Variable Learning Rate Adaptive Sliding Mode
Training Of Type-2 Fuzzy Neural Networks

Mojtaba Ahmadieh Khanesar*
Dept. of Control and Electrical Engineering, Semnan University
Semnan, Semnan, Iran
E-mail: ahmadieh@ieee.org

Okyay Kaynak
Dept. of Electrical and Electronics Engineering, Bogazici University
34342, Istanbul, Turkey
E-mail: okyay.kaynak@boun.edu.tr

Received (to be inserted
Revised by Publisher)

This paper proposes a novel training method for the parameters of a type-2 fuzzy neural network (T2FNN) using sliding mode control theory with an adaptive learning rate. The implemented control structure consists of a conventional (PD) controller in parallel with a T2FNN. The former is responsible to guarantee global asymptotic stability in compact space and to form a sliding behavior. The output of the conventional controller is used to update the parameters of the T2FNN. The use of sliding mode based training method with adaptive learning rate makes it possible to discard the dependency of the design of the controller to priori knowledge about upper bounds of the states of the system and their derivatives. An appropriate Lyapunov function is proposed to analyze the stability of the adaptation law of the parameters of T2FNN and the adaptive learning rate. Sufficient conditions to guarantee the boundedness of the parameters and the achievement of sliding behavior are derived. Using a two-link rigid robot manipulator as a benchmark, it is shown that the proposed structure improves the performance of the conventional PD controller and guides the states of the system towards sliding manifold. In addition, it is shown that T2FNN outperforms its type-1 counterpart which benefits from the same adaptation laws specially in the presence of high levels of noise.

1. Introduction

It is obvious that a sufficiently accurate mathematical modeling of a real world system is quite difficult if not impossible due to the complex dynamical relations of the elements, frictions, environmental changes, heat effects, ageing processes and wears and tears of the actuators, etc. When the mathematical model of the system is unknown, vague and/or uncertain, model-free control approaches are a preferred choice. The model free approaches are conceptually simple and they lessen the need for physical and mathematical knowledge of the system. Among model-free approaches, fuzzy logic systems are the most popular methods because of their power to model expert and experienced human knowledge. When the expert knowledge about a system is not complete and/or sufficient, fuzzy neural networks (FNNs) can be used to learn more about the system than is available through the expert. Basically, FNNs are learning machines that find the parameters of a fuzzy system (i.e., fuzzy sets, fuzzy rules) by exploiting approximation techniques from neural

*P.O. Box: 35131-19111 Pardis Shomare 1, Semnan University, Next to Sokan Park, Semnan, Iran
networks\textsuperscript{1}. In this way, FNNs can simultaneously use expert human knowledge and learn from measured input/output data\textsuperscript{2,3}.

There are two types of FNNs: type-1 and type-2 which differ in types of their membership functions (MFs). The membership functions (MFs) of type-1 fuzzy neural networks are totally crisp, while they are themselves fuzzy in type-2 Fuzzy Neural Networks (T2FNN). Therefore the antecedents and consequent parts in T2FNNs are interval rather than a single value and result in intermittent and uncertain rule bases. This uncertain rule base of T2FNN makes it possible to model and minimize the drawbacks of uncertainties of real world systems. Moreover, T2FNNs have more degrees of freedom and are capable of controlling the system with higher performance specially in the presence of measurement noise\textsuperscript{4,5,6,7,8}. A type-2 membership grade can take values in the closed interval of [0,1] which is called primary membership. There is a secondary membership value corresponding to each primary membership value which defines the grades for the primary memberships\textsuperscript{9,10}. Most researchers in the literature prefer interval T2FNNs in which secondary MFs can be either zero or one\textsuperscript{9,10} in the case which the system becomes computationally more manageable.

Stability is the most important consideration in the design of a controller. The earlier FNN controllers (regardless of using type-1 or type-2 MFs) suffer from the lack of rigorous stability analysis. This is the main reason why some researchers have made use of classical control approaches in the design of FNN based controllers. This combination makes it possible to simultaneously benefit from the flexibility and general function approximation property of FNNs and guaranteed stability analysis of classical control methods\textsuperscript{11}. Feedback-error-learning (FEL) is one of first introductions of classical methods to FNN controllers which is first developed by Kawato in an effort to establish a stable controller which can learn the inverse dynamic of the system under control\textsuperscript{12}. This structure comprises a fixed classical controller to ensure the stability of the system and an adaptive intelligent feed-forward controller in parallel to the fixed controller which improves the performance of the controller. The outputs of the fixed feedback controller is regarded as the error signal and is used to train the inverse model of the plant\textsuperscript{13}. From a control theoretic viewpoint, FEL falls in adaptive control technique categories\textsuperscript{14}. Stability analysis of FEL is considered in several papers. In\textsuperscript{15} the stability of FEL for stable and stably invertible linear systems is proved. In\textsuperscript{16} the stability property of FEL for a class of nonlinear dynamical systems is considered. In addition, stable FEL approaches based on sliding mode control (SMC) method are considered in several papers\textsuperscript{17,18}. To date several implementations of FEL in industrial plants have been reported. For example in\textsuperscript{7} and\textsuperscript{19}, FEL scheme is used to control a n-degree of freedom robotic manipulator and an anti-lock breaking system respectively.

