Machine Intelligence & Robotic Control, Vol. 1, No. 1, 27–41 (1999)

Paper

Design and Stability Analysis of a New Sliding-Mode Fuzzy Logic Controller of Reduced Complexity

Spyros G. Tzafestas† and Gerasimos G. Rigatos‡

Abstract: This paper derives and analyzes a new robust fuzzy-logic sliding-mode controller of the diagonal type, which does not need prior design of the rule base. The basic objective of the controller is to keep the system on the sliding surface so as to ensure the asymptotic stability of the closed-loop system. The control law consists of two rules: i) IF sgn($\boldsymbol{e}(t)\dot{\boldsymbol{e}}(t)$) < 0 THEN maintain the control action, and ii) IF sgn($\boldsymbol{e}(t)\dot{\boldsymbol{e}}(t)$) > 0 THEN change the control action, where $\boldsymbol{e}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_d(t)$ is the system state error, and the control action can be either an increase or decrease of the control signal, which is realized through the use of fuzzy rules. The proposed controller, which does not need the prior knowledge of the system model and the prior shape design of the membership functions, was tested, by simulation, on linear and nonlinear systems. The performance was in all cases satisfactory (very fast trajectory tracking, no chattering). Of course as in traditional control, there was a trade-off between the rise-time and the overshoot of the system response.

Keywords: Sliding-Mode Control, Fuzzy Logic Control, Sliding-Mode Fuzzy Logic Control, Sliding Surface, Iterative Learning Control, Rule Base

1. Introduction

FOR a large category of second-order systems, fuzzy logic controllers (FLCs) are designed using the fuzzy phase plane which is defined by the fuzzy values of e and $\dot{e}$ [1]–[4]. The fuzzy rules of an FLC of this type produce a fuzzy control signal u employing the fuzzy values of e and $\dot{e}$. The usual heuristic approach for the derivation of these control laws is to separate the fuzzy phase plane into two semi-planes with a sliding line. This means that the FLC will have a "diagonal" form. Each semi-plane is used to define only positive or only negative values of the fuzzy control signal u. The magnitude of a specific positive/negative fuzzy value of the control input u is deduced from the distance of the fuzzy state vector $[e, \dot{e}]^T$ from the sliding line. This implies that the absolute value of the control signal u increases/decreases with the increasing/decreasing distance of the state vector from the sliding line. This method of design is similar to the design of conventional *Sliding Mode Control* (SMC) with Boundary Layer (BL) which is a technique of robust control [5]–[12].

Because of the similarity between the diagonal FLC and SMC we can redefine the diagonal-FLC in terms of SMC with BL, and verify its stability and robustness. This is done in this paper.

Furthermore a modified approach to the design of Sliding-Mode-Fuzzy-Logic-Controllers (SMFLC) will be presented. The proposed controller, named *Reduced Complexity-SMFLC* (RC-SMFLC) is characterized by its simplicity, and can facilitate significantly the solution of control problems for nonlinear systems with model uncertainties, parameter fluctuations, and disturbances. Comparing with already existing schemes of SMFLC, it will be shown that the number of the required rules is drastically reduced. No previous knowledge about the system's model is required. Finally, the similarity between RC-SMFLC and other heuristic control techniques, like Iterative Learning Control is investigated. The efficiency of RC-SMFLC was verified in several test systems (linear and nonlinear). Here the results obtained by applying RC-SMFLC to an arc-welding system are demonstrated.

* Received September 7, 1999; accepted October 30, 1999.

† Intelligent Robotics and Automation Laboratory, Department of Electrical and Computer Engineering, National Technical University of Athens, Zografou, 15773, Athens, Greece. E-mail: tzafesta@softlab.ntua.gr

‡ Intelligent Robotics and Automation Laboratory, National Technical University of Athens, Zografou, 15773, Athens, Greece.

2. Brief Review of Sliding Mode Control

2.1 General principles

Consider the nonlinear, non-autonomous open-loop system:

$$x^{(n)}(t) = f(x,t) + b(x,t)u + \tilde{d}, \quad (x^{(n)} = d^n x/dt^n) \quad (1)$$

where $\boldsymbol{x}(t) = (x, \dot{x}, \cdots, x^{(n-1)})^T$ is the state vector, $\tilde{d}(x,t)$ are time-dependent disturbances with known upper bound, and $f(x,t)$ and $b(x,t)$ are nonlinear functions. Assume also that ν_u are the unmodeled frequencies of the system.

The tracking problem for Eq. (1) is to find a control law for a desirable trajectory $x_d(t)$ such that the tracking error $x(t) - x_d(t)$ tends to zero independently from the uncertainties of the systems.

The tracking error of the state vector is $\boldsymbol{e}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_d(t) = (e, \dot{e}, \cdots, e^{(n-1)})^T$ and we define a sliding surface $s(x,t) = 0$, where

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} e = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-1-k)} \quad (2)$$

with initial condition $\boldsymbol{e}(0) = \boldsymbol{0}$. This means that, by setting $s(x,t) = 0$, we have an homogenous differential equation which has a unique solution $\boldsymbol{e} = \boldsymbol{0}$. Consequently an appropriate control rule u is to keep the state vector $\boldsymbol{e}$ on the

Paper No. 1345–269X/99/010027-15

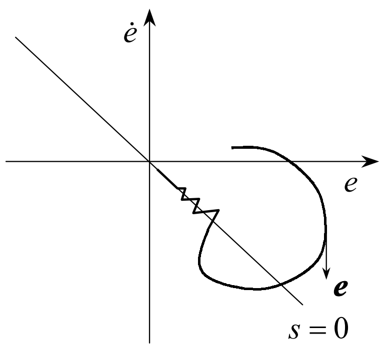

Fig. 1 Error state vector in the sliding mode

sliding surface $s(x,t) = 0$. To this end we define a Lyapunov function

$$V = \frac{1}{2}s^2$$

with $V(0) = 0$ and $V(s) > 0$ for $s > 0$. An efficient condition for the stability of the system is

$$\dot{V} = \frac{1}{2}\frac{d}{dt}s^2 \leq -\eta|s|$$

which leads to the convergence condition

$$s\dot{s} \leq -\eta|s| \Rightarrow s\dot{s} \leq -\eta \cdot \mathrm{sgn}(s)s \Rightarrow \dot{s} \cdot \mathrm{sgn}(s) \leq -\eta. \quad (3)$$

If $\eta > 0$, then the system is driven to the sliding mode. This means that if the state trajectory $[e, \dot{e}]^T$ has reached the sliding surface $s = 0$, then it remains on it while at the same time it slides to the origin $\boldsymbol{e} = 0$ independently of the system's parametric uncertainties and disturbances.

For a second-order system, convergence to the sliding mode is illustrated in the $(e, \dot{e})$-plane in **Fig. 1**.

The first step in the design of an SMC is the selection of the parameter λ. The linear differential equation of Eq. (2) can be considered as a chain of $n-1$ first-order low-pass filters as shown in **Fig. 2**, where the scalar s plays the role of the input, λ is the break frequency bandwidth and $\boldsymbol{e}$ is the output. The parameter λ should be selected such that the unmodeled frequencies of the system are to be rejected. From the elementary first-order filter $H(p) = 1/(\lambda + p)$, where $p = d/dt$, we observe that a sufficient condition for frequency rejection is $\lambda \ll p$. Thus in order to reject all unmodeled frequencies we select $\lambda \ll \nu_{u\,\min}$ where $\nu_{u\,\min}$ is the lower bound of the system's unmodeled frequencies ν_u.

The next step is to find the control law that will keep the system in sliding mode. Equation (3) gave us a sufficient condition for the asymptotic stability of the closed-loop system. Let us calculate the first derivative $\dot{s}$: since

$$\begin{aligned} s(x,t) &= \left(\frac{d}{dt} + \lambda\right)^{n-1} e = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-1-k)} \\ &= e^{(n-1)} + \binom{n-1}{1}\lambda e^{(n-2)} \\ &\quad + \binom{n-1}{2}\lambda^2 e^{(n-3)} + \cdots + \lambda^{(n-1)} e \end{aligned}$$

we have

$$\begin{aligned} \dot{s}(x,t) &= e^{(n)} + \binom{n-1}{1}\lambda e^{(n-1)} \\ &\quad + \binom{n-1}{2}\lambda^2 e^{(n-2)} + \cdots + \lambda^{(n-1)}\dot{e} \\ &= x^{(n)} - x_d^{(n)} + \sum_{k=1}^{n-1}\binom{n-1}{k}\lambda^k e^{(n-k)}. \quad (4) \end{aligned}$$

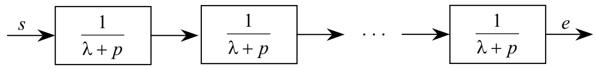

Fig. 2 SMC as a chain of low-pass filters

Substituting Eq. (4) in Eq. (3), and using Eq. (1), one gets

$$\left\{ [f(x,t) + b(x,t)u + \tilde{d}] - x_d^{(n)} + \sum_{k=1}^{n-1}\binom{n-1}{k}\lambda^k e^{(n-k)} \right\} \mathrm{sgn}(s) \leq -\eta. \quad (5)$$

The sliding control law is now defined via the following equations:

$$\begin{aligned} u &= \hat{b}^{-1}(\tilde{u} - \hat{f}) \\ \tilde{u} &= G\{\hat{u} - K(x,t)\mathrm{sgn}(s)\} \\ \hat{u} &= x_d^{(n)} - \sum_{k=1}^{n-1}\binom{n-1}{k}\lambda^k e^{(n-k)} \end{aligned} \quad (6)$$

where $K(x,t) > 0$, and $\hat{f}$ and $\hat{b}$ are estimates of the functions f and b respectively. To choose the multiplicative coefficient (gain) G we define the following bounds:

$$0 \leq \beta^{\min} \leq b\hat{b}^{-1} \leq \beta^{\max}.$$

Then G is defined as $G = (\beta^{\min}\beta^{\max})^{-1/2}$, and the gain margin β as $\beta = (\beta^{\max}/\beta^{\min})^{1/2}$.

