Soft Comput (2013) 17:2053–2064
DOI 10.1007/s00500-013-1123-6

FOCUS

Direct adaptive type-2 fuzzy neural network control for a generic hypersonic flight vehicle

Fang Yang · Ruyi Yuan · Jianqiang Yi · Guoliang Fan · Xiangmin Tan

Published online: 2 October 2013

Abstract A direct adaptive interval type-2 fuzzy neural network (IT2-FNN) controller is designed for the first time in hypersonic flight control. The generic hypersonic flight vehicle is a multi-input multi-output system whose longitudinal model is high-order, highly nonlinear, tight coupling and most of all includes big uncertainties. Interval type-2 fuzzy sets with Gaussian membership functions are used in antecedent and consequent parts of fuzzy rules. The IT2-FNN directly outputs elevator deflection and throttle setting which make the GHFV track the altitude command signal and meanwhile maintain its velocity. The parameter adaptive law of IT2-FNN is derived using backpropagation method. The deviation of the control signal from the nominal dynamic inversion control signal is used as the reference output signal of IT2-FNN. The tracking errors of velocity and altitude are used as inputs of IT2-FNN. Tracking differentiator is designed to form an arranged transition process (ATP) of the command signal as well as ATP's high-order derivatives. Nonlinear state observer is designed to get the approximations of velocity, altitude as well as their high-order derivatives. Simulation results validate the effectiveness and robustness of the proposed controller especially under big uncertainties.

Keywords Type-2 fuzzy logic · Fuzzy neural network · Hypersonic · Direct adaptive control · Uncertainty

Abbreviations

V	Velocity, m/s
q	Pitch rate, rad/s
γ	Flight path angle, rad
α	Angle of attack, rad
h	Altitude, m
M_y	Pitch moment, N m
I_y	Moment of inertia, kg m^2
r	Radial distance from Earth's center, m
μ	Gravitational constant
m	Mass, kg
s	Reference area, m^2
ρ	Density of air, kg/m^3
$\bar{c}$	Mean aerodynamic chord, m
R_E	Radius of the earth, m
β	Fuel equivalence ratio
δ_t	Throttle setting instruction
δ_e	Elevator deflection, rad
L	Lift, N
D	Drag, N
T	Thrust, N
C_L	Lift coefficient
C_D	Drag coefficient
C_T	Thrust coefficient
C_M	Pitch moment coefficient

1 Introduction

A generic hypersonic flight vehicle (GHFV) is in single-stage-to-orbit winged-cone configuration and is rocket-powered (Shaughnessy et al. 1990). Hypersonic flight refers

Communicated by C. Alippi, D. Zaho and D. Liu.

F. Yang · R. Yuan (✉) · J. Yi · G. Fan · X. Tan
Institute of Automation, Chinese Academy of Sciences, Beijing, China
e-mail: ruyiyuan@hotmail.com

to the flight with at least 5 Mach (that is 5 times the local speed of sound). GHFV has a large flight envelope and it flies within a complicated environment. For example, it can cruise at an altitude of about 33,000 m and Mach 15. Due to its high speed, high altitude and large thrust to weight ratio, GHFV can be used as a reusable orbital transport plane and intercontinental airliner. In spite of these beautiful future applications, its flight control law design is still highly challenging. The longitudinal model of the GHFV is multi-input multi-output (MIMO), high-order, highly nonlinear and tight coupling. It has strong interaction among the structural, aerodynamic and propulsive effects (Keshmiri et al. 2006; Parker et al. 2007; Shakiba and Serrani 2011). Furthermore, the state signals' high-order derivatives and the arranged transition process (ATP) of the command signal are difficult to get. The unmodeled dynamics, measurement noises and external disturbances all show the truth that there exist big uncertainties in hypersonic flight control. So it is of significant importance to design a robust controller to overcome various uncertain problems.

Many classic and intelligent hypersonic flight control methods have been proposed in recent years. A robust sliding mode controller (Xu et al. 2004) was designed for the feedback linearized model of the GHFV. It was robust with respect to parameter uncertainty but it had strong control chattering problem which should be avoided. A robust minimax linear quadratic regulator (LQR) (Rehman et al. 2009) was also designed for the linearized model. It was an optimal controller which minimized the maximum value of the quadratic cost function but the solving process of Riccati equation was not easy. The linearized model was decoupled into two subsystems with high-order active disturbance rejection controller (ADRC) (Yang et al. 2012). The extended state observer of ADRC could estimate the overall uncertain terms but there were too many parameters to be optimized. Adaptive neural network (Liu and Lu 2009a) was used to compensate the model error between the real plant and the nominal model before the dynamic inversion control (DIC) signals were obtained. Similarly a traditional adaptive fuzzy logic system was designed to approximate the uncertain terms of the linearized model (Liu and Lu 2009b), however, no simulation was conducted to validate the controller's effectiveness under big uncertainties.

Traditional fuzzy logic controller uses traditional type-1 fuzzy set (T1-FS) which can deal with uncertain problem to a certain extent. When the parameter uncertainty becomes bigger and there exist rule uncertainties, type-2 fuzzy set (T2-FS) can be more effective (Mendel 2007). Membership functions (MFs) of T2-FS are fuzzy whereas MF of T1-FS is crisp. The additional degree of freedom of T2-FS gives a possibility to deal with big parameter uncertainty and rule uncertainties (Li et al. 2013). Type-2 fuzzy logic system (T2-FLS) contains a type-reducer (TR) which converts the type-2 fuzzy inference output set to T1-FS. For a general T2-FLS, the computation of TR can be enormous. Interval type-2 fuzzy set (IT2-FS) is used in order to reduce the computation cost and therefore put type-2 fuzzy logic control into real application (Liang and Mendel 2000). An adaptive interval T2-FLS (IT2-FLS) was used to approximate the unknown nonlinear system online (Ougli et al. 2008). The fuzzy parameters were randomly initialized and it did not need any prior knowledge about the nonlinear terms. For a multivariable nonlinear system, a direct adaptive interval type-2 fuzzy controller (Lin et al. 2009) was designed which could satisfy the H_∞ tracking performance. IT2-FLS was also introduced into hypersonic flight control. IT2-FLS was used to approximate the nonlinear but unknown terms of the linearized model in DIC (Yang et al. 2013a). IT2-FLS was also used to approximate the partly unknown terms in backstepping control (Yang et al. 2013b). The global stability was guaranteed by Lyapunov theory. Other application of T2-FLS can be seen in (Wu and Mendel 2002; Zaheer and Kim 2011; Fazel Zarandi and Gamasaee 2012).

Fuzzy neural network (FNN) is one kind of intelligent systems which combines the advantages of fuzzy logic and neural network. Fuzzy logic adds prior knowledge via fuzzy rules, but it lacks self-learning capability. Neural network has the properties of parallel computation and strong self-learning capability, but it needs a large diversity of training data which result in a long time to converge to the optimal solution in real-time application. Furthermore, neural network is a black box system where weights are randomly initialized and their learning results are obscure for designers. Instead, the initialization, learning process and learning results of FNN have clear physical meaning and FNN is therefore easily understandable by engineers. The parameter learning process of FNN can be data driven or knowledge driven. Another advantage is that FNN has good fault-tolerance property. Traditional FNN which uses T1-FS is so called T1-FNN. Although T1-FNN can deal with many uncertain problems, it cannot handle rule uncertainties coming from the following aspects (Karnik et al. 1999; Mendel and John 2002; Pan et al. 2011): (1) noise in measurement and training data; (2) fuzzy language uncertainty, e.g. it has different meaning for different people; (3) consequent knowledge is extracted from a group of experts who do not all agree; (4) working environment and actuator characteristic uncertainty.

When IT2-FS is used in antecedent and/or consequent parts of FNN, it is so called IT2-FNN. Some works have been done on IT2-FNN control. IT2-FS was used in only antecedent part and a feedforward neural network was chosen as a consequent part (Singh et al. 2004). IT2-FNN was used in intelligent mobile navigation which was effective even with low cost range sensor and microprocessor (Nurmaini et al. 2009). IT2-FNN was systematically studied on the identification and control application in micro aircraft vehicle attitude

control (Chen 2012). But it had some disadvantages in the derivation process of adaptive law. A monotonic type-2 fuzzy neural network was used in application to thermal comfort prediction (Li et al. 2012). Simulation results showed its high accuracy in prediction.

