The Fusion of Computationally Intelligent Methodologies and Sliding-Mode Control—A Survey

Okyay Kaynak, *Senior Member, IEEE*, Kemalettin Erbatur, and Meliksah Ertugrul

***Abstract*—This paper surveys how some "intelligence" can be incorporated in sliding-mode controllers (SMCs) by the use of computational intelligence methodologies in order to alleviate the well-known problems met in practical implementations of SMCs. The use of variable-structure system theory in design and stability analysis of fuzzy controllers is also discussed by drawing parallels between fuzzy control and SMCs. An overview of the research and applications reported in the literature in this respect is presented.**

***Index Terms*—Computational intelligence, sliding-mode control, soft computing.**

I. Introduction

VARIABLE-STRUCTURE systems (VSSs) with a sliding mode were first proposed in early 1950s [1]–[3]. However, due to the implementation difficulties of high-speed switching, it was not until the 1970s that the approach received the attention it deserved. Sliding-mode controllers (SMCs) nowadays enjoy a wide variety of application areas, such as in general motion control applications and robotics, in process control, in aerospace applications, and in power converters [4]–[6]. The main reason for this popularity is the attractive properties that SMCs have, such as good control performance for nonlinear systems, applicability to multiple-input–multiple-output (MIMO) systems, and well-established design criteria for discrete-time systems. The most significant property of an SMC is its robustness. Loosely speaking, when a system is in a sliding mode, it is insensitive to parameter changes or external disturbances.

The primary characteristic of a VSS is that the feedback signal is discontinuous, switching on one or more manifolds in state space. When the state crosses each discontinuity surface, the structure of the feedback system is altered. Under certain circumstances, all motions in the neighborhood of the manifold are directed toward the manifold and, thus, a sliding motion on a predefined subspace of the state space is established in which the system state repeatedly crosses the switching surface [7]. This mode has useful invariance properties in the face of uncertainties in the plant model and, therefore, is a good candidate for tracking control of uncertain nonlinear systems. The theory is well developed, especially for single-input systems in controller canonical form.

The theory of VSSs with a sliding mode has been studied intensively by many researchers. A recent comprehensive survey is given in [4]. Motion control, especially in robotics, has been an area that has attracted particular attention and numerous reports have appeared in the literature [8]–[12]. One of the first experimental investigations that demonstrated the invariance property of a motion control system under a sliding mode is due to Kaynak *et al.* [13].

In practical applications, a pure SMC suffers from the following disadvantages. Firstly, there is the problem of chattering, which is the high-frequency oscillations of the controller output, brought about by the high-speed (ideally, at infinite frequency) switching necessary for the establishment of a sliding mode. In practical implementations, chattering is highly undesirable because it may excite unmodeled high-frequency plant dynamics, and this can result in unforeseen instabilities. It should here be mentioned that the switching type of control law may be acceptable, even advantageous in the case of pulsewidth modulation (PWM) control of electrical motors in which the control input is an electrical voltage rather than a mechanical torque or acceleration, as long as the switching frequency is beyond the frequency range of unmodeled dynamics.

Secondly, an SMC is extremely vulnerable to measurement noise since the input depends on the sign of a measured variable that is very close to zero [14]. Thirdly, the SMC may employ unnecessarily large control signals to overcome the parametric uncertainties. Last, but not least, there exists appreciable difficulty in the calculation of what is known as the equivalent control. A complete knowledge of the plant dynamics is required for this purpose [15]. To alleviate these difficulties, several modifications to the original sliding control law have been proposed [16], the most popular being the boundary-layer approach, which is, in essence, the application of a high-gain feedback when the motion of the system reaches ε-vicinity of a sliding manifold [9], [15]. This approach is based on the idea of the equivalence of the high-gain systems and the systems with sliding modes [17]. Another variation of the scheme is called provident control that combines variable-structure control (VSC) and variable-structure adaptation and performs hysteretic switching between the

Manuscript received March 15, 1999; revised October 14, 2000. Abstract published on the Internet November 15, 2000. This work was supported by TUBITAK Project Grant EEEAG-199E017 and by the Bogazici University Research Fund under Grant 99A202 and Grant 00A203D.

O. Kaynak is with the Faculty of Engineering, Bogazici University, 80815 Bebek-Istanbul, Turkey.

K. Erbatur was with the Electronic Systems Group, TUBITAK Marmara Research Centre, 41470 Gebze-Kocaeli, Turkey. He is now with the Faculty of Engineering and Natural Sciences, Sabanci University, 81474 Tuzla-Istanbul, Turkey.

M. Ertugrul is with the Electronic Systems Group, TUBITAK Marmara Research Centre, 41470 Gebze-Kocaeli, Turkey.

Publisher Item Identifier S 0278-0046(01)01114-5.

structures so as to avoid a sliding mode [18], [19]. Both approaches are based on the calculation of the equivalent control, requiring a good mathematical model of the plant. In [20], the use of an estimation technique is proposed to avoid the computational burden in the calculation of the equivalent control.

The technological developments of the recent decade have increased the use of high-speed computers in control applications. It is now possible and economically feasible to use complex model-based control paradigms in practical applications, using advanced control strategies derived from adaptive, nonlinear, and robust control theories. This has resulted in the development of the "intelligent control" field, and a host of new control approaches based on fuzzy logic (FL), neural networks (NNs), evolutionary computing, and other techniques adapted from artificial intelligence have come into common use. These methodologies provide an extensive freedom for control engineers to exploit their understanding of the problem, to deal with problems of vagueness, uncertainty, or imprecision, and to learn by experience and, therefore, they are good candidates for alleviating the problems associated with SMCs discussed above. A good deal of work is reported in the literature in this respect, some selective examples of which are cited in [21]–[107].

This survey paper first discusses the basic principle of VSSs and SMCs and the difficulties met in practical implementations. The popular measures suggested in the literature against the difficulties are then addressed. The paper continues with a discussion on how some "intelligence" can be incorporated in SMCs by the use of computational intelligence methodologies and presents an overview of the research and applications reported in the literature in this respect. Parallels are drawn between fuzzy control and VSSs with a sliding mode.

II. VSS Revisited

In the application of VSS theory to the control of nonlinear processes, it is argued that one only needs to drive the error to a "switching" or "sliding" surface, after which the system will stay in a "sliding mode" and will, therefore, not be affected by any modeling uncertainties and/or disturbances. Intuitively, VSS with a sliding mode is based on the argument that the control of first-order systems (i.e., systems described by first-order differential equations) is much easier, even when they are nonlinear or uncertain, than the control of general nth-order systems [15].

A. Description of the General Plant Dynamics under Control

Consider a nonlinear nonautonomous MIMO system of the form

$$x_i^{(k_i)} = f_i(X) + \sum_{j=1}^{m} b_{ij} u_j \tag{1}$$

where $x_i^{(k_i)}$ indicates the kth derivative of x_i and

$$X = [x_1 \quad \dot{x}_1 \quad \cdots \quad x_1^{k_1-1} \quad \cdots x_m \quad \dot{x}_m \quad \cdots \quad x_m^{k_m-1}]^T. \tag{2}$$

Defining

$$U = [u_1 \quad \cdots u_m]^T \tag{3}$$

and assuming that X is $(n \times 1)$, the system equation becomes

$$\dot{X}(t) = F(X) + BU(t) \tag{4}$$

where B is the $(n \times m)$ input gain matrix. Such systems are called square systems since they have as many control inputs as outputs x_i to be controlled [15].

B. Sliding Surface

For the systems given in (4), the sliding surface S $(m \times 1)$ is selected generally as

$$S(X,t) = G(X^d(t) - X(t)) - \phi(t) - S_a(X) \tag{5}$$

where

$$\phi(t) = GX^d(t) \quad \text{and} \quad S_a(X) = GX(t) \tag{6}$$

are the time- and the state-dependent parts of the sliding function. In (5), X^d represents the desired (reference) state vector and G is the $(m \times n)$ slope matrix of the sliding surface. Generally, the G matrix is selected such that the sliding surface function becomes

$$s_i = \left(\frac{d}{dt} + \lambda_i\right)^{k_i - 1} e_i \tag{7}$$

where e_i is the error for x_i $(e_i = x_i^d - x_i)$ and λ_is are selected as positive constants. Therefore, e_i goes to zero when s_i equals zero.

The objective in an SMC is to force the system states to the sliding surface. Once the states are on the sliding surface, the system errors converge to zero with an error dynamics dictated by the matrix G.

C. SMC Design

In the design of an SMC, there exists a number of approaches in front of the designer. The method described in Section I is based on the selection of a Lyapunov function. The control should be chosen such that the candidate Lyapunov function satisfies Lyapunov stability criteria.

The Lyapunov function is selected as

$$V(S) = \frac{S^T S}{2}. \tag{8}$$

It can be noted that this function is positive definite. $(V(S = 0) = 0$ and $V(S) > 0 \; \forall S \neq 0)$.

It is aimed that the derivative of the Lyapunov function is negative definite. This can be assured if one can assure that

$$\frac{dV(S)}{dt} = S^T \, D \, \text{sign}(S) \tag{9}$$

where D is the $(m \times m)$ positive-definite diagonal gain matrix and $\text{sign}(S)$ denotes the signum function, applied to each element of S, i.e.,

$$\text{sign}(S) = [\text{sign}(s_1) \quad \cdots \quad \text{sign}(s_m)]^T \tag{10}$$

and $\text{sign}(s_i)$ is defined as

$$\text{sign}(s_i) = \begin{cases} +1, & s_i > 0 \\ 0, & s_i = 0 \\ -1, & s_i < 0. \end{cases} \quad (11)$$

Taking the derivative of (8), and equating this to (9), one will obtain the following equation:

$$S^T \frac{dS}{dt} = -S^T\, D\, \text{sign}(S). \quad (12)$$

By taking the time derivative of (5) and using the plant equation

$$\frac{dS}{dt} = \frac{d\phi}{dt} - \frac{\partial S_a}{\partial X}\, \frac{dX}{dt} = \frac{d\phi}{dt} - G(F(X) + BU) \quad (13)$$

is obtained. By putting (13) into (12), the control input signal can be obtained as

$$U(t) = U_{\text{eq}}(t) + U_c(t) \quad (14)$$

where $U_{\text{eq}}(t)$ is the equivalent control and it is written as

$$U_{\text{eq}}(t) = -(GB)^{-1} \left(GF(X) - \frac{d\phi(t)}{dt} \right) \quad (15)$$

and $U_c(t)$ is the corrective control term and written as

$$U_c(t) = (GB)^{-1}\, D\, \text{sign}(S) = K\, \text{sign}(S). \quad (16)$$

Another design possibility is to ensure that

$$\frac{dV(S)}{dt} = -S^T DS \quad (17)$$

in which case the corrective term that adds onto $U_{\text{eq}}(t)$ is obtained as

$$U'_c(t) = (GB)^{-1} DS = K'S. \quad (18)$$

It is to be noted, however, that there is no switching in this case and the approach does not result in a VSS.

