

Self-Evolving Data Cloud-based PID-like Controller for Nonlinear Uncertain Systems

Zhao-Xu Yang, Hai-Jun Rong, *Member, IEEE*, Pak Kin Wong, Plamen Angelov, *Fellow, IEEE*, Zhi-Xin Yang, *Member, IEEE*, and Hang Wang

Abstract—In this paper, a novel self-evolving data cloud-based PID-like controller (SEDCPID) is proposed for uncertain nonlinear systems. The proposed SEDCPID controller is constructed by using fuzzy rules with non-parametric data cloud-based antecedence and PID-like consequence. The antecedent data clouds adopt the relative data density to represent the fuzzy firing strength of input variables instead of the explicit design of the membership functions in the classical sense. The proposed SEDCPID controller has the advantages of evolving structure and adapting parameter concurrently in an online manner. The density and distance information of data clouds are proposed to achieve the addition and deletion of data clouds and also a stable recursive method is proposed to update the parameters of the PID-like sub-controllers for the fast convergence performance. Based on the Lyapunov stability theory, the stability of the proposed controller is proven and the proof shows the tracking errors converge to a small neighborhood. Numerical and experimental results illustrate the effectiveness of the proposed controller in handling the uncertain nonlinear dynamic systems.

Index Terms—Self-evolving, Data Clouds, PID, Stability

I. INTRODUCTION

THE intensive significant achievements have been concentrated on developing nonlinear control system via fuzzy systems [1]-[3], which are represented as local linear subsystems connected by IF-THEN rules. During the controller design, fuzzy systems are generally utilized as fuzzy approximators based on their function approximation capabilities to approximate either the control law [4] or unknown system dynamics [5]. A lot of research work [6],[7] have demonstrated that fuzzy control systems provide promising

Manuscript received July 31, 2019; revised November 26, 2019; accepted March 6, 2020. This work was supported in part by the National Science Foundation of China under Grant No.61976172. This work was also supported by the UM Macao Distinguished Visiting Scholar program and the research grant of the University of Macau under Grant No. MYRG2016-00212-FST and the Science and Technology Development Fund of Macao S.A.R (FDCT) under MoST-FDCT Joint Grant No. 015/2015/AMJ.(Corresponding author: Hai-Jun Rong)

Z.-X. Yang and H.-J. Rong are with the State Key Laboratory for Strength and Vibration of Mechanical Structures, Shaanxi Key Laboratory of Environment and Control for Flight Vehicle, School of Aerospace Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: yangzhx@xjtu.edu.cn; hjrong@mail.xjtu.edu.cn).

P. K. Wong and H. Wang are with the Department of Electromechanical Engineering, Faculty of Science and Technology, University of Macau, Macau.

P. Angelov is with the Data Science Group, School of Computing and Communications, Lancaster University, Lancaster LA1 4WA, UK.

Zhi-Xin Yang is with State Key Laboratory of Internet of Things for Smart City, Department of Electromechanical Engineering, Faculty of Science and Technology, University of Macau, Macao.

control performance for the problems which are hardly analyzed and formulated in an accurate mathematical framework due to non-stationary uncertainties. However, in most of the existing fuzzy control approaches, the fuzzy approximators usually assume a fixed fuzzy system structure and mainly adjust its parameters to achieve the learning goal [8],[9]. Nonlinear systems generally appear dynamic changes due to nonstationary environments and external disturbances, which results in drifts and shifts of system states or new operating conditions and modes. From a practical point of view, there is no guarantee that a fixed controller structure has a satisfactory performance in online applications when the environment or the object of the controller changes. Once the system dynamics is prone to rapid and abrupt variations, the fixed controller structure makes the control schemes hard to follow the unexpected variations, even suffer from control failure after the variation.

In order to rapidly capture unexpected dynamic variations, some studies have developed evolving fuzzy system (EFS) for online learning and controller modeling [10]-[12]. The main advantage of these evolving fuzzy rule based controllers is that the function approximation capability is performed by the structure and parameter learning concurrently which relies on the data stream from the system input-output and states. Therefore, the controller design is converted into an approximation problem in terms of variable structure and adaptive parameters [13]. A fuzzy model reference adaptive control approach with evolving antecedent part is proposed in [14],[15] which uses an eFuMo method for the fuzzy-model identification. A fuzzy model reference adaptive control with leakage terms in the adaptive law is presented in [16] to guarantee the global stability. A fuzzy controller of Takagi-Sugeno (T-S) type with gradually evolving structure is designed in [17] and the controller parameters are trained in a non-iterative, recursive way. In [18], an online self-evolving fuzzy controller with global learning capabilities is proposed by utilizing the proper addition of membership functions to reach a desired accuracy level. In the self-evolution control method in [19], the error surface is analyzed to determine which cluster/rule produces the worst performance and needs to be further split. An evolving granular fuzzy control approach is proposed in [20], where the structure and parameters of the process model and controller are adapted according to the information extracted from uncertain data streams. In [21], a self-evolving probabilistic fuzzy neural controller with asymmetric membership function is presented to handle vagueness, randomness, and time-varying uncertainties of the

servo drive system during the control process. In [22],[23], the self-learning fuzzy controllers with adaptive structure and parameter learning are proposed to solve the control problems of back-to-turn missiles and hypersonic vehicles. These existing evolving fuzzy control approaches can effectively solve the unknown and time varying nonlinear system control problems by dynamically recruiting and modifying the rules. However, in these approaches specified types of membership functions are required to calculate the firing strength representing the antecedences of the rules. It is known that the membership functions depend on some design parameters that have to be properly defined/adjusted for better control performance, which could increase the online controller design complexity.

The novel type of evolving fuzzy rule based controller with *Data Clouds* offers a new way to construct the fuzzy rules without predefining the parameters in the antecedence. Defining the ‘IF’ part of rules without explicit membership functions and any fuzzy operators, the antecedences of fuzzy rules are formed upon the data clouds, which are the sets of samples attracted around the focal points [24]. In this case, the antecedences of fuzzy rules are presented in a parameter-free manner, while the consequence still remains the same T-S type. Regarding as a multi-model technique for modeling and approximation, the fuzzy systems, especially with T-S type consequence, is to decompose the process into linear sub-models in a state space correlated by membership functions or the above data clouds. This provides the possibility for developing an accumulated PID controller instead of utilizing a constant or a linear combination of input variables as consequence.