It now remains to find $K(x,t)$ so as to satisfy Eq. (3): $\dot{s} \cdot \mathrm{sgn}(s) \leq -\eta$. Introducing Eq. (6) into Eq. (5) yields:

$$\mathrm{sgn}(s)\left\{ f + b\hat{b}^{-1}(\tilde{u} - \hat{f}) + \tilde{d} - x_d^{(n)} + \sum_{k=1}^{n-1}\binom{n-1}{k}\lambda^k e^{(n-k)} \right\} \leq -\eta$$

i.e.,

$$\mathrm{sgn}(s)\left\{ f - b\hat{b}^{-1}\hat{f} + b\hat{b}^{-1}G\hat{u} - b\hat{b}GK(x,t)\mathrm{sgn}(s) + \tilde{d} - x_d^{(n)} + \sum_{k=1}^{n-1}\binom{n-1}{k}\lambda^k e^{(n-k)} \right\} \leq -\eta$$

whence

$$\{\Delta f + (b\hat{b}^{-1}G - 1)\hat{u} + \tilde{d}\}\mathrm{sgn}(s) - b\hat{b}^{-1}GK(x,t) \leq -\eta \quad (7)$$

where $\Delta f = f - b\hat{b}^{-1}\hat{f}$. The above inequality is satisfied if

$$b\hat{b}^{-1}GK(x,t) \geq |\Delta f + (b\hat{b}^{-1}G - 1)\hat{u} + \tilde{d}| + \eta$$

or

$$b\hat{b}^{-1}GK(x,t) \geq |\Delta f| + |(b\hat{b}^{-1}G - 1)||\hat{u}| + |\tilde{d}| + \eta.$$

Now if $b\hat{b}^{-1}$ is replaced by its lower bound $\beta^{\min}$, and the relation $\beta^{\min}G = (\beta^{\min}/\beta^{\max})^{1/2} = \beta^{-1}$ is used, one gets

$$\beta^{-1}K(x,t) \geq |\Delta f| + |1-\beta^{-1}||\hat{u}| + |\tilde{d}| + \eta$$

whence

$$K(x,t) \geq \beta\{|\Delta f| + (1-\beta^{-1})|\hat{u}| + |\tilde{d}| + \eta\}. \quad (8)$$

The upper bounds $\tilde{F}$, D and U:

$$|\Delta f| < \tilde{F}, \quad |\tilde{d}| < D, \quad |\hat{u}| < U$$

are supposed to be known from the system's analysis. Thus, a sufficient condition for the control law to make the sliding surface $s = 0$ a domain of attraction, is:

$$K(x,t) \geq \beta\{\tilde{F} + (1-\beta^{-1})U + D + \eta\}. \quad (9)$$

A characteristic feature of SMC compared to other nonlinear control methods is that when the system enters the sliding mode, then it has dynamics of the linear homogenous type:

$$\left(\frac{d}{dt} + \lambda\right)^{n-1} e = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} = 0$$

no matter what the system uncertainties and disturbances are.

2.2 SMC with boundary layer

An essential drawback of SMC is that, owing to the "signum" term $K(x,t)\text{sgn}(s)$, it causes abrupt changes (chattering) to the control signal u. However this can be avoided by introducing a Boundary Layer (BL) from both sides of the sliding surface $s = 0$. If the term $K(x,t)\text{sgn}(s)$ exceeds the width of the BL then it becomes saturated, and is assigned the maximum (minimum) permissible value. The width of BL is selected to be 2Φ.

Assume that $|s|$ is the distance between the state vector $\boldsymbol{e}$ and the sliding surface $s = 0$. Then, the state $\boldsymbol{e}$ is inside the boundary layer if $|s| < \Phi$, and is outside the BL if $|s| > \Phi$. If the BL is imported in the control law (6) we get:

$$\begin{aligned} u &= \hat{b}^{-1}(\tilde{u} - \hat{f}) \\ \tilde{u} &= G\{\hat{u} - K(x,t)\text{sat}(s/\Phi)\} \\ \hat{u} &= x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} \end{aligned} \quad (10)$$

where the saturation function $\text{sat}(\cdot)$ is defined as

$$\text{sat}(z) = \begin{cases} z & \text{if } |z| < 1 \\ \text{sgn}(z) & \text{if } |z| \geq 1. \end{cases}$$

Figure 3 depicts the BL for a second-order system.

If $K(x,t)$ is chosen according to Eq. (9), then the BL becomes a domain of attraction and the asymptotic stability of the closed-loop system is guaranteed. Obviously, this is a weaker requirement than making the sliding surface $s(x,t) = 0$ a domain of attraction. The result is that the BL reduces the chattering phenomenon, at the price to pay for that which is the increased tracking error.

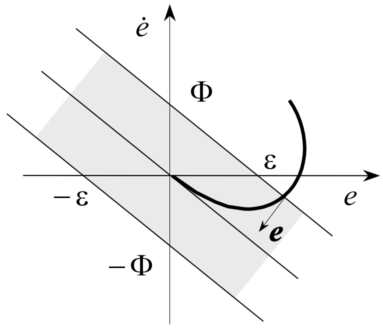

Fig. 3 SMC with BL

The next step in the design of SMC with BL is the selection of Φ. Equation (10) inside BL takes the form:

$$\begin{aligned} u &= \hat{b}^{-1}(\tilde{u} - \hat{f}) \\ \tilde{u} &= G\left\{\hat{u} - K(x,t)\frac{s}{\Phi}\right\} \\ \hat{u} &= x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}. \end{aligned} \quad (11)$$

Introducing Eq. (1) and $\hat{u}$ from Eq. (6) into Eq. (4) yields:

$$\begin{aligned} \dot{s}(x,t) &= f(x,t) + b(x,t)u + \tilde{d} - \hat{u} \\ &= f(x,t) + b(x,t)\{\hat{b}^{-1}(\tilde{u} - \hat{f})\} + \tilde{d} - \hat{u} \end{aligned}$$

or

$$\dot{s}(x,t) = b\hat{b}^{-1}\tilde{u} + (f - b\hat{b}^{-1}\hat{f}) + \tilde{d} - \hat{u}$$

or

$$\dot{s}(x,t) = b\hat{b}^{-1}\tilde{u} + \Delta f + \tilde{d} - \hat{u}.$$

Now using $\tilde{u}$ from Eq. (11) yields

$$\dot{s}(x,t) = b\hat{b}^{-1}G\left(\hat{u} - K\frac{s}{\Phi}\right) + \Delta f + \tilde{d} - \hat{u}$$

whence

$$\dot{s}(x,t) + \frac{b\hat{b}^{-1}GK}{\Phi}s = \hat{u}(b\hat{b}^{-1}G - 1) + \Delta f + \tilde{d}. \quad (12)$$

Equation (12) represents a low-pass filter with input $\hat{u}(b\hat{b}^{-1}G - 1) + \Delta f + \tilde{d}$, output s, and break frequency $b\hat{b}^{-1}GK/\Phi$. So far we have shown how to compute the numerator of the break frequency expression. It only remains to determine the width Φ of BL. There are two choices.

The first is to select Φ in proportion to the desirable tracking accuracy ϵ. According to Slotine and Li [5]

$$\epsilon = \frac{\Phi}{\lambda^{n-1}}. \quad (13)$$

The second choice is to select the bandwidth $b\hat{b}^{-1}GK/\Phi$ equal to λ. This choice is known as balanced condition,

$$\frac{b\hat{b}^{-1}GK}{\Phi} = \lambda. \quad (14)$$

From the above discussion we can see that SMC with BL is identical to the design of a simple SMC. The only additional step required is the selection of the width Φ which can be done either by Eq. (13) or (14).

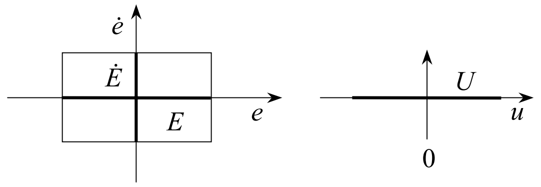

Fig. 4 Ranges of e, $\dot{e}$, and u

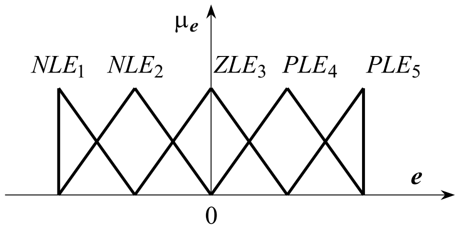

Fig. 5 Partition of the set E in fuzzy subsets

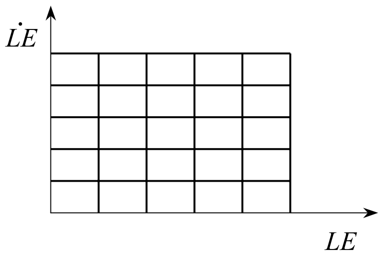

Fig. 6 Partition of the fuzzy phase plane in regions $\{LE^i, \dot{LE}^i\}$

3. Similarity Between FLC and SMC

3.1 The diagonal-type form FLC

Consider a second-order SISO nonlinear and non-autonomous system. In the case of a diagonal type FLC the controller inputs are the error e and the rate of change of error $\dot{e}$ while the controller output is u (see also [1]). The ranges of fluctuation of e, $\dot{e}$ and u are E, $\dot{E}$ and U, respectively, which are domains around zero (see **Fig. 4**).

As seen from **Fig. 5**, the fuzzy values of e, $\dot{e}$ and u belong to the fuzzy sets TE, $T\dot{E}$ and TU respectively, where

$$\begin{aligned} TE &= \{NLE_1, NLE_2, \cdots, NLE_m, ZLE_{m+1}, \\ &\quad PLE_{m+2}, \cdots, PLE_n\} \\ T\dot{E} &= \{N\dot{L}E_1, N\dot{L}E_2, \cdots, N\dot{L}E_m, Z\dot{L}E_{m+1}, \\ &\quad P\dot{L}E_{m+2}, \cdots, P\dot{L}E_n\} \\ TU &= \{NLU_1, NLU_2, \cdots, NLU_m, ZLU_{m+1}, \\ &\quad PLU_{m+2}, \cdots, PLU_n\}. \end{aligned}$$

The fuzzy phase plane is the set of all fuzzy state vectors $\{LE^i, \dot{LE}^i\}$ as shown in **Fig. 6**.

For the fuzzy region $(ZLE_{m+1}, Z\dot{L}E_{m+1})$, the controller output is 0 which implies that the system is in the steady state which is at the origin of the fuzzy phase plane. The set of all fuzzy vectors for which the fuzzy output of the controller becomes zero is:

$$\begin{aligned} &\{(PLE_n, N\dot{L}E_{n-1}), (PLE_{n-1}, N\dot{L}E_{n-2}), \cdots, \\ &(ZLE_{m+1}, Z\dot{L}E_{m+1}), \cdots, \\ &(NLE_2, P\dot{L}E_n), (NLE_1, P\dot{L}E_n)\}. \end{aligned}$$

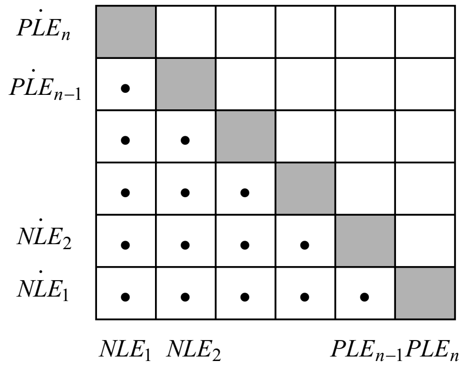

Fig. 7 Fuzzy regions below the diagonal

The regions where the controller output becomes zero lie on the diagonal that separates the fuzzy phase plane into two semi-planes. For all the fuzzy regions below the diagonal the controller's output takes a positive fuzzy value with a magnitude that depends on the distance between this fuzzy region and a particular zero-region on the diagonal, below which the given fuzzy region is located. The set of all fuzzy regions below the diagonal shown in **Fig. 7** is:

- $\{(NLE_1, P\dot{L}E_{n-1}), \cdots, (NLE_1, N\dot{L}E_1)$, i.e., the fuzzy regions below the zero region $(NLE_1, P\dot{L}E_n)$;
- $\{(NLE_2, P\dot{L}E_{n-2}), \cdots, (NLE_2, N\dot{L}E_1)$, i.e., the fuzzy regions below the zero region $(NLE_2, P\dot{L}E_{n-1})$;
- $(PLE_{n-1}, N\dot{L}E_1)$, i.e., the fuzzy region below the zero region $(PLE_{n-1}, N\dot{L}E_2)$.

