行政院國家科學委員會專題研究計畫　期中進度報告

子計畫四：高可靠度跟車系統之設計(1/2)

計畫類別：整合型計畫
計畫編號：NSC94-2213-E-009-126-
執行期間：94 年 08 月 01 日至 95 年 07 月 31 日
執行單位：國立交通大學電機與控制工程學系(所)

計畫主持人：李祖添

計畫參與人員：蔣欣翰,賴幼仙

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 95 年 5 月 26 日

行政院國家科學委員會補助專題研究計畫 □ 成 果 報 告
■期中進度報告

先進車輛控制及安全系統之設計與模擬-子計畫四:高可靠度跟車系統之設計(1/2)

計畫類別:□ 個別型計畫　■ 整合型計畫
計畫編號:NSC94－2213－E－009－126
執行期間:　2005 年 8 月 1 日至 2006 年 7 月 31 日

計畫主持人:李祖添 國家講座
計畫參與人員: 蔣欣翰, 賴幼仙

成果報告類型(依經費核定清單規定繳交):■精簡報告　□完整報告

本成果報告包括以下應繳交之附件:
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式:除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外,得立即公開查詢
□涉及專利或其他智慧財產權,□一年■二年後可公開查詢

執行單位:交通大學電機與控制學系

中　華　民　國　95 年　5　月　31 日

先進車輛控制及安全系統之設計與模擬

子計畫四：高可靠度跟車系統之設計(1/2)

計畫編號: NSC 94-2213-E-009-126
執行期限:94/08/01 - 95/07/31

主持人：李祖添 講座教授
參與人員：蔣欣翰, 賴幼仙
執行單位：國立交通大學電機與控制系

中文摘要

近幾年來隨著車輛數量的大幅成長，駕駛員所可能遇到的行車疲勞之負擔與行車安全之威脅也相對地越來越高，所以車輛的行車安全考量問題就顯得日益重要。本子計畫開發一自我建構模糊類神經網路控制(self-structuring fuzzy neural network control)法則，讓所有受控車輛都可以與前車保持固定車距，如此將能夠提高道路運輸效率並降低交通事故發生，所開發設計之控制法則包含一類神經控制器(neural control)與強健控制器(robust control)，其中類神經控制器使用一具有自我網路架構建構能力之模糊類神經網路，主要用來線上學習近似一理想控制法則，強健控制器主要用來克服類神經網路近似誤差所造成之影響。而所開發具有自我網路架構建構能力之模糊類神經網路，整個網路學習部分包誇類神經網路架構學習部分與類神經網路參數學習部分，且整個所設計之控制器參數均由李亞普諾夫穩定理論所推導之自我學習法則調整。最後，本研究子計畫經由模擬結果顯示所提出來的控制系統，除了可以得到良好且安全的自動跟車防撞效能，而且網路架構可以自我建構出所需之最少類神經元數目。

關鍵詞：自動跟車、網路架構自我建構、模糊類神經網路、強健控制

Abstract

Transportation technology is one of the most influential areas on the human life. Therefore, researchers have been involving in wide scope of related research activities aiming to enhance efficiency, comfort, and safety of transportation systems. Due to the ever growing number of vehicles on the roads, urban highways are congested and need additional capacity. Upon entering the intelligent automated highway system, the longitudinal control of the car-following collision prevention system will drive a vehicle along the fully automated highway. To achieve this objective, this subproject proposes a self-structuring fuzzy neural network control (SSFNNC) system for the vehicle platoons system. The proposed SSFNNC system is comprised of a neural controller and a robust controller. The neural controller using a self-structuring fuzzy neural network (SFNN) is designed to mimic an ideal controller, and the robust controller is designed to achieve L^2 tracking performance with desired attenuation level. The adaptation laws of the control system are derived in the sense of Lyapunov stability theorem, so that the stability of the control system can be guaranteed. Moreover, the learning phase of the SFNN is considered about structure and parameter learning phases. The structure learning phase possesses the ability of online generation and elimination of fuzzy rules to achieve optimal neural structure, and the parameter learning phase adjusts the interconnection weights of neural network to achieve favorable approximation performance. Finally, some simulation scenarios are examined to verify the effectiveness of the proposed SSFNNC system.

keywords：car-following, self-structuring learning phase, fuzzy neural network, robust control

I. Introduction

Transportation technology is one of the most influential areas in the human life. Many researchers have been involved in a wide scope of related research activities aiming to enhance efficiency, comfort, and safety of transportation systems. Among them, the traffic congestion is a global problem. One solution of this problem is to increase the traffic flow by decreasing the inter-vehicular spacing. To achieve this objective, car following for traffic safety has been an active area of research [1, 2]. However, human driving involves reaction time, delay, and human error that affect safe driving adversely. One way to eliminate human error and delay in vehicle following is to integrate an automated car-following control system in the driving system. Inside the vehicle platoon, all the vehicles follow the leading vehicle with a small intraplatoon separation. To enable this, each vehicle will be equipped with control systems which coordinate control between the brakes, engine and steering subsystems.

The neural-network-based control techniques have been used as an alternative design method for identification and control of dynamic systems [3-6]. The key element of the neural network is the capability of approximating mapping through choosing adequately learning method. Based on this property, the neural-network-based controllers have been developed to compensate for the effects of nonlinearities and system uncertainties, so that the stability, convergence and robustness of the control system can be improved. Although the control performances in [3-6] are acceptable, the learning algorithm only includes the parameter learning phase, and they have not considered the structure learning phase of the neural network. If the number of the hidden neurons is chosen too large, the computation loading is heavy so that they are not suitable for online practical applications. If the number of the hidden neurons is chosen too small, the learning performance may be not good enough to achieve desired control performance.

To solve this problem, several self-structure neural-network-based control have been developed [7-9]. However, some of them use the gradient descent method to derive the parameter learning algorithms which can’t guarantee the system stability. Some of them use the Lyapunov function to derive the parameter learning algorithms; however, the neurons in the hidden layer only can automatic spilt-up to achieve satisfactory performance without considering how to eliminate the neuron.