The fusion of SMC with intelligent control approaches based on neural-network, fuzzy-logic etc. has been widely applied to control nonlinear systems in recent years\textsuperscript{18,20,21,22,23}. SMC provides the desired performance of the system in terms of the sliding surface. The goal of this controller is to design the control law such that the states of the system approach the sliding surface and remain on it. This controller is widely known to be a robust control for systems with nonlinearities, uncertain parameters and bounded input disturbances\textsuperscript{24}. Although it shows robustness against nonlinear features of the system, SMC suffers from some drawbacks such as chattering, measurement noise and conservative control signal\textsuperscript{25,26}. In addition, the design of an ideal SMC needs an exact model of the system which is not readily on hand and/or it includes uncertainties almost in all cases\textsuperscript{23}. To avoid these problems, SMC is used in combination with intelligent approaches like FNNs. Because of proven general function approximation property, flexibility and capability of using human knowledge of FNNs, this structure is one of the most important structures used to overcome drawbacks of SMC\textsuperscript{23,17,18}. On the other hand, FNN can benefit from the mathematical stability analysis of SMC if it is used in combination with SMC.

In this study, a novel FEL scheme is proposed. As for many FEL structures, in the proposed approach a T2FNN works in parallel with a PD controller. The output of the PD controller is used to train the T2FNN. The T2FNN is assumed to have two inputs: the error and its derivative. The MFs considered for the system are Gaussian type-2 membership functions with uncertain variance. The sliding mode parameter update rules are derived for such a structure, and the stability of the learning algo-
Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks

The algorithm proposed is proved by using an appropriate Lyapunov function. The required conditions for the stability of the system are also derived. In addition, an adaptive learning rate for the training of the parameters of T2FNN is introduced. Using this adaptive learning rate, it is shown that a prior knowledge about the upper bound of the states of the system and their derivatives is no longer needed. This is the most important superiority of the current approach over similar SMC based training of FNN and T2FNN such as 17,18,27,28. Another benefit of the current approach over the similar methods reported in 17,18,28 is that the training method proposed in this paper does not include any derivative of states of the system. This is also very important feature because taking the derivatives of the states of the system amplifies the noise power significantly. Moreover since the proposed method uses type-2 MFs, it is expected that the system outperforms the type-1 counterpart specially when measurement noise is added to the system. The proposed approach is applied to the control of a two-link rigid robot manipulator. The control problem of two-link robotic manipulator is widely investigated in a number of different papers 29,30,31. The friction forces are also considered. The performance of the T2FNN is also compared with that of its type-1 counterpart in the presence of noise. It is shown T2FNN can control the system with less error when there is measurement noise in the system. It is also shown that the states of the system follow the predefined sliding motion.

2. The Adaptive Fuzzy Neural Network Control Approach

2.1. The Control Scheme and the Type-2 Fuzzy Neural Network Structure

In this section, the proposed sliding mode T2FNN is presented. Figure 1 shows the control scheme.

![Fig. 1. Block diagram of the proposed fuzzy neural network scheme](image-url)
The PD controller is provided to guarantee global asymptotic stability in compact space and as sliding manifold for the states of the system. The PD control law is described as follows:

\[ \tau_{ek} = k_{p} e_{k} + k_{d} \dot{e}_{k}, \quad k = 1, 2 \]  

(1)

in which \( e_{1} = x_{d1} - x_{1} \) and \( e_{2} = x_{d2} - x_{2} \) are the feedback errors, \( x_{d1}, x_{d2} \) are the target values, \( k_{p1}, k_{D1} \) and \( k_{p2}, k_{D2} \) are the gains of the PD controllers. As can be seen from the figure since robotic manipulator has two inputs, two independent FEL structures are used to control the system.

2.1.1 Type-2 Fuzzy Neural Network (T2FNN)

The T2FNN considered here benefits from type-2 membership functions in the premise part and crisp numbers for the consequent part. This structure is called A2-C0 fuzzy system \(^{32}\). The T2FNN has two inputs and one output serving as a feedback controller. Here Takagi-Sugeno (TS) fuzzy structure is used for T2FNN.

The fuzzy if-then rule \( R_{ij} \) of a zero-order TS model with two input variables where the consequent part has a constant value and is defined as follows:

\[ R_{ij;k}: \text{If } e_{k} \text{ is } \tilde{A}_{1i,k} \text{ and } \dot{e}_{k} \text{ is } \tilde{A}_{2j,k}, \text{ then } \tau_{jk} = f_{ij,k}, \quad k = 1, 2 \]

where \( f_{ij,k} \) are constant values which are updated during training. In this paper, for T2FNN, type-2 MF Gaussian with uncertain standard deviation are used. The mathematical expression for the membership function is expressed as:

\[ \tilde{\mu}(x) = \exp \left[ -\frac{(x - c)^2}{\sigma^2} \right] \]

(2)

where \( c \) and \( \sigma \) are the center and the standard deviation of the type-2 MF and \( x \) is the input.

