden layer and 50 hidden nodes. J = 0.00004.

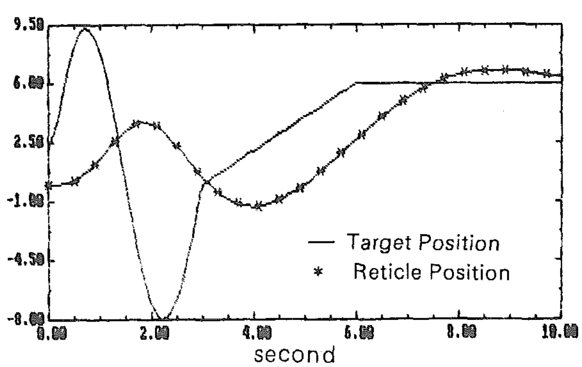

Fig.6 The simulation result of using the trained BPNN as an in–the–loop controller to chase a moving target.