This paper combines the advantages of the self-structuring fuzzy neural network (SFNN) approach and adaptive control technique to develop an intelligent longitudinal control for vehicle platoon systems. The proposed self-structuring fuzzy neural network control (SSFNNC) system is comprised of a neural controller and a robust controller. The neural controller exploiting a SFNN is the principal controller. The SFNN is used to online estimate the ideal controller with the structure and parameter learning phases, simultaneously. The structure learning phase possesses the ability of online generation and elimination of fuzzy rules to achieve optimal neural structure, and the parameter learning phase adjusts the interconnection weights of neural network to achieve favorable approximation performance. The robust controller is used to achieve L_2 tracking performance with desired attenuation level. Moreover, all the parameter learning algorithms are derived based on the Lyapunov function, thus the system stability can be guaranteed. Finally, two simulation scenarios (one-vehicle following scenario and multi-vehicles following scenario) are examined to verify the effectiveness of the proposed SSFNNC system.

II. Problem Formulation

A. Platoon Dynamic

Figure 1 describes a platoon of N vehicles following a lead vehicle on a straight lane of highway. The position of the rear bumper of the ith vehicle with respect to a fixed reference point O on the road is denoted by x_i. The position of the lead vehicle’s rear bumper with respect to the same fixed reference point is denoted by x_l. From the platoon configuration, the spacing error e_i can be written as

$$e_i = \begin{cases} x_l - x_1 - H_1 & for\ i = 1 \\ x_{i-1} - x_i - H_i, & for\ i = 2,3,...,N \end{cases}. \tag{1}$$

where H_i denotes the safety spacing of the ith vehicle in the platoon. In the following, the variables and parameters are assumed to be associated with the ith vehicle, unless subscripts indicate otherwise.

B. Vehicle Model

The dynamics of the car following system for the vehicle in a platoon are modeled as follows

$$\dot{\xi} = \frac{1}{\tau}(-\xi + u) \tag{2}$$

$$\ddot{x} = \frac{1}{m}(\xi - K_d \dot{x}^2 - d_m) \qquad (3)$$

where ξ denotes the driving force produced by the vehicle engine; τ denotes the engine time lag to the vehicle; u denotes the throttle command input to the vehicle's engine (if $u > 0$, then it represents a throttle input and if $u < 0$, it represents a brake input); m denotes the mass of the vehicle; K_d denotes the aerodynamic drag coefficient for the vehicle; and d_m denotes the vehicle's mechanical drag. Equation (2) represents the vehicle's engine dynamics, and (3) represents Newtons's second law applied to the vehicle modeled as a particle of mass. Differentiating both sides of (3) with respect to time and substituting the expression for $\dot{\xi}$ in term of v and a, yields

$$\dot{a} = f(v,a) + \mathrm{g}u \qquad (4)$$

where $f(v,a) = \frac{-1}{\tau}\left[a + \frac{K_d}{m}v^2 + \frac{d_m}{m}\right] - \frac{2K_d}{m}va$ is a nonlinear function, $\mathrm{g} = \frac{1}{m\tau}$ is a positive constant, v denotes the velocity of the vehicle, and a denotes the acceleration of the vehicle.

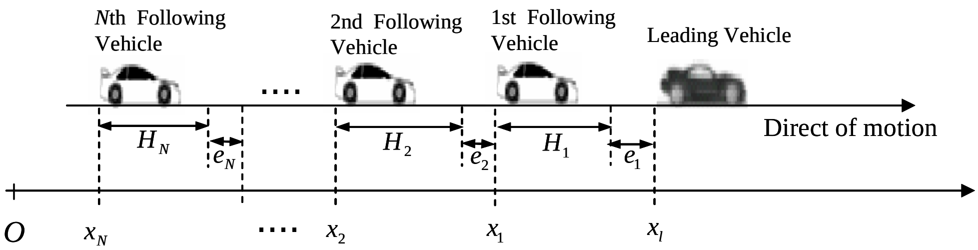


Fig. 1. Configuration of car-following platoon.

III. Control Algorithm Design

The control objective is to design a control system such that the tracking error e can be driven to zero. Assume that the parameters of the platoon system in (4) are well known, an ideal controller of the following vehicle can be constructed as [10]

$$u^* = \mathrm{g}^{-1}(-f + x_{i-1}^{(3)} + k_3\ddot{e} + k_2\dot{e} + k_1 e). \qquad (5)$$

Substituting (5) into (4), gives the following equation

$$e^{(3)} + k_3\ddot{e} + k_2\dot{e} + k_1 e = 0. \qquad (6)$$

If k_1, k_2 and k_3 are chosen to correspond to the coefficients of a Hurwitz polynomial, then it implies $\lim_{t\to\infty} e = 0$. However, the system dynamics f and g always cannot be precisely obtained in the real-time practical applications, thus the ideal controller u^* in (5) is always unachievable.

A. Structure of SSFNN

A self-structuring fuzzy neural network (SFNN) is shown in Fig. 2, which is comprised of the input, the membership, the rule and the output layers. The interactions for the layers are given as follows.

Layer 1 - Input layer: For every node *i* in this layer, the net input and the net output are represented as

$$net_i^1 = x_i^1 \qquad (7)$$

$$y_i^1 = f_i^1\left(net_i^1\right) = net_i^1, \quad i = 1,2,\ldots,l \qquad (8)$$

where x_i^1 represents the *i*-th input to the node of layer 1.

Layer 2 - Membership layer: In this layer, each node performs a membership function and acts as an element for membership degree calculation, where the Gaussian function is adopted as the membership function. For the *j*-th node, the reception and activation functions are written as

$$net_j^2 = -\frac{\left(x_i^2 - m_{ij}\right)^2}{\left(\sigma_{ij}\right)^2} \qquad (9)$$

$$y_j^2 = f_j^2\left(net_j^2\right) = \exp\left(net_j^2\right), \quad j = 1,2,...,m \tag{10}$$

where m_{ij} and σ_{ij} are the mean and standard deviation of the Gaussian function in the *j*-th term of the *i*-th input linguistic variable x_i^2, respectively, and m is the total number of the linguistic variables with respect to the input nodes.