Here, a direct adaptive interval type-2 fuzzy neural network controller is designed for the first time in the tracking control of hypersonic flight vehicle. We use IT2-FNN to directly obtain the control signals which approximate the nominal DIC signals. Gaussian MFs with the same center but different widths are used in IT2-FS. The parameter adaptive law for a MIMO IT2-FNN is derived using backpropagation (BP) method. For one pair of input-output data, BP runs the update course online to reduce the pre-defined error. Results show it is much more complicated than single-output FNN and it contains other properties different from other FNNs.

This paper is organized as follows: Sect. 2 describes the background of this paper in three aspects: GHFV model, T2-FS and type-2 fuzzy logic system, type-2 fuzzy neural network. Section 3 designs the controller in detail which includes IT2-FNN adaptive law, tracking differentiator and nonlinear state observer (NSO). In Sect. 4, simulations are conducted to validate the proposed controller. Conclusions are given in the final part. The main novelties and contributions of this study are listed as follows:

1. A direct adaptive IT2-FNN controller is designed for the first time in hypersonic flight control. It may develop the application of IT2-FNN in the field of complicated system control.
2. The parameter adaptive law for a MIMO IT2-FNN with Gaussian MFs is derived using backpropagation method. Results show it has new properties different from other FNNs.
3. Tracking differentiator and NSO are combined with IT2-FNN to get arranged trasition process, high-order derivatives and other useful variables.
4. Simulations under big parameter uncertainty and rule uncertainties are conducted. Simulation results validate the effectiveness and robustness of the proposed controller.

2 Background

In this section, we first present the mathematical model of a GHFV and the DIC law of it. Then, we give brief introduction of the T2-FS and type-2 fuzzy logic system. Finally, we discuss the structure and computation of MIMO interval type-2 fuzzy neural network system.

2.1 GHFV model and dynamic inversion control

The mathematical model of the longitudinal dynamics of a GHFV which was developed at NASA Langley Research Center (Shaughnessy et al. 1990) is as follows (Parker et al. 2007; Xu et al. 2004):

$$\dot{V} = \frac{T\cos\alpha - D}{m} - \frac{\mu\sin\gamma}{r^2} \tag{1}$$

$$\dot{\gamma} = \frac{L + T\sin\alpha}{mV} - \frac{(\mu - V^2 r)\cos\gamma}{Vr^2} \tag{2}$$

$$\dot{h} = V\sin\gamma \tag{3}$$

$$\dot{\alpha} = q - \dot{\gamma} \tag{4}$$

$$\dot{q} = \frac{M_y}{I_y} \tag{5}$$

where

$$L = \frac{1}{2}\rho V^2 s C_L \tag{6}$$

$$D = \frac{1}{2}\rho V^2 s C_D \tag{7}$$

$$T = \frac{1}{2}\rho V^2 s C_T \tag{8}$$

$$M_y = \frac{1}{2}\rho V^2 s\bar{c} C_M \tag{9}$$

$$C_M = c_e\delta_e + C_{M0} \tag{10}$$

$$C_T = c_t\beta + C_{T0} \tag{11}$$

$$r = h + R_E \tag{12}$$

where c_e, c_t are the linear coefficients and C_{M0}, C_{T0} are the remainders which do not contain the control variables. The engine dynamics are modeled by a second-order system as:

$$\ddot{\beta} = -2\xi_n\omega_n\dot{\beta} - \omega_n^2\beta + \omega_n^2\delta_t \tag{13}$$

where ξ_n is the damping ratio and ω_n is the natural frequency of the actuator. So the control variables become δ_e and δ_t. Here we use the simplified aerodynamic coefficients around the nominal cruising flight. Coefficient details can be seen in (Xu et al. 2004).

For the purpose of feedback linearization, we use Lie derivatives (Khalil 2001) to differentiate V three times and h four times separately until there appears the control variables δ_e and δ_t explicitly. Thus the relative degrees of V and h are $n_1 = 3$ and $n_2 = 4$, respectively, and the total relative degree is $n_t = n_1 + n_2 = 7$. Meanwhile, the total system order of the longitudinal model (1)–(5) and engine dynamics (13) is $n_s = 5+2 = 7$. Because $n_t = n_s$, the longitudinal dynamics can be completely input/output linearized as:

$$\begin{cases} \ddot{V} &= f_1 + g_{11}\delta_e + g_{12}\delta_t \\ h^{(4)} &= f_2 + g_{21}\delta_e + g_{22}\delta_t \end{cases} \tag{14}$$

where

$$f_1 = (W_1\ddot{X}_0 + \dot{X}^T W_2 \dot{X})/m \tag{15}$$

$$\begin{aligned} f_2 = {} & f_1 \sin\gamma + 3\ddot{V}\dot{\gamma}\cos\gamma - 3\dot{V}\dot{\gamma}^2\sin\gamma + 3\dot{V}\ddot{\gamma}\cos\gamma \\ & - 3V\dot{\gamma}\ddot{\gamma}\sin\gamma - V\dot{\gamma}^3\cos\gamma \\ & + V\cos\gamma(P_1\ddot{X}_0 + \dot{X}^T P_2\dot{X}) \end{aligned} \tag{16}$$

$$g_{11} = \left(\frac{\partial T}{\partial\alpha}\cos\alpha - T\sin\alpha - \frac{\partial D}{\partial\alpha}\right)c_e/mI_y \tag{17}$$

$$g_{12} = \frac{\partial T}{\partial\beta}\cos\alpha\cdot\omega_n^2/m \tag{18}$$

$$\begin{aligned} g_{21} = {} & \left(\frac{\partial T}{\partial\alpha}\sin(\alpha+\gamma) + T\cos(\alpha+\gamma)\right. \\ & \left. - \frac{\partial D}{\partial\alpha}\sin\gamma + \frac{\partial L}{\partial\alpha}\cos\gamma\right)c_e/mI_y \end{aligned} \tag{19}$$

$$g_{22} = \frac{\partial T}{\partial\beta}\sin(\alpha+\gamma)\omega_n^2/m \tag{20}$$

where $\mathbf{X} = [\mathrm{V}\ \gamma\ \alpha\ \beta\ \mathrm{h}]^T$ and $\ddot{\mathbf{X}}_0$ is the part without control variables in $\ddot{X}$. Details of (15)–(20) can be found in (Xu et al. 2004; Yang et al. 2012). We can see that f_1, f_2 are unknown but bounded nonlinearities and $g_{11}, g_{12}, g_{21}, g_{22}$ are approximately known but bounded nonlinearities. All their computing is quite complicated and time-consuming. We denote:

$$\begin{aligned} \mathbf{u} &\triangleq \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix},\ \mathbf{y} \triangleq \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} V \\ h \end{bmatrix} \\ \mathbf{y}_r &\triangleq \begin{bmatrix} y_{r1} \\ y_{r2} \end{bmatrix} = \begin{bmatrix} V_r \\ h_r \end{bmatrix},\ \mathbf{G} \triangleq \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}. \end{aligned} \tag{21}$$

where $\mathbf{u}$ is the system input, $\mathbf{y}$ is the system output , $\mathbf{y}_r$ is the reference command, and $\mathbf{G}$ is the control matrix. Hence system (14) can be rewritten as:

$$\begin{bmatrix} y_1^{(3)} \\ y_2^{(4)} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} + \mathbf{G}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{22}$$

Assume that $\mathbf{G}$ is nonsingular. When the system is free of uncertainty, by DIC theory, the nominal DIC signals can be:

$$\mathbf{u}_I \triangleq \begin{bmatrix} u_{I1} \\ u_{I2} \end{bmatrix} = \mathbf{G}^{-1}\left(-\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} - \begin{bmatrix} k_1^T\underline{e}_1 \\ k_2^T\underline{e}_2 \end{bmatrix} + \begin{bmatrix} y_{r1}^{(3)} \\ y_{r2}^{(4)} \end{bmatrix}\right) \tag{23}$$

where $\mathbf{e}_1 = y_1 - y_{r1}$, $\mathbf{e}_2 = y_2 - y_{r2}$, $\underline{\mathbf{e}}_1 = [e_1\ \dot{e_1}\ \ddot{e_1}]^T$, $\underline{\mathbf{e}}_2 = [e_2\ \dot{e_2}\ \ddot{e_2}\ \dddot{e_2}]^T$. $\mathbf{k}_i = [k_{i0}, \ldots, k_{i(n_i-1)}]^T$, $(i = 1, 2)$ are constant coefficients which guarantee the polynomials $\mathbf{k}_i^T\underline{\mathbf{e}}_i$ are Hurwitz.