As distance between a "fuzzy region below the diagonal and diagonal" is defined the distance between the "center of this region and the center of the zero region below which the given fuzzy region is located."

For all the fuzzy regions above the diagonal shown in **Fig. 8**, the controller output is assigned negative values with magnitude depending on the distance between the fuzzy region and the diagonal. The fuzzy regions that lie over the diagonal are:

- $\{(PLE_n, N\dot{L}E_2), \cdots, (PLE_n, P\dot{L}E_n)$, i.e., the fuzzy regions above the zero region $(PLE_n, N\dot{L}E_1)$;
- $\{(PLE_{n-1}, P\dot{L}E_2), \cdots, (PLE_n, P\dot{L}E_n)$, i.e., the fuzzy regions above the zero region $(PLE_{n-1}, N\dot{L}E_2)$;
- $(NLE_2, P\dot{L}E_n)$, i.e., the fuzzy region below the zero region $(NLE_2, P\dot{L}E_{n-1})$.

As distance between a "fuzzy region above the diagonal and the diagonal" is defined "the distance between the center of this region and the center of the zero region above which the given fuzzy region is located."

For example, if $\boldsymbol{e} = (e, \dot{e})^T = (LE^i, \dot{LE}^i)^T$, then the output of the controller should be $u = LU^i$, i.e.,

$$R_c^i : \text{IF } \boldsymbol{e} = (PLE_4, N\dot{L}E_1) \text{ THEN } u = PLU_4$$

where the magnitude of the control signal u is defined by the distance between the fuzzy region $(PLE_4, N\dot{L}E_1)$ and the diagonal.

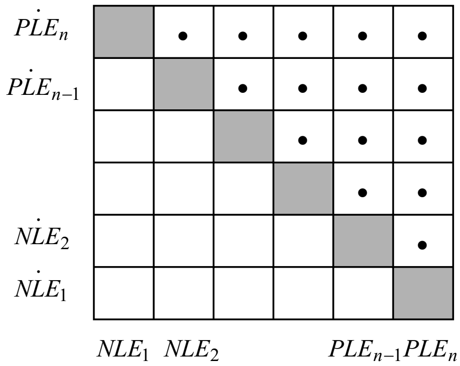

Fig. 8 Fuzzy regions above the diagonal

Assume that e^* and $\dot{e}^*$ are the crisp values of the FLC input. The computational structure of an FLC involves the following steps (see **Fig. 9**):

Step 1: Normalization, i.e., multiplication by normalization factors N_e and $N_{\dot{e}}$:

$$e_N^* = e^* N_e \text{ and } \dot{e}_N^* = \dot{e}^* N_{\dot{e}}. \tag{15}$$

Step 2: Fuzzification of the normalized inputs e_N^* and $\dot{e}_N^*$, i.e., calculation of the membership value $\mu^i(\boldsymbol{e}^*)$ of the input vector $\boldsymbol{e}^* = (e_N^*, \dot{e}_N^*)^T$ in the fuzzy region $(LE^i, \dot{LE}^i)$, $i = 1, 2, \cdots, n$.

Step 3: Inference through the use of $\mu^i(\boldsymbol{e}^*)$ and the rule base, i.e., computation of the membership value $\mu_{CLU^i}(u_N)$ of the controller output in the fuzzy set LU^i. For a multi-input/single-output FLC, the i-th fuzzy rule of the fuzzy rule base has the form:

$$R_c^i : \text{IF } \boldsymbol{e}^* = LE^i \text{ THEN } u = LU^i. \tag{16}$$

Step 4: Defuzzification, i.e., a mapping of the membership value $\mu_{CLU^i}(u_N)$ to a crisp point of the set LU^i.

Step 5: Denormalization of the controller crisp output u_N, i.e., multiplication by denormalization factors N_u^{-1}:

$$u = N_u^{-1} u_N. \tag{17}$$

3.2 Properties of the transfer characteristic of a diagonal type FLC

The transfer characteristic (control surface) of a diagonal FLC is a nonlinear mapping $u = h(e, \dot{e})$ which is defined by the operating points and the interpolation between them. An operating point $P^i(e^i, \dot{e}^i)$ is defined as follows: Assume that $\boldsymbol{e}^i = (e^i, \dot{e}^i)^T$ is the FLC input and u^i is the corresponding output. Assume also that the center of the fuzzy region LE^i is defined as $\boldsymbol{e}^i \in E^2$ where $(e^i, \dot{e}^i)^T$ are crisp values such that $\mu_{LE^i}(e) = 1$ and $\mu_{\dot{LE}^i}(\dot{e}) = 1$. Then, an operating point $P^i(e^i, \dot{e}^i)$ is a point for which $\boldsymbol{e}$ is located at the center of the fuzzy region LE^i, and u^i is the corresponding crisp output of the FLC. Thus when the input of the FLC coincides with an operating point, only one rule should be activated [1].

The quality of interpolation between the operating points (activation and aggregation of more than one fuzzy if-then rules) depends on the methods of inferring and defuzzification. A special feature of the control surface $u = h(e, \dot{e})$ is the diagonal $u = h(e, \dot{e}) = 0$, where u changes its sign. The diagonal-type FLC is designed such that for an increasing Euclidean distance $|s|$ between the state vector $\boldsymbol{e}$ and the diagonal $u = 0$, the absolute value $|u|$ of the controller output increases monotonically, i.e.,

$$|s_2| > |s_1| \text{ implies } |u(s_2)| > |u(s_1)|.$$

3.3 SMC with BL for a second-order system

The diagonal-type FLC was derived and discussed in Sections 3.1 and 3.2 for a second-order system. In order to show the similarity between the diagonal-type FLC and SMC with BL the latter control method is applied here to a second-order system [5]:

$$\ddot{x} = f(x, t) + b(x, t)u + \tilde{d} \tag{18}$$

where the sliding line is

$$s = \lambda e + \dot{e}. \tag{19}$$

The control law will be

$$\begin{aligned} u &= \hat{b}^{-1}(\tilde{u} - \hat{f}) \\ \tilde{u} &= G\{\hat{u} - K(x, t)\text{sat}(s/\Phi)\} \\ \hat{u} &= \ddot{x}_d - \lambda \dot{e} \end{aligned}$$

i.e.,

$$u = \hat{b}^{-1}\{-\hat{f} + G(\ddot{x}_d - \lambda \dot{e}) - GK(x, t)\text{sat}(s/\Phi)\}. \tag{20}$$

The above control law can be analyzed in the following terms:

1. *Compensation term*:

$$u_{comp} = -\hat{b}^{-1}\hat{f}. \tag{21}$$

2. *Filtering term*:

$$u_{filt} = -\hat{b}^{-1}G\lambda\dot{e}. \tag{22}$$

This term rejects the unmodeled frequencies of the system.

3. *Feedforward term*:

$$u_{ff} = \hat{b}^{-1}G\ddot{x}_d. \tag{23}$$

4. *Feedback control term*:

$$u_c = -\hat{b}^{-1}GK(x, t)\text{sat}(s/\Phi). \tag{24}$$

This term prevents the state vector $\boldsymbol{e}$ from moving away from the sliding surface $s = 0$. The negative sign indicates that the control action takes always place in the decrease direction of error.

The part $-K(x, t)\text{sat}(s/\Phi)$ is of diagonal form, with $s = 0$ being the diagonal line. Examining the diagonal part of u_c one gets:

$$u_{diag} = -K(x, t)\text{sat}(s/\Phi)$$

where

$$\begin{aligned} u_{diag} &> 0 \text{ for } s < 0 \\ u_{diag} &= 0 \text{ for } s = 0 \\ u_{diag} &< 0 \text{ for } s > 0. \end{aligned}$$

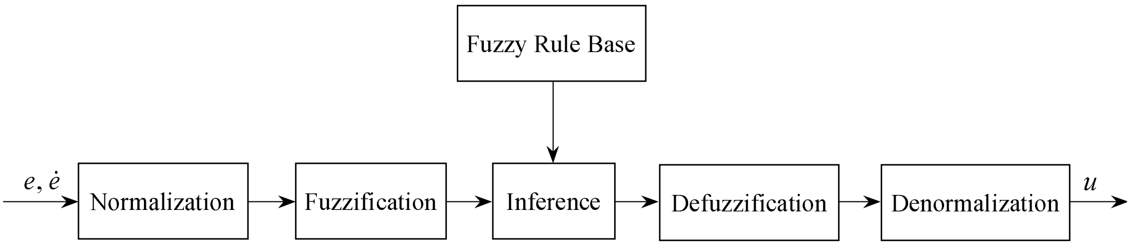

Fig. 9 Computational Structure of an FLC

Assuming that K is constant and that the state error vector $\boldsymbol{e}$ lies inside the BL then

$$u_{diag} = -K\frac{s}{\Phi} = -K\frac{|s|}{\Phi}\text{sat}(s). \tag{25}$$

The relation (25) shows that u_{diag} is proportional to $|s|$. Consequently the magnitude of u_{diag} increases when the distance of the fuzzy region $(e, \dot{e})$ from the sliding surface $s = 0$ increases and vice versa.

3.4 Properties of a diagonal-type FLC

The diagonal-type FLC provides a mapping from the crisp state vector $(e, \dot{e})$ to a crisp control output u and the magnitude of the fuzzy control signal is proportional to the distance of the fuzzy region $(e, \dot{e})$ from the diagonal. The states that are located on the diagonal have a significant role because there the control output changes sign. The diagonal for a second-order system is described by the equation

$$s = \lambda e + \dot{e} = 0.$$

Here, the rules of a diagonal-type FLC are selected such that [1]:

1. the states e and $\dot{e}$ are bounded
$$-e_{\max} \le e \le e_{\max} \text{ and } -\dot{e}_{\max} \le \dot{e} \le \dot{e}_{\max}; \tag{26}$$
2. the control signal u is bounded as
$$-u_{\max} \le u \le u_{\max}; \tag{27}$$
3. the states e and $\dot{e}$ that are located on the diagonal produce zero control signals;
4. the states e and $\dot{e}$ that are located below the diagonal produce positive control signals;
5. the states e and $\dot{e}$ that are located above the diagonal produce negative control signals;
6. the magnitude of the control signal $|u|$ increases when the distance from the diagonal increases, and vice-versa.

The properties 1. and 2. are inherent features of FLC. The properties 4. and 5. ensure that the control action tries to keep the state vector on the sliding surface (diagonal) and take place in the decrease direction of error.