Using uncertain values for \( \sigma \) the type-2 MF has a footprint of uncertainty which is bounded with an upper and lower MF. The upper and the lower MFs are denoted as: \( \tilde{\mu}(x) \) and \( \mu(x) \), respectively. The firing strength of the rule \( R_{ij,k} \) is obtained as a \( T \)-norm of the membership functions in the premise part (by using a multiplication operator):

\[ \tilde{W}_{ij,k} = \mu_{1i,k}(e_{k}) \mu_{2j,k}(\dot{e}_{k}), \quad k = 1, 2 \]

(3)

\[ W_{ij,k} = \pi_{1i,k}(e_{k}) \pi_{2j,k}(\dot{e}_{k}), \quad k = 1, 2 \]

(4)

The Gaussian membership functions \( \mu_{1i,k}(e_{k}) \), \( \pi_{1i,k}(e_{k}) \), \( \mu_{2j,k}(\dot{e}_{k}) \), and \( \pi_{2j,k}(\dot{e}_{k}) \) of the inputs \( e_{k} \) and \( \dot{e}_{k} \) in the above expression are of the following form:

\[ \mu_{1i,k}(e_{k}) = \exp \left[ -\frac{(e_{k} - c_{1i,k})^2}{\sigma_{1i,k}} \right], \quad k = 1, 2 \]

(5)

\[ \pi_{1i,k}(e_{k}) = \exp \left[ -\frac{(c_{1i,k} - e_{k})^2}{\sigma_{1i,k}} \right], \quad k = 1, 2 \]

(6)

\[ \mu_{2j,k}(\dot{e}_{k}) = \exp \left[ -\frac{c_{2j,k} - \dot{e}_{k}}{\sigma_{2j,k}} \right]^2, \quad k = 1, 2 \]

(7)

\[ \pi_{2j,k}(\dot{e}_{k}) = \exp \left[ -\frac{\dot{e}_{k} - c_{2j,k}}{\sigma_{2j,k}} \right]^2, \quad k = 1, 2 \]

(8)

where the real constants \( \sigma_{1,2}, \sigma > 0 \) and \( c \) are among the tunable parameters of the above T2FNN structure. Hence, (3) and (4) can be rewritten as follows:

\[ \tilde{W}_{ij,k} = \exp \left[ -\frac{(e_{k} - c_{1i,k})^2}{\sigma_{1i,k}} \right] - \frac{(\dot{e}_{k} - c_{2j,k})^2}{\sigma_{2j,k}}, \quad k = 1, 2 \]

(9)

\[ W_{ij,k} = \exp \left[ -\frac{(c_{1i,k} - e_{k})^2}{\sigma_{1i,k}} \right] - \frac{(c_{2j,k} - \dot{e}_{k})^2}{\sigma_{2j,k}}, \quad k = 1, 2 \]

(10)

The computational output of A2-C0 structure as it is proposed in \(^{33}\) is used to determine the output of TSK T2FNN.

\[ \tau_{jk} = \frac{q_{k} \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij,k} \tilde{W}_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} \tilde{W}_{ij,k}} + \frac{(1 - q_{k}) \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij,k} W_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}}, \quad k = 1, 2 \]

(11)

After the normalization of (11), the output signal of the fuzzy neural network will acquire the following form:

\[ \tau_{jk} = q_{k} \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij,k} \tilde{W}_{ij,k} \]

(12)

\[ + \frac{(1 - q_{k}) \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij,k} W_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}}, \quad k = 1, 2 \]

where \( \tilde{W}_{ij,k} \) and \( W_{ij,k} \) are the normalized values of the lower and the upper outputs corresponding to \( ij \), \( k^{th} \) node of the second hidden layer of the network.

\[ \tilde{W}_{ij,k} = \frac{W_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}} \]

(13)

\[ W_{ij,k} = \frac{W_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}} \]

(14)
The adaptation law for the tunable parameters is considered such that $\sigma$, $\bar{\sigma}$, and $c$ of the Gaussian membership functions are bounded as follows:

$$\|\sigma_{1,k}\| \leq \bar{B}_{\sigma}, \quad \|\sigma_{2,k}\| \leq \bar{B}_{\sigma}, \quad \|c_{1,k}\| \leq B_c, \quad \|c_{2,k}\| \leq B_c, \quad k = 1, 2$$

where $\bar{B}_{\sigma}$, $\bar{B}_{\sigma}$, $B_c$ and $B_f$ are some unknown positive constants.