Recently, considerable efforts have contributed to the problem of designing fuzzy PID controllers. A hybrid fuzzy-PID controller is proposed by combining a fuzzy logic controller and a PID controller under a switching condition [25]. Another hybrid fuzzy P+ID controller is constructed in [26] by replacing the proportional term in the conventional PID controller with an incremental fuzzy P controller to improve the transient response and adapt to load variations. In [27], a fuzzy self-tuning algorithm with a semi-global asymptotic stability proof is proposed to determine the proportional, integral, and derivative gains according to the actual states. In these fuzzy PID control schemes, the fuzzy systems mainly act as either an alternative controller, or PID controller components, or a PID parameter optimizer. Different from those studies, some data cloud based control schemes utilizing PID-like controllers as the consequence of rules have been proposed in recent years, along with the data clouds as the antecedence of rules. A fuzzy controller RECCo (Robust Evolving Cloud-based Controller) with normalized data space is presented in [28] which transforms the cloud-space into the space of the constant size. An improved adaptation law with absolute values in the starting phase is proposed in [29],[30], which speeds up convergence and reduces large transients when the initials are far away from the unknown parameters. These approaches offer the possibility of implementing subsets of PID-like control consequences such as PD [31] and PID-R [32]. The PID-like consequence in the above approaches can be regarded as a series of sub-PID controllers and then an accumulated PID-

like controller weighted by the firing strength of fuzzy rules is achieved accordingly. Based on non-parametric data cloud antecedence and PID-like consequence, these controllers have the advantages of real time evolvable structure and parameter adaption using online data streams. However, these control approaches still possess the following shortcomings: 1) fuzzy rules represented by data clouds only can be added without considering pruning of insignificant rules; 2) similar to most of the existing fuzzy control approaches [33], the dominating adaptation laws of the PID-like consequence parameters are gradient-based learning method [30],[34] and its improved method [35] with an inherent poor convergence performance caused by the learning rate; 3) a thorough stability analysis of the controlled system is lacked.

Although the stability proof presented in [36] is elaborated for the evolving fuzzy system based on data clouds, it is only applicable to the identification and prediction problems under the consideration of convergence of the adaptive parameters and the identification errors. In this case, the tracking error of the general controller exists the likelihood of divergence, i.e., it is difficult to ensure the controller stability. In this paper, a novel self-evolving data cloud-based PID-like control approach (SEDCPID) is proposed to solve the above drawbacks. First, the familiarity of data clouds represented by the normalization distance between local densities of two data clouds is introduced as the pruning criterion to remove some redundant data clouds and obtain a compact controller structure. Next, different from gradient-based learning algorithm, a recursion based algorithm with fast convergence performance is proposed to update the PID-like consequence parameters. Finally, the stability of the proposed controller is proven according to the Lyapunov theorem.

The rest of the paper is organized as follows. Section II introduces the considered dynamic nonlinear system. The proposed controller is described in Section III-A. The learning process of controller structure and parameters are given in Section III-B. Section IV illustrates the stability and convergence analysis. Section V shows the application results. Finally, the conclusions are made in Section VI.

II. PROBLEM FORMULATION

Without loss of generality, as in [37]-[40], we consider a class of n -order nonlinear system as follows,

$$\begin{aligned} \dot{x}_i &= x_{i+1}, i = 1, \dots, n-1 \\ \dot{x}_n &= F(\bar{x}_n, u) + \psi(\bar{x}_n) \end{aligned} \quad (1)$$

where $\bar{x}_n = [x_1, \dots, x_n]^T \in R^n$. $x_i \in R$ with $i = 1, \dots, n$, and $u \in R$ are system state variables and system control input, respectively. The system output is denoted as $y = x_1$. $F(\cdot) : R^{n+1} \rightarrow R$ with $F(\mathbf{0}, 0) = 0$ is unknown nonlinear function. $\psi(\cdot) : R^n \rightarrow R$ with $\psi_i(\mathbf{0}) = 0$ can be viewed as model uncertainties which include system parameter variations, external disturbance, manufacturing and assembly errors, measurement errors, et al. For instance, the friction is a kind of typical parameter variation. It is included in $\psi(\bar{x}_n)$. We can illustrate this by the inverted pendulum example. The angular acceleration of the pole in the inverted pendulum

system is driven by system actuating torque coupled with the friction torque. The friction torque is zero in ideal state. However, the loading condition of the moving contact of rotating components is difficult to obtain accurately when the friction interface suffers from complicated contact motion, which further causes the decline of modeling accuracy and control performance. It is hard to get the complicated friction model and thus it is defined in the unknown function $\psi(\bar{x}_n)$ to present the model uncertainties in this work, which is also one of major motivations of this work to overcome dependency of certain accurate model in control applications. By Euler's polygonal arc method, systems (1) can be expressed as,

$$\begin{aligned} x_{i,k+1} &= x_{i,k} + x_{i+1,k}, i = 1, \dots, n-1 \\ x_{n,k+1} &= \mathcal{F}(x_{n,k}, u_{k+1}) \end{aligned} \quad (2)$$

where $x_{j,k} \in R^p$ with $j = 1, \dots, n$, $u_k \in R^q$ and $y_k = x_{1,k} \in R^p$ denote state variables, control input and the system output, respectively. $\mathcal{F}(\cdot)$ is an unknown nonlinear function due to the modeling errors, external disturbance, modeling simplification and parameter variations. Subscript $k \in \mathbb{N}$ is the discrete time index.

Using the non-affine Nonlinear Auto Regressive Moving Average with eXogenous input (NARMAX) model [41],[42], the system in (1) can be written as follows

$$y_k = f(\mathbf{y}_{k-1}, \mathbf{u}_{k-1}, u_k) \quad (3)$$

where $\mathbf{y}_{k-1} = [y_{k-1}, \dots, y_{k-m}]$ with m past key outputs, and $\mathbf{u}_{k-1} = [u_{k-1}, \dots, u_{k-n}]$ with n past key inputs.

Remark 1: $x_{1,k}$ and $x_{2,k}$ in (1) generally represent the positional variable and velocity variable in kinematic systems, respectively. The other orders, by analogy, also have certain physical meaning, with $x_{3,k}$ denoting acceleration variable. The magnetic bearing systems and inverted pendulum system discussed in the study can be described by system (1). Many practical systems, for instance, automotive semi-active suspension system, belong to this form.

Assumption 1: System output y_k can be measured and its initial values y_0 are assumed to remain in a compact set Ω_{y_0} .

Assumption 2: There exists a positive constant ϵ such that $\epsilon < |\partial f / \partial u_k|$.