The analytical form of a diagonal-type FLC is:

$$u_{fuzz} = -K_{fuzz}(e, \dot{e}, \lambda)\text{sgn}(s) \tag{28}$$

with the following conditions:

- $-e_{\max} \le e \le e_{\max}$,
- $-\dot{e}_{\max} \le \dot{e} \le \dot{e}_{\max}$,

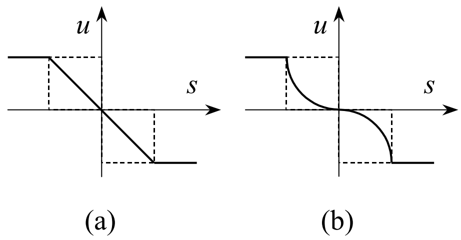

Fig. 10 Transfer characteristic of: (a) SMC, and (b) diagonal form FLC

- $\lambda > 0$,
- $0 \le K_{fuzz} < u_{\max} = K_{fuzz|\max}$,
- $K_{fuzz}(e_1, \dot{e}_1, \lambda) \le K_{fuzz}(e_2, \dot{e}_2, \lambda)$ for $|\lambda e_1 + \dot{e}_1| \le |\lambda e_2 + \dot{e}_2|$, which means that the greater the distance of $(e, \dot{e})$ from the sliding surface is, the greater the control signal becomes.

3.5 Comparison between SMC with BL and diagonal-type FLC

This comparison is also presented in [1] and will help to understand the properties of the RC-SMFLC controller, which is going to be presented in Section 4.

The diagonal control term in SMC with BL is (see Eq. (25))

$$u_{diag} = -K\frac{s}{\Phi} = -K\frac{|s|}{\Phi}\text{sat}(s)$$

while the control term in the diagonal-type FLC is (see Eq. (28))

$$u_{fuzz} = -K_{fuzz}(e, \dot{e}, \lambda)\text{sgn}(s).$$

From the above two equations the similarity between SMC with BL, and FLC is obvious. Additionally, the vicinity of the diagonal in a diagonal-type FLC can be viewed as a BL.

The main differences between the two controllers are as follows:

- The transfer characteristic $u_{fuzz} = f(s)$ of a diagonal-type FLC is nonlinear (due to the nonlinear nature of FLC), while the one of SMC with BL is linear (see **Fig. 10**).
- In the diagonal-type FLC, the state vector $\boldsymbol{e}$ is bounded (it is an inherent structural property of FLC) while this does not happen in SMC with BL, as shown in **Fig. 11**.

3.6 The basic principles of SMFLC

As it has already been mentioned, in the diagonal-type FLC the magnitude of the control signal u changes in proportion to the distance of $(e, \dot{e})$ from the diagonal. Thus the

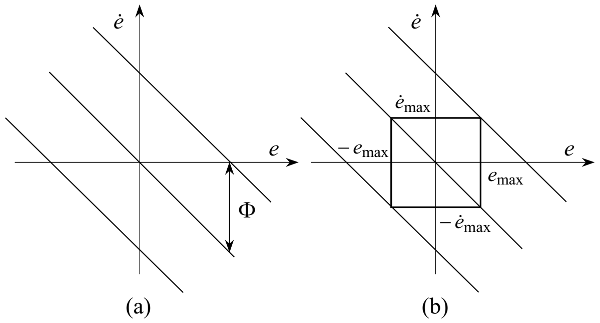

Fig. 11 Error state vector boundaries in: (a) SMC with BL, (b) diagonal-type FLC

control rules can be modified as:

$$R_c^i : \text{IF } s = LS^i \text{ THEN } u = LU^i$$

in which case the corresponding control law becomes

$$u_{fuzz} = -K_{fuzz}(|s|)\text{sgn}(s). \tag{29}$$

Equation (29) resembles even more to the control law of SMC with BL, and can be named "Sliding Mode FLC" (SMFLC). The main advantage of SMFLC compared to a typical diagonal-type FLC is that it reduces drastically the number of rules in the rule base. This happens because the diagonal form FLC uses as inputs the state variables e and $\dot{e}$, and so the number of the controller inputs is equal to the elements of the state vector $\boldsymbol{e}$. On the contrary, the only input of the SMFLC is the "signed" distance s from the diagonal. For an n-th-order system the distance s can be calculated by

$$s(x,t) = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-1-k)}.$$

In Section 4.1 the Reduced-Complexity SMFLC (RC-SMFLC) will be designed, which requires even less rules and introduces a simpler perspective in the design of diagonal-type FLCs.

3.7 Design of an SMFLC

As it has already been mentioned the control law of SMC is

$$u = \hat{b}^{-1}G\hat{u} - \hat{b}^{-1}\hat{f} - \hat{b}^{-1}GK(x,t)\text{sat}(s/\Phi)$$
$$\hat{u} = x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}.$$

After some modifications in the above control law, SMFLC is derived:

$$u = \hat{b}^{-1}G\hat{u} - \hat{b}^{-1}\hat{f} + \hat{b}^{-1}Gu_{fuzz}$$
$$u_{fuzz} = -K\{|s|\text{sgn}(s)\}. \tag{30}$$

The design of SMFLC is concentrated on the fuzzy part:

$$u_{fuzz} = -K\{|s|\text{sgn}(s)\}$$

while the choice of the previous terms can be done as in SMC. The choice of the transfer characteristic will be examined first.

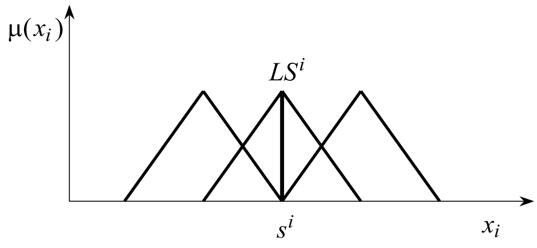

Fig. 12 Input s^i of an operating point $P(s^i, u^i)$

3.8 The transfer characteristic of an SMFLC

The crucial point in the design of SMFLC is the choice of the number of the fuzzy subsets for the inputs and the outputs of the controller and consequently the shape of the corresponding membership functions [1].

3.8.1 The number of the fuzzy subsets The operating point $P(s^i, u^i)$ of the transfer characteristic of an SMFLC is defined as follows. Consider a specific input s^i and the corresponding output u^i, and assume that the center of the fuzzy region LS^i is $s^i \in S$, where s^i is a crisp value such that $\mu_{LS^i}(s^i) = 1$ (see **Fig. 12**). Then, an operating point $P(s^i, u^i)$ is a point for which s^i is located in the middle of the fuzzy region LS^i and u^i is the corresponding output of SMFLC. This implies that for each operating point $P(s^i, u^i)$ only one rule is activated.

Depending on the type of fuzzy inference and the defuzzification method used, SMFLC provides either a linear or a nonlinear interpolation between two operating points. The number of operating points is equal to the number of discontinuities in the transfer characteristic $u = f(s)$.

3.8.2 The shape and the position of the membership functions A usual form for the membership functions is the triangular one. The problem of specifying the position of a membership function is related to the problem of placement of the operating points (centers of the membership functions). This is also related to the choice of the gain du/ds of the SMFLC, since the position of the operating points influences the slope of the transfer characteristic $u = f(s)$.

The operating points (i.e., the centers and widths of the membership functions) can be selected in two alternative ways:

1. Choose a small slope in the middle of region S and increase the slope for increasing values of $|s|$. Choose longer distances between the operating points in the middle of region S, as shown in **Fig. 13**. That is:

$$|s_2| > |s_1| \text{ implies } |du/ds|_{s_2} > |du/ds|_{s_1}.$$

2. Choose a high slope in the middle of region S and decrease the slope for increasing values of $|s|$. Choose shorter distances between the operating points in the middle of region S, as shown in **Fig. 14**. That is,

$$|s_2| > |s_1| \text{ implies } |du/ds|_{s_2} < |du/ds|_{s_1}.$$

3.8.3 Normalization of the input and denormalization of the output Normalization is the mapping from the input (physical) domain to a normalized domain, while denormalization is the mapping from a normalized domain to a

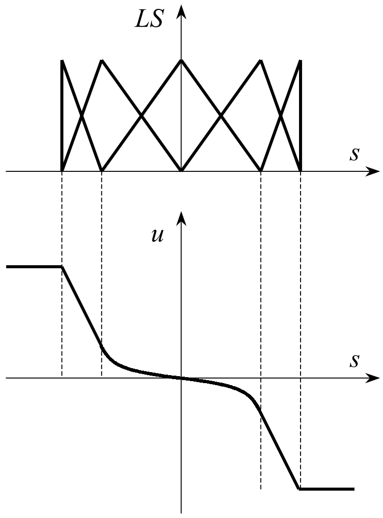

Fig. 13 Choice of the membership functions such that $|s_2| > |s_1| \Rightarrow |du/ds|_{s_2} > |du/ds|_{s_1}$

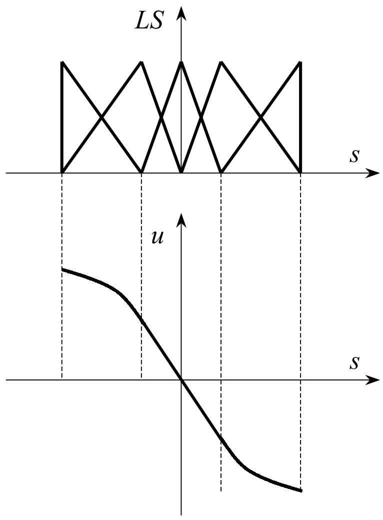

Fig. 14 Choice of the membership functions such that $|s_2| > |s_1| \Rightarrow |du/ds|_{s_2} < |du/ds|_{s_1}$

physical domain. The normalization affects the sensitivity of the controller and the gain in the vicinity of the operating point. The denormalization affects the total gain, and through it, the stability of the closed loop system.

Assume a second-order system with sliding line

$$s = \lambda e + \dot{e} = 0.$$

After normalization the sliding line will be transformed to

$$s = \lambda_N e_N + \dot{e}_N = 0 \tag{31}$$

where

$$e_N = eN_e, \quad \dot{e}_N = \dot{e}N_{\dot{e}}, \quad \lambda = \lambda_N \left(\frac{N_e}{N_{\dot{e}}}\right). \tag{32}$$

The parameter λ plays the role of the rejection frequency for all unmodeled frequencies of the system and has to be $\lambda \leq \nu_{s_u}$. Thus the normalized parameter λ_N should satisfy the condition

$$\frac{N_e}{N_{\dot{e}}} \leq \frac{\nu_{s_u}}{\lambda_N}. \tag{33}$$

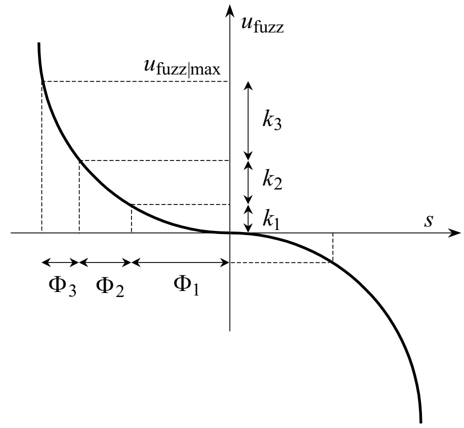

Fig. 15 SMFLC as a state dependent filter

For the denormalization of the output in the same second-order system one has:

$$u_N = N_u u \text{ implies } u = N_u^{-1} u_N.$$

The choice of N_u is important for the stability of the closed-loop system, and depends on the maximum value of K_{fuzz}. To achieve asymptotic stability of the closed-loop system we choose:

$$K_{fuzz|\max} \geq \beta\{\tilde{F} + (1 - \beta^{-1})\hat{U} + D + \eta\}$$

where

$$K_{fuzz|\max} = \max\{K_{fuzz}(|s|)\}. \tag{34}$$

From the relation $K_{fuzz_N|\max} = N_u K_{fuzz|\max}$, we can calculate the normalization coefficient N_u:

$$N_u = \frac{K_{fuzz_N|\max}}{K_{fuzz|\max}}. \tag{35}$$

Another approach to normalization and denormalization of SMFLCs would be to leave the coefficients of the equations unchanged and inflate or deflate directly the input or the output of the controller.