III. Computational Intelligence

In industrial applications, control engineers often have to deal with complex systems, having multiple variable and multiple parameter models with perhaps nonlinear coupling. The conventional approaches for understanding and predicting the behavior of such systems based on analytical techniques can prove to be inadequate, even at the initial stages of establishing an appropriate mathematical model. The computational environment used in such an analytical approach is perhaps too categoric and inflexible in order to cope with the intricacy and the complexity of the real world industrial systems. It turns out that, in dealing with such systems, one has to face a high degree of uncertainty and tolerate imprecision. Trying to increase precision can be very costly.

In the face of the difficulties stated above, Prof. L. A. Zadeh proposes a different approach to machine intelligence. He separates hard-computing-techniques-based artificial intelligence from soft-computing-techniques-based computational intelligence. Hard computing is oriented toward the analysis and design of physical processes and systems and has the characteristics of precision, formality, and categoricity. It is based on binary logic, crisp systems, numerical analysis, probability theory, differential equations, functional analysis, mathematical programming, approximation theory, and crisp software. On the other hand, soft computing is oriented toward the analysis and design of intelligent systems. It is based on FL, artificial NNs (ANNs), and evolutionary computing and has the attributes of approximation and dispositionality. Although in hard computing, imprecision and uncertainty are undesirable properties, in soft computing the tolerance for imprecision and uncertainty is exploited to achieve an acceptable solution at a low cost, tractability, and high machine intelligence quotient (MIQ). Zadeh argues that soft computing, rather than hard computing, should be viewed as the foundation of machine intelligence. A center established and directed by him at the University of California, Berkeley, the Berkeley Initiative for Soft Computing (BISC), devotes its activities to this concept. Soft computing, as he explains, is a consortium of methodologies providing a foundation for the conception and design of intelligent systems.

The principal constituents of soft computing are as follows:

- FL;
- ANNs;
- probabilistic reasoning (PR), including genetic algorithms (GAs), chaos theory, and parts of learning theory.

FL is mainly concerned with imprecision and approximate reasoning, NNs mainly with learning and curve fitting, and probabilistic reasoning mainly with uncertainty and propagation of belief. Table I, constructed by Fukuda and Shimojima [86], gives a comparison of their capabilities in different application areas, together with those of control theory and artificial intelligence. It is seen that the approaches are complementary rather than competitive, and there can be much to be gained in using them in a combined manner, rather than exclusively. For example, an integration of FL and neuro-computing has become very popular (known as neuro-fuzzy control) with many diverse applications, ranging from chemical process control to consumer goods. The synergy of NNs and fuzzy reasoning follows naturally. They are the best couple to mimic the structure and the reasoning of human brain. An NN accomplishes what a person does with data and FL realizes what a person does with language. The resulting controller is a nonlinear one, suitable to overcome the difficulties involved in using linear controllers for (naturally) nonlinear systems.

GAs, on the other hand, are parallel global searching algorithms based on the mechanism of natural selection and genetics, proposed by Holland in the early 1970s [87]. In this optimization scheme, the solution space is coded as gene-like strings. An objective function is used to evaluate the fitness of the string in the environment. The basic genetic algorithm uses three operators, namely, reproduction, crossover, and mutation, and effectively copies, swaps, or changes parts of the strings in a randomized manner. In every generation, a new set of artificial strings is thus created from the fittest of the previous generation.

GAs are especially effective in multipeak problems with local optimum solutions, since they search for a population of points rather than a single point. They are theoretically and empirically

TABLE I
COMPARISON OF CAPABILITIES OF DIFFERENT METHODOLOGIES [86]

	Mathematical Model	Learning Data	Operator Knowledge	Real Time	Knowledge Representation	Non-linearity	Optimization
Control Theory	Good or Suitable	Unsuitable	Needs other methods	Good or Suitable	Unsuitable	Unsuitable	Unsuitable
Neural Network	Unsuitable	Good or Suitable	Unsuitable	Good or Suitable	Unsuitable	Good or Suitable	Fair
Fuzzy Logic	Fair	Unsuitable	Good or Suitable	Good or Suitable	Needs other methods	Good or Suitable	Unsuitable
Artificial Intelligence	Needs other methods	Unsuitable	Good or Suitable	Unsuitable	Good or Suitable	Needs other methods	Unsuitable
Genetic Algorithms	Unsuitable	Good or Suitable	Unsuitable	Needs other methods	Unsuitable	Good or Suitable	Good or Suitable

proven to provide robust search in complex spaces. The main advantages of GAs are that they are computationally simple and powerful and they are not fundamentally limited by restrictive assumptions about the search space (assumptions concerning continuity, existence of derivatives, unimodality, and other matters).

GAs differ from conventional optimization and search procedures in the following ways, in that they:

1) work with a coding of the parameter set, not the parameters themselves;
2) search from a population of points, not a single point, and are capable of handling large search spaces;
3) use probabilistic transition rules, rather than deterministic ones.

The main disadvantage they have is that they may be too slow for real-time applications.

The use of a GA in a VSS is generally to fulfill a secondary function, such as for the tuning of a fuzzy controller (FC) and the determination of suboptimum controller parameters for SMCs.

IV. COMPUTATIONAL INTELLIGENCE AND SLIDING-MODE CONTROL

The fusion of soft-computing methodologies in sliding-mode control or, more correctly in VSSs, in general has the objective of alleviating the problems met in practical implementations of SMCs. Conversely, the use of VSS theory in systems based on soft-computing techniques has the goal of rigorous design and stability analysis. For example, in the case of ANNs, a sliding-mode approach can ensure convergence and stability of the learning algorithm. In the literature, a great variety of imaginative schemes has appeared [21]–[97] which cannot all be commented upon in this survey paper. In what follows, some representative work is cited and commented upon. The principal constituents of soft computing are handled in separate sections with regard to their integration with a VSS. The integration of neuro-FCs with a VSS is considered in a separate section.

A. FL and VSS

The integration of an FL system in an SMC is seen in many examples where an attempt to relieve the implementation difficulties of the SMC is made via the addition of the FL system. On the other hand, some significant research work has originated due to different difficulties, i.e., the difficulties in carrying out a rigorous stability analysis of FCs. In such studies, parallels are first drawn between the FC and the SMC, and then the well-developed theory of a VSS is utilized in discussing the performance characteristics and the stability of the FC. Whatever the main objective is, the approaches reported in the literature can be separated into two general classes: direct and indirect.

1) Indirect Approaches: In indirect approaches, the basic design and implementation philosophy of SMCs is followed to a great extent and FL systems are used to fulfil a secondary function. It may be there in order to either adapt the controller parameters (e.g., D and K in (16)) or to eliminate chattering in some ways or to mitigate the modeling difficulties and the consequent difficulty in the calculation of equivalent control $U_{\rm eq}$ in (14).

a) Use of FL in a smoothing filter: One of the earliest works seen in the literature on the integration of VSS theory with FCs proposes to smooth the control input in a VSS and, thus, prevent chattering by the use of a low-pass filter as follows [73]:

$$\dot{U}_f = \lambda(U_f - U) \tag{19}$$

where

U_f filtered control;
U unfiltered one, calculated from (14);
λ bandwidth of the filter.

When λ is small, abrupt changes in U can be prevented. However, if λ is too small, then the difference between U and U_f may become too large and the deviation of the system from the ideal sliding mode becomes more pronounced. Intuitively, when the state is within the vicinity of the sliding surface λ should be

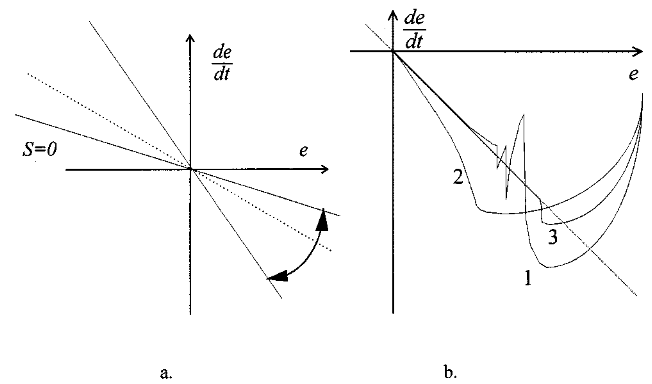

Fig. 1. Effects of variations in G and D.

Fig. 2. Experimental two-link direct-drive arm.

small since the change in U is expected to be abrupt. Otherwise, λ can be made large so that the advantages of VSS with a sliding mode are not lost. This can be done by a fuzzy set of rules [73]. The advantages of this approach as compared to conventional chattering avoidance techniques based on "hard computing" are not discussed in the paper and is open to discussion.

b) Use of FL for the tuning of SMC parameters: Consider the SMC as defined by (14)–(16) or (18). The effects of the design parameters G and D on the system performance are shown in Fig. 1(a) and (b) for the two-dimensional case. The parameter G determines the slope of the sliding line and, therefore, the larger it is, the faster will the system response be, but a too large value of G can cause overshoot, or even instability. It would, therefore, be advantageous to adaptively vary the slope in such a way that the slope is increased as the magnitude of the error gets smaller.

The effect of the design parameter D on the performance of the system is shown in Fig. 1(b). The curve labeled "1" corresponds to the case when D is large. The system states reach the sliding line in a short time, but overshoot it by a considerable amount. The curve labeled "2" reflects the case with a small D parameter. Neither curve 1 nor curve 2 is very desirable. The third curve in the phase plane can be obtained via a fuzzy adaptation algorithm in which the parameter D is increased only when the states are close to the sliding line. Erbatur *et al.*follows such an approach in [51] for trajectory control of a robotic manipulator. The absolute value of the position error, the absolute value of the sliding function S, and a chattering variable defined as the cumulative absolute change in the control input over the last 30 control periods are used as the input linguistic variables. The fuzzy adaptation scheme outputs the values of D, G and a gain parameter K that is a multiplicative term in front of $U_c(t)$. In that particular SMC scheme, $U_c(t)$ is a function of S and $\dot{S}$. The gain K balances the chattering and the error in the system. The fuzzy system tunes the parameters in such a way as to get the best tracking performance without chattering.