Defining the pre-arranged desired trajectory y_k^r as the first-order reference, we have the tracking error as $e_k = x_{1,k} - y_k^r$, corresponding to the position tracking error. Here, the virtual second-order reference v_k^r , which can generally be regarded as the velocity reference trajectory, is constructed by

$$v_k^r = -x_{1,k} + c(x_{1,k} - y_k^r) + y_{k+1}^r \quad (4)$$

where $c > 1$ is an appropriate gain. Defining the second-order tracking error as $\varepsilon_k = x_{2,k} - v_k^r$, we obtain

$$e_{k+1} = x_{1,k+1} - y_{k+1}^r = x_{1,k} + x_{2,k} - y_{k+1}^r = ce_k + \varepsilon_k \quad (5)$$

The virtual reference is determined to meet the stability requirements which is illustrated later.

Assumption 3: The ideal control input is bounded, with $|u_k| < P$, $P > 0$.

The control objective is to design control signal u_k to drive $x_{1,k}$, $x_{2,k}$ to follow y_k^r , v_k^r , respectively. If the stability of

the uncertain dynamic together with well-defined past key memorization ensures the existence of a control input u_k^* that can make the system follow any arbitrary y_k^r , such that

$$f(\mathbf{y}_{k-1}, \mathbf{u}_{k-1}, u_k^*) - y_k^r = 0 \quad (6)$$

Based on implicit function theorem, we have the following lemma to establish the ideal controller u_k^* .

Lemma 1 [43]: If the partial derivative $|\partial f / \partial u_k|$ is bounded below with positive lower bound which is stated in Assumption 2, there exists a unique and discrete function

$$u_k^* = G(\mathbf{y}_{k-1}, \mathbf{u}_{k-1}, y_k^r) \quad (7)$$

such that (6) holds.

By substituting (7) into itself and repeating the process until the superscript of variables becomes zero, the control input u_k^* in (7) can link to the sequences of system outputs and desired trajectories. If the system is able to track any reference, we can further express (7) with the preceding variables by

$$\chi_k = [e_{k-1}, \dots, e_{k-j}] \quad (8)$$

with j past key errors as

$$u_k^* = \mathcal{G}(\chi_k) \quad (9)$$

It is noteworthy that the tracking errors are acted as the characterization factor among the outputs and reference. Here $j \geq 2$ is an integer which could be chosen as large as possible. In this paper, j is set to 6 for all the cases.

(9) demonstrates that u_k^* is expressed as a NARMAX model with an appropriate j past tacking errors. Exact estimation of the controller $\mathcal{G}(\cdot)$ is the core goal of our work while the prior knowledge is sorely lacking under the demands of self-learning and model-free adaption. To achieve this, the control law (9) is implemented by the following self-evolving data cloud-based fuzzy system with PID-like consequence.

III. SELF-EVOLVING DATA CLOUD-BASED PID-LIKE CONTROLLER

A. Controller Structure

The proposed SEDCPID controller framework is depicted in Fig. 1, where the data clouds and PID-like controllers are used as the antecedence and the consequence, respectively. The proposed SEDCPID controller diagram consists of six submodules. It is expressed in the following form:

Rule i : IF $(\chi_k \sim \xi_k^i)$ THEN $(\hat{u}_k \text{ is } \hat{u}_k^i)$

where \sim denotes that the fuzzy membership is expressed linguistically as "is associated with", ξ_k^i is the focal point of the i th data cloud in the input space. The density includes the information of all previous data samples directly and exactly. Each data sample belongs to certain data cloud with a different degree of density, described as $\gamma_k^i \in [0, 1]$. $\hat{u}_k^i (i = 1, 2, \dots)$ represents the crisp consequence of the i th rule that can be a constant or a linear combination of input variables. The proposed controller is required to bring the process output to the desired reference value as soon as possible. The scheme of extracting knowledge from the data stream of the tacking errors implies a great potential for constructing the controller.

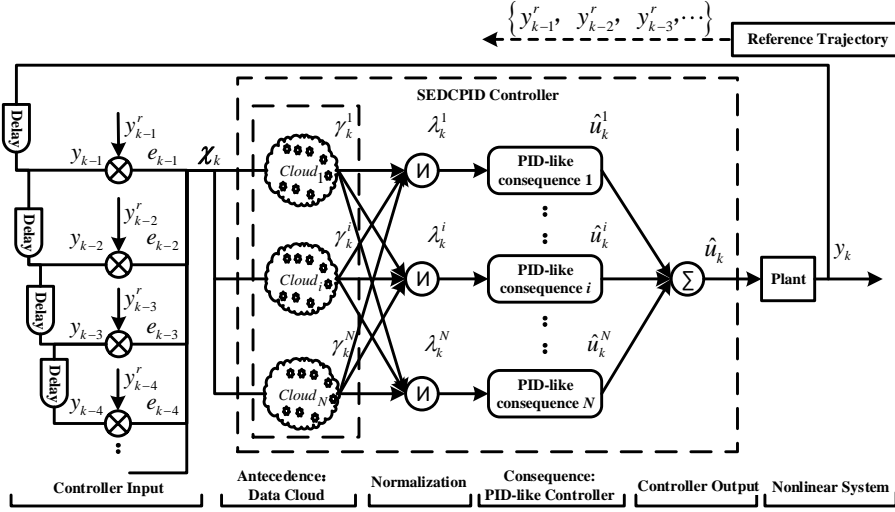


Fig. 1. Block diagram of the proposed controller

In our study, the PID-like consequence is utilized and denoted as

$$\hat{u}_k^i = P_k^i e_{k-1} + I_k^i \sum_{\kappa=k-j}^e e_{\kappa} + D_k^i \Delta_k^e + R_k^i \quad (10)$$

where P_k^i , I_k^i , D_k^i are the controller gains, R_k^i is the compensation of the operating point for each rule. The approach offers the possibility of implementing several subsets of PID-like controllers such as PI, PD, P, via setting either I_k^i , or D_k^i , or both of them as zero. $\sum_{\kappa=k-j}^e$ and Δ_k^e denote the discrete-time integral and derivative of the tracking error, which are described as follows,

$$\sum_{\kappa=k-j}^e e_{\kappa} = \sum_{\kappa=k-j}^{k-1} e_{\kappa} \quad (11)$$

$$\Delta_k^e = e_k - e_{k-1} \quad (12)$$

What we call PID-like consequence is that the integral term contains only a finite number j of tracking errors which is different from that of the conventional discrete PID controller with all past errors.