The adaptation law for the tunable parameters is considered such that $\sigma$, $\bar{\sigma}$, and $c$ of the Gaussian membership functions are bounded as follows:

$$\|\sigma_{1,k}\| \leq \bar{B}_{\sigma}, \quad \|\sigma_{2,k}\| \leq \bar{B}_{\sigma}, \quad \|c_{1,k}\| \leq B_c, \quad \|c_{2,k}\| \leq B_c, \quad k = 1, 2$$

$$\|\sigma_{1,k}\| \leq \bar{B}_{\sigma}, \quad \|\sigma_{2,k}\| \leq \bar{B}_{\sigma}, \quad \|c_{1,k}\| \leq B_c, \quad \|c_{2,k}\| \leq B_c, \quad k = 1, 2$$

where $\bar{B}_{\sigma}$, $\bar{B}_{\sigma}$, $B_c$ and $B_f$ are some unknown positive constants.

Similar to the previous case, it follows that $0 < \bar{W}_{ij,k} < 1$ and $0 < \bar{W}_{ij,k} < 1$. In addition, it can be easily seen that $\sum_{i=1}^{t} \sum_{j=1}^{t} \bar{W}_{ij,k} = 1$ and $\sum_{i=1}^{t} \sum_{j=1}^{t} \bar{W}_{ij,k} = 1$. It is also considered that, $\tau_k$ and $\bar{\tau}_k$ are bounded signals too, i.e.

$$|\tau_k(t)| \leq B_{\tau}, \quad |\bar{\tau}_k(t)| \leq B_{\tau} \quad \forall t \quad \text{and} \quad k = 1, 2$$

where $B_{\tau}$ and $B_{\bar{\tau}}$ are two unknown positive constants. In similar previous papers e.g. \cite{17,18,27,28} it is assumed that the upper bounds defined by (15), (18), (19) and (20) are known. But in the current paper, using an adaptive learning rate, these parameters are no more needed to be known. This is the main improvement of the current approach with respect to above mentioned papers.

### 2.2. The Sliding Mode Learning Algorithm

Using the sliding mode control theory principles the zero value of the learning error coordinate $\tau_c(t)$ can be defined as time-varying sliding surface, i.e.,

$$S_{ck}(\tau_{fk}, \tau_k) = \tau_{ck}(t) = \tau_{fk}(t) + \tau_k(t) \quad k = 1, 2$$

which is the condition that the fuzzy neural network is trained to become a nonlinear regulator to obtain
the desired response during the tracking-error convergence movement by compensating for the nonlinearity of the controlled plant.

The sliding surface for the nonlinear system under control $S_{p1}(e_1, \dot{e}_1)$ and $S_{p2}(e_2, \dot{e}_2)$ are defined as:

$$S_{pk}(e_k, \dot{e}_k) = \dot{e}_k + \chi_k e_k, \quad k = 1, 2 \quad (22)$$

with $\chi_k$ being a constant determining the slope of the sliding surface.

**Definition:** A sliding motion will appear on the sliding surface $S_{ck}(\tau_{jk}, \tau_k) = \tau_{ck}(t) = 0$ after a time $t_h$, if the condition $S_{ck}(t)\dot{S}_{ck}(t) = \tau_{ck}(t) \dot{\tau}_{ck}(t) < 0$ is satisfied for all $t$ in some nontrivial semi-open subinterval of time of the form $[t, t_h) \subset (-\infty, t_h)$.

It is desired to design an online learning algorithm for the parameters of FNN such that the sliding mode condition of the above definition is enforced.

### 2.2.1. The Parameter Update Rules For T2FNN

**Theorem 1:** If the adaptation laws for the parameters of the T2FNN are chosen as:

$$\dot{\hat{c}}_{1i,k} = -\frac{\sigma^2_{1i,k}}{e_k - c_{1i,k}} \beta_{1k} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (23)$$

$$\dot{\hat{c}}_{2i,k} = -\frac{\sigma^2_{2i,k}}{e_k - c_{2i,k}} \beta_{1k} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (24)$$

$$\dot{\hat{\sigma}}_{1i,k} = -\beta_{1k} \frac{\sigma^2_{1i,k}}{(e_k - c_{1i,k})^2} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (25)$$

$$\dot{\hat{\sigma}}_{2i,k} = -\beta_{1k} \frac{\sigma^2_{2i,k}}{(e_k - c_{2i,k})^2} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (26)$$

$$\dot{\hat{\sigma}}_{1i,k} = -\beta_{1k} \frac{\sigma^3_{1i,k}}{(e_k - c_{1i,k})^3} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (27)$$

$$\dot{\hat{\sigma}}_{2i,k} = -\beta_{1k} \frac{(\sigma_{2i,k})^3}{(e_k - c_{2i,k})^3} \text{sgn}(\tau_{ck}), \quad k = 1, 2 \quad (28)$$

The dynamics of a robot manipulator describes how the robot moves in response to actuator torques applied to the joints of the robot. An two-link rigid robot manipulator without considering friction or other disturbances can be described by:

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F_C(q\dot{\dot{q}}) = \tau \quad (32)$$

where $q \in \mathbb{R}^2$ is a vector of generalized coordinates, $D(q)$ is the $n \times n$ inertia matrix, $C(q, \dot{q})\dot{q}$ is the $n$-vector of centrifugal and Coriolis forces, $G(q)$ is the gravitational force vector, $F_C$ is the $n \times 1$ positive definite diagonal matrix which contains the viscous friction forces of the robot joints, and $\tau$ is the control torque vector. Consider a planar robot with two rigid links and two rigid revolute joints shown in Fig. 2. With reference to the symbols listed in Table 1, we present below the entries of the robot dynamics.