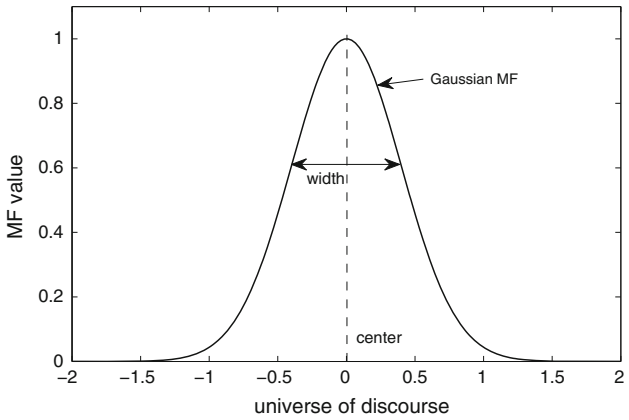

Fig. 1 MF example of T1-FS

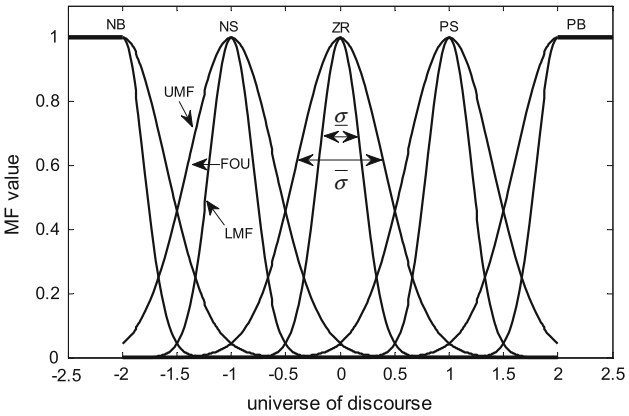

Fig. 2 MF example of IT2-FS

2.2 Type-2 fuzzy set and type-2 fuzzy logic system

The membership function (MF) of type-1 fuzzy set (T1-FS) is crisp as shown in Fig. 1. Each point in the universe of discourse has only one MF value. For Gaussian MF, the MF value is only determined by its center c and width σ. T2-FS is characterized by membership functions (primary MFs) that are themselves fuzzy as shown in Fig. 2. Each point in the universe of discourse has at least one MF value and each value also has a corresponding MF (secondary MF) value. All the possible values form an area. The area between the upper MF (UMF) and the lower MF (LMF) is called footprint of uncertainty (FOU), which adds the second degree of freedom to handle rule uncertainties (Mendel 2007). For IT2-FS, all secondary MF values equal 1. So IT2-FS can be and only be represented by LMF and UMF which are only determined by center c and widths $\underline{\sigma}$, $\overline{\sigma}$ (Castillo and Melin 2008).

The structure of interval type-2 fuzzy logic system (IT2-FLS) is shown in Fig. 3. The only difference between IT2-FLS and T1-FLS is that T2-FLS has a type reducer (TR). TR converts interval type-2 fuzzy inference output set to T1-FS.

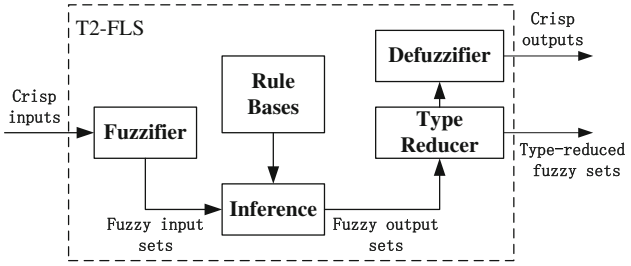

Fig. 3 The structure of IT2-FLS

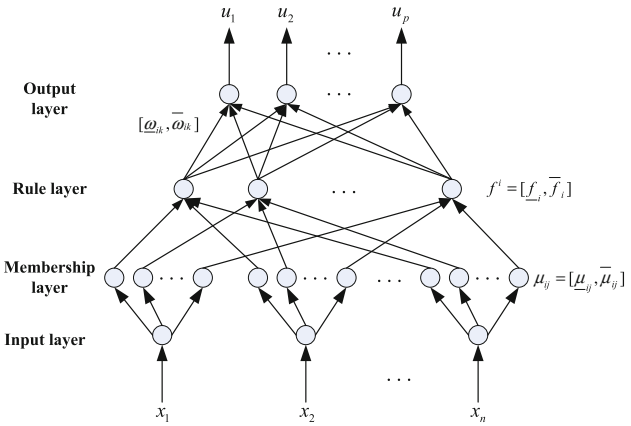

Fig. 4 General architecture of IT2-FNN

The TR computation of IT2-FLS just involves its LMF and UMF and therefore has low cost than general T2-FLS.

2.3 MIMO interval type-2 fuzzy neural network system

For a MIMO IT2-FNN system, we consider the following architecture as shown in Fig. 4.

It is a four-layer n input/p output IT2-FNN. The input layer transmits values of input linguistic variable x_j, $(j = 1, \ldots, n)$ to the membership layer directly. The membership layer computes MF values of each input variable belonging to each fuzzy set. Here we use Gaussian MF with the same center but different widths to fully describe the IT2-FS. Each node in the rule layer represents a possible IF-part of the fuzzy rules. The output of the ith $(i = 1, \ldots, M)$ rule is the fire strength f^i. Each node in the output layer represents an output linguistic variable u_k, $(k = 1, \ldots, p)$. The output of one node in this layer is the weighted summation of the fire strength. For IT2-FNN, the fire strength and the weight are in a range $[\underline{f}^i, \overline{f}^i]$ and $[\underline{\omega}_k^i, \overline{\omega}_k^i]$.

The fuzzy rule bases of IT2-FNN consist of a collection of IF-THEN rules in the following form:

Rule i: If x_1 is $\widetilde{A}_1^i$ and $\cdots$ and x_n is $\widetilde{A}_n^i$, Then u_1 is $[\underline{\omega}_1^i, \overline{\omega}_1^i]$ and $\cdots$ and u_p is $[\underline{\omega}_p^i, \overline{\omega}_p^i]$

where $i = 1, \ldots, M$ and M is the number of rules. $\widetilde{A}_j^i$, $(j = 1, \ldots, n)$ is the antecedent interval T2-FS. $\underline{\omega}_k^i$ and $\overline{\omega}_k^i$, $(k = 1, \ldots, p)$ are the lower and upper weights of the output u_k, respectively. Denote $\mathbf{x} = [x_1 \cdots x_n]^T$. The firing set $F^i(\mathrm{x})$ and the firing strength f_i associated with the ith rule are:

$$f^i \in F^i(\mathrm{x}) = \prod_{j=1}^{n} \mu_{\widetilde{A}_j^i}(x_j) = [\underline{f}^i, \overline{f}^i] \tag{24}$$

where

$$\underline{f}^i = \prod_{j=1}^{n} \underline{\mu}_{\widetilde{A}_j^i}(x_j) \tag{25}$$

$$\overline{f}^i = \prod_{j=1}^{n} \overline{\mu}_{\widetilde{A}_j^i}(x_j) \tag{26}$$

where $\underline{\mu}_{\widetilde{A}_j^i}(x_j)$ and $\overline{\mu}_{\widetilde{A}_j^i}(x_j)$ are the lower and upper MFs of $\mu_{\widetilde{A}_j^i}(x_j)$, respectively. By using the singleton fuzzification, product inference, center-average defuzzification and the center-of-sets (COS) (Karnik and Mendel 1998) type reducer, the type-reduced set is given by (Liang and Mendel 2000):

$$U_{cos}^k = \int_{u_k^1} \cdots \int_{u_k^M} \int_{f^1} \cdots \int_{f^M} \prod_{i=1}^{M} \mu_{F^i}(f^i) \Big/ \frac{\sum_{i=1}^{M} f^i u_k^i}{\sum_{i=1}^{M} f^i} \tag{27}$$