3.8.4 SMFLC as a state dependent filter SMFLC, like SMC, can be regarded as a filter function. If the transfer characteristic between two operating points i and $i+1$ is approximately considered to be a linear segment, then a gain k_i/ϕ_i can be attached to this i-th segment. Since the gain changes take place from segment to segment one obtains a state dependent filter function [1] (see **Fig. 15**).

For the i-th segment of the transfer characteristic the resulting filter function is similar to the one for SMC:

$$\dot{s} + b\hat{b}^{-1}G\frac{k_i}{\phi_i}s = b\hat{b}^{-1}Gu_i\text{sgn}(s) + \hat{u}(b\hat{b}^{-1}G - 1) + \Delta f + \tilde{d} \tag{36}$$

with

$$u_i = \begin{cases} -\sum_{\nu=1}^{i-1}\left(k_\nu + \frac{k_i \sum_{\nu=1}^{i-1}\phi_\nu}{\phi_i}\right) & \text{if } i \geq 2 \\ 0 & \text{if } i = 1. \end{cases} \tag{37}$$

It is a filter with break frequency $b\hat{b}^{-1}G(k_i/\phi_i)$, which also depends on the state vector $s = (e, \dot{e})$. For a large distance $|s|$ between the sliding line $s = 0$ and the state vector $\boldsymbol{e}$, there is no unmodeled frequency that can cause a change of sign to the control signal (since $|s|$ is big, the impact that unmodeled frequencies can have on sgn(s) is negligible). Thus for a big $|s|$ we select a bigger control gain than for small $|s|$. On the contrary, in the neighborhood of the sliding line we choose a small control gain.

It will be shown that in RC-SMFLC the choice of the appropriate control gain, according to the distance from the sliding line, is done implicitly.

To achieve a big control gain far from the sliding line and a small control gain near the sliding line, the following inequalities must hold:

$$\frac{k_1}{\phi_1} \leq \frac{k_2}{\phi_2} \leq \cdots \leq \frac{k_n}{\phi_n}. \tag{38}$$

It should be pointed out that the shape and the position of the membership functions of the controller inputs and outputs determine the slopes of the segments of the nonlinear transfer characteristic.

Compared to the conventional SMC, the balance condition

$$b\hat{b}^{-1}G\frac{k_i}{\phi_i} \leq \lambda, \quad (\lambda \leq \nu_{u_{\min}})$$

must be satisfied only in the vicinity of the origin of the phase plane (since away from the origin the unmodeled frequencies cannot change much sgn(s)). The quality of tracking is guaranteed by the maximum values

$$K_{fuzz|\max} = \sum_{\nu=1}^{i-1} k_\nu \text{ and } \Phi_{\max} = \sum_{\nu=1}^{i-1} \phi_\nu$$

as long as

$$b\hat{b}^{-1}G\frac{K_{fuzz|\max}}{\Phi_{\max}} \leq \lambda.$$

In RC-SMFLC tracking quality will not be of primary importance.

3.8.5 SMFLC for an n-th order system The design of an SMFLC for a second-order system can be extended to an n-th order system. The crucial point is to produce the normalization factors $N_e, N_{\dot{e}}, \cdots, N_{e^{(n-1)}}$ for each one of the states $e, \dot{e}, \cdots, e^{(n-1)}$. The unnormalized sliding surface is determined by [1]:

$$\left(\frac{d}{dt}+\lambda\right)^{(n-1)} e = e^{(n-1)} + \binom{n-1}{1}\lambda e^{(n-2)} + \cdots + \lambda^{(n-1)}e = 0,$$

and the normalized one by

$$\left(\frac{d}{dt}+\lambda\right)^{(n-1)} e_N = e_N^{(n-1)} + \binom{n-1}{1}\lambda_N e_N^{(n-2)} + \cdots + \lambda_N^{(n-1)}e_N = 0.$$

With the use of normalization factors, one gets

$$\left(\frac{d}{dt}+\lambda\right)^{(n-1)} e_N = e_N^{(n-1)} + \binom{n-1}{1}\lambda_N \frac{N_{e^{(n-2)}}}{N_{e^{(n-1)}}} e^{(n-2)} + \cdots + \lambda_N^{(n-1)}\frac{N_e}{N_{e^{(n-1)}}}e = 0.$$

Comparing the coefficients of the above two equations yields :

$$\lambda = \lambda_N \frac{N_{e^{(n-2)}}}{N_{e^{(n-1)}}}, \quad \lambda^2 = \lambda_N^2 \frac{N_{e^{(n-3)}}}{N_{e^{(n-1)}}}, \cdots \quad \lambda^{n-1} = \lambda_N^{n-1}\frac{N_e}{N_{e^{(n-1)}}} \tag{39}$$

$$\frac{N_{e^{(n-2)}}}{N_{e^{(n-1)}}} = \cdots = \frac{N_e}{N_{\dot{e}}} = \frac{\lambda}{\lambda_N}. \tag{40}$$

This means that, if λ, λ_N and N_e are known one can calculate all the normalization factors.

In RC-SMFLC one has just to normalize the scalar control input (i.e., to use only one normalization factor) and proceed to the design of the SMFLC in the normalized region of input values.

4. Design of an RC-SMFLC

4.1 The concept

For the design of a *Reduced-Complexity* SMFLC the basic conditions for the convergence of a system to the desirable set-point are taken into account. Consider an n-th-order nonlinear non-autonomous system

$$x^{(n)}(t) = f(x,t) + b(x,t)u + \tilde{d}$$

with output $x(t)$ and desirable set-point $x_d(t)$. The tracking error is $e(t) = x(t) - x_d(t)$ and the rate of error change is $\dot{e}(t) = \dot{x}(t) - \dot{x}_d(t)$. Generally, we can assume the following relations:

- If $e(t) > 0$ and $\dot{e}(t) < 0$ then $x(t) \to x_d(t)$ implies $e(t) \to 0$.
- If $e(t) > 0$ and $\dot{e}(t) > 0$ then $x(t)$ deviates from $x_d(t)$.
- If $e(t) < 0$ and $\dot{e}(t) < 0$ then $x(t)$ deviates from $x_d(t)$.
- If $e(t) < 0$ and $\dot{e}(t) > 0$ then $x(t) \to x_d(t)$ implies $e(t) \to 0$.

The above four convergence conditions can be merged as (see also [13], [14]):

$$\text{If } e(t)\dot{e}(t) < 0 \text{ then } x(t) \to x_d(t) \Rightarrow e(t) \to 0; \tag{41}$$

$$\text{If } e(t)\dot{e}(t) > 0 \text{ then } x(t) \text{ deviates from } x_d(t). \tag{42}$$

The same conditions could have been derived from the Lyapunov function

$$V = \frac{1}{2}e^2 \Rightarrow \dot{V} = e\dot{e}.$$

From Eqs. (41) and (42) it is obvious that the greater part of the information, needed to achieve convergence to the desirable set-point, is contained in $e(t)$ and $\dot{e}(t)$.

Define now the sliding surface $s(x,t)$

$$s(x,t) = e(t)\dot{e}(t) < 0. \tag{43}$$

Then the control law can be expressed as follows:

- If sgn$(e(t)\dot{e}(t)) < 0$, then the control action leads to convergence and should be maintained.
- If sgn$(e(t)\dot{e}(t)) > 0$, then the control action leads to divergence and should be altered.

Once the state vector $[e(t), \dot{e}(t)]^T$ is found in the semi-plane $s(x,t) = e(t)\dot{e}(t) < 0$, it gradually approaches the null vector $[0,0]^T$. Thus, the goal is to find a control law u that will be able to keep the state vector in the semi-plane $s(x,t) = e(t)\dot{e}(t) < 0$.

4.2 The rule base of RC-SMFLC

There are two possible control actions: increase or decrease the control signal u. That is, we can state:

- IF sgn$(e(t)\dot{e}(t)) < 0$ AND the previous control action was to increase the control signal THEN keep on increasing.
- IF sgn$(e(t)\dot{e}(t)) < 0$ AND the previous control action was to decrease the control signal THEN keep on decreasing.
- IF sgn$(e(t)\dot{e}(t)) > 0$ AND the previous control action was to increase the control signal THEN now decrease it.
- IF sgn$(e(t)\dot{e}(t)) > 0$ AND the previous control action was to decrease the control signal THEN now increase it.

An equivalent formulation is as follows:

$$\begin{aligned}
&\text{IF } \mathrm{sgn}(e(t)\dot{e}(t)) < 0 \text{ AND } \Delta u_k > 0 \text{ THEN } \Delta_{k+1} > 0\\
&\text{IF } \mathrm{sgn}(e(t)\dot{e}(t)) < 0 \text{ AND } \Delta u_k < 0 \text{ THEN } \Delta_{k+1} < 0\\
&\text{IF } \mathrm{sgn}(e(t)\dot{e}(t)) > 0 \text{ AND } \Delta u_k > 0 \text{ THEN } \Delta_{k+1} < 0\\
&\text{IF } \mathrm{sgn}(e(t)\dot{e}(t)) > 0 \text{ AND } \Delta u_k < 0 \text{ THEN } \Delta_{k+1} > 0
\end{aligned}$$

where Δu_k is the change in the control signal at the k-th iteration of the algorithm.

To ensure that the control signal is increased with the use of the FLC the following rules are employed:

$$\begin{aligned}
&\text{IF } u_k \text{ is } U_1 \text{ THEN } u_{k+1} \text{ is } U_2\\
&\text{IF } u_k \text{ is } U_2 \text{ THEN } u_{k+1} \text{ is } U_3\\
&\quad\vdots\\
&\text{IF } u_k \text{ is } U_{n-1} \text{ THEN } u_{k+1} \text{ is } U_n.
\end{aligned}$$

To ensure that the control signal is decreased with the use of the FLC the rules that must be used are:

$$\begin{aligned}
&\text{IF } u_k \text{ is } U_2 \text{ THEN } u_{k+1} \text{ is } U_1\\
&\text{IF } u_k \text{ is } U_3 \text{ THEN } u_{k+1} \text{ is } U_2\\
&\quad\vdots\\
&\text{IF } u_k \text{ is } U_n \text{ THEN } u_{k+1} \text{ is } U_{n-1}
\end{aligned}$$

where $U_1, U_2, \cdots, U_{n-1}, U_n$ are the fuzzy subsets in which the fuzzy phase plane of the control input is divided.