In an FL system, outputs are computed by a mechanism of IF-THEN rules. The general type IF-THEN rules that are used in the fuzzy adaptation scheme are of the form

$$R^{(k)}: \quad \text{IF } x_1 \text{ is } F_1^k \text{ and} \cdots \text{and } x_n \text{ is } F_n^k \quad \text{THEN } y^k = c^k. \tag{20}$$

Here, x_i are real-valued input variables, F_i^k are fuzzy sets specified by membership functions $\mu_{F_i^k}(x)$, c^k is a real-valued constant, and y^k is the system output from rule $R^{(k)}$. If there are m rules, $R^{(1)}, \ldots, R^{(m)}$ in the rule base, the output of the fuzzy adaptive system is computed as

$$y = \Lambda \frac{\sum_{k=1}^{m} w^k y^k}{\sum_{k=1}^{m} w^k}. \tag{21}$$

In this weighted sum, the weights w^k are computed from

$$w^k = \prod_{i=1}^{n} \mu_{F_i^k}(x) \tag{22}$$

and Λ is introduced as a scaling and tuning term and is very useful in an implementation. The fuzzy adaptive scheme described above is implemented for trajectory control of a two-link arm shown in Fig. 2. Experimental results obtained demonstrate the feasibility of the approach for good trajectory following performance without chattering [51].

The tuning of SMC parameters via fuzzy systems is also used in [33], where a discrete-time fuzzy SMC (FSMC) is put to use for vibration control of a smart structure featuring a piezo film actuator. First, a discrete-time model with mismatched uncertainties is considered for the design of a discrete-time SMC. It has two parts: an equivalent part and a discontinuous part. The FSMC is formulated by employing a fuzzy technique to appropriately determine control parameters such as the discontinuous feedback gain. The controller is then used in experiments to demonstrate the effectiveness of the proposed method.

c) Use of FL for modeling uncertainties: One of the main difficulties in the design of an SMC is the fact that an exact knowledge of the plant is rarely (if ever) available. Even the bounds of the uncertainties may not be known. This may result in an overconservative design, i.e., a larger K in (18) than necessary. A number of researchers have, therefore, proposed the use of adaptive FL identifiers for the uncertainties. For example, in [22], a fuzzy system architecture is employed to adaptively model the plant nonlinearities which have unknown uncertainties. In the proposed scheme, the bound of the modeling error, which results from the error between the fuzzy system and the actual nonlinear plant (an inverted pendulum system) is identified adaptively. Using this bound, a sliding control input is calculated.

The approach proposed in [23] is similar; a nonlinear system is first linearized around a number of operating points and then FL principles are used to aggregate each locally linearized model into a global model representing the nonlinear system. Finally, a robust SMC is proposed that guarantees the asymptotic stability of the system.

The use of fuzzy approximators [99], [100] in modeling uncertainties is also seen in the literature. For example, in [96], two adaptive SMC schemes with FL systems (approximators) are designed. FL systems are used to approximate the unknown system functions. In the first scheme, an FL system approximates the unknown function f of the nonlinear system $\dot{X} = f(X) + bU$. A robust adaptive law is employed to minimize the approximation errors between the real system functions and the fuzzy approximators. In the second scheme, two FL systems are used to approximate f and b, respectively. Stability proofs of the control schemes are given.

The schemes described above somewhat fall in the direct approach class, too, because VSS theory is utilized in stability analysis. Similarly, the work reported in [25] combines the direct and indirect approaches in the sense that, firstly, FL systems are employed to approximate the unknown dynamics in each subsystem of an interconnected nonlinear system (NNs can equivalently be used for this purpose). Then, an FSMC is developed to compensate for the fuzzy approximating errors and to attenuate the interactions between subsystems. Global asymptotic stability is established in the Lyapunov sense, with the tracking errors converging to a neighborhood of zero.

d) FL controller complementary to SMC: Some approaches seen in the literature include FL systems as complementary controllers to SMC schemes. First, SMCs are designed. For performance enhancement and chattering elimination, additional fuzzy control terms are used together with the SMC output.

In [34], such a scheme is presented for linearized systems suffering from uncertainties. SMC combined with fuzzy tuning is used to compensate for the influence of unmodeled dynamics and chattering. The control law is of the form

$$u = u_{\text{eq}} + u_r + u_{ft} \tag{23}$$

where u_{eq} is the equivalent control, u_r stands for the reaching control part of a sliding controller, and u_{ft} is the complementary fuzzy control. For the linearized system model, considering matching conditions and bounds on the uncertainties in the model, the equivalent and the reaching control terms are designed and a stability proof with Lyapunov approach is given. Following this, u_{ft} is designed to accelerate the reaching phase and to reduce chattering while maintaining sliding behavior. The FC is based on the following ideas.

When the state trajectories are far from the sliding hyperplane ($|S|$ is large), the switching gain should be increased. When the state trajectories deviate from the sliding surface ($S\dot{S} > 0$), if $|S|$ is large the switching gain should be increased to force the trajectories back. When the state trajectories approach the sliding surface ($S\dot{S} < 0$), if $|\dot{S}|$ is large the switching gain should be decreased to reduce chattering. Introducing the scaled modulus S_* and $\dot{S}_*$, the following fuzzy rules are proposed:

1) if S_* is large, then w_s is large;
2) if S_* is small, then w_s is small;
3) if $S\dot{S} > 0$ and $\dot{S}_*$ is large, then u_{ft} is large;
4) if $S\dot{S} > 0$ and $\dot{S}_*$ is small, then u_{ft} is small;
5) if $S\dot{S} < 0$ and $\dot{S}_*$ is large, then u_{ft} is small;
6) if $S\dot{S} < 0$ and $\dot{S}_*$ is small, then u_{ft} is large.

The term w_s is a weighting factor depending on S. Using singletons for u_{ft} and w_s and centroid method for defuzzification, fuzzy control contribution is computed as follows:

$$u_{ft} = \frac{\mu_{S\_large} w_s + \mu_{S\_small} 0}{\mu_{S\_large} + \mu_{S\_small}}. \tag{24}$$

The membership functions used are exponential ones. Simulations on a two-mass servo system indicate superior performance over pole placement and pure sliding-mode schemes in that high robustness against parameter and load variations is obtained.

The approach in [34] is further generalized to a class of nonlinear systems in [55] where the simulations on a robotic manipulator are presented.

2) Direct Approaches: An appreciable amount of work is seen in the literature in which a computational intelligence methodology is used directly in the design of a VSS-theory-based scheme (generally, for control purposes) or, conversely, the VSS theory is used in the computationally intelligent architecture for parameter adaptation or for a robust and stable design. In this survey paper, such schemes are classified as direct schemes and some exemplary work is briefly described below.

a) Similarity between FC and SMC: In order to draw parallels between FC and SMC, let us consider a single-input–single-output FC and suppose that we have a set of rules as follows:

$$\begin{aligned}
&\text{R1:} \quad \text{IF } x = \text{NB THEN } y = \text{BIGGER}\\
&\text{R2:} \quad \text{IF } x = \text{NS THEN } y = \text{BIG}\\
&\text{R3:} \quad \text{Z THEN } y = \text{MEDIUM}\\
&\text{R4:} \quad \text{PS THEN } y = \text{SMALL}\\
&\text{R5:} \quad \text{IF } x = \text{PB THEN } y = \text{SMALLER.}
\end{aligned} \tag{25}$$

In (25), x is the input, y is the outpu, t and NB, NS, Z, PS, and PB are the labels of fuzzy sets, which are negative big, negative small, zero, positive small, and positive big, respectively. Let the universe of discourses of x and y be partitioned as shown in Fig. 3.

The fuzzy inference performs a mapping from the fuzzy sets in X to the fuzzy sets in Y, based on the rule base and compositional rule of inference for fuzzy reasoning. Let X_s be a fuzzy set in X. Then, each of the rules R_i in the rule base determines a fuzzy set $X_s \circ R_i$ in Y according to the compositional rule of inference. Let X_s be a fuzzy singleton with support α, i.e.,

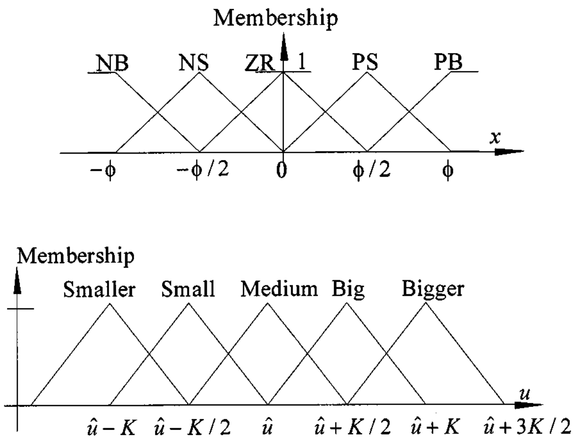

Fig. 3. Universe of discourse for x and u.

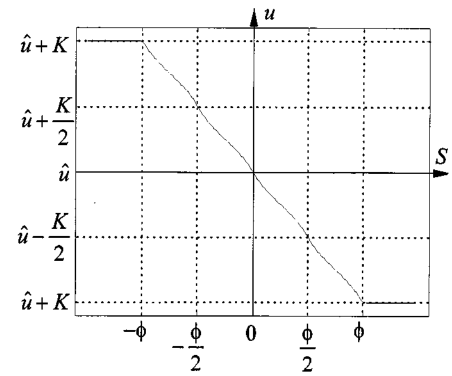

Fig. 4. Nonlinear operating line in an FSMC, resembling the saturation function used in conventional SMC.