The antecedence of the controller is represented by a data cloud that describes a certain sub-set of the entire data set. Thus, the proposed approach replaces the scalar membership functions with a nonparametric function that is represented by the local densities. Considering Euclidean type of distance, in this paper, the local density of the i th data cloud is defined as follows [44]:

$$\gamma_k^i = \frac{1}{1 + \frac{\|\mathbf{X}_k - \Gamma_k^i\|^2}{\Xi_k^i - \|\Gamma_k^i\|^2}} \quad (13)$$

Here Γ_k^i and Ξ_k^i are the mean and scalar product of the data samples within the i th data cloud.

Assuming there are N data clouds, the normalization value of the local density for each data cloud is equal to

$$\lambda_k^i = \frac{\gamma_k^i}{\sum_{i=1}^N \gamma_k^i}, i = 1, 2, \dots, N \quad (14)$$

Substituting (11) and (12) into (10), we can get

$$\hat{u}_k^i = [1 \ e_{k-1} \ e_{k-2} \ e_{k-3} \ \dots \ e_{k-j}] \begin{bmatrix} R_k^i & P_k^i + I_k^i + D_k^i & I_k^i - D_k^i & I_k^i & \dots & I_k^i \end{bmatrix}^T \quad (15)$$

Then the controller output is obtained as follows:

$$\hat{u}_k = \sum_{i=1}^N \lambda_k^i \hat{u}_k^i, \hat{u}_k = \boldsymbol{\chi}_e^T \hat{\mathbf{q}}_k^i \quad (16)$$

where $\boldsymbol{\chi}_e = [1, \boldsymbol{\chi}_k^T]_{(j+1) \times 1}^T$ is extended input vector by appending the input vector $\boldsymbol{\chi}_k$ with 1. $\hat{\mathbf{q}}_k^i$ is the vector representing the consequent parameters of the k th rule and given by $\hat{\mathbf{q}}_k^i = [R_k^i \ P_k^i + I_k^i + D_k^i \ I_k^i - D_k^i \ I_k^i \ \dots \ I_k^i]_{(j+1) \times 1}^T$.

The output \hat{u}_k in (16) is further rewritten in a compact matrix form as,

$$\hat{u}_k = \mathbf{H}_k^T \hat{\mathbf{Q}}_k \quad (17)$$

where $\hat{\mathbf{Q}}_k$ is the consequent parameter vector of all N rules and \mathbf{H}_k is the inputs weighted by the normalized local density. They are reformulated as $\hat{\mathbf{Q}}_k = [\hat{\mathbf{q}}_k^1, \dots, \hat{\mathbf{q}}_k^N]_{(j+1)N \times 1}^T$ and $\mathbf{H}_k = [\boldsymbol{\chi}_e^T \lambda_k^1, \dots, \boldsymbol{\chi}_e^T \lambda_k^N]_{(j+1)N \times 1}^T$.

Remark 2: Different from the existing data cloud-based control approaches inspired by the PID-type control consequence [44] which suffer from the duplicated counting of integral and derivative of the tracking error at each executing step, the proposed SEDCPID with PID-like consequence is represented as finite historical tracking errors and corresponding variations of the three classical parameters in a concise form. This can be nearly expressed by a linear polynomial which can relatively reduce the computational complexity.

Remark 3: Although the final controller is rewritten in a compact matrix form of (17) that looks like a linear controller, the control system is nonlinear, in which the antecedent parameter is calculated via the concept of the nonlinear relative data density between the current data sample and the data cloud. The controller inherits the practicability and simplicity of the typical PID controller, and is further strengthened with the multiple parallel operating PID-like consequence. With the self-learning ability of the structure and parameters, the

significant changed system dynamics can be captured well by integrating new knowledge brought by the new system behavior and operating conditions. The better adaptability is achieved via the combination of structure adjustment and parameter updating instead of self-tuning PID controller with only parameter tuning mechanism. This can be explained from the viewpoint of machine learning that (9) constitutes a nonlinear function approximation problem. A typical approach for approximating complex nonlinear functions is to compose them out of basis functions in order to reduce complexity [45] as shown in (16). The same idea can be applied to generalize learning policies that a complicated policy could be learned from the combination of some simple (ideal, global and valid) policies. Indeed, similar ideas have been suggested in various fields of researches, for instance in behavior-based or reactive mobile robots [46].

B. Self-Evolving Learning Process

The self-evolving learning process of the proposed SEDCPID controller includes the mechanism to evolve the structure via forming the new data cloud and deleting redundant data clouds, and update the consequent parameters utilizing the least-square method. The former employs the global density and distance information as criteria to trigger whether a data cloud needs to be formed or deleted. Instead of the gradient-based learning method adopted generally in most of fuzzy control approaches, a modified recursive least square parameter update law is proposed to further guarantee the fast convergence.

1) *Formation of Data Clouds*: The proposed SEDCPID controller starts with zero data clouds and forms its data clouds from the online data stream. The first data sample is used to construct the first data cloud and assigned as its focal point ξ_1 that is the center of the data cloud. With more data samples coming, more data clouds are formed based on the following criteria. The first criterion is the global densities that are calculated at the focal points of the existing data clouds according to empirical data analysis [44]. The global density of the i th data cloud at the k th time instance is given by

$$\gamma_k^{i(G)} = \frac{1}{1 + \frac{\|\Gamma_k^i - \Gamma_k^G\|^2}{\Xi_k^G - \|\Gamma_k^G\|^2}} \quad (18)$$

where Γ_k^G and Ξ_k^G are the global mean and the global scalar product of the observed data samples. The global density $\gamma_k^{i(G)}$ is similar to the local density, but Γ_k^G and Ξ_k^G consider all the samples including χ_k at the current time instant, k and all the previously observed samples $\chi_j, j = 1, 2, \dots, k-1$.