For IT2-FLS, F^i is T1-FS, $\mu_{F^i}(f^i) = 1$, so

$$U_{cos}^k = \int_{u_k^1} \cdots \int_{u_k^M} \int_{f^1} \cdots \int_{f^M} 1 \Big/ \frac{\sum_{i=1}^{M} f^i u_k^i}{\sum_{i=1}^{M} f^i} = [u_{kl}, u_{kr}] \tag{28}$$

where

$$u_{kl} = \frac{\sum_{i=1}^{M} f_{kl}^i \underline{\omega}_k^i}{\sum_{i=1}^{M} f_{kl}^i}, \qquad u_{kr} = \frac{\sum_{i=1}^{M} f_{kr}^i \overline{\omega}_k^i}{\sum_{i=1}^{M} f_{kr}^i} \tag{29}$$

where f_{kl}^i, f_{kr}^i denote the firing values used to compute the bounds u_{kl}, u_{kr}, respectively, which can be obtained using the Karnik–Mendel algorithm (KMA) (Liang and Mendel 2000). Because U_{cos}^k is T1-FS with equal MF value 1, the output can be its mean arithmetical value:

$$u_k = \frac{u_{kl} + u_{kr}}{2} \tag{30}$$

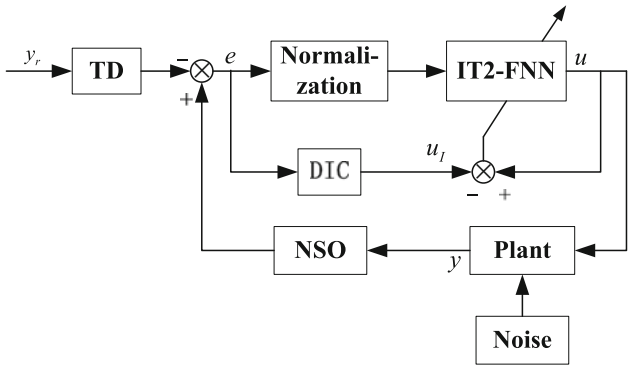

Fig. 5 The overall control scheme of the direct adaptive IT2-FNN

3 Control design

In this section, we first give an overall view on the control scheme. Then, the parameter adaptive law is derived in detail using backpropagation method. Algorithms of tracking differentiator and NSO are also given. Finally, the algorithm flow of IT2-FNN control is shown.

3.1 Overall control scheme

The overall control scheme is shown in Fig. 5. High-order derivatives of $\mathbf{y}_r$ and $\mathbf{y}$ are obtained by tracking differentiator (TD) and NSO, respectively. TD also forms the ATP of the command signal. The DIC outputs the nominal DIC signal $\mathbf{u}_I$. Output errors $\mathbf{e}$ are normalized to the corresponding universe of discourse before they are sent to IT2-FNN. Gaussian white noise (GWN) can be added to the plant to verify the controller's robustness.

3.2 IT2-FNN control law and adaptive law

The direct adaptive IT2-FNN controller is used to get the fuzzy control signal $\mathbf{u}$ online. Error $\mathbf{e}$ is used as fuzzy input. The deviation of the control signal $\mathbf{u}$ from the nominal DIC signal $\mathbf{u}_I$ is used as reference output signal. $\mathbf{e}$ is normalized before applied to the fuzzy rules. As is discussed in the MIMO IT2-FNN system, the fuzzy inputs are chosen as $x_1 = \Delta V$ and $x_2 = \Delta h$, the fuzzy outputs are $\mathbf{u} = [u_1\ u_2]^T = [\Delta\delta_e\ \Delta\delta_t]^T$. The defuzzified outputs are obtained by (28)–(30). Next we will discuss the derivation process of the parameter adaptive law.

Define the overall output error $E = \frac{1}{2}\sum_{k=1}^{m}(u_k - u_{Ik})^2$. The parameter adaptive law is designed to be (31) using backpropagation method (Wang 1996):

$$\begin{cases} \underline{\omega}_k^i(N+1) = \underline{\omega}_k^i(N) - \alpha_{1k}\dfrac{\partial E}{\partial \underline{\omega}_k^i} \\ \overline{\omega}_k^i(N+1) = \overline{\omega}_k^i(N) - \alpha_{1k}\dfrac{\partial E}{\partial \overline{\omega}_k^i} \\ c_{ij}(N+1) = c_{ij}(N) - \alpha_2\dfrac{\partial E}{\partial c_{ij}} \\ \underline{\sigma}_{ij}(N+1) = \underline{\sigma}_{ij}(N) - \alpha_3\dfrac{\partial E}{\partial \underline{\sigma}_{ij}} \\ \overline{\sigma}_{ij}(N+1) = \overline{\sigma}_{ij}(N) - \alpha_3\dfrac{\partial E}{\partial \overline{\sigma}_{ij}} \end{cases} \tag{31}$$

where $\alpha_{1k}, \alpha_2, \alpha_3$ are learning rates of the weights, the centers and the widths, respectively, N is the number of iteration in BP. The partial derivatives of E to $\underline{\omega}_k^i$, $\overline{\omega}_k^i$, and c_{ij} are easy to get as (32) whereas it is not obvious to $\underline{\sigma}_{ij}$ and $\overline{\sigma}_{ij}$ as (33).

$$\begin{cases} \dfrac{\partial E}{\partial \underline{\omega}_k^i} = \dfrac{u_k - u_{Ik}}{2\sum_{i=1}^{M} f_{kl}^i} f_{kl}^i, \dfrac{\partial E}{\partial \overline{\omega}_k^i} = \dfrac{u_k - u_{Ik}}{2\sum_{i=1}^{M} f_{kr}^i} f_{kr}^i \\ \dfrac{\partial E}{\partial c_{ij}} = \sum_{k=1}^{M}\left[\dfrac{(u_k - u_{Ik})(x_j - c_{ij})}{2}\left(\dfrac{(\underline{\omega}_k^i - u_{kl})f_{kl}^i}{\underline{\sigma}_{ij}^2\sum_{i=1}^{M} f_{kl}^i}\right.\right. \\ \qquad \left.\left. + \dfrac{(\overline{\omega}_k^i - u_{kr})f_{kr}^i}{\overline{\sigma}_{ij}^2\sum_{i=1}^{M} f_{kr}^i}\right)\right] \end{cases} \tag{32}$$

$$\begin{cases} \dfrac{\partial E}{\partial \underline{\sigma}_{ij}} = \sum_{k=1}^{M}\dfrac{\partial E}{\partial u_k}\dfrac{\partial u_k}{\partial \underline{\sigma}_{ij}} = \sum_{k=1}^{M}\dfrac{u_k - u_{Ik}}{2}\left(\dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} + \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}}\right) \\ \dfrac{\partial E}{\partial \overline{\sigma}_{ij}} = \sum_{k=1}^{M}\dfrac{\partial E}{\partial u_k}\dfrac{\partial u_k}{\partial \overline{\sigma}_{ij}} = \sum_{k=1}^{M}\dfrac{u_k - u_{Ik}}{2}\left(\dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} + \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}}\right) \end{cases} \tag{33}$$

In order to obtain the partial derivatives $\dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}}$, $\dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}}$ and $\dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}}$, $\dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}}$, we come back to the KMA type-reduction method. KMA returns two important integers L and R, where for $i \le L$, $f_{kl}^i = \overline{f}^i$, for $i > L$, $f_{kl}^i = \underline{f}^i$, for $i \le R$, $f_{kr}^i = \underline{f}^i$, for $i > R$, $f_{kr}^i = \overline{f}^i$. So u_{kl}, u_{kr} can be written as (Liang and Mendel 2000):

$$u_{kl} = u_{kl}\left(\overline{f}^1, \ldots, \overline{f}^L, \underline{f}^{L+1}, \ldots, \underline{f}^M, \underline{\omega}_k^1, \ldots, \underline{\omega}_k^M\right) \tag{34}$$

$$u_{kr} = u_{kr}\left(\underline{f}^1, \ldots, \underline{f}^R, \overline{f}^{R+1}, \ldots, \overline{f}^M, \overline{\omega}_k^1, \ldots, \overline{\omega}_k^M\right) \tag{35}$$

From expression (34) and (35), the partial derivatives of u_{kl} to $\underline{\sigma}_{ij}$, $\overline{\sigma}_{ij}$ and u_{kr} to $\underline{\sigma}_{ij}$, $\overline{\sigma}_{ij}$ have something to do with L and R. Further analysis shows that the size relationship between L and R also influences the partial derivatives. After a complicated derivation, the partial derivatives are as (36)–(41) show:

If $L \le R$, when $i \le L$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = 0 \end{cases} \tag{36}$$

when $L < R$ and $L < i \le R$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = 0 \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = 0 \end{cases} \tag{37}$$

when $i > R$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = 0 \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \end{cases} \tag{38}$$

If $L > R$, when $i \le R$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = 0 \end{cases} \tag{39}$$

when $R < i \le L$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \end{cases} \tag{40}$$

when $i > L$,

$$\begin{cases} \dfrac{\partial u_{kl}}{\partial \underline{\sigma}_{ij}} = \dfrac{f_{kl}^i(\underline{\omega}_k^i - u_{kl})}{\sum_{i=1}^M f_{kl}^i} \dfrac{(x_j - c_{ij})^2}{\underline{\sigma}_{ij}^3}, & \dfrac{\partial u_{kl}}{\partial \overline{\sigma}_{ij}} = 0 \\ \dfrac{\partial u_{kr}}{\partial \underline{\sigma}_{ij}} = 0, & \dfrac{\partial u_{kr}}{\partial \overline{\sigma}_{ij}} = \dfrac{f_{kr}^i(\overline{\omega}_k^i - u_{kr})}{\sum_{i=1}^M f_{kr}^i} \dfrac{(x_j - c_{ij})^2}{\overline{\sigma}_{ij}^3} \end{cases} \tag{41}$$

3.3 Tracking differentiator and nonlinear state observer

3.3.1 Tracking differentiator

Tracking differentiator (TD) is used to form the ATP of the reference signals and also to filter out high frequency noises. The discrete algorithms implemented in TD are as follows (Han 2009): velocity channel:

$$\begin{cases} fs_1(K) = -\lambda(\lambda(\lambda(\nu_{11}(K) - V_r) \\ \qquad +3\nu_{12}(K)) + 3\nu_{13}(K)) \\ \nu_{11}(K+1) = \nu_{11}(K) + \tau * \nu_{12}(K) \\ \nu_{12}(K+1) = \nu_{12}(K) + \tau * \nu_{13}(K) \\ \nu_{13}(K+1) = \nu_{13}(K) + \tau * fs_1(K) \end{cases} \tag{42}$$

Altitude channel:

$$\begin{cases} fs_2(K) = -\lambda(\lambda(\lambda(\lambda(\nu_{21}(K) - h_r) \\ \qquad +4\nu_{22}(K)) + 6\nu_{23}(K)) + 4\nu_{24}) \\ \nu_{21}(K+1) = \nu_{21}(K) + \tau * \nu_{22}(K) \\ \nu_{22}(K+1) = \nu_{22}(K) + \tau * \nu_{23}(K) \\ \nu_{23}(K+1) = \nu_{23}(K) + \tau * \nu_{24}(K) \\ \nu_{24}(K+1) = \nu_{24}(K) + \tau * fs_2(K) \end{cases} \tag{43}$$

where "velocity factor" λ and time step τ are taken as 3 and 0.01, respectively. $\nu_{11}, \nu_{12}, \nu_{13}$ represent V_r, $\dot{V}_r$, $\ddot{V}_r$ and $\nu_{21}, \nu_{22}, \nu_{23}, \nu_{24}$ represent h_r, $\dot{h}_r$, $\ddot{h}_r$, $\dddot{h}_r$. K is the number of iteration.

3.3.2 Nonlinear state observer

Nonlinear state observer can estimate the exact flight states and their high-order derivatives online which are difficult to measure in hypersonic flight condition. The discrete algorithms implemented in NSO are as follows:
Velocity channel:

$$\begin{cases} e_1 = z_{11}(K) - V \\ z_{11}(K+1) = z_{11}(K) + \tau * (z_{12}(K) - \beta_{11} * e_1) \\ z_{12}(K+1) = z_{12}(K) + \tau * (z_{13}(K) \\ \qquad -\beta_{12} * fal(e_1, 0.5, \tau)) \\ z_{13}(K+1) = z_{13}(K) + \tau \\ \qquad *(-\beta_{13} * fal(e_1, 0.25, \tau) + U_1) \end{cases} \tag{44}$$

Altitude channel:

$$\begin{cases} e_2 = z_{21}(K) - h \\ z_{21}(K+1) = z_{21}(K) + \tau * (z_{22}(K) - \beta_{21} * e_2) \\ z_{22}(K+1) = z_{22}(K) + \tau * (z_{23}(K) \\ \qquad -\beta_{22} * fal(e_2, 0.5, \tau)) \\ z_{23}(K+1) = z_{23}(K) + \tau * (z_{24}(K) \\ \qquad -\beta_{23} * fal(e_2, 0.25, \tau)) \\ z_{24}(K+1) = z_{24}(K) + \tau * \\ \qquad (-\beta_{24} * fal(e_1, 0.125, \tau) + U_2) \end{cases} \tag{45}$$

where z_{11}, z_{12}, z_{13} represent V, $\dot{V}$, $\ddot{V}$ and $z_{21}, z_{22}, z_{23}, z_{24}$ represent $h, \dot{h}, \ddot{h}, \dddot{h}$. $U_1 = g_{11}\delta_e + g_{12}\delta_t$ and $U_2 = g_{21}\delta_e + g_{22}\delta_t$. Parameters of NSO are chosen as $\beta_{11} = 100$, $\beta_{12} = 300$, $\beta_{13} = 2000$, $\beta_{21} = 100$, $\beta_{22} = 300$, $\beta_{23} = 1000$, $\beta_{24} = 1800$. Nonlinear functions used in (44) and (45) are

defined as:

$$\begin{cases} f_{db}(x,\delta) = (sgn(x+\delta) + sgn(x-\delta))/2 \\ f_{sg}(x,\delta) = (sgn(x+\delta) - sgn(x-\delta))/2 \\ fal(x,a,\delta) = xf_{sg}(x,\delta)/\delta^{1-a} + |f_{db}(x,\delta)||x|^a sgn(x) \end{cases} \tag{46}$$

where $sgn(\mathrm{x})$ is the sign function.

3.4 Algorithm flow

The algorithm flow of the IT2-FNN control is as follows

Step 1 Determine the IT2-FNN structure and initialize the corresponding parameters and fuzzy rules;

Step 2 Start the simulation iteration N;

Step 3 Compute the IT2-FNN inputs x_1, x_2 and the reference outputs u_{I1}, u_{I2};

Step 4 For x_1, x_2, u_{I1}, u_{I2}, start the BP iteration K;

Step 5 Compute the partial derivatives $\frac{\partial E}{\partial \underline{\omega}_k^i}$, $\frac{\partial E}{\partial \overline{\omega}_k^i}$, $\frac{\partial E}{\partial c_{ij}}$, $\frac{\partial E}{\partial \underline{\sigma}_{ij}}$, $\frac{\partial E}{\partial \overline{\sigma}_{ij}}$ like (32)–(41);

Step 6 Update parameters $\underline{\omega}_k^i$, $\overline{\omega}_k^i$, c_{ij}, $\underline{\sigma}_{ij}$, $\overline{\sigma}_{ij}$ like (31) and compute the fuzzy reference outputs u_1, u_2;

Step 7 If the terminate condition satisfies, goto Step 8; else $K = K + 1$ and go back to Step 5;

Step 8 Output u_1, u_2 as control variables to the plant;

Step 9 If the simulation time is over, go to Step 10; else $N = N + 1$ and go back to Step 3;

Step 10 Simulation over.

4 Simulations

In this section, we conduct simulations in two cases (with and without uncertainties) to validate the effectiveness and robustness of the proposed IT2-FNN controller.

4.1 Simulation conditions and initialization

The trimmed condition of the cruise flight of GHFV is chosen as Table 1. The command signal is chosen as a sine wave $\Delta h = h_r - h_0 = 600\sin(0.04\pi t)$ whose period is 50 s. The control objective is to track the sinusoidal altitude signal and meanwhile maintain its velocity, so $\Delta V = V_r - V_0 = 0$.