### 3. The Dynamics Of The Robotic Manipulator

<table>
<thead>
<tr>
<th>Table 1. Values of the parameters of the manipulator and their definitions.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_{1i,k}$</td>
</tr>
<tr>
<td>$\gamma_{1k}$</td>
</tr>
</tbody>
</table>
The entries of the gravitational torque vector $G(q)$ are given by

$$G_1 = m_1 l_1 g \sin q_1 + m_2 g [l_2 \sin (q_1 + q_2) + l_1 \sin q_1]$$
$$G_2 = m_2 l_2 g \sin (q_1 + q_2)$$

(36)

The coefficients of the viscous friction are

$$b_1 = 2.288, \quad b_2 = 0.175, \quad f_{c2} = 1.734,$$
$$f_{c1} = \begin{cases} 7.17 & \text{if } \dot{q}_1 > 0 \\ 8.049 & \text{if } \dot{q}_1 < 0 \end{cases}$$

(37)

Experiments showed that there exist static and Coulomb friction at the motor joints and they depend in a complex manner on the joint position and velocity.

4. Simulation Studies

The proposed approach is applied to control the robotic manipulator described in previous section to test its implementability and effectiveness. The sampling period for the simulations is selected as 1ms. The implemented control structure is as is shown in Fig. 1. As it is mentioned earlier the PD controller is provided both to guarantee global asymptotic stability in compact space and as the sliding manifold of the controlled system. The output of PD controller acts as a guideline for the training of T2FNN. In this way the T2FNN is able to gradually replace the conventional controller.

The tracking performances of the T2FNN working in parallel with a PD controller and a PD controller alone are compared in Fig. 3. The performance of the proposed learning method is compared with the case in which the PD controller is working alone. As can be seen from this figure, when the T2FNN is used in parallel with a conventional PD controller, the system does not have any overshoot, steady state error and the settling time of the system is decreasing simultaneously.
Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks

Fig. 3. a,b: The comparison of the step response of the proposed control algorithm and PD controller. c,d: The comparison of tracking the reference signal of the proposed control algorithm and PD controller.

Fig. 4. a: The output of classical PD controller ($\tau_1$) and output of T2FNN ($\tau_{f1}$) in the proposed control scheme. b: The output of classical PD controller ($\tau_2$) and output of T2FNN ($\tau_{f2}$) in the proposed control scheme.
Figure 4 shows the control signal of the PD controller and T2FNN in the proposed control algorithm. As can be seen from the figure, the signal $\tau_{c1}$ and $\tau_{c2}$ which are the outputs of classical PD controllers tend to zero. When the output of the PD controller is near zero, T2FNN takes the responsibility of controlling the system.

Figure 5 shows the phase portrait of the error dynamics of the system. This figure indicates that the error dynamics can reach the sliding lines and remain on them when the proposed method is used.

As mentioned earlier, the proposed control scheme benefits from an adaptive learning rate. Figure 6 shows how the learning rates ($\alpha_1$ and $\alpha_2$) are updated during training. As can be seen from the figure, the initial value of the learning rates are small positive number. Using the adaptation law as is proposed in (30), the learning rates are updated. In this way, unnecessarily large values for learning rate ($\alpha$) are prevented and no priori knowledge about the upper bound of states of the system and their derivatives are needed. This is the main contribution of the current work over 17,18,27,28 in which the selection of the learning rate need a priori knowledge about the mentioned upper bounds.
Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks

Figure 6. a: The updated values of adaptive learning rate ($\alpha$) of controller for $q_1$. b: The updated values of adaptive learning rate ($\alpha$) of controller for $q_2$

Figure 7 shows the percentage improvement of T2FNN over type-1 counterpart as more noise is added to the measured angles. As can be seen from the figure when the level of the noise power is low the performance of T2FNN and T1FNN are very similar. But as the level of the measurement noise increases T2FNN acts much better when compared to T1FNN. This is an expected results and is quite consistent with what is previously reported in type-2 literature specially $^6,^5$.