When the new data sample comes, the global density produced by the new observation χ_k is given as,

$$\Upsilon_k = \frac{1}{1 + \frac{\|\chi_k - \Gamma_k^G\|^2}{\Xi_k^G - \|\Gamma_k^G\|^2}} \quad (19)$$

Then the global density of the new data sample is compared to the global densities of the existing data clouds to determine if a new data cloud needs to be formed, which is expressed as

$$(\Upsilon_k - \gamma_k^{i(G)} > 0) \text{OR} (\Upsilon_k - \gamma_k^{i(G)} < 0); \forall i \in 1, \dots, N \quad (20)$$

The second criterion is the distance information denoted as,

$$\zeta_{ki} > \rho_k^i; \forall i \in 1, \dots, N \quad (21)$$

This indicates that the new data χ_k is required to be sufficiently far from the focal points of the existing data clouds. Here ζ_{ki} denotes the distance between the current input data χ_k and the focal point ξ_i , that is $\|\chi_k - \xi_i\|$. ρ_k^i represents the radius describing the spread of the data cloud. It is updated as $\rho_k^i = \frac{1}{2}(\rho_{k-1}^i + \varrho_k^i)$, where $\rho_0^i = 1$, and ϱ_k^i represents the local scatter of the i th data cloud over the input data space at the k th time instance and is expressed as $\varrho_k^i = \sqrt{\Xi_k^i - \|\Gamma_k^i\|^2}$.

When the criteria (20) and (21) are both satisfied, a new data cloud is formed and the new data is assigned as the focal point, $\xi_{N+1} = \chi_k$. If they are not satisfied, the focal point of the nearest data cloud is updated by the new data, that is $\xi_f = \chi_k; f = \arg \min_{i=1}^N \|\chi_k - \xi_i\|$.

2) *Deletion of Data Clouds*: During learning, the distance between two local densities of data clouds may become low, which is denoted as

$$D_k^{ij} = \|\Gamma_k^i - \Gamma_k^j\| \quad (22)$$

In this case, the redundant rule need to be removed promptly which has a familiar summarization power with duplicate narration. The duplicated data cloud is generated according to the following condition

$$\text{cloud label} = \arg \min_{i=1}^N \min_{j=1, i \neq j}^N \|\Gamma_k^i - \Gamma_k^j\| \quad (23)$$

The normalization distance of the proposed duplicated data cloud is expressed as

$$d_k^i = D_k^{ij} / \sum_{j=1}^N D_k^{ij} \quad (24)$$

The i th data cloud is pruned if the normalization distance d_k^i falls below a threshold value d_e .

3) *Parameter Learning*: Based on the universal approximation property of fuzzy systems, optimal parameters \mathbf{Q}^* exist for approximating the control law by satisfying

$$u_k^* = \mathbf{H}_k^T \mathbf{Q}^* + v_k \quad (25)$$

Here v_k is the inherent approximation error. With the number of data clouds increasing, the inherent approximation error can be reduced arbitrarily. Therefore, based on Assumption 3, it is reasonable to assume that the inherent approximation error v_k is bounded with the constant δ , which is given by

$$|v_k| \leq \delta \quad (26)$$

According to (17) and (25), the controller approximation error becomes

$$\varpi_k = u_k^* - \hat{u}_k = \mathbf{H}_k^T \tilde{\mathbf{Q}}_k + v_k \quad (27)$$

where $\tilde{\mathbf{Q}}_k (= \mathbf{Q}^* - \hat{\mathbf{Q}}_k)$ is the parameter error. In our work, the error ϖ_k is represented with a linear combination of the tracking errors of each order, that is $\varpi_k = \sigma(cc_k + \varepsilon_k)$, where $\sigma > 0$ is chosen as an estimated proportional operator.

Without any antecedent parameters to be adjusted, the parameter optimization of this proposed controller is simply

equivalent to finding a least-squares solution of the consequent parameters $\hat{\mathbf{Q}}_k$ which can be denoted as a linear regression model. To ensure fast convergence and stability of the proposed controller, a stable recursive least squares algorithm is proposed to update the parameters ($\hat{\mathbf{Q}}_k$), which is defined as

$$\hat{\mathbf{Q}}_{k+1} = \hat{\mathbf{Q}}_k + \sigma \alpha_k \Sigma_k \mathbf{H}_k E_k \quad (28a)$$

$$\Sigma_{k+1} = \Sigma_k - \alpha_k \Sigma_k \mathbf{H}_k \mathbf{H}_k^T \Sigma_k \quad (28b)$$

where $E_k = ce_k + \varepsilon_k (= e_{k+1})$, and α_k is the time-varying learning rate defined as $\alpha_k = \frac{1}{\eta + \mathbf{H}_k^T \Sigma_k \mathbf{H}_k}$, $\eta \geq 1$.

When a data cloud is added, the consequent parameters $\hat{\mathbf{Q}}_k$ becomes $\hat{\mathbf{Q}}_k = [\hat{\mathbf{Q}}_{K-1}, \hat{\mathbf{q}}_{N+1}]^T$ with the parameters of the new data cloud determined by $\hat{\mathbf{q}}_{(N+1)i} = \mathbf{0}$, $i = 0, 1, 2, \dots, j$. The added diagonal elements of the covariance matrix Σ_k are $p_0 I_{(j+1) \times (j+1)}$, where p_0 is an initial value of the uncertainty assigned to the newly allocated rule.

Remark 4: Different from the existing work where only the first-order tracking error is applied, the proposed SEDCPID controller adopts the linear combination of the first-order and second-order tracking errors in the parameter update laws, which can ensure the good learning performance and also the stability of the controlled system to be described below. Besides, compared with the existing fuzzy control approaches, especially the existing data cloud based controllers with gradient-based learning methods, the proposed SEDCPID control approach utilizes a stable recursive least square parameter update method with fast convergence performance.

IV. STABILITY AND CONVERGENCE ANALYSIS

The stability of the proposed SEDCPID controller is investigated based on the Lyapunov stability theory with the consideration of the stable adaptation law. The uniform stability analysis result is summarized in the following theorem.

Theorem 1: Consider the SEDCPID controller described by (17) with the evolving data cloud as the antecedent and the PID-like control action as the consequence. The updating equations for the parameters $\hat{\mathbf{Q}}_k$ are described by (28a). Then, the uniform stability of the controller described by (17) is ensured. The error between the system output and the reference output converges to a small neighborhood in which the average identification error satisfies $\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^T e_k^2 \leq \frac{1}{\eta} (\delta/\tau)^2$. Here, τ is the lower bound of $\sqrt{\sigma^2 \alpha_k - 1}$, and δ is the upper bound of the uncertainty v_k satisfying $|v_k| \leq \delta$.

To prove the theorem, the following Lemma is required.

Lemma 2 [36]: Define $V(s_k) : R^n \rightarrow R \geq 0$ as a Lyapunov function. If there exists K_∞ functions $\delta_1(\cdot)$, $\delta_2(\cdot)$ and K function $\delta_3(\cdot)$, and for any $s_k \in R^n$, $\sigma \in R$ satisfies

$$\delta_1(\|s_k\|) \leq V_k(s) \leq \delta_2(\|s_k\|) \quad (29a)$$

$$V_{k+1} - V_k = \Delta V_k \leq -\delta_3(\|s_k\|) + \delta_3(\sigma) \quad (29b)$$

then the nonlinear system is uniform stable.