$\mu_{X_s}(x) = 1$ for $x = \alpha$ and $i_{X_s} = 0$, otherwise, and let us use the center-of-area defuzzifier, i.e.,

$$y = \frac{\int x\mu_{X_s}(x)\, dx}{\int \mu_{X_s}(x)\, dx}. \tag{26}$$

Under these conditions, it is very easy to find out what the output will be for the corner values of the membership functions of x, i.e., $x = -\Phi, -\Phi/2, 0, +\Phi/2$ and $+\Phi$. In between the corner values, nonlinear functions determine the crisp y value as shown in Fig. 4. Kim and Lee [65] show that the result of inference for every x is given by

$$y = \overline{u} - K\ sig(x/\Phi) \tag{27}$$

where the sigmoid function $\mathrm{sig}(z)$ is defined as

$$\mathrm{sig}(z) = \begin{cases} -1, & \text{if } z \le -1 \\ -\dfrac{1}{2}\,\dfrac{(2z+3)(3z+1)}{4z^2+6z+1}, & \text{if } -1 \le z \le -\dfrac{1}{2} \\ -\dfrac{1}{2}\,\dfrac{z(2z+3)}{4z^2+2z-1}, & \text{if } -\dfrac{1}{2} \le z \le 0 \\ \dfrac{1}{2}\,\dfrac{z(2z-3)}{4z^2-2z-1}, & \text{if } 0 \le z \le \dfrac{1}{2} \\ \dfrac{1}{2}\,\dfrac{(2z-3)(3z-1)}{4z^2-6z+1}, & \text{if } \dfrac{1}{2} \le z \le 1 \\ 1, & \text{if } z \ge 1. \end{cases} \tag{28}$$

The shape of the function shown in Fig. 4 is very much like the saturation function used in a classical SMC and (27) is of the form (14) with $\overline{u} = U_{\mathrm{eq}}$. The FSMC is, therefore, very much like an SMC, and theorems that are developed for the latter have corresponding ones in an FSMC. This makes the performance and the stability analysis of FC possible. It should here be pointed out that the partition of the universe of discourse for both s and u need not be symmetrical, as shown in Fig. 4. The amount of overlap and the shape of the membership functions can be chosen at will (no need to say, the analytical expressions given in (28) will then not be valid).

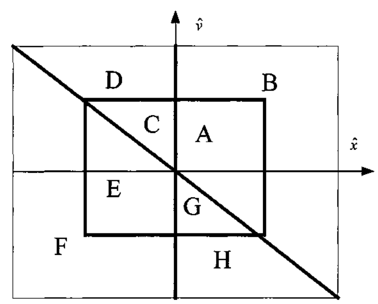

Fig. 5. Phase space partitioning in [93].

As is done with the normal FCs, the use of other computational intelligence methodologies for the tuning of the membership functions is also a possibility. In [27], two methods are used for this purpose, namely, Taguchi's method and GAs.

The similarity between an FC and an SMC is also addressed by Palm [54], [78] by pointing out that most FCs for nonlinear second-order systems are designed with a two-dimensional phase plane in mind. A fuzzy value for the control variable is determined with respect to fuzzy values of error and change of error. In the general approach to the control design, the phase plane is divided into two semiplanes by an effective switching line. Within the semiplanes, positive and negative control outputs are produced, the magnitude of which depends on the distance of the state vector from the switching line. The FC is, therefore, very much like an SMC and this explains why FCs are so successful, especially in the presence of disturbances and ill-defined knowledge about the system. By tracing the FC back to the principle of an SMC, evidence about the stability of the closed-loop system can be obtained. Some insight to the scaling factors for the crisp inputs and outputs can also be gained by the comparison of the FC with the SMC.

The similarity of sliding-mode systems and FCs is also addressed by Cao in [82], in which a fuzzy compensator scheme for a stick–slip friction is developed considering the effects of the fuzzy rules in the phase plane. The phase plane is divided into regions as shown in Fig. 5 where $\hat{x}$ is the normalized displacement value and $\hat{v}$ is the normalized velocity. Such a partitioning results in a switching line passing through the origin. Eight rules are used in the controller corresponding to the eight regions. Using the tuning parameters, the slope of this switching line is adjusted to obtain the desired dynamics.

Another direct approach seen in the literature is based on the so-called suction control method proposed in [98] as a remedy to the chattering problem. In this quasi-SMC, a boundary layer is designed into which the state vector is "sucked." Such a suction controller is proposed in [31] based on an FL controller (FLC). It is shown that, if the linguistic rules are properly designed, there exists a switching line in the FLC, and the slope of the switching line is determined by scaling factors. Using Lyapunov design, sufficient conditions for the fuzzy control system (FCS) to be globally asymptotically stable is derived. Based on these conditions, a design procedure, which guarantees that the FCS is globally asymptotically stable, is developed. It is shown that the FLC is similar to sliding-mode control and that a suction controller can be obtained through fuzzy systems. It is also shown that the rise time of the FCS can be adjusted by the scaling factors. In the simulations presented, the state vector is sucked toward the switching line and slide along it toward the desired state.

b) Rule bases on S and $\dot{S}$: In some approaches seen in the literature, rule bases constructed on variables S and $\dot{S}$, the derivative of $\mathbf{S}$, are employed to drive system states to a sliding manifold. In [85], Hwang and Lin propose such an approach for the design of an FSMC. The inputs of the proposed FC are fuzzified form of S and $\dot{S}$. The output of the FC is ΔU which is the fuzzified variable of Δu (change of u). All the universes of discourse of these variables are arranged from -1 to 1; thus, the range of nonfuzzy variables must be scaled to fit the universe of discourse of fuzzified variables with scaling factors. The selection of the scaling factors is not wholly subjective, since their magnitudes are a compromise between the speed of response and steady-state accuracy. Each fuzzy variable is quantized into seven qualitative fuzzy variables: 1) PB—Positive Big; 2) PM—Positive Medium; (3) PS—Positive Small; (4) ZE—Zero; (5) NS—Negative Small; (6) NM—Negative Medium; and (7) NB—Negative Big. For simplicity, triangular-type membership functions are chosen for the above-stated fuzzy variables. Fuzzy rules are designed in an attempt to satisfy $S\dot{S} < 0$. The resulting rule base is shown in Table II. A case study of an inverted pendulum system with successful results is presented to demonstrate the effectiveness of the scheme.

For large-scale systems composed of interconnected smaller systems, an important problem is the interactions between the subsystems. An approach to alleviate interaction problems is to use decentralized control methods based on local information. In [79], Yeh presents a systematic methodology to the design of a decentralized FLC for large-scale nonlinear systems. A performance index of sliding-mode control is used to design fuzzy control rules that use S and $\dot{S}$ as input. First, sliding variables S_i are defined for each of the subsystems. An iterative learning algorithm is devised to adjust control inputs. For the ith subsystems, the learning structure is

$$u_i^k = u_i^{k-1} + \delta u_i^k. \tag{29}$$

An expression for δu_i^k is obtained defining a performance index I_i as a function of S_i and $\dot{S}_i$ and minimizing this function using a gradient approach. The resulting form for δu_i^k is as follows:

$$\delta u(k) = \eta_i f_1(S_i, \dot{S}_i)\dot{S}_i + \eta_i f_2(S_i, \dot{S}_i)S_i \tag{30}$$

where η_i is a learning rate and f_1 and f_2 are highly nonlinear functions. To obtain still higher robustness, the expression above is computed via a fuzzy system. Seven triangular membership functions are defined on each of $S_i, \dot{S}_i$ and $\delta u_i(k)$. The output values of the 49 rules of the FLC are assigned using a fuzzy computation of this expression. The resulting rule base for a particular choice of the performance index is given in Table III.

This controller is tested in [79] via simulations on a double-inverted pendulum system and a two-link manipulator. Smaller residual error and robustness against nonlinear interactions are obtained.

c) Fuzzy system design via sliding surfaces consisting of nonisoclinal line segments: The shape of the sliding surface can be used as a guide to design FCs. Wang and Lin [97] follow such an approach and propose an FL control for a linear system for trajectory tracking in the phase plane. The system's state $(e, \dot{e})$ is forced to track the prespecified trajectory composed of several nonisoclinal segments in the phase plane (Fig. 6). Each segment corresponds to a relation between the tracking error $e$and the error change $\dot{e}$ in a particular region. The prespecified trajectory is regarded as a sliding surface. Trajectory tracking is completed region by region.

TABLE II
RULE BASE CONSTRUCTED ON VARIABLES S AND $\dot{S}$ [85]

	$\dot{S}$						
S	NB	NM	NS	ZE	PS	PM	PB
PB	ZE	PS	PM	PB	PB	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PS	NM	NS	ZE	PS	PM	PB	PB
ZE	NB	NM	NS	ZE	PS	PM	PB
NS	NB	NB	NM	NS	ZE	PS	PM
NM	NB	NB	NB	NM	NS	ZE	PS
NB	NB	NB	NB	NB	NM	NS	ZE

TABLE III
RULE BASE DERIVED BY THE MINIMIZATION OF A PERFORMANCE INDEX

$\dot{S}_i$	S_i						
	PB_3	PM_2	PS_1	ZO_0	NS_{-1}	NM_{-2}	NB_{-3}
PB_3	NB	NM	NM	NM	NS	PS	PM
PM_2	NB	NM	NS	NS	ZO	PS	PB
PS_1	NB	NM	NS	NS	PS	PM	PB
ZO_0	NB	NM	NS	ZO	PS	PM	PB
NS_{-1}	NB	NM	NS	PS	PS	PM	PB
NM_{-2}	NB	NS	ZO	PS	PS	PM	PB
NB_{-3}	NM	NS	PS	PM	PM	PM	PB

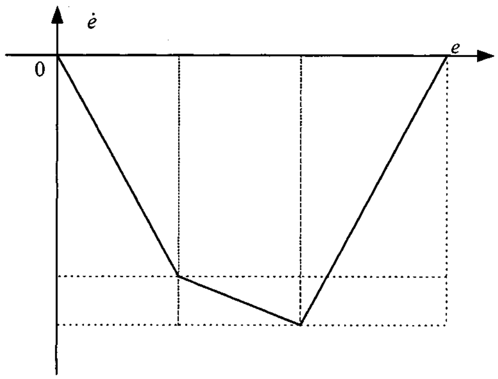

Fig. 6. Regions and sliding surfaces [97].

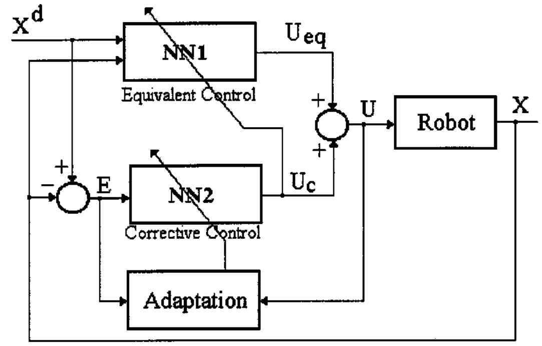

Fig. 7. The neuro-SMC architecture proposed by Ertugrul and Kaynak [41].