If the fuzzy phase plane U is partitioned by n triangular membership functions with equal widths and slopes, it can easily be verified that the above rule base can lead the system to oscillations around the desirable set-point. Consequently, in order to achieve convergence, the nonlinear transfer characteristic of the fuzzy controller should be such that the smaller the distance from the set-point is the smaller the change of the control signal becomes.

4.3 Analytical description of the RC-SMFLC

In the diagonal-type FLC it was necessary to produce a fuzzy control signal that would be proportional to the distance from the diagonal, where the diagonal was the sliding surface $s(x,t) = \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}$. In the RC-SMFLC it is necessary to produce a fuzzy control signal that will be proportional to the distance from the set-point $e = 0$. Thus in this case $e = 0$ plays the role of the diagonal.

Therefore there are two requirements for the control law in RC-SMFLC:

- To keep the error state vector $[e(t), \dot{e}(t)]^T$ inside the sliding surface $s(x,t) = e(t)\dot{e}(t) < 0$.
- Its magnitude to be proportional to the distance from the diagonal $e = 0$.

The RC-SMFLC has similar properties to the ones of a diagonal-type FLC as described in Section 3.4, i.e.,

1. the states e and $\dot{e}$ are bounded;
2. the control signal u is bounded;
3. the states e and $\dot{e}$ that are located on the diagonal produce zero control signals;
4. the states e and $\dot{e}$ that are located above the diagonal produce negative control signals (because surpass of the diagonal is a result of an "increase" control action, which means that the next control action will be "decrease");
5. the states e and $\dot{e}$ that are located below the diagonal produce positive control signals.

It remains to show that RC-SMFLC also satisfies the property 6. in Section 3.4, i.e., the magnitude of the control signal $|u|$ decreases when the distance from the diagonal decreases. The desirable control law should be similar to the one of a diagonal-type FLC:

$$u_{fuzz} = -K_{fuzz}(e, \dot{e}, \lambda)\mathrm{sgn}(s)$$

or to the control law of SMFLC:

$$u_{fuzz} = -K\{|s|\mathrm{sgn}(s)\}$$

where $|s|$ is the distance from the diagonal.

The use of membership functions with the same shape (e.g. triangular with the same width and slopes) produces a control law of the form:

$$u_{fuzz} = -K\mathrm{sgn}(s)$$

where K is static. To overcome this problem, the width of the membership functions should be modified at every crossing of the diagonal $e = 0$. The last two control signals u_{k-1} and u_k are taken into account:

- u_{k-1} is the last control signal below (above) the diagonal;
- u_k is the last control signal above (below) the diagonal.

Recalling the bisection method, the control signal u^* that will produce zero error should be searched in the range $[u_{k-1}, u_k]$.

The new fuzzy subsets $U_1, U_2, \cdots, U_{n-1}, U_n$ correspond to the division of the interval between these two control signals $[u_{k-1}, u_k]$ in n equal segments.

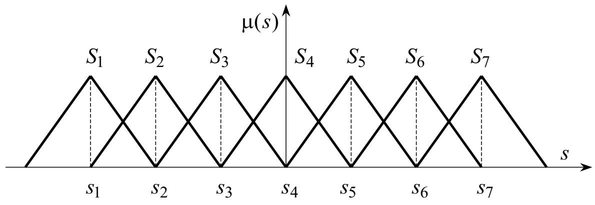

Fig. 16 Partition of the RC-SMFLC into fuzzy subsets

Thus the operating points $P(s^i, u^i)$ are in this case moving, and the nonlinear transfer characteristic $u = f(s)$ of the fuzzy controller changes. As the diagonal is approached, the width of the membership functions is reduced, and consequently the control gain K is reduced too. In this way a control law of the form

$$u_{fuzz} = -K_{fuzz}\text{sgn}(e_k e_{k+1})\text{sgn}(s), \quad s = e\dot{e} \tag{44}$$

where e_k is the error at the k-th step of the algorithm, and e_{k+1} is the error at the $(k+1)$-th step of the algorithm, is derived, and the similarity between RC-SMFLC and the diagonal-type FLC or the conventional SMFLC becomes clear.

4.4 The transfer characteristic of RC-SMFLC

As it has already been discussed the universe of discourse of the input of the RC-SMFLC can be analyzed in the fuzzy subsets of **Fig. 16**.

The center of each fuzzy region S_i is denoted by s_i. According to Wang [3], a fuzzy rule base that consists of M fuzzy IF-THEN rules of the type:

$R^{(l)}$: IF x_1 is F_1^l AND $\cdots$ AND x_n is F_n^l THEN y is G^l

and uses singleton fuzzifier, product inference rule, Gaussian membership function, and center of gravity defuzzifier produces an output of the form:

$$f(x) = \frac{\sum_{l=1}^{M} \bar{y}^l \left[\prod_{i=1}^{n} a_i^l \exp\left\{-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l}\right)^2\right\}\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} a_i^l \exp\left\{-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l}\right)^2\right\}\right]} \tag{45}$$

where $\bar{y}^l$ is the center of the l-th output fuzzy subset, and $\bar{x}_i^l$ is the center of the i-th input fuzzy subset. The above equation indicates that, for example, in the "increase" control mode one should expect $f(s_1) = s_2$. Therefore the operating points of RC-SMFLC are:

(a) In the increase control mode: (s_1, s_2), (s_2, s_3), (s_3, s_4), (s_4, s_5), (s_5, s_6) and (s_6, s_7).
(b) In the decrease control mode: (s_2, s_1), (s_3, s_2), (s_4, s_3), (s_5, s_4), (s_6, s_5) and (s_7, s_6).

Equation (45) also reveals the nonlinear nature of a fuzzy controller.

As it can be observed from Fig. 16, at the beginning of the controller's operation the fuzzy subsets have equal width and identical shape. Therefore a rough approximation of the transfer characteristic $u_{k+1} = f(u_k)$ would be given by **Fig. 17**. Of course, the transfer characteristic between two adjacent operating points is not described by a first order linear function, but what the above diagram tries

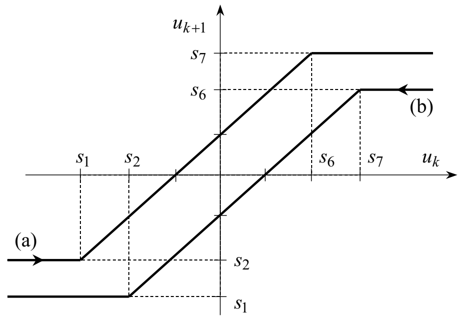

Fig. 17 Approximation of the transfer characteristic of RC-SMFLC (a) increase control action, (b) decrease control action

to stress out is that the change of the magnitude of the control signal does not tie directly to the size of the controller's input s:

$$|s_2| > |s_1| \text{ implies } |du/ds|_{s_2} = |du/ds|_{s_1}$$

or

$$|s_2| < |s_1| \text{ implies } |du/ds|_{s_2} = |du/ds|_{s_1}.$$

While the algorithm evolves, each change of sgn$(e_k e_{k+1})$ clips the ranges of fluctuation of the control signal. This causes a reduction of the parameter K_{fyzz}, which is proportional to $|du/ds|$. However, between two successive changes of sgn$(e_k e_{k+1})$, $|du/ds|$ remains unaffected, and therefore K_{fuzz} remains unchanged too. Unlike SMFLC, the study of RC-SMFLC as a state-dependent filter shows that no additional effort is needed in order to achieve different control gains in different areas of the transfer characteristic. Between two successive changes of the sign of $e_k e_{k+1}$, the local control gains satisfy the equality

$$\frac{k_1}{\phi_1} = \frac{k_2}{\phi_2} = \cdots = \frac{k_n}{\phi_n}.$$

The transition from the one side of the diagonal $e = 0$ to the other, results in an equal diminution of all the local gains k_i/ϕ_i. In SMFLC the reduction of the local control gains, as the output approaches the diagonal $e = 0$, is designed beforehand. On the contrary in RC-SMFLC this reduction occurs during the evolution of the algorithm.

If the input space is partitioned in more fuzzy subsets as shown in **Fig. 18**, then the output of the closed-loop system will appear to have a lower overshoot. Of course the price for this would be the increase of rise-time.

4.5 Formulation of RC-SMFLC

The control law in conventional SMC is

$$u = \hat{b}^{-1}(\tilde{u} - \hat{f})$$
$$\tilde{u} = G\{\hat{u} - K(x,t)\text{sat}(s/\Phi)\}$$
$$\hat{u} = x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}.$$

The control law in RC-SMFLC will be the same with only one change concerning the control term $K(x,t)\text{sat}(s/\Phi)$ which now becomes $K\text{sgn}(e_k e_{k+1})\text{sgn}(e_k \dot{e}_k)$, i.e.,

$$u = \hat{b}^{-1}(\tilde{u} - \hat{f})$$

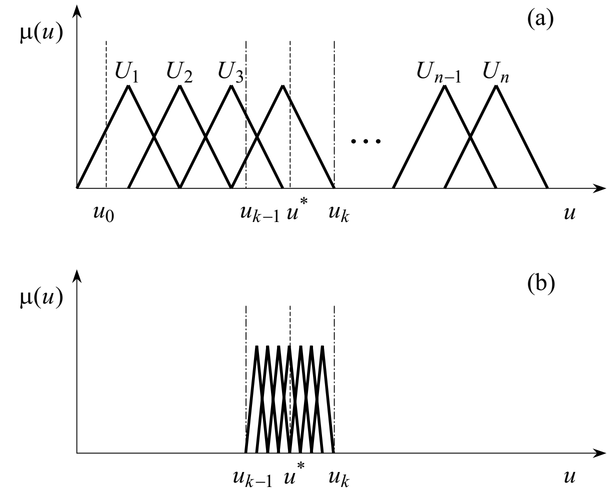

Fig. 18 (a) width of the fuzzy subsets before the change of $\mathrm{sgn}(e_k e_{k+1})$, (b)width of the fuzzy subsets after the change of $\mathrm{sgn}(e_k e_{k+1})$

$$\tilde{u} = G\{\hat{u} - K\mathrm{sgn}(e_k e_{k+1})\mathrm{sgn}(e_k \dot{e}_k)\}$$
$$\hat{u} = x_d^{(n)} - \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}.$$

Is a previous estimation of the system's parameters $f(x,t)$ and $b(x,t)$ always necessary? Assume that the system is stable and $\hat{b}^{-1} = 1$ and $\hat{f} = 0$. Assume also that G and $\hat{u}$ are selected as:

$$G = 1 \text{ and } \hat{u} = u_0$$

where u_0 is a randomly selected value in the interval of the permitted input values. Introducing these values in the above RC-SMFLC equations yields:

$$\begin{aligned} u &= \tilde{u} \\ \tilde{u} &= \hat{u} - K\mathrm{sgn}(e_k e_{k+1})\mathrm{sgn}(e_k \dot{e}_k) \\ \hat{u} &= u_0. \end{aligned} \tag{46}$$

The absence of estimation of the system's parameters $f(x,t)$ and $b(x,t)$ means that RC-SMFLC will have to handle additional parameter uncertainties. However, this additional uncertainty can be viewed as a perturbation that can be compensated by the robustness property possessed by the fuzzy controller. Neglecting the filtering term $u_{filt} = -\hat{b}^{-1} G \sum_{k=1}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)}$, makes the system susceptible to unmodeled high frequencies and consequently this should be done with caution.