Layer 3 - Rule layer: Each node *k* in this layer is denoted by $\prod$, which multiplies the incoming signals and outputs the result of the product. For the *k*-th rule node

$$net_k^3 = \prod_j x_j^3 \tag{11}$$

$$y_k^3 = f_k^3\left(net_k^3\right) = net_k^3, \quad k = 1,2,...,n \tag{12}$$

where x_j^3 represents the *j*-th input to the node of layer 3.

Layer 4 - Output layer: The single node *o* in this layer is labeled as Σ, which computes the overall output as the summation of all incoming signals

$$net_o^4 = \sum_k w_k^4 x_k^4 \tag{13}$$

$$y_o^4 = f_o^4\left(net_o^4\right) = net_o^4 \tag{14}$$

where the link weight w_k^4 is the output action strength associated with the *k*-th rule, x_k^4 represents the *k*-th input to the node of layer 4, and y_o^4 is the output of the SOFNN. For ease of notation, by defining vectors $\mathbf{m}$ and $\boldsymbol{\sigma}$ collecting all parameters of SOFNN, the output of the SOFNN can be represented in a vector form

$$y_o^4 = \mathbf{w}^T \boldsymbol{\Phi}(\mathbf{m}, \boldsymbol{\sigma}) \tag{15}$$

where $\mathbf{w} = \left[w_1^4\ w_2^4 ... w_n^4\right]^T$ and $\boldsymbol{\Phi} = \left[x_1^4\ x_2^4 ... x_n^4\right]^T = [\Phi_1\ \Phi_2 ... \Phi_n]^T$

B. Structure Learning of SFNN

The first step in the structure learning phase is to determine whether or not to add a new node (membership function) in layer 2 and the associated fuzzy rule in layer 3, respectively. In the rule generating process, the mathematical description of the existing rules can be expressed as a cluster. Since one cluster formed in the input space corresponds to one potential fuzzy logic rule, the firing strength of a rule for each incoming data x_i^1 can be represented as the degree that the incoming data belong to the cluster. The firing strength obtained from (11) is used as the degree measure

$$\beta_k = y_k^3, \quad k = 1,2,...,n(t) \tag{16}$$

where $n(t)$ is the number of the existing rules at the time *t*. According to the degree measure, the criterion of generating a new fuzzy rule for new incoming data is described as follows. Find the maximum degree $\beta_{\max}$ defined as

$$\beta_{\max} = \max_{1 \le k \le n(t)} \beta_k . \tag{17}$$

It can be observed that if the maximum degree $\beta_{\max}$ is smaller as the incoming data is far away the existing fuzzy rules. If $\beta_{\max} \le \beta_{th}$ is satisfied, where $\beta_{th} \in (0,1)$ a pre-given threshold, then a new membership function is generated. The mean and the standard deviation of the new membership function and the output action strength are selected as follows:

$$m_i^{new} = x_i^1 \tag{18}$$

$$\sigma_i^{new} = \sigma_i \tag{19}$$

$$w^{new} = 0 \tag{20}$$

where x_i is the new incoming data and σ_i is a pre-specified constant.

Next the structure learning phase is considered to determine whether or not to eliminate the existing fuzzy rules which are inappropriate. A significance index is determined for the importance of the *k*-th rules can be given as follows

$$I_k(t+1) = \begin{cases} I_k(t)\exp(-\tau), & if\ \beta_k < \rho \\ I_k(t) & ,\ if\ \beta_k \ge \rho \end{cases}, \quad k = 1, ,2,..., n(t) \tag{21}$$

where I_k is the significance index of the *k*-th rule its initial value is 1, ρ is the elimination threshold value and τ is the elimination speed constant. The proposed elimination algorithm is derived from the observation that if the significance index gets fading when the rule firing weight β_k is smaller than elimination threshold value ρ.

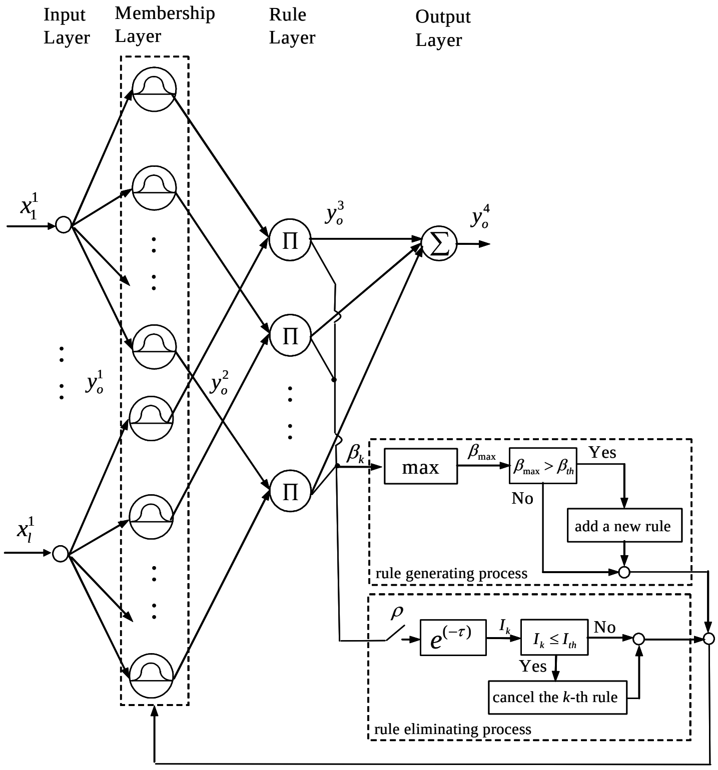


Fig. 2. Self-structuring fuzzy neural network.