B. Integration of an NN with an SMC

The integration of an NN with an SMC can be classified into two main categories:

1) use of NN in feedback (or feedforward) control loop with a SMC, either in parallel or to compute the equivalent control part;
2) use of NN for online adaptation of the SMC parameters.

1) Use of NN in Parallel with SMC: Various schemes are seen in the literature that propose the use of NNs together with an SMC (or a variable-structure controller) either in parallel to avoid the chattering effect inside the boundary layer, or to compute the equivalent control part. Equivalent control computation requires exact knowledge of the system dynamics and parameters and, obviously, only an approximate value can be arrived at for partly known or uncertain systems. Computation of the equivalent control by NNs can be a good solution for this kind of system. In the literature, NNs are widely used, and successful results are obtained for computation of the dynamics, or inverse dynamics, of the systems [88].

a) Neuro-sliding-mode control architecture: In one approach seen in the literature (named "neuro-sliding-mode control") [41], two NNs in parallel are used to realize the equivalent control and the corrective control terms of the SMC (see Fig. 7). In [41], two similarities are pointed out. The first is that the equivalent control and the inverse dynamics have similar effects while the system is in sliding mode. The second similarity is between the corrective term of the SMC and a proposed neuro-controller structure. Based on the first one, a two-layer feedforward NN is proposed to compute the equivalent control and the weights are adapted to minimize the square of the corrective term. This adaptation is based on the fact that if the NN learns the equivalent control, the corrective control term goes to zero when the system is on the sliding surface, and any difference between the equivalent control and the NN output is reflected as a nonzero corrective control. The structure of the NN to compute the equivalent control is presented in Fig. 8. The inputs (designated as Z) to the net consist of desired and actual states

$$Z = [(X^d)^T \quad X^T]^T. \tag{31}$$

The net sum and the output of the hidden layer are designated as Ynet and Yout, respectively. Similarly, the net sum and the output of the output layer are designated as Unet and Ueq, respectively.

b) Use of a particular type of NN architecture to compute the corrective term of SMC: The NN structure presented in Fig. 9 directly computes the corrective term of SMC design if state errors are selected as inputs to the NN, G and K matrices are selected as the weights for the hidden and the output layers, and sliding function (S) and $h(S)$ are selected as the net sum and the output for the hidden layer [41]. The goal of showing this similarity is the use of NN weight adaptation techniques to adapt the gains of the SMC such that chattering is eliminated and the control signal is optimized. The described approach can, therefore, be classified as the use of the NN for online adaptation of the SMC parameters.

According to Ertugrul and Kaynak [41], the neuro-SMC algorithm has mainly the following advantages.

1) Learning is achieved online, i.e. learning and the derivation of the control signal are achieved simultaneously.
2) Error performance is improved.
3) Chattering is eliminated.
4) There is no need to compute the inertia (or inverse) matrix to compute the equivalent control.
5) The NN structure determination problem is solved for the corrective term. In other words, number of layers, number of neurons, and connections are well defined from SMC design.
6) The weights of NN2 do not have to be randomly initialized; the required performance from the SMCs allows the designer to fix the initial weights.
7) It is a robust neuro-controller where robustness comes up as a result of the SMC design.

The use of an NN for the calculation of the equivalent control term of an SMC is also proposed by Jezernik *et al.* [44]. In their approach, shown in Fig. 10, a VSC-type feedback controller [63] is used in parallel with an NN to obtain a robust control action. In this architecture, a feedforward NN with one hidden layer is used. The inputs to the net are selected as the sine of the desired angular positions, desired velocities, position errors, and velocity errors. A gradient-type learning algorithm is pro-

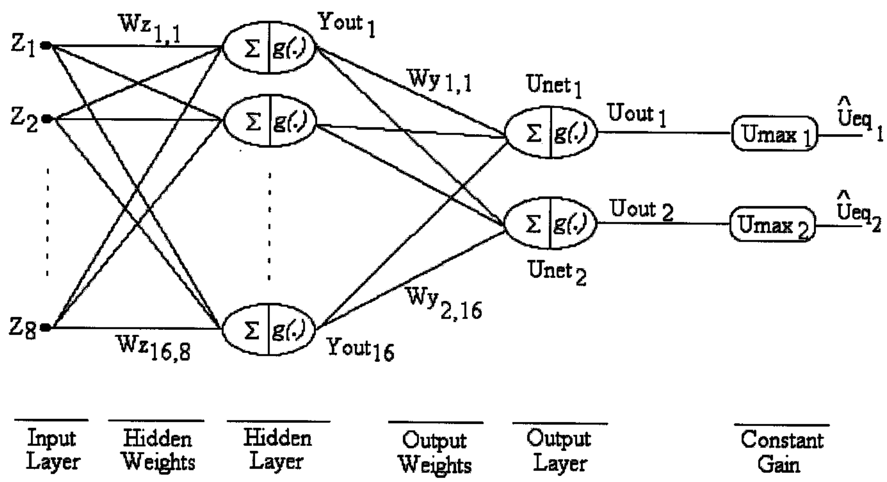

Fig. 8. NN1 structure for a 2-DOF robot to compute the equivalent control [41].

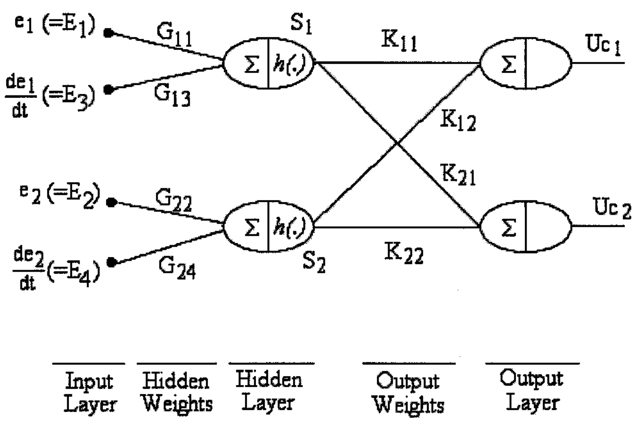

Fig. 9. NN2 structure for a 2-DOF robot to compute the corrective control [41].

posed to adapt the weights of the NN. The aim of the weight adaptation is to minimize the cost function

$$J = \frac{1}{2}(DS + \dot{S})^2. \tag{32}$$

According to Jezernik *et al.*, the main reason of this selection is that the Lyapunov design results as $DS + \dot{S} = 0$, and to assure the stability the cost in (32) should be minimized down to zero with the aid of the learning process of the NN. The algorithm is verified by experiments where an inverted pendulum with the additional mass-spring damper load is used.

c) Locally activated and stable neural-controller architecture: A somewhat different approach to the use of an NN in parallel with an SMC is that proposed by Kim and Oh for a class of nonlinear dynamic systems [68]. The basic architecture is as shown in Fig. 11.

The method employs a hybrid control architecture in which the NN approximates the plant nonlinearities using plant input–output data and tracking errors, while sliding-mode control used in parallel ensures uniform stability before the

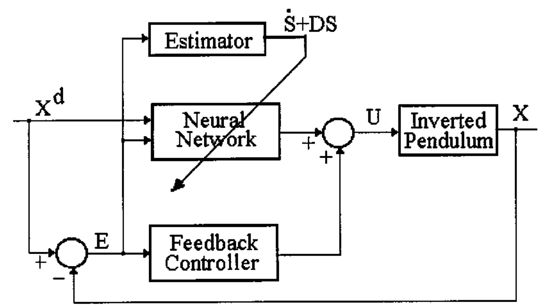

Fig. 10. Architecture proposed by Jezernik *et al.* [44].

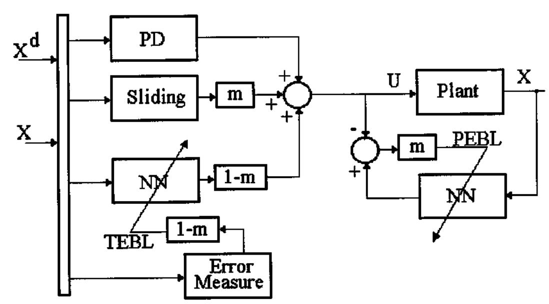

Fig. 11. Architecture proposed by Kim and Oh [68].

neurocontroller has had enough time to learn the system dynamics. A linear state feedback (PD) controller is also used to avoid chattering of the SMC controller inside the boundary layer. An activation scheme is determined based on the approximation error of the inverse dynamics as shown in Fig. 12. The radii of the circles are determined based on the stability and chattering elimination.

2) Use of NN for Adaptation of the SMC Parameters: In this section, a summary of the approaches seen in the literature on how the SMC parameters, such as the slope of the sliding surface (G) and the controller gain $(D$ or $K)$ are progressively updated, is presented. An example is the one proposed by Karakasoglu and Sundareshan [89] in which an NN is first trained to learn

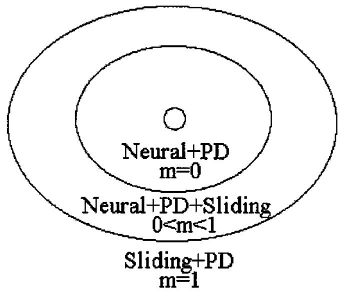

Fig. 12. Illustration of the activation scheme of the controllers based on the approximation error of the inverse dynamics [68].

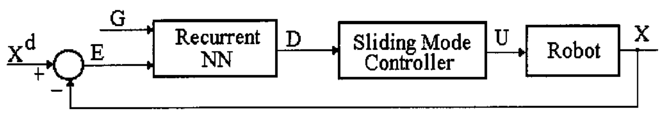

Fig. 13. Architecture proposed by Karakasoglu and Sundareshan [89].