Proof of Theorem 1: Define the following Lyapunov candidate function,

$$V_k = \tilde{\mathbf{Q}}_k^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_k + e_k^2 \quad (30)$$

According to the matrix inversion lemma [47], one can get

$$\Sigma_{k+1}^{-1} = \Sigma_k^{-1} + \frac{1}{\eta} \mathbf{H}_k \mathbf{H}_k^T \quad (31)$$

Substituting (31) to (30), we get

$$\begin{aligned} V_{k+1} &= \tilde{\mathbf{Q}}_{k+1}^T \Sigma_{k+1}^{-1} \tilde{\mathbf{Q}}_{k+1} + e_{k+1}^2 \\ &= \tilde{\mathbf{Q}}_{k+1}^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_{k+1} + \frac{1}{\eta} (\tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k)^2 + (ce_k + \varepsilon_k)^2 \end{aligned} \quad (32)$$

From (28a), we can easily obtain

$$\tilde{\mathbf{Q}}_{k+1} = \tilde{\mathbf{Q}}_k - \sigma \alpha_k \Sigma_k \mathbf{H}_k E_k \quad (33)$$

Then, substituting (33) to (32), we get

$$\begin{aligned} V_{k+1} &= V_k + \sigma^2 \alpha_k^2 E_k^2 \mathbf{H}_k^T \Sigma_k \mathbf{H}_k - 2\sigma \alpha_k E_k \tilde{\mathbf{Q}}_k^T \mathbf{H}_k \\ &\quad + \frac{1}{\eta} (\tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k)^2 + E_k^2 - E_{k-1}^2 \end{aligned} \quad (34)$$

Due to $\tilde{\mathbf{Q}}_k = \tilde{\mathbf{Q}}_{k+1} + \sigma \alpha_k \Sigma_k \mathbf{H}_k E_k$ transformed by (33),

$$\begin{aligned} V_{k+1} &= V_k - \sigma^2 \alpha_k^2 E_k^2 \mathbf{H}_k^T \Sigma_k \mathbf{H}_k - 2\sigma \alpha_k E_k \tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k \\ &\quad + \frac{1}{\eta} (\tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k)^2 + E_k^2 - E_{k-1}^2 \end{aligned} \quad (35)$$

From (27) and (33), the following is obtained as

$$\mathbf{H}_k^T \tilde{\mathbf{Q}}_{k+1} + v_k = \varpi_k - \sigma \alpha_k \mathbf{H}_k^T \Sigma_k \mathbf{H}_k E_k = \sigma \eta \alpha_k E_k \quad (36)$$

Substituting (36) obtains

$$\begin{aligned} \Delta V_k &\leq -\sigma^2 \alpha_k^2 E_k^2 \mathbf{H}_k^T \Sigma_k \mathbf{H}_k + \frac{1}{\eta} v_k^2 - \sigma^2 \eta (\alpha_k E_k)^2 + E_k^2 \\ &\leq (1 - \sigma^2 \alpha_k) E_k^2 + \frac{1}{\eta} v_k^2 \end{aligned} \quad (37)$$

Then (37) becomes $\Delta V_k \leq -\psi_k E_k^2 + \frac{1}{\eta} \delta^2$, where $\psi_k = \sigma^2 \alpha_k - 1$. For the K_∞ functions $\delta_1(\cdot)$ and $\delta_2(\cdot)$

$$\begin{aligned} \delta_1(\cdot) &= N(j+1) \min(\|\tilde{\mathbf{Q}}_k\|^2) + e_k^2 \\ \delta_2(\cdot) &= N(j+1) \max(\|\tilde{\mathbf{Q}}_k\|^2) + e_k^2 \end{aligned} \quad (38)$$

the following inequality holds

$$N(j+1) \min(\|\tilde{\mathbf{Q}}_k\|^2) + e_k^2 \leq V_k \leq N(j+1) \max(\|\tilde{\mathbf{Q}}_k\|^2) + e_k^2$$

By choosing $\sigma \geq \sqrt{\eta + \mathbf{H}_k^T \Sigma_k \mathbf{H}_k}$, $\psi_k E_k^2$ is a K_∞ function. v_k^2 is a K function. According to Lemma 2, the uniform stability of the proposed controller is ensured. Summing up both sides of inequality from 1 up to T yields

$$\sum_{k=1}^T (\psi_k E_k^2 - \frac{1}{\eta} \delta^2) \leq V_1 - V_T \quad (39)$$

Since $V_T > 0$ is bounded, (39) is rewritten as,

$$\frac{1}{T} \sum_{k=1}^T \psi_k E_k^2 \leq \frac{1}{T} V_1 + \frac{1}{\eta} \delta^2 \quad (40)$$

As $T \rightarrow \infty$, we obtain

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^T \psi_k E_k^2 \leq \frac{1}{\eta} \delta^2 \quad (41)$$

Taking $\tau = \min(\psi_k)$, one can get

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^T E_k^2 \leq \frac{1}{\eta} (\delta/\tau)^2 \quad (42)$$

Since $E_k = e_{k+1}$, we can obtain that the average tracking error is bounded as follow,

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^T e_{k+1}^2 = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=1}^T E_k^2 \leq \frac{1}{\eta} (\delta/\tau)^2 \quad (43)$$

When the square of tracking error e_k^2 is bigger than the uncertainty $\frac{1}{\eta} (\delta/\tau)^2$, the parameter approximation error is discussed as follows.