The number of inputs is $n = 2$ (the normalized error of velocity and altitude) and the number of outputs is $p = 2$ (the control increment of the elevator deflection and throttle setting). Assume each of the inputs x_1 and x_2 is divided into five IT2-FS. Therefore there are 25 fuzzy rules. Here we form $M = 25$ fuzzy rules according to the prior knowledge. The fuzzy input universe of discourse is chosen as $[-2,\ 2]$, the output universe of discourse of $u_1(\Delta\delta_e)$ is chosen as $[-0.2,\ 0.2]\ rad$ and the output universe of discourse of u_2 $(\Delta\delta_t)$ is chosen as $[-0.7,\ 0.7]$. The initialization of the interval type-2 antecedent fuzzy sets is shown in Table 2 (x_1 and x_2 have the same fuzzy division) and the initialized fuzzy rules are shown in Table 3. The learning rates are chosen as $\alpha_{11} = \alpha_{12} = \alpha_2 = \alpha_3 = 0.1$. The feedback coefficients are chosen as $k_{10} = 1, k_{11} = 3, k_{12} = 3, k_{20} = 1, k_{21} = 4, k_{22} = 6, k_{23} = 4$. The learning rates and feedback coefficients do not change in the simulation process. The parameter uncertainties are chosen as:

Table 1 Trimmed condition of GHFV

States	Value
V_0	4,590.3 m/s
h_0	33,528 m
γ_0	0 rad
α_0	0.04799 rad
q_0	0 rad/s
Control variables	**Value**
δ_{e0}	−0.5507°
δ_{t0}	0.2124

Table 2 Initialization of antecedent fuzzy sets

Fuzzy sets	Center	Lower width	Upper width
NB	−2	0.2	0.5
NS	−1	0.2	0.5
ZR	0	0.2	0.5
PS	1	0.2	0.5
PB	2	0.2	0.5

$$\begin{cases} m = m_0(1 + U_f + U_g * GWN) \\ I_y = I_{y0}(1 + U_f + U_g * GWN) \\ \rho = \rho_0(1 + U_f + U_g * GWN) \\ s = s_0(1 + U_f + U_g * GWN) \\ \bar{c} = \bar{c}_0(1 + U_f + U_g * GWN) \end{cases} \tag{47}$$

where $m_0, I_{y0}, \rho_0, s_0, \bar{c}_0$ are the nominal values, U_f is the fixed parameter uncertainty and U_g is the strength of the Gaussian white noise (GWN) whose power is 0.002.

4.2 Simulation results *without* uncertainty

Set $U_f = 0, U_g = 0$. So there exists no fixed parameter uncertainty and no noise. Then all parameters are in their nominal values. The simulation is conducted for 100 s (2 periods)in two cases: IT2-FNN and T1-FNN. The simulation results are shown in Figs. 6, 7, 8, 9, 10 and 11.

Table 3 Initialization of the fuzzy rules with prior knowledge

	NB		*NB*		[0.14, 0.2]		[0.5, 0.7]
	NB		*NS*		[0.14, 0.2]		[0.2, 0.4]
	NB		*ZR*		[0.14, 0.2]		[−0.1, 0.1]
	NB		*PS*		[0.14, 0.2]		[−0.4, −0.2]
	NB		*PB*		[0.14, 0.2]		[−0.7, −0.5]
	NS		*NB*		[0.06, 0.12]		[0.5, 0.7]
	NS		*NS*		[0.06, 0.12]		[0.2, 0.4]
	NS		*ZR*		[0.06, 0.12]		[−0.1, 0.1]
	NS		*PS*		[0.06, 0.12]		[−0.4, −0.2]
	NS		*PB*		[0.06, 0.12]		[−0.7, −0.5]
	ZR		*NB*		[−0.03, 0.03]		[0.5, 0.7]
	ZR		*NS*		[−0.03, 0.03]		[0.2, 0.4]
If x_1 is	*ZR*	and x_2 is	*ZR*	then u_1 is	[−0.03, 0.03]	and u_2 is	[−0.1, 0.1]
	ZR		*PS*		[−0.03, 0.03]		[−0.4, −0.2]
	ZR		*PB*		[−0.03, 0.03]		[−0.7, −0.5]
	PS		*NB*		[−0.12, −0.06]		[0.5, 0.7]
	PS		*NS*		[−0.12, −0.06]		[0.2, 0.4]
	PS		*ZR*		[−0.12, −0.06]		[−0.1, 0.1]
	PS		*PS*		[−0.12, −0.06]		[−0.4, −0.2]
	PS		*PB*		[−0.12, −0.06]		[−0.7, −0.5]
	PB		*NB*		[−0.2, −0.14]		[0.5, 0.7]
	PB		*NS*		[−0.2, −0.14]		[0.2, 0.4]
	PB		*ZR*		[−0.2, −0.14]		[−0.1, 0.1]
	PB		*PS*		[−0.2, −0.14]		[−0.4, −0.2]
	PB		*PB*		[−0.2, −0.14]		[−0.7, −0.5]

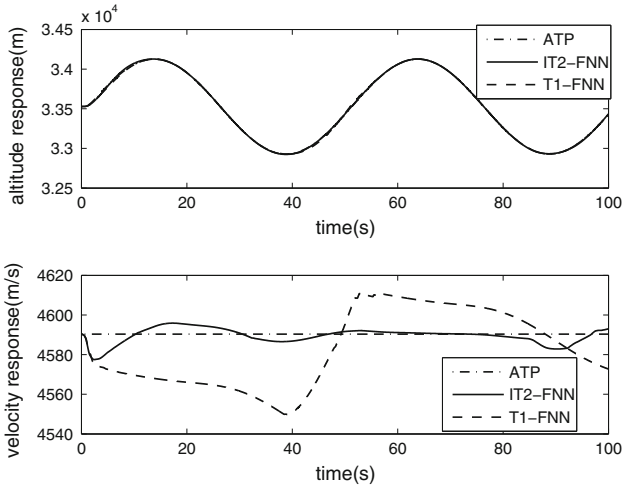

Fig. 6 Command response without uncertainty

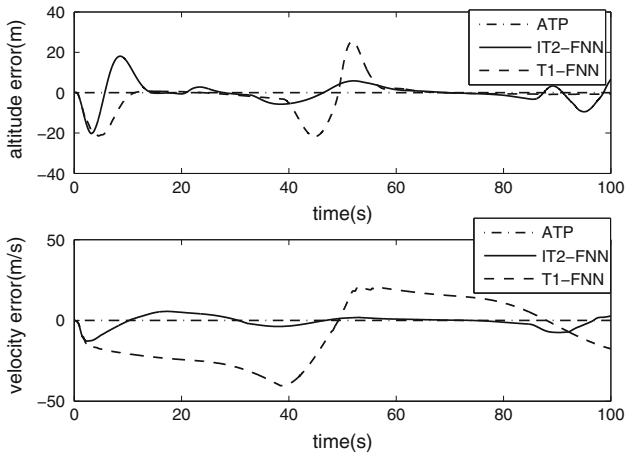

Fig. 7 Tracking error without uncertainty

As shown in Figs. 6 and 7, IT2-FNN and T1-FNN both can track the sinusoidal altitude command signal when there exists no uncertainty. Quantitatively, IT2-FNN has much less tracking error and overshoot than T1-FNN. Figure 8 shows that IT2-FNN has better actuator dynamics than T1-FNN. Control variables of IT2-FNN almost have no saturation whereas δ_e of T1-FNN has chattering and saturation at the beginning 10 s and δ_t of T1-FNN saturates almost in the entire course.This property of IT2-FNN is beneficial for GHFV because the actuators of GHFV often work in a severe condition. The above three figures also demonstrate a phenomenon that it is difficult to maintain the velocity and meanwhile track the altitude. This can be explained by energy conservation law. The variance of geopotential energy due to altitude vari-

