A New Method of Self-Adapted Learning Mechanism Based on Neural Network

Hu Hua

School of Mathematics and Computer Science, NingXia University
Yin Chuan, NingXia, P. R. China, 750021
E-mail: huahunum@126.com

Abstract ─**This paper describes a new method to neural network model based self-adapted learning mechanism for deriving a complex system. This Learning scheme is organized based on a multi-agent-based self-adapted learning method as well as a multi-agent-based self-adapted method, while an Neural Network Decision Mechanism (NNDM) is used to identify the best agent-based self-adapted method, at each instant of time. In order to demonstrate the validity of the proposed Learning scheme, the single linear model based generalized self-adapted learning mechanism is used as a basic method. At last, the appropriate tracking performance of the proposed Learning scheme is easily outperformed in comparison with previous one.**

Keywords-self-adapted learning, multi-agent-based method, Neural Network Decision Mechanism (NNDM)

I. INTRODUCTION

Nowadays, the linear model based self-adapted learning theory has been widely used in so many applications, in the past two decades. This Learning theory guarantees that the future error signals will be methoded to zero by optimizing the specific cost function, if the linear model approximation of a complex system could accurately be identified. Based on the matter presented here, a complex system cannot actually be dealt with, while we are using the linear model based self-adapted learning theory. In order to improve the present results, a new Learning scheme is now proposed to derive a severe nonlinear and time variant system in accordance with this Learning theory. The advantage of the proposed scheme is to realize an neural network Learning method in the absence of any linear models of the system. In the Learning scheme proposed, a multi-agent-based self-adapted model-self-adapted learning method so called KeXing scheme in this paper is organized to cope with an industrial tubular heat ex-changer system as a severe nonlinear and time variant system. To realize the Learning scheme, at first, agent-based self-adapted method and subsequently agent-based self-adapted learning method need to be correspondingly realized at a specific operating point of the system. After that, the multi-agent-based self-adapted method and then the multi-agent-based self-adapted learning method are correspondingly implemented on the system to cover all the operating points. The rest of the proposed Learning algorithm is based on an neural network decision mechanism (NNDM) realization to choose the best agent-based method and also the corresponding agent-based self-adapted learning method, at each instant of time. In fact, the system behavior is fully covered via the NNDM, provided that the multi-agent-based self-adapted method as well as the multi-agent-based self-adapted learning method is suitably organized. It means that the best agent-based self-adapted method and therefore the best agent-based self-adapted learning method are accurately identified by the NNDM, at each instant of time, while the system parameters are abruptly varied with respect to time [1–3]. Based on the Learning strategy, the new self-adapted learning could be so flexible to derive a highly nonlinear system in association with the traditional linear model based self-adapted learning theory.

The rest of the paper is organized as follows. The proposed Learning scheme is given in Section 2. The simulation results and concluding remark are finally de-scribed in Sections 3 and 4, respectively.

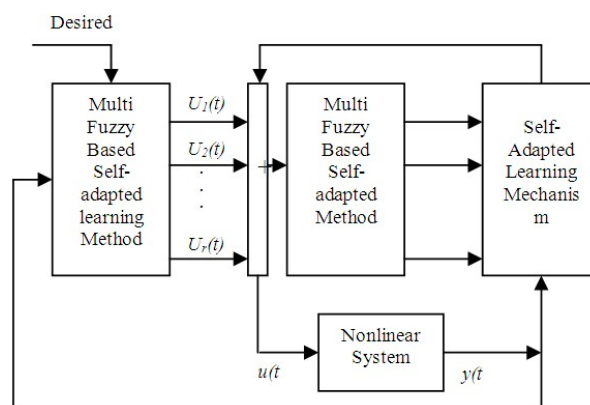## II. THE PROPOSED LEARNING SCHEME

Figure 1. The scheme of the proposed learning strategy

The proposed Learning scheme, as shown in Figure 1 is organized based on a multi-agent-based Learning method as well as a multi-agent-based self-adapted method and also an neural network decision mechanism (NNDM) to derive a severe nonlinear and time variant system. This Learning scheme is realized based on the self-adapted learning theory to calculate the future Learning action by optimizing the specific quadratic cost function, given by:

$$J = \sum_{i=N_1}^{N_2} (y(t+i) - w(t+i))^2 + \sum_{i=1}^{N_u} \lambda_i \Delta u(t+i-1)^2 \qquad (1)$$

where N_u is given as the Learning horizon, $N_2 - N_1 + 1$ is given as the prediction horizon, i is given as step ahead predictor, $y(t)$ is given as the system output, $w(t)$ is given as the desired trajectory of the future output, $\Delta u(t)$ is given as the manipulated variable and finally λ is given as the Learning weight factor. In order to realize the Learning strategy, the number of models to cover the different operating points of the system is strongly needed and consequently the corresponding Learning method needs to be designed for each one of them. If the operating points of the system are not well chosen, the outcomes corresponding to the required tracking performance cannot actually be acquired.