5. Comments on RC-SMFLC

5.1 RC-SMFLC and the problem of self-constructing rule bases

A great advantage of the RC-SMFLC method is that its rule base relates the output only to the sign of error e and rate of error's change $\dot{e}$, and not to their magnitude. In conventional incremental fuzzy controllers rules of the form:

$$\text{IF } e \text{ is } E \text{ and } \dot{e} \text{ is } \tilde{E} \text{ THEN } \Delta u \text{ is } U \tag{47}$$

are contained. This kind of rules is the core of the majority of fuzzy controllers such as fuzzy PD, fuzzy PI or fuzzy PID ones, or of the SMFLC that was described before.

On the contrary in RC-SMFLC rules of the form

$$\text{IF } \mathrm{sgn}(e_k \dot{e}_k) < 0 \text{ AND } \Delta u_k > 0 \text{ THEN } \Delta u_{k+1} > 0 \tag{48}$$

are contained in which the sign of $e_k \dot{e}_k$ is related to the sign of the change of the control signal change, $\Delta u_{k+1} > 0$ or $\Delta u_{k+1} < 0$. The change of the control signal magnitude is defined by the transitions above or below the diagonal $e = 0$ and is declared by the term $K\mathrm{sgn}(e_k e_{k-1})$.

A question arising from real implementation of rule-based control systems is how a set of control rules of type (47) can be derived. The success and performance of rule-based control systems depend largely on the availability and the performance of the rule-base. When there are no experts or skilled operators available to supply necessary knowledge, it is necessary to construct the rule-base by directly operating the process being controlled. It is also desirable to refine and improve the rough rule-base derived from experts that may be incomplete, inconsistent or even partly incorrect, especially when the operating condition is changed.

An approach to self-constructing rule-bases would be the implementation of neuro-fuzzy techniques like training of the fuzzy logic controller using back-propagation, orthogonal least squares, nearest neighborhood clustering or genetic algorithms [3]. These techniques consist of initializing the rule-base with low confidence level rules provided by experts. While the algorithm goes on, the rules are modified according to the deviation of the output from the desirable set-point. Finally, the optimal widths and centers of the fuzzy subsets of the rules are selected.

However, according to the previous analysis, to avoid the implementation of neuro-fuzzy adaptive control systems, one uses the RC-SMFLC algorithm. Therefore RC-SMFLC can be considered as an alternative solution to the problem of self-constructing rule-bases.

5.2 Similarity with Iterative Learning Control (ILC)

As the name implies, the correct control action in ILC is progressively learned and hence the desired performance is progressively achieved by repeated trial. The modification of the present control is based on the error information obtained during previous trials. The ILC concept and algorithm were formally proposed by Arimoto [15].

The objective of the learning control is to determine the control input u by repetitive trial such that the error $e(t) = y_d(t) - y(t)$ tends asymptotically to zero. The following algorithms have been proposed:

(a) Error correction algorithm:

$$u_{k+1}(t) = u_k(t) + g_1 e_k(t) \tag{49}$$

(b) Error and error-derivative correction algorithm:

$$u_{k+1}(t) = u_k(t) + p_1 e_k(t) + q_1 \dot{e}_k(t) \tag{50}$$

where k denotes the number of iterations, and g_1, p_1, q_1 are learning gains. The above algorithms may be represented compact matrix forms as:

$$\boldsymbol{u}_{k+1}(t) = \boldsymbol{u}_k(t) + P\boldsymbol{e}_k(t) + Q\dot{\boldsymbol{e}}_k(t)$$

where P and Q are learning matrices.

The change of the control signal u takes place at each iteration k, and in the interval between two successive iterations k and $k+1$ the controller tries to lead the system to the desirable output using the same control signal $u_k(t)$. This reminds directly of the RC-SMFLC where between two successive crossings of the diagonal $e = 0$ the slope of the transfer characteristic $u_{k+1} = f(u_k)$ remains unchanged, while in this interval, $u_k(t)$ is determined by the increase or decrease control mode.

Comparing Iterative Learning Algorithm to RC-SMFLC one can note that although at first look they seem to have some properties in common, they also have subtle differences. Both in the Iterative Learning Algorithm and in the RC-SMFLC no adaptation of controller parameters (e.g. feedforward and feedback gains) takes place and the control input is directly updated. Additionally, both the Iterative Learning Algorithm and RC-SMFLC are intended to eliminate the tracking error uniformly in a time interval of interest and this objective is achieved with the increase of the iteration number.

6. Simulation Results

A case study of RC-SMFLC was done for the linearized problem of arc-welding. This difficult from a control point of view application offered a good chance to test the capabilities and advantages of the proposed control scheme. The tests were also expanded to systems of high order and nonlinear systems.

6.1 The linearized model of arc-welding

Arc-welding is a highly nonlinear process that is subject to many disturbances and parametric changes due to the severe environmental conditions. However, its model can be linearized in small regions around specific operation points and thus its geometrical and thermal characteristics can locally be described by linear transfer functions [16]–[18].

The linearized first-order models for the geometrical characteristics (weight W, height H and depth D) are:

$$\frac{W(s)}{U(s)} = \frac{K_w}{\tau_w s + 1}$$
$$\frac{H(s)}{U(s)} = \frac{K_h}{\tau_h s + 1}$$
$$\frac{D(s)}{U(s)} = \frac{K_d}{\tau_d + 1}$$

where the corresponding control inputs are the thermal power of the torch, the velocity of the torch and the wire-feed rate.

The arc-welding thermal model considers as outputs the weld nugget cross section NS, the heat affected zone HZ, and the centerline cooling rate CR, and as inputs the thermal power of the torch, which are given by

$$\frac{NS(s)}{U(s)} = \frac{K_a}{\tau_a s + 1}$$
$$\frac{HZ(s)}{U(s)} = \frac{K_b(\tau_b s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)}$$
$$\frac{CR(s)}{U(s)} = \frac{K_c}{(\tau_a s + 1)(\tau_b s + 1)}.$$

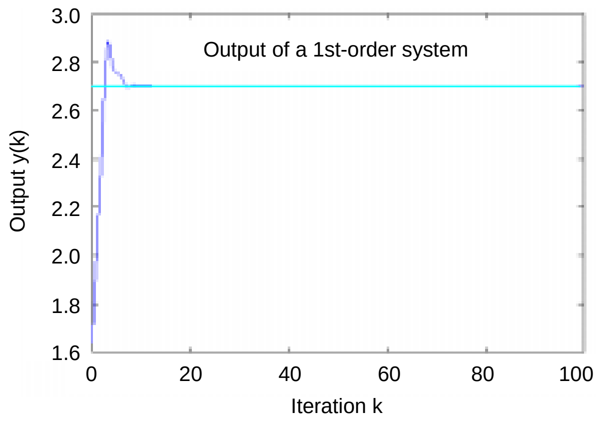

Fig. 19 Control of the 1st-order linear system (a)

The transfer function of NS is of first order, while the transfer function of HZ is given by a second-order one with relative degree one. Finally, the linearized model of CR is of second-order with relative degree two.

6.2 The systems under control

The following systems were considered:

(a) The 1st-order linear system (input: thermal power of the torch Q; output: W)

$$G_m(z) = \frac{0.5(z+1)}{(z-0.77)}.$$

(b) The 2nd-order linear system (input: thermal power of the torch Q; output: CR)

$$G_m(z) = \frac{0.11(z+1)^2}{(z-0.96)(z-0.63)}.$$

(c) The 3rd-order linear system

$$G_m(z) = \frac{0.5(z+1)^2(z-0.4)}{(z-0.5)(z-0.6)(z-0.7)}.$$

(d) The nonlinear system:

$$y(k+1) = 0.95y(k) + 0.025y^2(k) + 0.05u(k).$$

The system models are assumed to be unknown, and the RC-SMFLC had to operate under total parametric uncertainty. The behavior of the controllers was tested both for stabilization and tracking. The first two systems were derived from the linearized model of the arc-welding process [16]–[18] while the other two systems were introduced in order to prove the controller's efficiency in compensating successfully nonlinear and high-order systems. Moreover, the tracking capability of RC-SMFLC was evaluated in the case of the second-order linear model (b) where the aim was to follow a ramp set-point with either positive or negative slope.

As can be seen from **Fig. 19–Fig. 24**, the RC-SMFLC controller showed a superb control performance for the first-and second-order linear models, and was also shown a very good tracking response. A remarkable performance was also shown in the control of high-order and nonlinear processes.

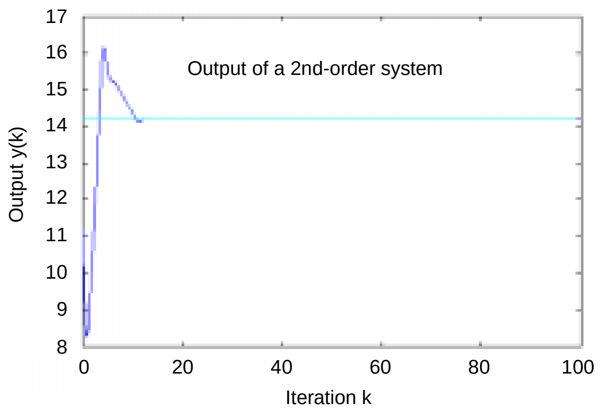

Fig. 20 Control of the 2nd-order linear system (b)

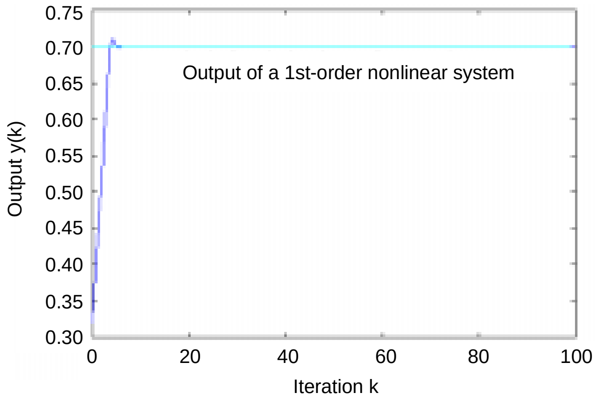

Fig. 21 Control of the 1st-order nonlinear system (d)