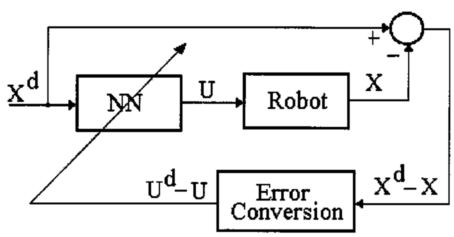

Fig. 14. Modified training architecture used in [89].

the inverse dynamics of the plant with the VSC, and then used as a cascade controller in the configuration shown in Fig. 13. It must be emphasized that, in this figure, the control computation is shown as a separate block in order to identify the output of the neural controller as the control gain D. A multilayer recurrent NN with a hidden layer is proposed to adjust the controller gains to keep the motion on the sliding manifolds.

Various architectures are described in the literature for teaching the inverse plant dynamics to an NN [91]. For example, in [89], the network is initially trained by a general learning architecture with a large range of input values and then the modified architecture presented in Fig. 14 is utilized for specialized training.

C. Use of VSS Theory to Introduce Robustness in NN Learning

In what is described above, NNs are used to improve the performance of an SMC. Conversely, the theories of the VSS and SMC can be used to improve the performance of NNs.

1) Sliding-Mode Strategy for Aadaptive Learning in Adalines: A dynamical sliding-mode control approach is proposed by Ramirez and Morles [93] for robust adaptive learning in analog adaptive linear elements (Adalines). The zero-level set of the learning error variable is regarded as a sliding surface in the space of learning parameters. Sliding-mode invariance conditions determine a least-squares characterization of the adaptive weights average dynamics whose stability features may be studied using standard time-varying linear systems results. The paper presents some simulation examples dealing with applications of the proposed algorithm to forward and inverse plant dynamics identification.

2) Sliding-Mode Algorithm for Training Multilayer ANNs: A new online learning algorithm based on sliding-mode control is proposed by Parma *et al.* in [94]. The approach is similar to that described by Ramirez, but the sliding surface is defined as in the classical way; a constant times the output error of NN is summed with the derivative of it; on the other hand, in the approach followed by Ramirez, the output error of the NN is used as a sliding surface. Another distinctive property of the proposed algorithm is that the algorithm is generic and can be applied to any configuration of multilayer NN; in other words, its potential application is not restricted to Adalines.

3) Stabilization and Robustification of Learning Process in Computational Intelligence: Noise rejection, handling the plant-model mismatches, and alleviation of structured or unstructured uncertainties constitute prime challenges that are frequently encountered in the practice of systems and control engineering. Particularly, in the training of computationally intelligent structures, the use of VSS theory can suitably alleviate the mentioned difficulties.

In parameter tuning of computationally intelligent structures the error backpropagation technique and the Levenberg–Marquardt optimization algorithm are very commonly used. However, the use of such approaches in noisy environments and under the existence of abruptly changing dynamics in the system under investigation requires special care. The reason for this is the fact that the mentioned effects may excite the high-frequency dynamics of the chosen training strategy, which is nonlinear, and the desired behavior can be observed only for slowly changing stimuli. For these reasons, the idea of incorporating a conventional approach with the VSS theory or the idea of designing a training strategy based directly on VSS theory can robustify the training mechanism and handle the problems arising through the impreciseness and noisy observations.

Several studies utilizing VSS theory in training of computationally intelligent structures are reported in the literature. The pioneering ones of these primarily discuss the use of an SMC in learning in NNs [101]–[103]. Particular interest has been directed toward the use of Gaussian networks, as they have a single hidden layer with mathematically tractable nonlinear activation functions. In what follows, more recent studies accounting for the training performance for a wide class of intelligent structures are briefly considered.

In [104], it is demonstrated that the objectives of the design can be achieved by defining two different cost functions, namely, one for measuring the learning performance and the other for measuring the stability in the parameter space. Based on these performance criteria, two different parameter tuning signals are derived. The ultimate form of the training information is constructed by mixing the two parameter tuning signals in a suitably weighted manner. The method proposed has been tested on robotic manipulators [104], [105]. It is seen that it performs well under adverse conditions, such as varying payload and a considerable amount of noise corrupting the state information to be used by the intelligent controller, whereas the performance of a pure error backpropagation technique is far from satisfactory.

Another study by Efe, Kaynak, and Yu focuses on the direct construction of the parameter tuning signal [106]. In this paper, a parameter tuning strategy is proposed and is shown to be stable in the sense of Lyapunov. Furthermore, the parametric evolution is proved to take place in a finite volume, which, more explicitly, corresponds to the parametric stability. In [106], the use of the algorithm for control engineering applications is discussed. Specifically, the analysis of sliding-mode learning inside the controller and the relevance of this with the sliding-mode control of the plant is discussed with analytical results.

In [107], another scheme based on an augmented switching manifold is discussed for the purpose of minimizing the realization error together with a suitable minimization in the magnitudes of the sensitivity derivatives of squared-error measure with respect to the adjustable parameters. In this approach, the parameter vector is driven toward a sliding manifold and the parameter drift problem of error backpropagation technique is alleviated. The modular representation of the proposed learning strategy allows the designer to achieve a soft transition between dynamic forms of Gauss–Newton method, Gradient Descent, and Levenberg–Marquardt technique.

D. Tuning of Sliding-Mode Parameters Using GAs

The integration of GAs and VSS control is of an indirect nature in that the former tunes the control parameters of the latter. A number of reports have appeared in the literature in this respect. For example, [58] describes the difficulties in SMC design and gives guidelines on GAs. In that paper, two practical and illustrative examples on the use of GAs in SMC construction are presented in detail.

An FSMC structure in which the antecedents are fuzzy sets on the sliding variable and the consequents are control outputs is considered in [46]. Two types of GA-based FSMC design methods are studied. In Type I, only the parameters in the THEN part are learned, while in Type II, all the parameters in both the IF part and the THEN part are considered. The Type-I design approach has a shorter string length and a smaller search space such that its convergence rate is faster than Type II. The disadvantage is that the IF part of the rules has to be defined heuristically. The Type-II approach presented has a longer string length and better optimization capability.

V. CONCLUSIONS

There is a growing amount of interest in the use soft-computing methodologies in SMCs. In some of these, the control structure is derived using the conventional design techniques and the computational intelligence is introduced in a complementary manner for a variety of purposes, ranging from fuzzy identification of the plant-model mismatches to the development of a global fuzzy model for the plant. Such applications are termed as being indirect in this survey paper. The more promising application approach, however, appears to be the direct one in which FL principles are used for the design of the sliding hyperplane. The theory of nonlinear systems, in general, and that of VSS in particular, can then be utilized for the performance and stability analysis of the FC. The theory of VSS can help the FC designer also on the scaling of the crisp inputs and outputs. The use of some adaptation or learning by an adaptive fuzzy system or an NN or by evolutionary computing is also a possibility.

More recently, an increasing amount of work is seen in the literature that proposes the use of VSS theory in parameter adaptation of computationally intelligent architectures, such as fuzzy inference systems (FISs), NN structures, or adaptive neuro-fuzzy inference systems (ANFIS), and so on. The simulation and experimental results obtained indicate that such an approach can be very powerful in alleviating the difficulties encountered in noisy or varying dynamical conditions.

REFERENCES

[1] S. V. Emelyanov, "Control of first order delay systems by means of an astatic controller and nonlinear correction," *Autom. Remote Control*, no. 8, pp. 983–991, 1959.

[2] V. I. Utkin, "Variable structure systems with sliding modes," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 212–222, 1977.

[3] K. D. Young, V. I. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE Trans. Contr. Syst. Technol.*, vol. 7, pp. 328–342, May 1999.

[4] J. Y. Huang, W. Gao, and J. C. Hung, "Variable structure control: A survey," *IEEE Trans. Ind. Electron.*, vol. 40, pp. 1–9, Feb. 1993.

[5] A. S. I. Zinober, *Variable Structure and Lyapunov Control*. Berlin, Germany: Springer-Verlag, 1994.

[6] K. K. Young, *Variable Structure Control for Robotics and Aerospace Systems*. Amsterdam, The Netherlands: Elsevier, 1993.

[7] V. I. Utkin, *Sliding Modes in Control Optimization*. Berlin, Germany: Springer-Verlag, 1992.

[8] K. D. Young, "Controller design for a manipulator using theory of variable structure systems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, no. 2, pp. 210–218, 1978.

[9] J. J. Slotine and S. S. Sastry, "Trackikng control of nonlinear systems using sliding surfaces with application to robot manipulators," *Int. J. Control*, vol. 38, pp. 465–492, 1983.

[10] H. Hashimoto, K. Maruyama, and F. Harashima, "A microprocessor based robot manipulator control with sliding mode," *IEEE Trans. Ind. Electron.*, vol. IE-34, pp. 11–18, Feb. 1987.

[11] S. W. Wijesoma, "Robust trajectory following of robots using computed torque structure with VSS," *Int. J. Control*, vol. 52, pp. 935–962, 1990.

[12] A. Denker and O. Kaynak, "Application of VSC in motion control systems," in *Variable Structure and Lyapunov Control*, A. S. I. Zinober, Ed. London, U.K.: Springer-Verlag, 1994, ch. 17, pp. 365–383.

[13] O. Kaynak, F. Harashima, and H. Hashimoto, "Variable structure systems theory, as applied to sub-time optimal position control with an invariant trajectory," *Trans. Inst. Elect. Eng. Jpn. E*, vol. 104, pp. 47–52, 1984.

[14] A. Bartoszewicz, "On the robustness of variable structure systems in the presence of measurement noise," in *Proc. IEEE IECON'99*. Aachen, Germany, Aug. 31–Sept. 4, 1988, pp. 1733–1736.

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

[16] H. Elmali and N. Olgac, "Robust output tracking control of nonlinear MIMO systems via sliding mode technique," *Automatica*, vol. 28, pp. 145–151, 1992.

[17] D. B. Izosimov and V. I. Utkin, "On equivalence of systems with large coefficients and systems with nonlinear control," *Autom. Remote Control*, vol. 11, pp. 189–191, 1981.

[18] I. Tunay and O. Kaynak, "Provident control of an electrohydraulic servo with experimental resultws," *Mechatron.*, vol. 6, no. 3, pp. 249–260, 1996.

[19] ——, "A new variable structure controller for affine nonlinear systems with nonmatching uncertainties," *Int. J. Control*, vol. 62, no. 4, pp. 917–939, 1995.

[20] M. Ertugrul, O. Kaynak, and A. Sabanovic, "A comparison of various VSS techniques on the control of automated guided vehicles," in *Proc. IEEE ISIE'95*, vol. 2, 1995, pp. 837–842.