5. CONCLUSIONS

This paper investigates a novel FEL structure. The proposed control structure consists of a conventional PD controller serving in parallel with a T2FNN controller. The PD controller is responsible to guarantee the global asymptotic stability of the system, as well as introducing a sliding line for the controlled system. The parameters of the T2FNN are updated using the output of the PD controller. As the parameters of the T2FNN are updated, it takes the responsibility of controlling the system and the output of the PD controller tends to zero. The proposed new learning algorithm makes direct use of the sliding mode theory and establishes a sliding motion in terms of the T2FNN parameters, leading the classical control signal towards zero. Since there exists a direct relationship between the output of the classical controller and the sliding manifold, when the output of the PD controller tends to zero an sliding behavior for the system is achieved. In addition, the proposed control structure benefits from an adaptive learning rate. Using the adaptive learning rate, the existence of upper bounds for the states of
the system is enough and the exact values of them is no longer needed. Furthermore, it is possible to set the value of learning rate to a small positive value and its desired value is obtained automatically during training. In this way, unnecessarily large values for the learning rates can be avoided. The proposed method is tested on a robotic manipulator with friction. Since this system is a MIMO system with two inputs, two FEL structures are used. The simulation studies show that when the T2FNNs are used in parallel with conventional PD controllers, the steady state error of PD controller becomes zero and the system performs faster. In addition, in the simulation part it is shown that the states of the system tend to the sliding line and remain on it in a finite time. In order to investigate the effect of using type-2 MFs, a uniformly distributed noise is added to the measured values of angles of the manipulator. It is shown that when the power of noise is small the performance of T2FNN and its type-1 counterpart are quite similar. But as more noise is injected to the system T2FNN outperforms T1FNN in terms of integral of squared error. These result suggest that when the noise level in a process is low T1FNN may be a better choice because of its simplicity and easier implementation. But if the noise level in a process is high, it is more logical to use T2FNN, because its more degrees of freedom results in a higher performance for the closed loop system.

Appendix A

For simplicity, in order to have shorter equations the indices $k$ is used instead of 1 and 2 in the following equations. For instance, instead of using $A_{11,1}$ and $A_{11,2}$ the notation $A_{11,k}$ is used and instead of using $e_1$ and $e_2$, $e_k$ is used and so on. The stability analysis of the two controllers T2FNN1 and T2FNN2 are quite similar and this makes it possible to use the above mentioned notation. Considering (5)-(8) we have:

$$\dot{\lambda}_{11,k}^{(e_k)} = -2A_{11,k}\dot{A}_{11,k}^{(e_k)}$$  \hspace{1cm} (38)$$

$$\ddot{\lambda}_{11,k}^{(e_k)} = -2U_{11,k}\dot{U}_{11,k}^{(e_k)}$$  \hspace{1cm} (39)$$

$$\dot{\tau}_{2j,k}^{(\dot{e}_k)} = -2\tau_{2j,k}^{(\dot{e}_k)}$$  \hspace{1cm} (40)$$

$$\ddot{\tau}_{2j,k}^{(e_k)} = -2U_{2j,k}\dot{U}_{2j,k}^{(e_k)}$$  \hspace{1cm} (41)$$

In addition considering (13) and (14), the time derivatives of $\dot{W}_{ij,k}$ and $\ddot{W}_{ij,k}$ are obtained as:

$$\dot{W}_{ij,k} = \frac{W_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}} \Rightarrow \dot{\dot{W}}_{ij,k} = \frac{(\mu_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k))' (\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})}{(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})^2}
- \frac{(W_{ij,k})' (\sum_{i=1}^{I} \sum_{j=1}^{J} \mu_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k))'}{(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})^2}
\hspace{1cm} (42)$$

$$\ddot{W}_{ij,k} = \frac{\ddot{W}_{ij,k}}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}} \Rightarrow \dddot{W}_{ij,k} = \frac{(\dot{\mu}_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k) + \mu_{11,k}(e_k)\dot{\mu}_{2j,k}(\dot{e}_k))' (\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})}{(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})^2}
- \frac{(\dot{W}_{ij,k})' (\sum_{i=1}^{I} \sum_{j=1}^{J} \mu_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k))'}{(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})^2}
\hspace{1cm} (43)$$

Since $\dddot{W}_{ij,k} = (\dddot{W}_{ij,k})/(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})$ and $\dddot{W}_{ij,k} = (\dddot{W}_{ij,k})/(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})$ we have:

$$\dddot{W}_{ij,k} = \frac{\dot{\mu}_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k) + \mu_{11,k}(e_k)\dot{\mu}_{2j,k}(\dot{e}_k)}{\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k}}
- \frac{(\dddot{W}_{ij,k})' (\sum_{i=1}^{I} \sum_{j=1}^{J} \mu_{11,k}(e_k)\mu_{2j,k}(\dot{e}_k))'}{(\sum_{i=1}^{I} \sum_{j=1}^{J} W_{ij,k})^2}$$

Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks
Using (48), we have:

$$
\frac{\dot{x}_1 - \dot{c}_{11,k}}{\sigma_{11,k}} = \frac{\dot{c}_k - \dot{c}_{11,k}}{\sigma_{11,k} + \sigma_{11,k}} = \frac{\dot{c}_k - \dot{c}_{11,k}}{\sigma_{11,k}} \left( 1 - \sigma_{11,k} - \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

and:

$$
\frac{e_k - c_{11,k}}{\sigma_{11,k}} = \frac{e_k - c_{11,k}}{\sigma_{11,k} + \sigma_{11,k}} = \frac{e_k - c_{11,k}}{\sigma_{11,k}} \left( 1 - \sigma_{11,k} + \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