According to (33), we have

$$\begin{aligned} & \tilde{\mathbf{Q}}_{k+1}^T \Sigma_{k+1}^{-1} \tilde{\mathbf{Q}}_{k+1} - \tilde{\mathbf{Q}}_k^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_k \\ &= -\sigma^2 \alpha_k^2 E_k^2 \mathbf{H}_k^T \Sigma_k \mathbf{H}_k - 2\sigma \alpha_k E_k \tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k \\ & \quad + \frac{1}{\eta} (\tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k)^2 \end{aligned} \quad (44)$$

and further obtain that

$$\begin{aligned} & \tilde{\mathbf{Q}}_{k+1}^T \Sigma_{k+1}^{-1} \tilde{\mathbf{Q}}_{k+1} - \tilde{\mathbf{Q}}_k^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_k \\ & \leq -2\sigma \alpha_k E_k \tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k + \frac{1}{\eta} (\tilde{\mathbf{Q}}_{k+1}^T \mathbf{H}_k)^2 \end{aligned} \quad (45)$$

Substituting (34) obtains

$$\begin{aligned} & \tilde{\mathbf{Q}}_{k+1}^T \Sigma_{k+1}^{-1} \tilde{\mathbf{Q}}_{k+1} - \tilde{\mathbf{Q}}_k^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_k \\ & \leq -2\sigma \alpha_k E_k (\sigma \eta \alpha_k E_k - v_k) + \frac{1}{\eta} (\sigma \eta \alpha_k E_k - v_k)^2 \\ & = -\eta (\sigma \alpha_k E_k)^2 + \frac{1}{\eta} v_k^2 \end{aligned} \quad (46)$$

By choosing $\sigma \leq \sqrt{\frac{1}{1-\alpha_k \eta} (\eta + \mathbf{H}_k^T \Sigma_k \mathbf{H}_k)}$

$$\begin{aligned} & -\eta (\sigma \alpha_k E_k)^2 + \frac{1}{\eta} v_k^2 \leq -\eta (\sigma \alpha_k)^2 \frac{1}{\eta} (\delta/\tau)^2 + \frac{1}{\eta} \delta_k^2 \\ & = \frac{1}{\eta} \delta_k^2 \left(1 - \frac{\eta (\sigma \alpha_k)^2}{\tau^2} \right) \leq \frac{1}{\eta} \delta_k^2 \left(1 - \frac{\eta (\sigma \alpha_k)^2}{\sigma^2 \alpha_k - 1} \right) \leq 0 \end{aligned} \quad (47)$$

Considering that $e_{k'+1}^2 \geq \frac{1}{\eta} (\delta/\tau)^2$ for $k' \in [1, k]$ is true, it gives

$$\tilde{\mathbf{Q}}_{k+1}^T \Sigma_{k+1}^{-1} \tilde{\mathbf{Q}}_{k+1} \leq \tilde{\mathbf{Q}}_k^T \Sigma_k^{-1} \tilde{\mathbf{Q}}_k \dots \leq \tilde{\mathbf{Q}}_1^T \Sigma_1^{-1} \tilde{\mathbf{Q}}_1 \quad (48)$$

From (43), the the average tracking error is bounded, and from (48), the parameter approximation error is bound, i.e., the proposed controller to train a fuzzy rule based system is uniform stable.

V. SIMULATION AND EXPERIMENTAL RESULTS

A. Example 1

Consider the control of the inverted pendulum system, x_1 and x_2 are denoted as the angular displacement and velocity of the pendulum, respectively. Due to the page limit, the detailed model description in [48] is not presented here. The uncertainty ($d = 0.2 \sin(4\pi t)$) is inserted in the system to verify the validity under uncertain dynamics. The proposed SEDCPID control approach is compared with other fuzzy

control methods based on the RECCo, decomposed fuzzy system (DFS), simplified DFS (SDFS) [48], and the traditional fuzzy system (FS), in which the same uncertainty dynamic is utilized. For the inverted pendulum, the control performance is usually tested by the set-point signal, square wave signal, and sinusoidal signal. It is noteworthy that the sinusoidal signal is valuable in engineering applications. As described in [49], the sinusoidal desired trajectory is used to present the stepping and walking motion of the inverted pendulum of humanoid robots. Therefore, the tracking trajectory is designed as $x_d = \sin(t)$ whose cyclic time is about 6.28s and thus the time of one cycle is $T_f = 6.28s$. The simulation time is 120s and there are about 19 adaption cycles. The simulation results from different control approaches are listed in Table I for comparison. In the table, $\|e_i\|_{L_2}^2 = \int_{(i-1)T}^{iT} |e|^2 dt$ is the L_2 -norm of the error vector at the i th adaption cycle in order to depict the learning speed and the tracking performance. $\|u_i\|_{L_2}^2 = \int_0^t |u|^2 d\tau$ is the L_2 -norm of control input u and represents the magnitude of the control input. Table I shows that the proposed SEDCPID controller achieves the smallest tracking error, root mean square error, and standard deviation of error. The tracking results of angular displacement and velocity of SEDCPID are illustrated in Fig. 2(a), Fig. 2(b) and the control inputs and achieved rule evolution are shown in Fig. 2(c) and Fig. 2(d).

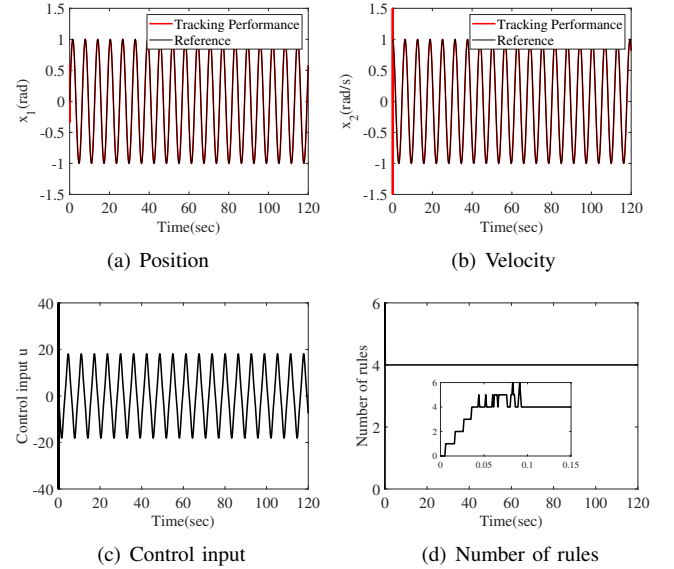


Fig. 2. Tracking results of inverted pendulum system of SEDCPID

TABLE I
PERFORMANCE COMPARISON FOR EXAMPLE 1.