C. SSFNNC Design

The proposed SSFNNC system, comprised of a neural controller and a robust controller, for the vehicle platoon is shown in Fig. 3, where a tracking error index is defined as

$$s = \ddot{e} + k_3\dot{e} + k_2 e + k_1\int_0^t e\,d\tau\,. \tag{22}$$

The control law of the intelligent longitudinal controller is taken as

$$u = u_{nc} + u_{rc} \tag{23}$$

where u_{nc} is the neural controller and u_{rc} is the robust controller. By substituting (23) into (4), it is revealed that

$$\dot{a} = f + \mathrm{g}(u_{nc} + u_{rc})\,. \tag{24}$$

Multiplying (5) with g_i^{-1}, adding to (20) and using (7), the error equation can be obtained as

$$\dot{s} = g\,(u^* - u_{nc} - u_{rc})\,. \tag{25}$$

By the universal approximation theorem, an optimal SOFNN can be designed to approximate the controlled system dynamics, such that [11]

$$u^* = u_{nc}^* + \Delta = \mathbf{w}^{*T}\mathbf{\Phi}^*(\mathbf{m}^*,\boldsymbol{\sigma}^*) + \Delta \tag{26}$$

where Δ is the approximation error, $\mathbf{w}^*$ and $\mathbf{\Phi}^*$ are the optimal parameter vectors of $\mathbf{w}$ and $\mathbf{\Phi}$, respectively, and $\mathbf{m}^*$ and $\boldsymbol{\sigma}^*$ are the optimal parameters of $\mathbf{m}$ and $\boldsymbol{\sigma}$, respectively. Let the number of optimal neurons be n^* and the neurons be divided into two parts. The first part contains n neurons which are the activated part and the secondary part contains $n^* - n$ neurons which do not exist yet. Thus, the optimal weights $\mathbf{w}^*$, $\mathbf{\Phi}^*$, $\mathbf{m}^*$ and $\boldsymbol{\sigma}^*$ are classified in two parts such as

$$\mathbf{w}^* = \begin{bmatrix}\mathbf{w}_\mathbf{a}^*\\ \mathbf{w}_\mathbf{i}^*\end{bmatrix},\ \ \mathbf{\Phi}^* = \begin{bmatrix}\mathbf{\Phi}_\mathbf{a}^*\\ \mathbf{\Phi}_\mathbf{i}^*\end{bmatrix},\ \ \mathbf{m}^* = \begin{bmatrix}\mathbf{m}_\mathbf{a}^*\\ \mathbf{m}_\mathbf{i}^*\end{bmatrix}\ \text{ and }\ \boldsymbol{\sigma}^* = \begin{bmatrix}\boldsymbol{\sigma}_\mathbf{a}^*\\ \boldsymbol{\sigma}_\mathbf{i}^*\end{bmatrix} \tag{27}$$

where $\mathbf{w}_\mathbf{a}^*$, $\mathbf{\Phi}_\mathbf{a}^*$, $\mathbf{m}_\mathbf{a}^*$ and $\boldsymbol{\sigma}_\mathbf{a}^*$ are activated parts, and $\mathbf{w}_\mathbf{i}^*$, $\mathbf{\Phi}_\mathbf{i}^*$, $\mathbf{m}_\mathbf{i}^*$ and $\boldsymbol{\sigma}_\mathbf{i}^*$ are inactivated parts, respectively. Since these optimal parameters are unobtainable, a SFNN is defined as

$$\hat{u}_{nc} = \hat{\mathbf{w}}_\mathbf{a}^T \hat{\mathbf{\Phi}}_\mathbf{a}(\hat{\mathbf{m}}_\mathbf{a}, \hat{\boldsymbol{\sigma}}_\mathbf{a}) \tag{28}$$

where $\hat{\mathbf{w}}_\mathbf{a}$, $\hat{\mathbf{\Phi}}_\mathbf{a}$, $\hat{\mathbf{m}}_\mathbf{a}$ and $\hat{\boldsymbol{\sigma}}_\mathbf{a}$ are the estimated values of $\mathbf{w}_\mathbf{a}^*$, $\mathbf{\Phi}_\mathbf{a}^*$, $\mathbf{m}_\mathbf{a}^*$ and $\boldsymbol{\sigma}_\mathbf{a}^*$, respectively. Define the estimated error $\tilde{u}$ as

$$\begin{aligned}\tilde{u} &= u^* - \hat{u}_{nc} \\ &= \mathbf{w}_\mathbf{a}^{*T}\mathbf{\Phi}_\mathbf{a}^* + \mathbf{w}_\mathbf{i}^{*T}\mathbf{\Phi}_\mathbf{i}^* - \hat{\mathbf{w}}_\mathbf{a}^T\hat{\mathbf{\Phi}}_\mathbf{a} + \Delta \\ &= \tilde{\mathbf{w}}_\mathbf{a}^T\hat{\mathbf{\Phi}}_\mathbf{a} + \hat{\mathbf{w}}_\mathbf{a}^T\tilde{\mathbf{\Phi}}_\mathbf{a} + \tilde{\mathbf{w}}_\mathbf{a}^T\tilde{\mathbf{\Phi}}_\mathbf{a} + \mathbf{w}_\mathbf{i}^{*T}\mathbf{\Phi}_\mathbf{i}^* + \Delta\end{aligned} \tag{29}$$

where $\tilde{\mathbf{w}}_\mathbf{a} = \mathbf{w}_\mathbf{a}^* - \hat{\mathbf{w}}_\mathbf{a}$ and $\tilde{\mathbf{\Phi}}_\mathbf{a} = \mathbf{\Phi}_\mathbf{a}^* - \hat{\mathbf{\Phi}}_\mathbf{a}$. Some adaptive laws will be proposed to on-line tune the mean and standard deviation of the Gaussian function of the SFNN to achieve favorable estimation of the dynamic function. The Taylor expansion linearization technique is employed to transform the nonlinear radial basis function into a partially linear form, i.e.