[21] M. Ertugrul and O. Kaynak, "Neural network adaptive sliding mode control and its application to scara type robot manipulator," in *Proc. IEEE ICRA'97*, 1997, pp. 2932–2937.

[22] C. S. Chen and W. L. Chen, "Robust adaptive sliding-mode control using fuzzy modeling for an inverted-pendulum system," *IEEE Trans. Ind. Electron.*, vol. 45, pp. 297–306, Apr. 1998.

[23] X. H. Yu, Z. H. Man, and B. L. Wu, "Design of fuzzy sliding-mode control systems," *Fuzzy Sets Syst.*, vol. 95, no. 3, pp. 295–306, 1998.

[24] F. J. Lin and C. L. Chiu, "Adaptive fuzzy sliding-mode control for PM synchronous servo motor drives," *Proc. IEE—Control Theory Applicat.*, vol. 145, no. 1, pp. 63–72, 1998.

[25] S. C. Tong and T. Y. Chai, "Fuzzy indirect adaptive control for a class of decentralized nonlinear systems," *Int. J. Syst. Sci.*, vol. 29, no. 2, pp. 149–157, 1998.

[26] Z. H. Man and X. H. Yu, "Adaptive terminal sliding mode tracking control for rigid robotic manipulators with uncertain dynamics," *JSME Int. J. Ser. C—Mech. Syst. Mach. Elements Manuf.*, vol. 40, no. 3, pp. 493–502, 1997.

[27] C. L. Chen and M. H. Chang, "Optimal design of fuzzy sliding-mode control: A comparative study," *Fuzzy Sets Syst.*, vol. 93, no. 1, pp. 37–48, 1998.

[28] M. B. Ghalia, "Modeling and robust control of uncertain dynamical systems using fuzzy set theory," *Int. J. Control*, vol. 68, no. 6, pp. 1367–1395, 1997.

[29] S. C. Tong, T. Y. Chai, and Q. G. Li, "Fuzzy indirect adaptive control for a class of decentralized nonlinear systems," *Cybern. Syst.*, vol. 28, no. 8, pp. 653–673, 1997.

[30] T. P. Zhang and C. B. Feng, "Decentralized adaptive fuzzy control for large-scale nonlinear systems," *Fuzzy Sets Syst.*, vol. 92, no. 1, pp. 61–70, 1997.

[31] T. T. Lee, K. Y. Tu, and W. J. Wang, "Design of a fuzzy logic controller as a suction controller," *Fuzzy Sets Syst.*, vol. 91, no. 3, pp. 305–317, 1997.

[32] C. L. Hwang, M. H. Wei, and W. J. Jieng, "Non-circular cutting with a lathe using a three-stage intelligent controller," *Robot. Comput.-Integr. Manuf.*, vol. 13, no. 3, pp. 181–191, 1997.

[33] S. B. Choi and M. S. Kim, "New discrete-time, fuzzy-sliding-mode control with application to smart structures," *J. Guid. Control Dynam.*, vol. 20, no. 5, pp. 857–864, 1997.

[34] Q. P. Ha, "Robust sliding mode controller with fuzzy tuning," *Electron. Lett.*, vol. 32, no. 17, pp. 1626–1628, 1996.

[35] J. M. Carrasco, J. M. Quero, F. P. Ridao, M. A. Perales, and L. G. Franquelo, "Sliding mode control of a dc/dc PWM converter with PFC implemented by neural networks," *IEEE Trans. Circuits Syst. I*, vol. 44, pp. 743–749, Aug. 1997.

[36] A. TrebiOllennu and B. A. White, "Multiobjective fuzzy genetic algorithm optimization approach to nonlinear control system design," in *Proc. IEE—Control Theory Applicat.*, vol. 144, 1997, pp. 137–142.

[37] J. C. Chio, M. C. Hwang, S. D. Wu, and J. Y. Yang, "Robust attitude control of spacecraft using sliding mode control and productive networks," *Int. J. Syst. Sci.*, vol. 28, no. 5, pp. 435–446, 1997.

[38] C. L. Hwang and C. V. Hsu, "A thin and deep hole drilling using a fuzzy discrete sliding mode control with a woodpeckering strategy," in *Proc. Inst. Mech. Eng. I—J. Syst. Control Eng.*, vol. 209, 1995, pp. 281–292.

[39] T. J. Spooner and K. M. Passino, "Fault-tolerant control for automated highway systems," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 770–785, Aug. 1997.

[40] S. G. Tzafestas and P. A. Prokopiou, "Compensation of teleoperator modeling uncertainties with a sliding mode controller," *Robot. Comput.-Integr. Manuf.*, vol. 13, no. 1, pp. 9–20, 1997.

[41] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatron.*, vol. 10, no. 1–2, pp. 243–267, 2000.

[42] T. P. Zhang and C. B. Feng, "Fuzzy variable structure control via output feedback," *Int. J. Syst. Sci.*, vol. 28, no. 3, pp. 309–319, 1997.

[43] A. Sabanovic, K. Jezernik, K. Erbatur, and O. Kaynak, "Soft computing technics in discrete-time sliding mode control systems," *Automatika*, vol. 38, pp. 7–14, 1997.

[44] K. Jezernik, M. Rodic, R. Safaric, and B. Curk, "Neural network sliding mode robot control," *Robotica*, vol. 15, pp. 23–30, 1997.

[45] S. B. Choi and J. S. Kim, "A fuzzy-sliding mode controller for robust tracking of robotic manipulators," *Mechatron.*, vol. 7, no. 2, pp. 199–216, 1997.

[46] S. C. Lin and Y. Y. Chen, "Design of self-learning fuzzy sliding mode controllers based on genetic algorithms," *Fuzzy Sets Syst.*, vol. 86, no. 2, pp. 139–153, 1997.

[47] J. S. Glower and J. Munighan, "Designing fuzzy controllers from a variable structures standpoint," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 138–144, Feb. 1997.

[48] A. TreibOllennu, B. A. Stacey, and B. A. White, "Multivariable decoupling design of an ROV depth control system (a direct adaptive fuzzy SMC approach)," *Trans. ASME—J. Dynam. Syst., Meas., Control*, vol. 119, no. 1, pp. 89–94, 1997.

[49] C. L. Hwang, F. Y. Sung, and M. H. Wei, "Improved fuzzy sliding-mode control for a linear servo motor system," *Control Eng. Practice*, vol. 5, no. 2, pp. 219–227, 1997.

[50] Z. M. Yeh, "Adaptive multivariable fuzzy logic controller," *Fuzzy Sets Syst.*, vol. 86, no. 1, pp. 43–60, 1997.

[51] K. Erbatur, O. Kaynak, A. Sabanovic, and I. Rudas, "Fuzzy adaptive sliding mode control of a direct drive robot," *Robot. Auton. Syst.*, vol. 19, no. 2, pp. 215–227, 1996.

[52] J. C. Wu and T. S. Liu, "Fuzzy control stabilization with applications to motorcycle control," *IEEE Trans. Syst., Man, Cybern. B*, vol. 26, pp. 836–847, Dec. 1996.

[53] A. TrebiOllennu and B. A. White, "Non-linear robust control designs for a remotely operated underwater vehicle depth control system," *Proc. IME I—J. Syst. Control Eng.*, vol. 210, no. 3, pp. 201–214, 1996.

[54] R. Palm, "On the compatibility of fuzzy control and conventional control techniques," in *Fuzzy Logic: Implementation and Applications*, M. J. Patyra and D. M. Mlynek, Eds. New York: Wiley-Teubner, 1996, ch. 3, pp. 63–116.

[55] Q. P. Ha, "Sliding performance enhancement with fuzzy tuning," *Electron. Lett.*, vol. 33, no. 16, pp. 1421–1423, 1997.

[56] T. H. Liu and M. T. Lin, "A fuzzy sliding-mode controller design for a synchronous reluctance motor drive," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, pp. 1065–1076, July 1996.

[57] Y. Ohtani and T. Yoshimura, "Fuzzy control of a manipulator using the concept of sliding mode," *Int. J. Syst. Sci.*, vol. 27, no. 2, pp. 179–186, 1996.

[58] Y. Li, K. C. Ng, D. J. MurraySmith, G. J. Gray, and K. C. Sharman, "Genetic algorithm automated approach to the design of sliding mode control systems," *Int. J. Control*, vol. 63, no. 4, pp. 721–739, 1996.

[59] S. Y. Wang, C. M. Hong, C. C. Liu, and W. T. Yang, "Design of a static reactive power compensator using fuzzy sliding mode control," *Int. J. Control*, vol. 63, no. 2, pp. 393–413, 1996.

[60] J. C. Wu and T. S. Liu, "A sliding-mode approach to fuzzy control design," *IEEE Trans. Contr. Syst. Technol.*, vol. 4, pp. 141–151, Mar. 1996.

[61] F. Zhao and V. I. Utkin, "Adaptive simulation and control of variable-structure control-systems in sliding regimes," *Automatica*, vol. 32, pp. 1037–1042, 1996.

[62] C. S. Ting, T. H. S. Li, and F. C. Kung, "An approach to systematic design of the fuzzy control system," *Fuzzy Sets Syst.*, vol. 77, no. 2, pp. 151–166, 1996.

[63] A. Sabanovic, K. Jezernik, and M. Rodic, "Neural network application in sliding mode control systems," in *Proc. IEEE VSS'96*, 1996, pp. 143–147.

[64] C. L. P. Swamy, B. Singh, and B. P. Singh, "Fuzzy based sliding mode control of permanent magnet brushless DC motor," *J. Inst. Electron. Telecommun. Eng.*, vol. 41, no. 4, pp. 245–252, 1995.

[65] S. W. Kim and J. J. Lee, "Design of a fuzzy controller with fuzzy sliding surface," *Fuzzy Sets Syst.*, vol. 71, pp. 359–367, 1995.

[66] T. Senjyu and K. Uezato, "Enhancement of transient stability of power-systems by using fuzzy controller," *J. Intell. Fuzzy Syst.*, vol. 3, no. 3, pp. 239–246, 1995.

[67] Z. M. Yeh, Y. S. Tarng, and C. Y. Nian, "A self-organizing neural fuzzy-logic controller for turning operations," *Int. J. Mach. Tools Manuf.*, vol. 353, no. 10, pp. 1363–1374, 1995.