In the proposed Learning scheme, the multi-agent-based self-adapted method is organized in accordance with the several agent-based self-adapted method, where each one of them is realized at a specific operating point of the system. And then the multi-agent-based self-adapted learning method is organized in accordance with the several agent-based self-adapted learning method, where each one of them is realized in line with the corresponding agent-based self-adapted method. The proposed Learning scheme is an appropriate method to cope with a complex system, when we are using the self-adapted learning theory. In fact, the proposed Learning scheme operates in multiple operating points, which may change from one to another abruptly. On the other hand, the operating points regions of a nonlinear and time variant system have been extended, where a linear fixed model may not really lead to the expected performance. In correspondence with the proposed Learning scheme, r agent-based self-adapted method are used to identify the complex system at r operating points and subsequently r agent-based self-adapted learning method are used to derive the system presented at corresponding operating points. In designing the Learning scheme proposed, $u(t)$ is given as the Learning action of the multi-agent-based self-adapted learning method, given by

$$u(t) = \sum_{k=1}^{r} P_k(t) u_k(t) \qquad (2)$$

Where

$$\sum_{k=1}^{r} P_k(t) = 1 \qquad (3)$$

and r is given as the number of the operating points of the system. Hereinafter, $P_k(t)$ and $u_k(t)$ are given as the appropriate weight and the Learning action of the kth local agent-based self-adapted learning method, respectively.

Realization of the agent based self-adapted learning method, the agent based self-adapted method and finally the NNDM are now described in the proceeding sections.

A   The Agent-Based Self-adapted learning Method

The agent-based self-adapted learning method is given as one of the local Learning in the multi-agent-based self-adapted learning method, in the Learning scheme presented. The main concept in realizing the agent-based self-adapted learning method is described by the following agent rule base

$$Rule.i : IF.w(t+N).is.D_j^i.and$$
$$.y_{mk}(t+N).is.E_j^i$$
$$THEN.\Delta u_k(t).is.F_j^i.$$

Where i is given as the rule number, j is given as the agent set number and finally D_j^i, E_j^i, F_j^i are given as the input-output agent sets. In this Learning strategy, $w(t+N)$ denote the kth desired trajectory of the future output, $y_{mk}(t+N)$ denotes the agent-based self-adapted method output and finally $\Delta u_k(t)$ denotes the manipulated variable of the kth agent-based self-adapted learning method, which is directly related to the following future error signal:

$$e_k(t+N) = w(t+N) - y_{mk}(t+N) \qquad (4)$$

The agent set of the agent-based self-adapted learning method could easily be acquired by the following

$$u_k(t) = \frac{1}{1-z^{-1}} \Delta u_k(t) \qquad (5)$$

1) Agent Sets Tuning

In the Learning scheme proposed, the agent sets used in realizing the multi-agent-based self-adapted learning mechanism are tuned in an offline manner. In fact, the proposed tuning algorithm which is already based on the GA algorithm is realized for all the agent-based self-adapted learning method. With this purpose, these agent sets are first parameterized, using $a_i, b_i, c_i, d_j; i=1,2,3, j=1,3$. In this way, some populations are chosen, while the first population has nine chromosomes $(V_j; j=1,2,K,9)$, which is now realized by the following forms

$$V_j = \{g_{1j}, g_{2j}, g_{3j}, g_{4j}, g_{5j}, g_{6j}, g_{7j}, g_{8j}, g_{9j}, g_{10j}, g_{11j}\}$$

where we could have the following

$$\begin{cases} g_{1j}=a_1+\delta_{a_{1j}} & g_{2j}=a_2+\delta_{a_{2j}} & g_{3j}=a_3+\delta_{a_{3j}} \\ g_{4j}=b_1+\delta_{b_{1j}} & g_{5j}=b_2+\delta_{b_{2j}} & g_{6j}=b_3+\delta_{b_{3j}} \\ g_{7j}=c_1+\delta_{c_{1j}} & g_{8j}=c_2+\delta_{c_{2j}} & g_{9j}=c_3+\delta_{c_{3j}} \\ g_{10j}=d_1+\delta_{d_{1j}} & g_{11j}=d_3+\delta_{d_{3j}} \end{cases} \qquad (6)$$