$$\tilde{\mathbf{\Phi}}_\mathbf{a} = \mathbf{A}^T\tilde{\mathbf{m}}_\mathbf{a} + \mathbf{B}^T\tilde{\boldsymbol{\sigma}}_\mathbf{a} + \mathbf{h} \tag{30}$$

where $\mathbf{A} = \left[\frac{\partial \Phi_1}{\partial \mathbf{m}_\mathbf{a}} \cdots \frac{\partial \Phi_n}{\partial \mathbf{m}_\mathbf{a}}\right]\Big|_{\mathbf{m}_\mathbf{a}=\hat{\mathbf{m}}_\mathbf{a}}$, $\mathbf{B} = \left[\frac{\partial \Phi_1}{\partial \boldsymbol{\sigma}_\mathbf{a}} \cdots \frac{\partial \Phi_n}{\partial \boldsymbol{\sigma}_\mathbf{a}}\right]\Big|_{\boldsymbol{\sigma}_\mathbf{a}=\hat{\boldsymbol{\sigma}}_\mathbf{a}}$, $\mathbf{h}$ is a vector of higher-order terms, $\tilde{\mathbf{m}}_\mathbf{a} = \mathbf{m}_\mathbf{a}^* - \hat{\mathbf{m}}_\mathbf{a}$, $\tilde{\boldsymbol{\sigma}}_\mathbf{a} = \boldsymbol{\sigma}_\mathbf{a}^* - \hat{\boldsymbol{\sigma}}_\mathbf{a}$, and $\frac{\partial \Phi_k}{\partial \mathbf{m}_\mathbf{a}}$ and $\frac{\partial \Phi_k}{\partial \boldsymbol{\sigma}_\mathbf{a}}$ are defined as

$$\left[\frac{\partial \Phi_k}{\partial \mathbf{m}_\mathbf{a}}\right]^T = \left[\underbrace{0\cdots0}_{(k-1)\times l}\ \frac{\partial \Phi_k}{\partial m_{1k}} \cdots \frac{\partial \Phi_k}{\partial m_{lk}}\ \underbrace{0\cdots0}_{(m-k)\times l}\right] \tag{31}$$

$$\left[\frac{\partial \Phi_k}{\partial \boldsymbol{\sigma}_\mathbf{a}}\right]^T = \left[\underbrace{0\cdots0}_{(k-1)\times l}\ \frac{\partial \Phi_k}{\partial \boldsymbol{\sigma}_{1k}} \cdots \frac{\partial \Phi_k}{\partial \boldsymbol{\sigma}_{lk}}\ \underbrace{0\cdots0}_{(m-k)\times l}\right]. \tag{32}$$

Substituting (30) into (29), it is obtained that

$$\begin{aligned}\tilde{u} &= \tilde{\mathbf{w}}_\mathbf{a}^T\hat{\mathbf{\Phi}}_\mathbf{a} + \hat{\mathbf{w}}_\mathbf{a}^T(\mathbf{A}^T\tilde{\mathbf{m}}_\mathbf{a} + \mathbf{B}^T\tilde{\boldsymbol{\sigma}}_\mathbf{a} + \mathbf{h}) + \tilde{\mathbf{w}}_\mathbf{a}^T\tilde{\mathbf{\Phi}}_\mathbf{a} + \Delta \\ &= \tilde{\mathbf{w}}_\mathbf{a}^T\hat{\mathbf{\Phi}}_\mathbf{a} + \tilde{\mathbf{m}}_\mathbf{a}^T\mathbf{A}\hat{\mathbf{w}}_\mathbf{a} + \tilde{\boldsymbol{\sigma}}_\mathbf{a}^T\mathbf{B}\hat{\mathbf{w}}_\mathbf{a} + \varepsilon\end{aligned} \tag{33}$$

where $\hat{\mathbf{w}}_\mathbf{a}^T\mathbf{A}^T\tilde{\mathbf{m}}_\mathbf{a} = \tilde{\mathbf{m}}_\mathbf{a}^T\mathbf{A}\hat{\mathbf{w}}_\mathbf{a}$ and $\hat{\mathbf{w}}_\mathbf{a}^T\mathbf{B}^T\tilde{\boldsymbol{\sigma}}_\mathbf{a} = \tilde{\boldsymbol{\sigma}}_\mathbf{a}^T\mathbf{B}\hat{\mathbf{w}}_\mathbf{a}$ are used since they are scales; and the uncertain term $\varepsilon \equiv \hat{\mathbf{w}}_\mathbf{a}^T\mathbf{h} + \tilde{\mathbf{w}}_\mathbf{a}^T\tilde{\mathbf{\Phi}}_\mathbf{a} + \mathbf{w}_\mathbf{i}^{*T}\mathbf{\Phi}_\mathbf{i}^* + \Delta$.

From (25), the error equation can be rewritten as

$$\dot{s} = g(\tilde{\mathbf{w}}_\mathbf{a}^T\hat{\mathbf{\Phi}}_\mathbf{a} + \tilde{\mathbf{m}}_\mathbf{a}^T\mathbf{A}\hat{\mathbf{w}}_\mathbf{a} + \tilde{\boldsymbol{\sigma}}_\mathbf{a}^T\mathbf{B}\hat{\mathbf{w}}_\mathbf{a} + \varepsilon - u_{rc}). \tag{34}$$

Then, the following theorem can be stated and proven.

Theorem 1: Consider a car-following system represented by (4). The vehicle's control law is designed as $u = u_{nc} + u_{rc}$. The neural controller u_{nc} is designed as (28), in which the parameter vectors are tuned by

$$\dot{\tilde{\mathbf{w}}}_\mathbf{a} = -\dot{\hat{\mathbf{w}}}_\mathbf{a} = -\eta_1 s\hat{\mathbf{\Phi}}_\mathbf{a} \tag{35}$$

$$\dot{\tilde{\mathbf{m}}}_\mathbf{a} = -\dot{\hat{\mathbf{m}}}_\mathbf{a} = -\eta_2 s\mathbf{A}\hat{\mathbf{w}}_\mathbf{a} \tag{36}$$

$$\dot{\tilde{\boldsymbol{\sigma}}}_\mathbf{a} = -\dot{\hat{\boldsymbol{\sigma}}}_\mathbf{a}^T = -\eta_3 s\mathbf{B}\hat{\mathbf{w}}_\mathbf{a} \tag{37}$$

where η_1, η_2 and η_3 are the learning-rates. The robust controller u_{rc} is designed as

$$u_{rc} = \frac{(\delta^2+1)s}{2\delta^2} \tag{38}$$

where δ is a prescribed attenuation constant. Then the stability of the intelligent longitudinal control system can be guaranteed.