[68] H. G. Kim and S. Y. Oh, "Locally activated neural networks and stable neural controller-design for nonlinear dynamic-systems," *Int. J. Neural Syst.*, vol. 6, no. 1, pp. 91–106, 1995.

[69] C. S. Ting, T. H. Li, and F. C. Kung, "Design of fuzzy controller for active suspension system," *Mechatron.*, vol. 5, no. 4, pp. 365–383, 1995.

[70] J. Y. Chen, "Fuzzy sliding mode controller design: Indirect adaptive approach," *Cybern. Syst.*, vol. 30, no. 1, pp. 9–27, 1999.

[71] M. C. Shih and C. S. Lu, "Fuzzy sliding mode position control of a ball screw driven by pneumatic servomotor," *Mechatron.*, vol. 5, no. 4, pp. 421–431, 1995.

[72] T. Senjyu, N. Gibo, and K. Uezato, "Cooperative fuzzy control of avr and gov based on sliding mode to improve transient stability of power-systems," *Trans. Inst. Elect. Eng. Jpn.*, vol. 115, no. 2, pp. 78–88, 1995.

[73] Y. R. Hwang and M. Tomizuka, "Fuzzy smoothing algorithms for variable structure systems," *IEEE Trans. Fuzzy Syst.*, vol. 2, pp. 277–284, Nov. 1994.

[74] D. P. Filev and R. R. Yager, "On the analysis of fuzzy-logic controllers," *Fuzzy Sets Syst.*, vol. 68, no. 1, pp. 39–66, 1994.

[75] L. K. Wong, F. H. F. Leung, and P. Tam, "Lyapunov-function-based design of fuzzy logic controllers and its application on combining controllers," *IEEE Trans. Ind. Electron.*, vol. 45, pp. 502–509, June 1998.
[76] P. K. Dash, A. C. Liew, and A. Routray, "High-performance controllers for HVDC transmission links," *Proc. IEE—Generation, Transmission Distrib.*, vol. 141, no. 5, pp. 422–428, 1994.
[77] Y. Lu and J. S. Chen, "A self-organizing fuzzy sliding-mode controller-design for a class of nonlinear servo systems," *IEEE Trans. Ind. Electron.*, vol. 41, pp. 492–502, Oct. 1994.
[78] R. Palm, "Robust-control by fuzzy sliding mode," *Automatica*, vol. 30, no. 9, pp. 1429–1437, 1994.
[79] Z. M. Yeh, "A performance approach to fuzzy control design for nonlinear-systems," *Fuzzy Sets Syst.*, vol. 64, no. 3, pp. 339–352, 1994.
[80] H. Hellendoorn and R. Palm, "Fuzzy system technologies at Siemens research-and-development," *Fuzzy Sets Syst.*, vol. 63, no. 3, pp. 245–269, 1994.
[81] N. S. Kam, J. S. Kim, and W. S. Yoo, "Fuzy sliding-mode control of gasoline fuel-injection system with oxygen sensor," *JSME Int. J. Ser. C—Dynam. Control Robot. Des. Manuf.*, vol. 37, no. 1, pp. 100–106, 1994.
[82] C. T. Cao, "Fuzzy compensator for stick-slip friction," *Mechatron.*, vol. 3, no. 6, pp. 783–794, 1993.
[83] A. Ishigame, T. Furukawa, S. Kawamoto, and T. Taniguchi, "Sliding mode controller-design based on fuzzy inference for nonlinear-systems," *IEEE Trans. Ind. Electron.*, vol. 40, pp. 64–70, Feb. 1993.
[84] I. Takahashi and S. Itoh, "A simple positioning servo system by breaking control," *IEEE Trans. Ind. Applicat.*, vol. 28, pp. 1314–1321, Nov./Dec. 1992.
[85] G. C. Hwang and S. C. Lin, "A stability approach to fuzzy control design for nonlinear-systems," *Fuzzy Sets Syst.*, vol. 48, no. 3, pp. 279–287, 1992.
[86] T. Fukuda and K. Shimojima, "Intelligent robotic systems based on soft computing—Adaptation, learning and evolution," in *Computational Intelligence–Soft Computing and Fuzzy-Neuro Integration with Applications*. Berlin, Germany: Springer-Verlag, 1998, pp. 458–489.
[87] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. of Michigan Press, 1975.
[88] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*, Singapore: World Scientific, 1998.
[89] A. Karakasoglu and M. K. Sundareshan, "A recurrent neural network-based adaptive variable structure model-following control of robotic manipulators," *Automatica*, vol. 31, no. 10, pp. 1495–1507, 1995.
[90] M. K. Sundareshan and C. Askew, "Neural network-assisted variable structure control scheme for control of a flexible manipulator arm," *Automatica*, vol. 33, no. 9, pp. 1699–1710, 1997.
[91] D. Psaltis, A. Sideris, and A. A. Yamamura, "A multilayered neural network controller," *IEEE Contr. Syst. Mag.*, vol. 8, pp. 17–21, Feb. 1988.
[92] F. Sun, Z. Sun, and P. Y. Woo, "Stable neural-network-based adaptive control for sampled-data nonlinear systems," *IEEE Trans. Neural Networks*, vol. 9, no. 5, pp. 956–968, 1998.
[93] H. S. Ramirez and E. C. Morles, "A sliding mode strategy for adaptive learning in adalines," *IEEE Trans. Circuits Syst. I*, vol. 42, pp. 1001–1012, Dec. 1995.
[94] G. G. Parma, B. R. Menezes, and A. P. Braga, "Sliding mode algorithm for training multilayer artificial neural networks," *Electron. Lett.*, vol. 34, no. 1, pp. 97–98, 1998.
[95] K. Erbatur, O. Kaynak, and I. Rudas, "Design automation of a fuzzy PID robot control algorithm via genetic computing," in *Proc. 2nd Asian Control Conf.*, vol. III, Seoul, Korea, July 22–25, 1997, pp. 275–278.
[96] B. Yoo and W. Ham, "Adaptive fuzzy sliding mode control of nonlinear system," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 315–321, May 1998.
[97] W.-J. Wang and H.-R. Lin, "Fuzzy control design for the trajectory tracking in phase plane," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 710–719, Sept. 1998.
[98] J. J. E. Slotine, "Sliding controller design for nonlinear systems," *Int. J. Control*, vol. 40, pp. 421–434, 1984.
[99] X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning," *IEEE Trans. Neural Networks*, vol. 3, no. 5, pp. 807–814, 1992.
[100] X. Wang, *Adaptive Fuzzy Systems and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.
[101] R. N. Sanner and J. J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, pp. 837–863, Nov. 1992.
[102] L. Hsu and J. A. Real, "Dual mode adaptive control using gaussian neural networks," in *Proc. 36th Conf. Decision and Control*, New Orleans, LA, 1997, pp. 4032–4037.
[103] ——, "Dual mode adaptive control," in *Proc. IFAC World Congr.*, vol. K, Beijing, China, 1999, pp. 333–337.
[104] M. O. Efe, O. Kaynak, and B. M. Wilamowski, "Stable training of computationally intelligent systems by using variable structure systems technique," *IEEE Trans. Ind. Electron.*, vol. 47, pp. 487–496, Apr. 2000.
[105] M. O. Efe and O. Kaynak, "Stabilizing and robustifying the learning mechanisms of artificial neural networks in control engineering applications," *Int. J. Intell. Syst.*, vol. 15, no. 5, pp. 365–388, 2000.
[106] M. O. Efe, O. Kaynak, and X. Yu, "Sliding mode control of a three degrees of freedom anthropoid robot by driving the controller parameters to an equivalent regime," *Trans. ASME—J. Dynam. Syst., Meas. Control*, vol. 122, no. 4, pp. 783–792, 2000.
[107] M. O. Efe, "Variable structure systems theory based training strategies for computationally intelligent systems," Ph.D. dissertation, Dep. Elect. Electron. Eng., Bogazici University, Istanbul, Turkey, 2000.

Okyay Kaynak (M'80–SM'90) received the B.Sc. (first class honors) and Ph.D. degrees in electronic and electrical engineering from the University of Birmingham, Birmingham, U.K., in 1969 and 1972, respectively.

After spending seven years in industry, in January 1979, he joined the Department of Electrical and Electronic Engineering, Bogazici University, Istanbul, Turkey, where he is presently a Full Professor. He served as the Chairman of the Computer Engineering Department for three years, as Chairman of the Electrical and Electronic Engineering Department for two years, and as the Director of the Biomedical Engineering Institute for one year. Currently, he is the holder of the UNESCO Chair on Mechatronics and Director of the Mechatronics Research and Application Center. He has hold long-term Visiting Professor/Scholar positions at various institutions in Japan, Germany, the U.S., and Singapore. His current research interests are in the fields of intelligent control and mechatronics. He has authored three books and edited three. He has also authored or coauthored almost 200 papers which have appeared in various journals and conference proceedings.

Dr. Kaynak is presently the President Elect of the IEEE Industrial Electronics Society and an Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.

Kemalettin Erbatur received the B.S. (honors) degree in electronic engineering and mathematics from Bogazici University, Istanbul, Turkey, the M.Sc. degree in control systems from Imperial College, London, U.K, and the Ph.D. degree from Bogazici University, in 1992, 1993, and 2000, respectively.

In 1993, he joined the Robotics and Automation Group, TUBITAK Marmara Research Centre, Gebze-Kocaeli, Turkey. During 1997–1998, he was the Leader of this group. In the period 1998–2000, he was an Expert Researcher in the Electronic Systems Group. In 2000, he joined the Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey. He is currently a Visiting Researcher in the New Industry Research Organization, Kobe, Japan. His primary research interest is intelligent motion control. He is the author of 25 papers which have been published in international conference proceedings and journals.

Meliksah Ertugrul received the B.Sc., M.Sc., and Ph.D. degrees in electronic engineering from Bogazici University, Istanbul, Turkey, in 1992, 1994, and 1999, respectively.

Since 1993, he has been a Senior Researcher in the Information Technologies Research Institute, TUBITAK Marmara Research Centre, Gebze-Kocaeli, Turkey, where has worked on robotics, intelligent systems, control theory and applications, visual software, and microprocessor-based hardware design. He has worked with several industrial companies, where he was involved in microprocessor-based electronic hardware design and the writing of related software in assembly language. He has held a Visiting Lecturer position in the Computer Engineering Department, S. Demirel University, Kazakhstan. He is the author of more than 33 publications in journals, books, and conference proceedings.