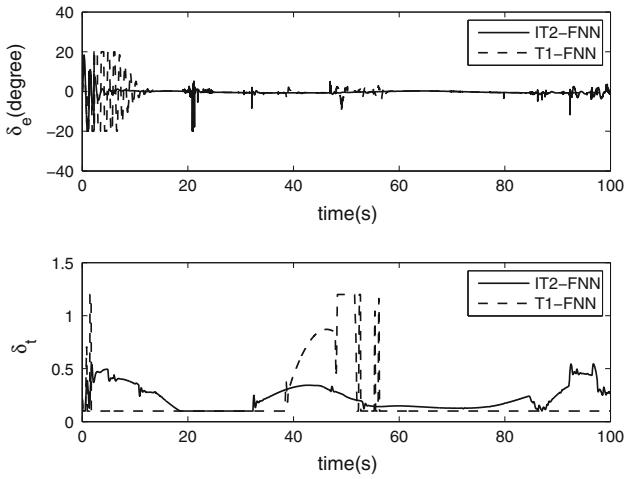

Fig. 8 Control response without uncertainty

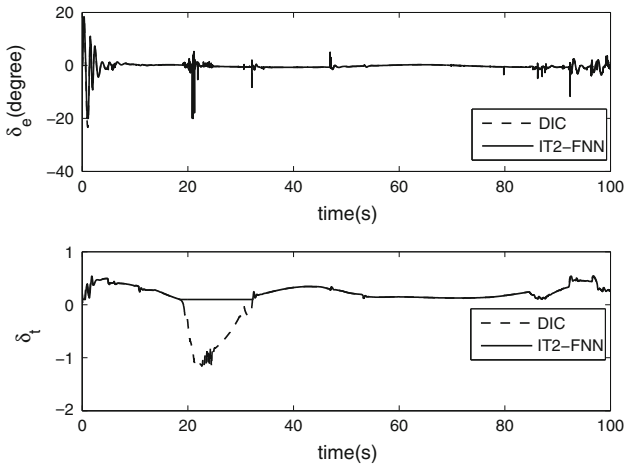

Fig. 9 IT2-FNN approximation effect

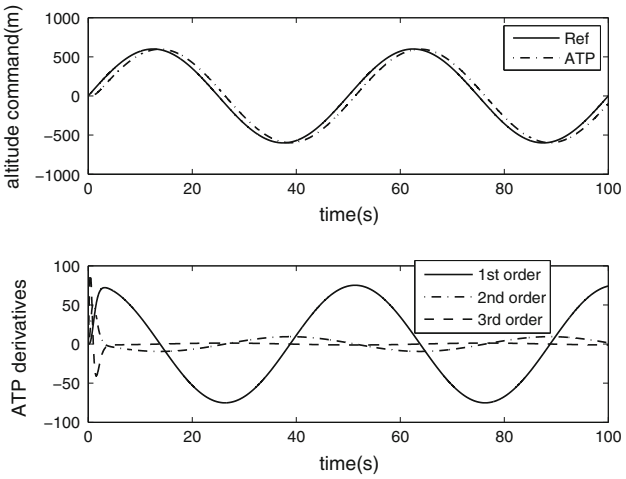

Fig. 10 ATP of altitude and its high-order derivatives

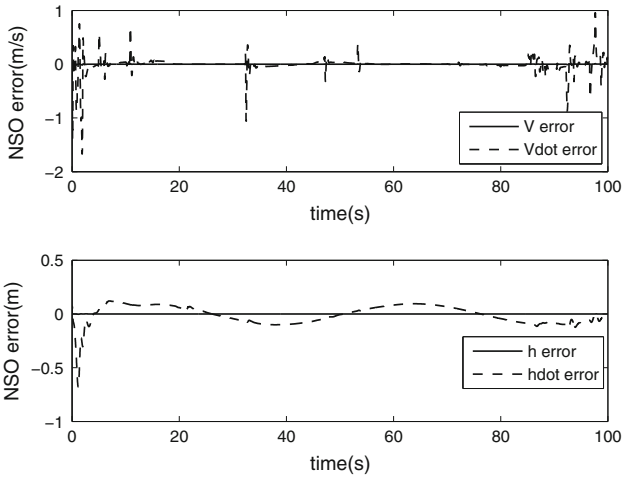

Fig. 11 Obsered values of V and h from NSO

ance can only be compensated by fuel energy and resistance loss. So it is not easy to maintain velocity and meanwhile track the altitude.

Figure 9 shows the fuzzy approximation effect of IT2-FNN only. We can see IT2-FNN can approximate the DIC signals almost immediately and IT2-FNN can smooth the control signals.

Figure 10 shows the outputs of the tracking differentiator (TD). TD can form an ATP which can reduce the initial tracking error and therefore result in smooth actuator movements. Meanwhile, TD can filter out high frequency noise when it gets high-order derivatives. ATP and its derivatives will be used in error feedback control (Huang et al. 2001; Han 2009).

Figure 11 shows part of the NSO outputs. V, h approximate their true values very well. $\dot{V}$, $\dot{h}$ also approximate their differential values very well considering their huge basic values. NSO is also very useful in hypersonic flight condition as it is nearly impossible to measure some states, let alone their high-order derivatives (Shao et al. 2008; Han 2009).

4.3 Simulation results *with* uncertainty

Set $U_f = -0.2$, $U_g = 0.8$. So all parameters are corrupted by fixed uncertainties and Gaussian white noises. It means there exist -20 % fixed parameter uncertainties and large Gaussian white noise. Figure 12 shows the amplitude and frequency example of $0.8 * GWN$. This combination of uncertainties is actually rather big and it can simulate most of the severe flight conditions under big uncertainties including rule uncertainties. Simulation are conducted similarly for 100 s in two cases: IT2-FNN and T1-FNN. The simulation results are shown in Figs. 13, 14 and 15.

When the uncertainty becomes bigger as chosen in this paper, T1-FNN can no longer track the command signal anymore (thus its simulation results are not shown in Figs. 13, 14 and 15) whereas IT2-FNN can still work although less