Proof: Consider a Lyapunov function candidate in the following form

$$V(s, \tilde{\mathbf{w}}_\mathbf{a}, \tilde{\mathbf{m}}_\mathbf{a}, \tilde{\boldsymbol{\sigma}}_\mathbf{a}) = \frac{1}{2}s^2 + g\left(\frac{\tilde{\mathbf{w}}_\mathbf{a}^T\tilde{\mathbf{w}}_\mathbf{a}}{2\eta_1} + \frac{\tilde{\mathbf{m}}_\mathbf{a}^T\tilde{\mathbf{m}}_\mathbf{a}}{2\eta_2} + \frac{\tilde{\boldsymbol{\sigma}}_\mathbf{a}^T\tilde{\boldsymbol{\sigma}}_\mathbf{a}}{2\eta_3}\right). \tag{39}$$

Differentiating (39) with respect to time and using (34), it gives

$$
\begin{aligned}
\dot{V} &= s\dot{s} + g(\frac{\tilde{\mathbf{w}}_{\mathbf{a}}^T\dot{\tilde{\mathbf{w}}}_{\mathbf{a}}}{\eta_1} + \frac{\tilde{\mathbf{m}}_{\mathbf{a}}^T\dot{\tilde{\mathbf{m}}}_{\mathbf{a}}}{\eta_2} + \frac{\tilde{\boldsymbol{\sigma}}_{\mathbf{a}}^T\dot{\tilde{\boldsymbol{\sigma}}}_{\mathbf{a}}}{\eta_3}) \\
&= s(\tilde{\mathbf{w}}_{\mathbf{a}}^T\hat{\boldsymbol{\Phi}}_{\mathbf{a}} + \tilde{\mathbf{m}}_{\mathbf{a}}^T\mathbf{A}\hat{\mathbf{w}}_{\mathbf{a}} + \tilde{\boldsymbol{\sigma}}_{\mathbf{a}}^T\mathbf{B}\hat{\mathbf{w}}_{\mathbf{a}} + \varepsilon - u_{rc}) + g(\frac{\tilde{\mathbf{w}}_{\mathbf{a}}^T\dot{\tilde{\mathbf{w}}}_{\mathbf{a}}}{\eta_1} + \frac{\tilde{\mathbf{m}}_{\mathbf{a}}^T\dot{\tilde{\mathbf{m}}}_{\mathbf{a}}}{\eta_2} + \frac{\tilde{\boldsymbol{\sigma}}_{\mathbf{a}}^T\dot{\tilde{\boldsymbol{\sigma}}}_{\mathbf{a}}}{\eta_3}) \\
&= \tilde{\mathbf{w}}_{\mathbf{a}}^T(s\hat{\boldsymbol{\Phi}}_{\mathbf{a}} + \frac{\dot{\tilde{\mathbf{w}}}_{\mathbf{a}}}{\eta_1}) + \tilde{\mathbf{m}}_{\mathbf{a}}^T(s\mathbf{A}\hat{\mathbf{w}}_{\mathbf{a}} + \frac{\dot{\tilde{\mathbf{m}}}_{\mathbf{a}}}{\eta_2}) + \tilde{\boldsymbol{\sigma}}_{\mathbf{a}}^T(s\mathbf{B}\hat{\mathbf{w}}_{\mathbf{a}} + \frac{\dot{\tilde{\boldsymbol{\sigma}}}_{\mathbf{a}}}{\eta_3}) + s(\varepsilon - u_{rc}) \\
&= s\varepsilon - \frac{(\delta^2+1)s^2}{2\delta^2} \\
&= -\frac{s^2}{2} - \frac{1}{2}(\frac{s}{\delta} - \delta\varepsilon)^2 + \frac{1}{2}\delta^2\varepsilon^2 \\
&\le -\frac{s^2}{2} + \frac{1}{2}\delta^2\varepsilon^2
\end{aligned} \tag{40}
$$

Assume $\varepsilon \in L_2[0,T)$, $\forall T \in [0,\infty)$, integrating the above equation from $t=0$ to $t=T$, yields

$$V(T) - V(0) \le -\frac{1}{2}\int_0^T s^2\, dt + \frac{1}{2}\delta^2\int_0^T \varepsilon^2\, dt \tag{41}$$

Since $V(T) \ge 0$, the above inequality implies the following inequality

$$\frac{1}{2}\int_0^T s^2\, dt \le V(0) + \frac{1}{2}\delta^2\int_0^T \varepsilon^2\, dt \tag{42}$$

If the system starts with initial conditions $s(0)=0$, $\tilde{\mathbf{w}}_{\mathbf{a}} = 0$, $\tilde{\mathbf{m}}_{\mathbf{a}}(0) = 0$ and $\tilde{\boldsymbol{\sigma}}_{\mathbf{a}}(0) = 0$, the L_2 tracking performance can be rewritten as

$$\sup_{\varepsilon \in L_2[0,T]} \frac{\| s \|}{\| \varepsilon \|} \le \delta \tag{43}$$

where $\| s \|^2 = \int_0^T s^2(t)dt$ and $\| \varepsilon \|^2 = \int_0^T \varepsilon^2(t)dt$.