In such a way, with a specific mutation rate; $P_c =6.0$, the

2nd population have five chromosomes and the 3rd population have three chromosomes. Also the 4th population has two chromosomes and the 5th populations have one chromosome. Obviously, the agent sets parameters could be optimized to appropriate values by the superior genes.

B The Agent-Based Self-adapted Method

The agent-based self-adapted method is realized as one of models in the multi-agent-based self-adapted method. In case of this method, $u(t - \tau)$ is given as a valid input of the system, while τ is given as its delay. Also, $y_{mk}(t)$ is given as the kth agent-based model output. The schematic of this method is shown in Figure 2. Here, the agent-based model #N could be placed in sequence with the agent-based model #(N-1) to provide the agent-based self-adapted method. In fact, the prediction of model output could be improved, while the number of the present agent-based models is increased. The agent rule based for this method is now given as follows
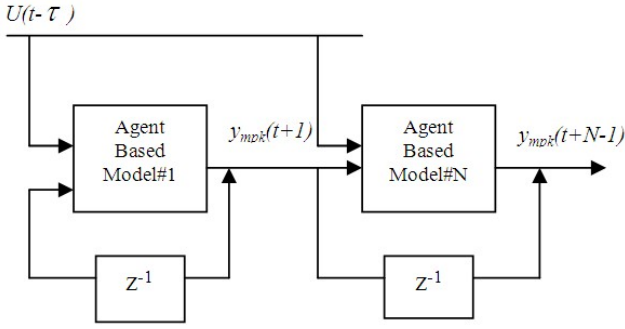

Figure 2. The Agent-based self-adapted method

$$Rule\, i: \quad IF.u(t - \tau)\, is.A_j^i.and$$
$$.y_{mk}.(t-1).is.B_j^i \quad THEN.y_{mk}(t).is.C_j^i$$

Where i is given as the rule number, j is given as the agent set number and finally A_j^i, B_j^i, C_j^i are given as the input-output agent sets.

C The NNDM Realization

In the Learning scheme proposed, the best agent-based self-adapted method in the multi-agent-based self-adapted method is accurately identified by using the NNDM, at each instant of time. Now, by assuming only three agent-based self-adapted method, for simplicity, the NNDM can be summarized as follows

Defining the specific performance indices; $J_i(t); i = 1, 2, 3,$ for each one of the agent-based self-adapted method, correspondingly, given by:

$$J_i(t) = a e_i^2(t) + \beta \int_0^t e^{-\lambda(t-\tau)} e_i^2(\tau) d\tau, a \geq 0, \beta, \lambda > 0 \quad (7)$$

Where $\qquad e_i(t) = y(t) - y_i(t) \qquad (8)$

In the NNDM proposed. α, β and λ are given as the weighting factors on the instantaneous measures, the long term accuracy and the forgetting factor, respectively.

. Defining the input-output agent sets.

. Defining the agent rules based, which can be given by the following

Rule#1:
$$IF.J_1(t).is.S_j^1.AND.J_2(t).is.L_j^1.AND.J_3(t).is.L_j^1 \quad THEN.P_1(t).is.L_P^1$$

Rule#2:
$$IF.J_1(t).is.L_j^2.AND.J_2(t).is.S_j^2.AND.J_3(t).is.L_j^2 \quad THEN.P_2(t).is.L_P^2$$

Rule#3:
$$IF.J_1(t).is.L_j^3.AND.J_2(t).is.L_j^3.AND.J_3(t).is.S_j^3 \quad THEN.P_3(t).is.L_P^3$$

III. SIMULATION RESULTS

In order to verify the applicability of the proposed Learning scheme so called KeXing scheme in this paper, an industrial tubular heat exchanger system is used to Learning [1–5]. To analyze the proposed Learning scheme in comparison with other Learning techniques which are fully investigated by the Authors in the several publications, this complex system is realized the same specification as these research works, where all the research papers could easily be found by the readers in detail.