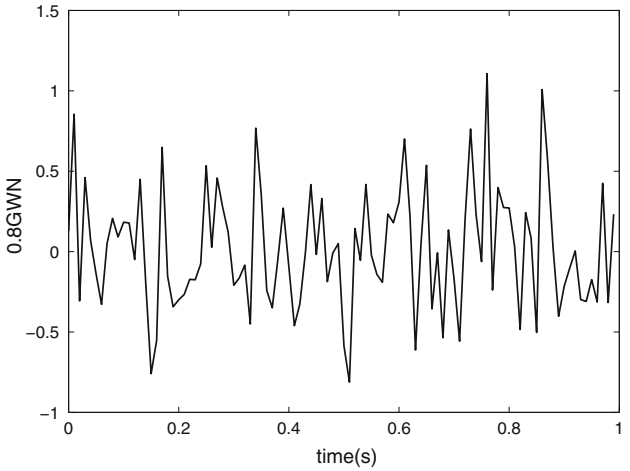

Fig. 12 0.8∗ Gaussian white nosie in 1 s

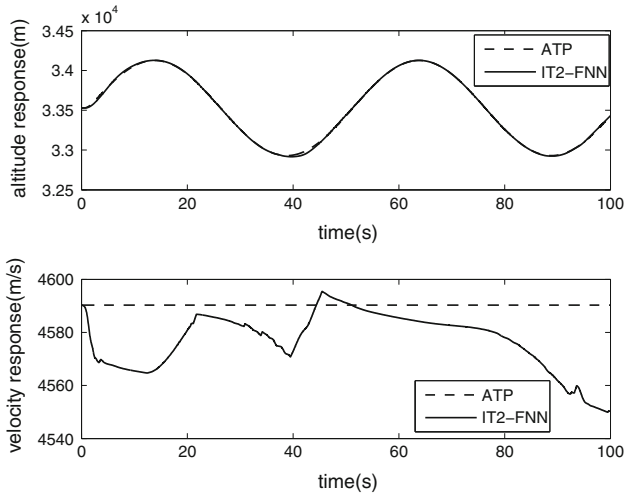

Fig. 13 Command response with uncertainty

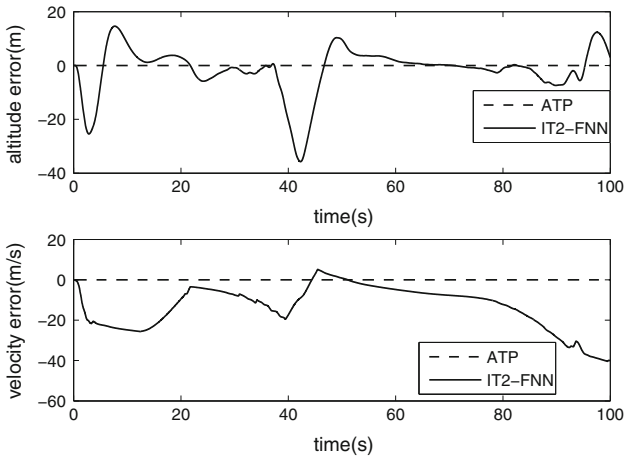

Fig. 14 Tracking error with uncertainty

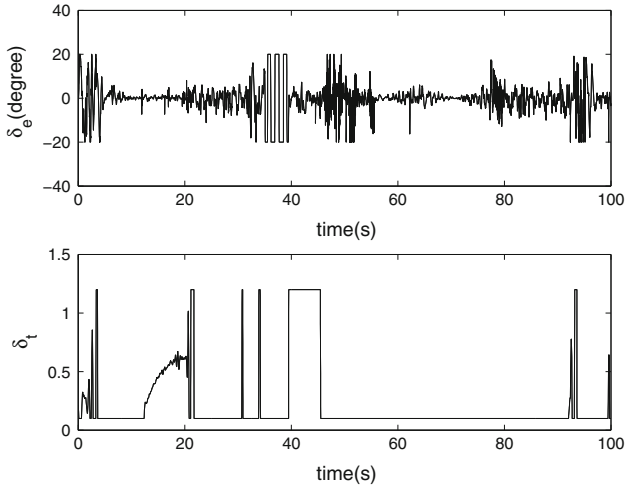

Fig. 15 Control response with uncertainty

effectively. Figures 13 and 14 show that IT2-FNN can still track the altitude command signal but with bigger error both in velocity and altitude. From Fig. 15 we can see δ_e contains strong chattering and δ_t chatters between the upper and lower bound. The chattering is mainly caused by the strong Gaussian white noise.

5 Conclusion

In this paper, a direct interval type-2 fuzzy neural network controller is designed for the first time in hypersonic flight control. When Gaussian MFs with the same center but different widths are used to represent IT2-FS, for a MIMO IT2-FNN, its parameter adaptive law is difficult to obtain. We derive and analyse the adaptive law in detail. The derivation of the adaptive law has something to do with the KMA type reduction method. Tracking differentiator is also introduced to form the ATP and get high-order derivatives of the command signal. Nonlinear state observer is introduced to get the states and their high-order derivatives which are nearly impossible to measure in hypersonic flight condition. Finally simulations are conducted for IT2-FNN and T1-FNN in two cases: without uncertainty and with big uncertainties. Simulation results validate the effectiveness and robustness of the proposed direct adaptive IT2-FNN controller.

In this paper, we just consider the longitudinal rigid body dynamics of the GHFV. The flexible dynamics will be considered in our next research. Meanwhile, we will explore other type reduction method to further reduce the computation cost of IT2-FNN.

Acknowledgments This work was supported by National Natural Science Foundation of China under Grant 61203003, 61273149 and 60904006, Knowledge Innovation Program of the Chinese Academy of Sciences under Grant YYYJ-1122, and Innovation Method Fund of China under Grant 2012IM010200, and B1320133020.

References

Castillo O, Melin P (2008) Type-2 fuzzy logic theory and applications. Springer, Berlin

Chen X (2012) Intelligent control method research for attitude control of mav. PhD thesis, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Science

Fazel Zarandi MH, Gamasaee R (2012) Type-2 fuzzy hybrid expert system for prediction of tardiness in scheduling of steel continuous casting process. Soft Comput 16(8):1287–1302. doi:10.1007/s00500-012-0812-x

Han J (2009) Active disturbance rejection control technique-the technique for estimating and compensating the uncertainties. National Defense Industry Press, Beijing

Huang H, Wan H, Han J (2001) Arranging the transient process is an effective method improved the "robustness, adaptability and stability" of closded-loop system. Control Theory Appl 18(Supply):89–94

Karnik NN, Mendel JM (1998) Type-2 fuzzy logic systems: type-reduction. In: IEEE international conference on systems, man, and cybernetics, pp 2046–2051

Karnik NN, Mendel JM, Liang Q (1999) Type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 7(6):643–658

Keshmiri S, Colgren R, Mirmirani M (2006) Six-dof modeling and simulation of a generic hypersonic vehicle for control and navigation purposes. In: AIAA guidance, navigation, and control conference and exhibit, AIAA

Khalil H (2001) Nonlinear system, 3rd edn. Prentice Hall, NJ

Li C, Yi J, Wang M, Guiqing Z (2012) Monotonic type-2 fuzzy neural network and its application to thermal comfort prediction. Neural Comput Appl. doi:10.1007/s00521-012-1140-x

Li C, Zhang G, Yi J, Wang M (2013) Uncertainty degree and modeling of interval type-2 fuzzy sets: definition, method and application. Comput Math Appl. doi:10.1016/j.camwa.2013.07.021

Liang Q, Mendel JM (2000) Interval type-2 fuzzy logic systems: theory and design. IEEE Trans Fuzzy Syst 8(5):535–550

Lin TC, Liu HL, Kuo MJ (2009) Direct adaptive interval type-2 fuzzy control of multivariable nonlinear systems. Eng Appl Artif Intell 22(3):420–430

Liu Y, Lu Y (2009a) Nonlinear adaptive inversion control with neural network compensation for a longitudinal hypersonic vehicle model. Int Conf Intell Comput Intell Syst, IEEE, pp 264–268

Liu Y, Lu Y (2009b) Nonlinear fuzzy robust adaptive control of a longitudinal hypersonic aircraft model. In: International conference on artificial intelligence and, computational intelligence, pp 31–35

Mendel JM (2007) Type-2 fuzzy sets and systems: an overiew. IEEE Comput Intell Mag 2(1):20–29

Mendel JM, John RI (2002) Type-2 fuzzy sets made simple. IEEE Trans Fuzzy Syst 10(2):117–127

Nurmaini S, Zaiton S, Norhayati D (2009) An embedded interval type-2 neuro-fuzzy controller for mobile robot navigation. In: IEEE international conference on systems, man, and cybernetics, pp 4315–4321

Ougli AE, Lagrat I, Boumhidi I (2008) A type-2 fuzzy adaptive controller of a class of nonlinear system. Int J Inf Math Sci 4(4):282–288

Pan Y, Huang D, Sun Z (2011) Overview of type-2 fuzzy logic control. Control Theory Appl 28(1):13–23

Parker JT, Serrani A, Yurkovich S, Bolender MA, Doman DB (2007) Control-oriented modeling of an air-breathing hypersonic vehicle. J Guid Control Dyn 30(3):846–869

Rehman OU, Fidan B, Petersen IR (2009) Robust minimax optimal control of nonlinear uncertain systems using feedback linearization with application to hypersonic flight vehicles. In: Joint 48th IEEE conference on decision and control and 28th Chinese control conference, IEEE, pp 720–726

Shakiba M, Serrani A (2011) Control oriented modeling of 6-dof hypersonic vehicle dynamics. In: AIAA guidance, navigation, and control conference, AIAA, pp 1–27

Shao L, Liao X, Xia Y, Han J (2008) Stability analysis and synthesis of third order discrete extended state observer. Inf Control 37(2):135–139

Shaughnessy JD, Pinckney SZ, McMinn JD, Cruz CI, Kelley ML (1990) Hypersonic vehicle simulation model: winged-cone configuration. Technical report, NASA Langley Research Center

Singh M, Srivastava S, Gupta JRP, Hanmandlu M (2004) A type-2 fuzzy neural model based control of a nonlinear system. In: IEEE cybernetics and intelligent systems, pp 1352–1356

Wang LX (1996) A course in fuzzy systems & control. Prentice Hall, NJ

Wu H, Mendel JM (2002) Uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems. IEEE Trans Fuzzy Syst 10(5):622–639

Xu H, Mirmirani M, Ioannou P (2004) Adaptive sliding mode control design for a hypersonic flight vehicle. AIAA JGCD 27(5):829–838

Yang F, Yi J, Tan X, Yuan R (2012) Parameter-optimized high-order active disturbance rejection controllers for a generic hypersonic flight vehicle. Commun Inf Sci Manag Eng 2(10):6–12

Yang F, Tan X, Yi J (2013a) A type-2 adaptive fuzzy logic controller for a generic hypersonic flight vehicle. ICIC Express Lett 7(5):1583–1588

Yang F, Yuan R, Yi J, Fan G, Tan X (2013b) Backstepping based type-2 adaptive fuzzy control for a generic hypersonic flight vehicle. In: Chinese intelligent automation conference, pp 180–188

Zaheer SA, Kim JH (2011) Type-2 fuzzy airplane altitude control: a comparative study. In: IEEE international conference on fuzzy systems, pp 2170–2176