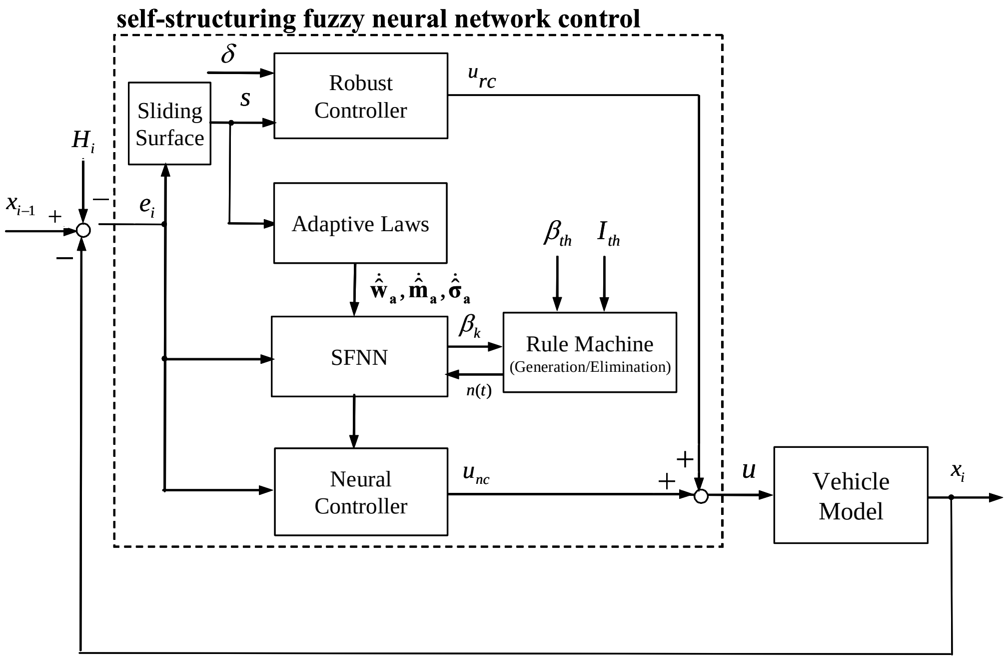


Fig. 3. Self-structuring fuzzy neural network control for the longitudinal system.

IV. Simulation Results

To investigate the effectiveness of the proposed intelligent longitudinal control system, two simulation scenarios are carried out. The specific constants of the vehicle parameters used in this paper are chosen as $\tau = 0.2$, $m = 916kg$, $K_d = 0.44Ns^2/m^2$ and $d_m = 67.7$ Nm. For both scenarios, the control parameters of SSFNNC are selected as $k_1 = 2$, $k_2 = 5$, $k_3 = 4$,

$\eta_1 = \eta_2 = \eta_3 = 1000$, $\delta = 0.5$, $\sigma_i = 2.0$, $\beta_{th} = 0.6$, $\tau = 0.01$, $\rho = 0.3$, and $I_{th} = 0.01$. These parameters are chosen through some trials to achieve satisfactory transient control performance.

In scenario 1, assumes that one following vehicle (FV) follows the leading vehicle (LV). The safety spacing is initialized with $H_1 = 10m$ first, and after the 15th, 30th, 45th, 60th, and 75th seconds the safety space is changed between $H_1 = 5m$ and $H_1 = 10m$, respectively. The initial values of the LV and FV are chosen as $v_l(0) = 20m/\sec$, $a_l(0) = 0m/\sec^2$, $v_1(0) = 20m/\sec$ and $a_1(0) = 0m/\sec^2$ and the LV in the platoon has no acceleration. The simulation results of intelligent longitudinal control for scenario 1 are shown in Fig. 4. The safety spacing of the FV is shown in Fig. 4(a), the vehicle of the FV is shown in Fig. 4(b), the control input of FV is shown in Fig. 4(c), and the rule number of SFNN is shown in Fig. 4(d). From the simulation results, it can be seen that the proposed SSFNNC system can achieve satisfactory performance for the one-vehicle following system even in the change of the safety spacing command.

In scenario 2, assumes that three FVs follow the LV with the safety space $H_i = 15m$. The vehicle acceleration and velocity of the LV are shown in Fig. 5(a) and 5(b), respectively. For numerical simulations, the initial values of the vehicle following system are chosen as $v_l(0) = 20m/\sec$, $a_l(0) = 0m/\sec^2$, $v_i(0) = 20m/\sec$ and $a_i(0) = 0m/\sec^2$. The simulation results of intelligent longitudinal control for scenario 2 are shown in Fig. 6. The safety spacing of the FV is shown in Fig. 6(a), the vehicle of the FV is shown in Fig. 6(b), the acceleration of the FV is shown in Fig. 6(c), the control input of FV is shown in Fig. 6(d), and the rule number of SFNN is shown in Fig. 6(e). From the simulation results, it can be seen that the proposed SSFNNC system can also achieve satisfactory performance even in the changes of acceleration and velocity of the LV.

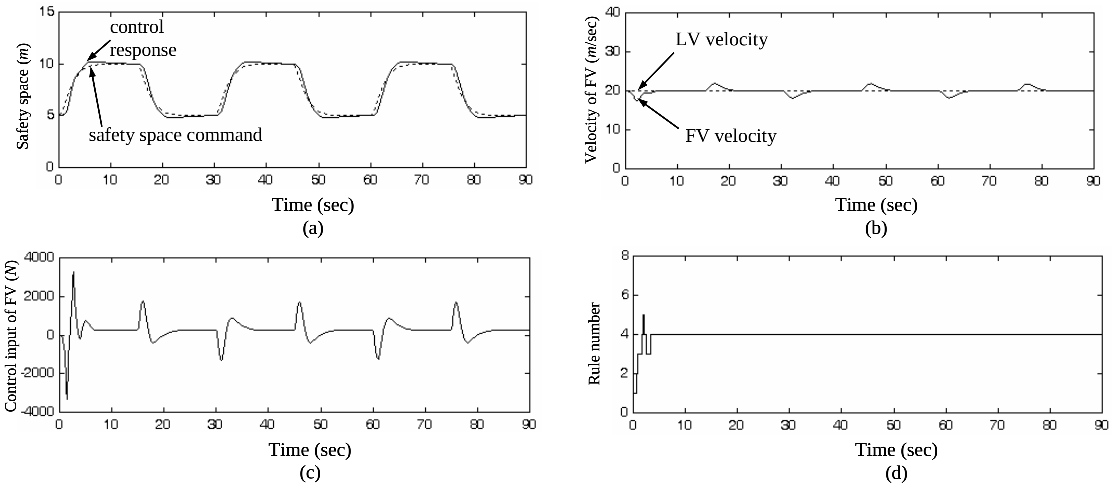


Fig. 4. Simulation results for scenario 1.