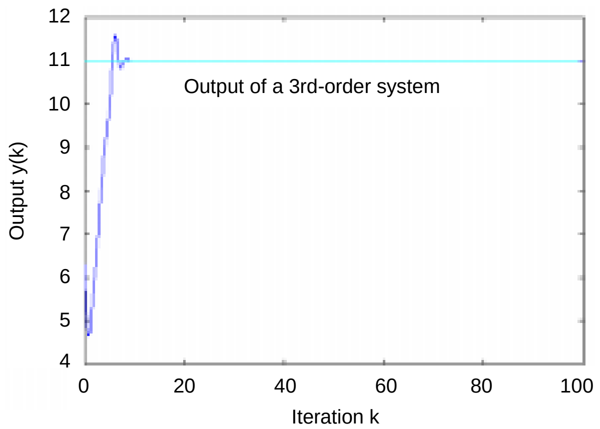

Fig. 22 Control of the third-order linear system (c)

7. Conclusions

A new approach to Sliding-Mode Fuzzy Control was presented in this paper. It combines the basic principles of diagonal-type fuzzy controllers with sliding mode theory, and has the additional advantage that no prior design of the rule base is required. Instead of trying to keep the system on a sliding surface of the form $s(x,t) = \sum_{k=0}^{n-1} \binom{n-1}{k} \lambda^k e^{(n-k)} = 0$, the goal now is to find a control law u which will keep the system in the semiplane $s(x,t) = e\dot{e} < 0$. If the system remains in this semipalne then zero error will be achieved and the asymptotic stability

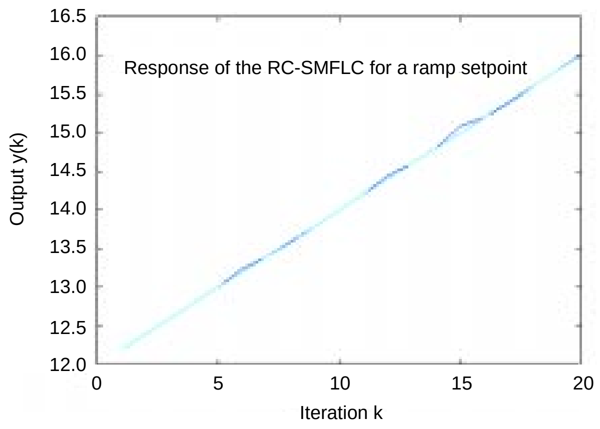

Fig. 23 Tracking of a ramp set-point with positive slope

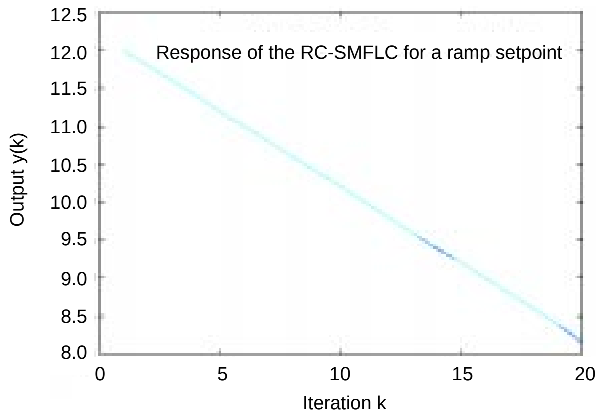

Fig. 24 Tracking of a ramp set-point with negative slope

of the closed-loop system will be guaranteed. The control law that satisfies the above requirements is described by the rules:

- IF $\text{sgn}(e(t)\dot{e}(t)) < 0$ THEN maintain the control action
- IF $\text{sgn}(e(t)\dot{e}(t)) > 0$ THEN change the control action

where the control action can be either an increase or a decrease of the control signal. The increase or decrease of the control signal is realized via the use of fuzzy linguistic rules.

The problem now is to determine changes in the control signal that will become smaller as the system's output approaches the diagonal $e = 0$. This is achieved by changing the widths of the fuzzy subsets in which the control signal can be analyzed. The adaptation of the widths is performed each time the system's output crosses the diagonal $e = 0$, either in a positive or a negative direction.

In contrast to the conventional SMFLC, there is no need to design in advance the shape of the membership functions. All membership functions are identical triangular functions, and therefore the transfer characteristic of RC-SMFLC is roughly approximated by a first-order linear function. Unlike SMFLC, in RC-SMFLC no previous existence of a rule base, connecting the error output to the control signal, is necessary. Thus the problem of unavailable expert's knowledge is solved. The RC-SMFLC algorithm bares a noticeable similarity to other adaptive control techniques such as the Iterative Learning Algorithm [15]. The underlying affinity between RC-SMFLC and other robust or adaptive control schemes becomes clear.

The performance of the model-free controller RC-SMFLC has been tested for several systems, both linear and nonlinear. RC-SMFLC was proved to be excellent. No previous knowledge of the system's model was required. The RC-SMFLC control scheme was proved to be robust for all kind of disturbances and parameter changes of the system. Very fast convergence was achieved. As it happens in conventional control techniques, there is a trade-off between the rise time and the overshoot. A great advantage of the method compared to SMC is the elimination of chattering around the operating point. RC-SMFLC can be viewed as a simple but efficient tool for handling difficult nonlinear control problems with strong parametric uncertainties and disturbances. Its application could cover a wide range of industrial processes such as arc-welding, injection moulding, robotic manipulators, etc.

References

[1] R. Palm, D. Driankov, and H. Hellendoorn, *Model Based Fuzzy Control*. Berlin: Springer-Verlag, 1996, pp. 75–113.

[2] S. G. Tzafestas and C. S. Tzafestas, "Fuzzy and neural control: Basic principles and architectures," in *Methods and Applications of Intelligent Control*, S. G. Tzafestas, Ed. Dordrecht/Boston: Kluwer, 1997, pp. 25–60.

[3] L. X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Englewood Cliffs, NJ: Prentice Hall, 1994, pp. 9–28.

[4] J. Nie and D. Linkens, *Fuzzy-Neural Control: Principles, Algorithms and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1995, pp. 69–86.

[5] J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice Hall, 1991.

[6] J. C. Wu and T. S. Liu, "A sliding-mode approach to fuzzy control design," *IEEE Trans. on Control Systems Technology,* vol. 4, no. 2, pp. 141–151, 1996.

[7] S. C. Lin and C. C. Kung, "Linguistic fuzzy sliding mode controller," in *Proc. 1992 Amer. Contr. Conf.,* Chicago, IL, 1992, pp. 1904–1905.

[8] J. K. Pieper and K. R. Goheen, "Discrete-time sliding mode control via input-output models," in *Proc. Amer. Contr. Conf.,* San Fransisco, CA, 1993, pp. 964–965.

[9] S. G. Tzafestas, "Digital PID and self-tuning control," in *Applied Digital Control*, S. G. Tzafestas, Ed. Amsterdam: North Holland, 1985, pp. 1–49.

[10] W. Li, "A method for design of a hybrid neuro-fuzzy control system based on behavior modeling," *IEEE Trans. on Fuzzy Systems*, vol. 5, no. 1, pp. 128–137, 1997.

[11] H. Liand and H. B. Gatland, "Conventional fuzzy control and its enhancement," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 26, no. 5, pp. 791–801, 1996.

[12] S. G. Tzafestas and N. P. Papanikolopoulos, "Incremental fuzzy expert PID control," *IEEE Trans. on Industrial Electronics*, vol. 37, no. 5, pp. 365–371, 1990.

[13] H. A. Malki, H. Li, and G. Chen, "New design and stability analysis of fuzzy proportional-derivative control systems," *IEEE Trans. on Fuzzy Systems*, vol. 4, no. 4, pp. 245–254, 1994.

[14] H. Malki, D. Misir, D. Feigenspan, and G. Chen, "Fuzzy PID control of a flexible-joint robot arm with uncertainties from time-varying loads," *IEEE Trans. on Control Systems Technology*, vol. 5, no. 3, pp. 371–378, 1997.

[15] S. Arimoto, "Learning control theory for robotic motion," *Int. J. Adaptive Control and Signal processing*, vol. 4, pp. 543–564, 1990.

[16] S. G. Tzafestas, E. Kyriannakis, and G. Rigatos, "Adaptive, model based predictive and neural control of the welding process," in *Methods and Applications of Intelligent Control*, S. G. Tzafestas, Ed. Dordrecht/Boston: Kluwer, 1997, pp. 509–548.

[17] C. Doumanidis and D. E. Hardt, "A model for in-process control of thermal properties during welding," *ASME J. Dynamic Systems Meas. and Control*, vol. 111, pp. 40–50, 1989.

[18] D. V. Nishar, J. L. Schiano, W. R. Perkins, and R. A. Weber, "Adaptive control of temperature in arc-welding," *IEEE Control Systems Magazine*, pp. 4–12, 1994.

Biographies

Spyros G. Tzafestas received the B.Sc. degree in Physics and Electronics from Athens University, the D.I.C. degree in Electrical Engineering from Imperial College, the M.Sc. degree in Automatic Control Systems from London University, and the Ph.D. and D.Sc. degrees in Systems, Control and Automation from Southampton University.

He is currently Professor of Control and Robotics, Director of the Intelligent Robotics and Automation Laboratory and Director of the Institute of Communication and Computer Systems at the National Technical University, Athens, Greece. From 1969 to 1973, he was leader of the computer control group at the Computer Science Department of the Nuclear Research Center "Democritos" in Athens, and from 1973 to 1984, he was Professor of Systems and Control Engineering at Patras University, Patras, Greece. His current interests include robust and adaptive control, robot modeling and control, knowledge-based systems, neural-fuzzy control systems and computer-aided manufacturing systems. He has published 30 books and over 600 journal and conference technical papers. He is an active reviewer of 20 journals, associate editor of 6 journals, the Editor-in-Chief of the Journal of Intelligent and Robotic Systems, and the Editor of the Kluwer book series entitled Microprocessor-Based and Intelligent Systems Engineering. He has served as guest editor for fifteen special issues on distributed systems, large-scale systems, orthogonal expansions, robotics, neural networks and soft-computing and has organized and/or chaired many international conferences on systems, computing, robotics and control.

Dr. Tzafestas is a member of the New York Academy of Sciences, a Fellow of the IEEE (CSS, CS, IES, RAS), a Fellow of IEE (London), a member of ASME (N.Y.) and Chairman of the Control Systems and Robotics Committee of the International Association for Mathematics and Computers in Simulation (IMACS). He is recipient of a D.Sc. (Hon) from the International University Foundation and an Honorary Doctorate (Dr.-Ing. E.h) from the Technical University of Munich, Honorary President of SIRES (Society of Intelligent Robotics and Expert Systems, Brussels), and a member of the IFAC SECOM and MIM Technical Committees. He is coordinator for the part of NTUA in several European Community Projects (ESPRIT CIM, Stimulation Action, BRITE, TIDE, STRIDE, INTAS, Erasmus, Comett, Socrates) on Information Systems Technology.

Gerasimos G. Rigatos received a degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece in 1995. Currently he is working towards the completion of his Ph.D. thesis at the Intelligent Robotics and Automation Laboratory of the NTUA. His research work is mainly concerned with intelligent control techniques using neural networks, fuzzy systems, genetic algorithms and learning automata. He has participated in several research projects on neuro-fuzzy techniques for robot path planning and control, and on genetic algorithms for warehouse management systems. He has published 8 papers and he is a member of the Technical Chamber of Greece.