Using (48), we have:

$$
U_{11,k} \dot{U}_{11,k} = \frac{(\dot{x}_1 - \dot{c}_{11,k}) (e_k - c_{11,k})}{\sigma_{11,k}^2} \left( 1 - \sigma_{11,k} + \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

and further:

$$
U_{11,k} \dot{U}_{11,k} = \frac{(e_k - c_{11,k}) (e_k - c_{11,k})}{\sigma_{11,k}^2} \left( -\sigma_{11,k} + \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks

$$
\dot{A}_{11,k} = \frac{(\dot{c}_k - \dot{c}_{11,k}) (e_k - c_{11,k})}{\sigma_{11,k}^2} \left( 1 - \sigma_{11,k} - \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

and further:

$$
\dot{W}_{ij,k} = -\dot{W}_{ij,k} + \dot{W}_{ij,k} + \dot{W}_{ij,k} \sum_{i=1}^{I} \sum_{j=1}^{J} \dot{W}_{ij,k} \dot{N}_{ij,k}
$$

$$
\dot{W}_{ij,k} = -\dot{W}_{ij,k} + \dot{W}_{ij,k} + \dot{W}_{ij,k} \sum_{i=1}^{I} \sum_{j=1}^{J} \dot{W}_{ij,k} \dot{N}_{ij,k}
$$

in which:

$$
\dot{N}_{ij,k} = 2A_{11} \dot{A}_{11,k} + 2A_{2j,k} \dot{A}_{2j,k} \dot{N}_{ij,k} = 2U_{11,k} \dot{U}_{11,k} + 2U_{2j,k} \dot{U}_{2j,k}
$$

$$
\dot{A}_{11,k} = \frac{(\dot{c}_k - \dot{c}_{11,k}) (e_k - c_{11,k})}{\sigma_{11,k}^2} \left( 1 - \sigma_{11,k} - \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

$$
\dot{U}_{11,k} = \frac{(\dot{c}_k - \dot{c}_{11,k}) (e_k - c_{11,k})}{\sigma_{11,k}^2} \left( 1 - \sigma_{11,k} + \sigma_{11,k} + \frac{(\sigma_{11,k} - \sigma_{11,k})^2}{\sigma_{11,k}} + H.O.T \right)
$$

it is possible to use Taylor series expansion to obtain following equations.
and similarly using (49) we obtain that:

\[ U_{2j,k} \dot{U}_{2j,k} = A_{2j,k} \dot{A}_{2j,k} + \frac{(\dot{e}_k - c_{2j,k}) (e_k - c_{2j,k})}{\sigma_{2j,k}} \left( -\frac{\sigma_{2j,k} - \sigma_{2j,k}}{\sigma_{2j,k}} + \frac{(\sigma_{2j,k} - \sigma_{2j,k})^2}{\sigma_{2j,k}} + H.O.T. \right) \]

\[-\frac{(\dot{e}_k - c_{2j,k}) \sigma_{2j,k}}{\sigma_{2j,k}} \left( \frac{\sigma_{2j,k} - \sigma_{2j,k}}{\sigma_{2j,k}} + \frac{(\sigma_{2j,k} - \sigma_{2j,k})^2}{\sigma_{2j,k}} + H.O.T. \right) \]

In order to analyze the stability of the controller with adaptive learning rate, the following Lyapunov function is proposed.

\[ V_{c,k} = \frac{1}{2} \tau_{ck}(t) + \frac{1}{2\gamma_{k}} (\alpha_k - \alpha_k^*)^2 \] (50)

The time derivative of the Lyapunov function (50) is derived as:

\[ \dot{V}_{c,k} = \tau_{ck} \dot{\tau}_{ck} = \tau_{ck} (\dot{\tau}_{fk} + \dot{\tau}_k) + \frac{\dot{\alpha}_k}{\gamma_{k}} (\alpha_k - \alpha_k^*) \] (51)

Since:

\[ \tau_{fk} = \frac{q_k \sum_{i=1}^J \sum_{j=1}^J f_{ij,k} \dot{W}_{ij,k}}{\sum_{i=1}^J \sum_{j=1}^J W_{ij,k}} + \frac{(1 - q_k) \sum_{i=1}^J \sum_{j=1}^J f_{ij,k} \dot{W}_{ij,k}}{\sum_{i=1}^J \sum_{j=1}^J W_{ij,k}} \]

\[ = q_k \sum_{i=1}^J \sum_{j=1}^J f_{ij,k} \dot{W}_{ij,k} + (1 - q_k) \sum_{i=1}^J \sum_{j=1}^J f_{ij,k} \dot{W}_{ij,k} \] (52)

and:

\[ \dot{\tau}_{fk} = q_k \sum_{i=1}^J \sum_{j=1}^J (\dot{f}_{ij,k} \dot{W}_{ij,k} + f_{ij,k} \ddot{W}_{ij,k}) + (1 - q_k) \sum_{i=1}^J \sum_{j=1}^J (\dot{f}_{ij,k} \dot{W}_{ij,k} + f_{ij,k} \ddot{W}_{ij,k}) \] (53)

one obtains:

\[ \dot{\tau}_k = q_k \sum_{i=1}^J \sum_{j=1}^J \left( \frac{-\dot{W}_{ij,k} \dot{K}_{ij,k} + \dot{W}_{ij,k} \sum_{i=1}^J \sum_{j=1}^J \dot{W}_{ij,k} \dot{K}_{ij,k}}{\sum_{i=1}^J \sum_{j=1}^J \dot{W}_{ij,k}} \right) f_{ij,k} + \dot{W}_{ij,k} \dot{f}_{ij,k} \]

\[ + (1 - q_k) \sum_{i=1}^J \sum_{j=1}^J \left( \frac{-\dot{W}_{ij,k} \dot{K}_{ij,k} + \dot{W}_{ij,k} \sum_{i=1}^J \sum_{j=1}^J \dot{W}_{ij,k} \dot{K}_{ij,k}}{\sum_{i=1}^J \sum_{j=1}^J \dot{W}_{ij,k}} \right) f_{ij,k} + \dot{W}_{ij,k} \dot{f}_{ij,k} \] (54)

Considering adaptation for sigma and center we have:

\[ \dot{V}_{c,k} = \tau_{ck} \left( \sum_{i=1}^J \sum_{j=1}^J \left( 2 \left( -\dot{W}_{ij,k} \frac{\dot{e}_k}{\sigma_{1i,k}} A_{1i,k} + \frac{\dot{e}_k}{\sigma_{2j,k}} A_{2j,k} \right) f_{ij,k} + \dot{W}_{ij,k} \dot{f}_{ij,k} \right) + \left( 2 \left( -\dot{W}_{ij,k} \frac{\dot{e}_k}{\sigma_{1i,k}} A_{1i,k} + \frac{\dot{e}_k}{\sigma_{2j,k}} A_{2j,k} \right) f_{ij,k} + \dot{W}_{ij,k} \dot{f}_{ij,k} \right) + (1 - q_k) \sum_{i=1}^J \sum_{j=1}^J \left( 2 \left( -\dot{W}_{ij,k} \frac{\dot{e}_k}{\sigma_{1i,k}} A_{1i,k} + \frac{\dot{e}_k}{\sigma_{2j,k}} A_{2j,k} \right) f_{ij,k} + \dot{W}_{ij,k} \dot{f}_{ij,k} \right) + \frac{\dot{\alpha}_k}{\gamma_{k}} (\alpha_k - \alpha_k^*) \right) \] (55)
Considering assumptions (18) and (19), (55) can be rewritten as:

\[
\dot{V}_{c,k} \leq 4B_r |\tau_{ck}| + \tau_{ck} \left( \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij,k}(q\overline{W}_{ij,k} + (1 - q_k)\overline{W}_{ij,k}) + \hat{r} \right) + \frac{\hat{\alpha}_k}{\gamma_{1k}}(\alpha_k - \alpha^*_k) \\
\leq 4B_r |\tau_{ck}| - \alpha^*_k |\tau_{ck}| + (\alpha^*_k - \alpha_k) |\tau_{ck}| + |\tau_{ck}| B_r + \frac{\hat{\alpha}_k}{\gamma_{1k}}(\alpha_k - \alpha^*_k) < 0 \tag{56}
\]

in which:

\[
B_r = B_f \left( \frac{B_e^2 + B_e^2 + B_e B_e + B_e B_e}{B_e^2} \right) \tag{57}
\]

using the adaptation law for \( \alpha \) as:

\[
\hat{\alpha}_k = \gamma_{1k} |\tau_{ck}| - \nu_k \gamma_{1k} \alpha_k
\]

and taking \( \alpha^*_k \) as:

\[
B_e + 4B_r < \frac{1}{2} \alpha^*_k
\]

we have:

\[
\dot{V}_{c,k} \leq -\frac{1}{2} \alpha^*_k |\tau_{ck}| + \nu_k \alpha_k(\alpha_k - \alpha^*_k)
\]

\[
= -\frac{1}{2} \alpha^*_k |\tau_{ck}| + \nu_k (\alpha_k - \alpha^*_k)^2 - \frac{\nu_k \alpha^2_k}{4} \tag{58}
\]

Furthermore:

\[
\dot{V}_{c,k} \leq -\frac{1}{2} \alpha^*_k |\tau_{ck}| + \nu_k (\alpha_k - \alpha^*_k)^2 \tag{59}
\]

Therefore, the Lyapunov function \( \tau_{ck} \) converges exponentially until \( |\tau_{ck}| \leq \frac{2\nu_k}{\alpha^*_k}(\alpha_k - \alpha^*_k)^2 \) and the parameters of the controller are bounded. Consequently the system states converge to a compact set \( \mathcal{R} \) in which:

\[
\mathcal{R} = \left\{ \tau_{ck} | |\tau_{ck}| \leq \frac{2\nu_k}{\alpha^*_k}(\alpha_k - \alpha^*_k)^2 \right\} \tag{60}
\]

It should be noted that this region can be chosen to be as small as desired by choosing a proper value for \( \nu_k \). Consequently, \( \tau_{ck} \) can be made as small as desired.

References
Variable Learning Rate Adaptive Sliding Mode Training Of Type-2 Fuzzy Neural Networks