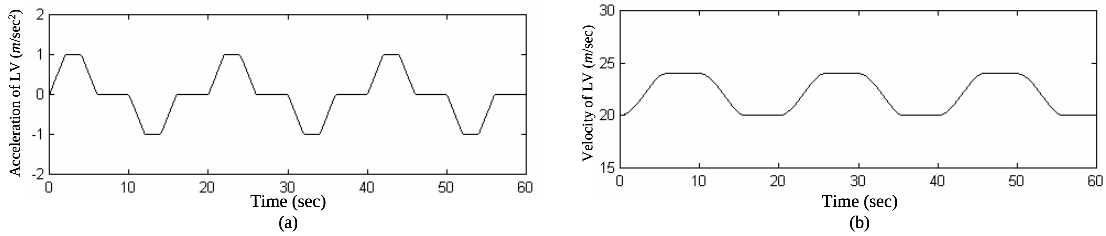
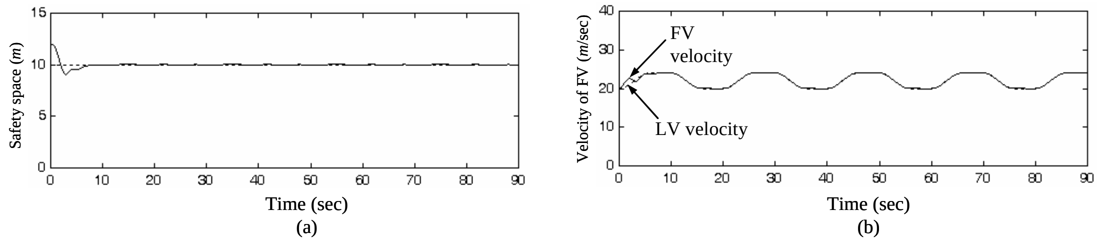


Fig. 5. Leading vehicle’s acceleration and velocity time profile for scenario 2.



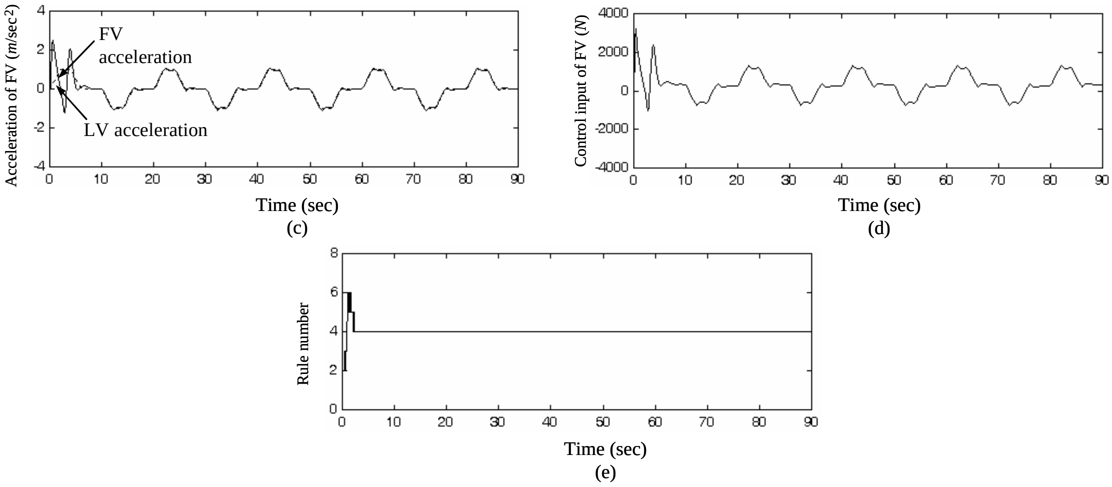


Fig. 6. Simulation results for scenario 2.

V. Conclusions

This paper has successfully developed an intelligent longitudinal control system via the self-structuring fuzzy neural network (SFNN) approach and adaptive control approach for the vehicle-following system. The on-line adaptation laws of the proposed self-structuring fuzzy neural network control scheme are derived based on the Lyapunov stability theorem, so that the tracking stability can be guaranteed for the control system. In the SFNN design, a dynamic rule generating/elimination mechanism is developed to cope with the tradeoff between the approximation accuracy and computational loading. Finally, two different simulation scenarios are carried out and simulation results have demonstrated that the proposed control system can achieve favorable tracking performance for the vehicle-following control even under the leading vehicles safety space change and acceleration maneuver.

VI. References

[1] **Sheikholeslam, S and Desoer, C A** Longitudinal control of a platoon of vehicles with no communication of lead vehicle information: a system level study, IEEE Trans. Veh. Technol., 42 (4) (November 1993) 546-554

[2] **Yang; Y and Farrell, J A** Magnetometer and differential carrier phase GPS-aided INS for advanced vehicle control**,** IEEE Trans. Robotics and Automation, **19** (2) (April 2003) 269-282

[3] **Omidvar, O and Elliott, D L** Neural Systems for Control, Academic Press (1997)

[4] **Wang, H and Wang, Y** Neural-network-based fault-tolerant control of unknown nonlinear systems, IEE Proc., Contr. Theory Appl., **146** (5) (September 1999) 389-398

[5] **Lin, C M and Hsu, C F** Neural network hybrid control for antilock braking systems, IEEE Trans. Neural Netw., **14** (2) (March 2003) 351-359

[6] **Hsu, C F Lin, C M and Chen, T Y** Neural-network-identification-based adaptive control of wing rock motion, IEE Proc., Contr. Theory Appl., **152** (1) (January 2005) 65-71

[7] **Lin, F J Lin, C H and Shen P H** Self-constructing fuzzy neural network speed controller for permanent-magnet synchronous motor drive, IEEE Trans. Fuzzy Syst., **9** (5) (October 2001) 751-759

[8] **Lin, F J and Lin, C H** A permanent-magnet synchronous motor servo drive using self-constructing fuzzy neural network controller, IEEE Trans. Energy Conversion, **19** (1), (March 2004) 66-72

[9] **Park, J H Huh, S H Kim, S H, Seo, S J and Park, G T** Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks, IEEE Trans. Neural Networks, **16** (2), (March 2005) 414-422

[10] **Slotine, J J E and Li, W** Applied Nonlinear Control, Englewood Cliffs, New Jersey (1991)

[11] **Wang, L X** Adaptive Fuzzy Systems and Control: Design and Stability Analysis, Englewood Cliffs, NJ: Prentice-Hall (1994)