With this purpose, we have carried out the KeXing scheme for this system using the Simulink-Matlab programming language, while the multi-agent-based self-adapted method is organized based on the three agent-based self-adapted method and subsequently the multi-agent-based self-adapted learning method is correspondingly organized based on the three agent-based self-adapted method. In these simulations, the single model linear generalized self-adapted learning entitled KeXing-2 scheme is used as a basic method here. At first, the desired trajectory of the future output is given as 60°C at 5 sec., while it is abruptly varied to 55°C and also 40°C at 54 sec. and 104 sec., respectively. In these simulations, the KeXing-2 scheme is carried out by the following Learning parameters

$$\begin{cases} N_2 - N_1 + 1 = 3 \\ N_u = 3 \end{cases} \quad (9)$$

And the SELF-ADAPTED LEARNING model is also given by self-adapted method outputs are given as below. Here the Learning actions are all multiplied by its weights. As is easily obvious from the outcomes, the tracking performance of the KeXing scheme is quite outperformed with respect to its basic method A

$$A(q^{-1})y(k) = B(q^{-1})u(k-1) + \frac{e(k)}{\Delta(q^{-1})} \quad (10)$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + K + a_p q^{-P}; p = 4 \quad (11)$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + K\, b_m q^{-m}; m = p \quad (12)$$

IV. CONCLUSIONS

A main problem in realizing the linear model based self-adapted learning theory to derive a severe nonlinear system has been thoroughly solved, in this paper. Based on the proposed Learning strategy so called HuaXing scheme by the Authors, a nonlinear system must first be represented via the multi-agent-based self-adapted method. And then the multi-agent-based self-adapted learning method is correspondingly realized to derive the nonlinear system presented.

TABLE 1. THE COEFFICIENTS OF THE KEXING MODEL

j	aj	bj
1	-0.942	0.2606e-3
2	-0.415	0.3628e-3
3	0.0068	0.5375e-3
4	0.4312	0.1910e-3

Besides, the neural network decision mechanism (NNDM) is used to identify the best agent-based self-adapted method and the corresponding agent-based self-adapted learning method, at each instant of time. The advantage of the proposed Learning scheme over other related Learning techniques is to realize the strategy in the absence of any linear model approximation of the complex system. It points out that the proposed KeXing scheme is the generalized version of the traditional model based self-adapted learning theory. Also the present Learning structure aims us to develop it for all the highly nonlinear and time variant systems in both real and academic environments.

REFERENCES

[1] Berners-Lee, T. (1996). The world wide web: Past, present and future. IEEE Computer Special Issue IEEE Computer Society, 29(10), 69–77.

[2] Jeong, D., et al. (2008). View-based storage-independent model for SPARQL-to-SQL translation algorithms in semantic grid environment. In The 11th IEEE international conference on computational science and engineering (CSE'08), pp. 381–386.

[3] Kim, J.-D., et al. (2008). Jena storage plug-in providing an improved query processing performance for semantic grid environment. In The 11th IEEE international conference on computational science and engineering (CSE'08), pp. 398–398.

[4] W3C. (2009). Resource description framework. http://www.w3.org/RDF/, 2009.

[5] Manola,F.,& Miller,E.(2004).RDF Primer, W3C Recommendation. http://www.w3.org /TR/rdf-primer/.

[6] Beckett, D. (2004). RDF/XML Syntax specification (Revised), W3C recommendation. http://www. w3.org/TR/rdf-syntax-grammar/.123 144 D. Jeong et al.

[7] Xue, G., Pan, Q., & Li, M. (2007). A new semantic-based query processing architecture. In 2007 International conference on parallel processing workshops (ICPPW 2007), p. 63.

[8] Ibrahim, I.K., et al. (2005). A semantic solution for data integration in mixed concept networks. Computer Communications, 28(13), 1564–1574. Elsevier.

[9] Goodwin, C., & Russomanno, D. J. (2007). An ontology-based concept network prototype environment. In The 5th international conference on information processing in concept networks (Poster), pp. 1–2.

[10] Imai, M., et al. (2006). Semantic concept network for physically grounded applications. In The 9th international conference on control, automation, robotics and vision (ICARCV 2006), pp. 1637–1642.