Performance Index	FS	DFS	SDFS	RECCo	SEDCPID
Rule Number	25	81	27	16	10
Final $\ e_i\ _{L_2}^2 (\times 10^{-4})$	8.98	1.94	2.53	3.03	0.16
Cycle # when error $< 10^{-3}$	6	4	4	4	6
Final $\ u_i\ _{L_2}^2$	1464	723	733	818	971
Root mean square error ($\times 10^{-2}$)	7.55	2.13	2.52	0.13	0.052
Standard deviation of error ($\times 10^{-2}$)	2.68	0.76	0.21	0.13	0.051

B. Example 2

Consider the tracking control problem of a thrust active magnetic bearing system (TAMB), the control current, axial position and velocity of the rotor are denoted as u , z and \dot{z} , respectively. The considered TAMB system [50] is given by the following continuous form

$$\dot{z}(t) = a(\mathbf{z}; t)\dot{z}(t) + b(\mathbf{z}; t)z(t) + g(\mathbf{z}; t)u(t) + d(\mathbf{z}; t) \quad (49)$$

where $a(\mathbf{z}; t) = -(c(t)/m)$, $b(\mathbf{z}; t) = Q_z(t)/m$, $g(\mathbf{z}; t) = Q_i(t)/m$, $d(\mathbf{z}; t) = f_{dz}(t)/m$, $\mathbf{z} = [z \ \dot{z}]$. System disturbance setting as a combination of a small sinusoidal term $0.01 \sin(2t) + 0.01$ and a random value in the range of $[-0.001, 0.001]$ are considered. These parameters are time-varying during real applications. Also, the system parameter perturbations, $c(t) = c_0(1 + 0.3 \sin(2t))$, $Q_z(t) = Q_{z0}(1 + 0.2 \sin(2t))$, and $Q_i(t) = Q_{i0}(1 + 0.3 \cos(t))$ are considered, where the parameters of the TAMB system are set as $m = 7.35 \text{ kg}$, $c_0 = 3.5 \times 10^4$, $Q_{i0} = 212.82 \text{ N/A}$ and $Q_{z0} = 7.3 \times 10^5 \text{ N/m}$. Here a periodic sinusoidal command with two operating frequencies is applied as the reference tracking trajectory and given as

$$\begin{cases} z_d = 0 & t \leq 1 \\ z_d = 0.001 \sin(2\pi/5(t-1)) & 1 < t \leq 6 \\ z_d = 0.001 \sin(2\pi/2(t-6)) & 6 < t \leq 12 \end{cases} \quad (50)$$

In this example, a conventional PID controller and neural controllers based on the recurrent neural network (RNN) and hermite polynomial-based recurrent neural network (HPBRNN) [50] are used for comparison. The average tracking error and the standard deviation of the tracking error are utilized as the performance comparison indexes. Table II gives the comparison results. It can be seen that the proposed SEDCPID controller obtains better control performance as compared with the PID, RNN and HPBRNN, in which the same uncertainty and distance conditions are applied. Although the standard deviation and root mean square error of the proposed controller is not the lowest, the average of the tracking error achieves one order lower than the other controllers. Figs. 3(a)-3(d) show the simulation results achieved by the proposed SEDCPID controller in terms of position and velocity trajectory, control input and rule evolution. From these figures, one can observe that the proposed SEDCPID controller can achieve superior control performance by capturing different control conditions via the online self-evolving learning capability.

TABLE II
PERFORMANCE COMPARISON FOR EXAMPLE 2.

Tracking errors ($\times 10^{-5} \text{ m}$)	PID	RNN	HPBRNN	SEDCPID
Average	4.51	4.36	2.49	0.23
Standard deviation	5.29	5.00	2.99	4.90
Root mean square error	4.52	4.12	3.99	4.21

C. Example 3

To experimentally exam the merit of the SEDCPID controller, a practical sprung mass and unsprung mass system of a quarter car is considered [51], which is shown in Fig.4,

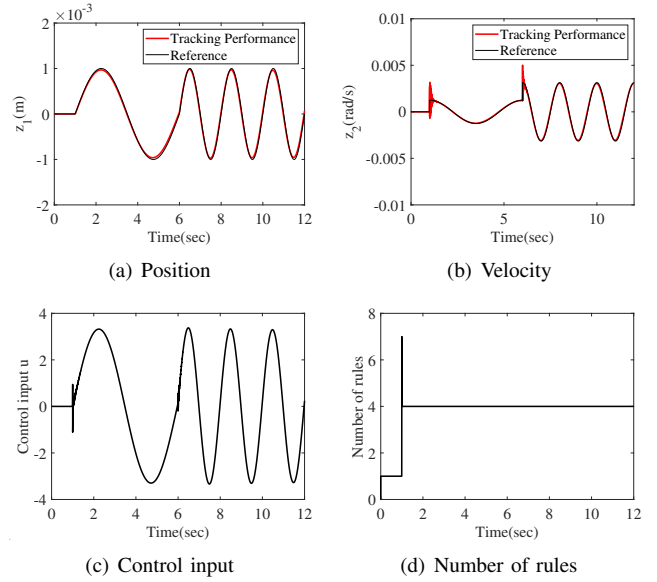


Fig. 3. Tracking results of TAMB of SEDCPID

where m_s and m_u denote the sprung mass and unsprung mass, respectively; k_s and c_s are the spring stiffness and initial damping coefficient of the damper, respectively; k_t stands for the tire stiffness; z_s , z_u and z_r represent the displacement of sprung mass, unsprung mass and road disturbance, respectively; and $u(t)$ is the control damping force. The system is known as the semi-active suspension system. The experimental setup consists of a quarter car test rig(QCTR) and a stepper motor driven hydraulic adjustable damper(SMDHAD). The damping force of the SMDHAD is adjusted by rotating an internal valve, so that the opening area of the oil passage inside the damper can be regulated. The rotation of the valve is achieved by the stepper motor which is driven by the controller. As for the QCTR part, a road excitation is generated via a LabVIEW program. The controller is also implemented in the LabVIEW program. Then the road excitation and the desired motor angle calculated in the computational part are entered into the QCTR and the nonlinear hydraulic adjustable damper(HAD) driven by a stepper motor to generate desired damping force, respectively. At the same time, the piston displacement and piston speed of HAD are measured by sensors and fed back to the computational part.

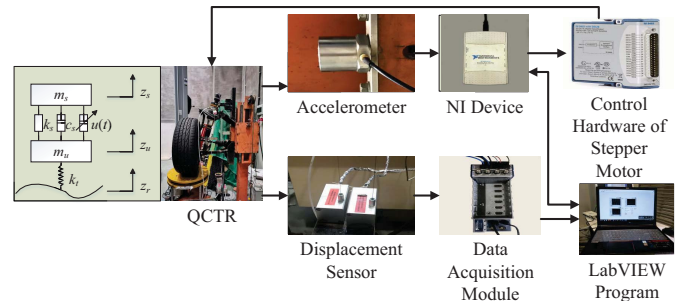


Fig. 4. Experimental setup of a semi-active suspension system

An online hardware-in-the-loop (HIL) test as shown in Fig.4

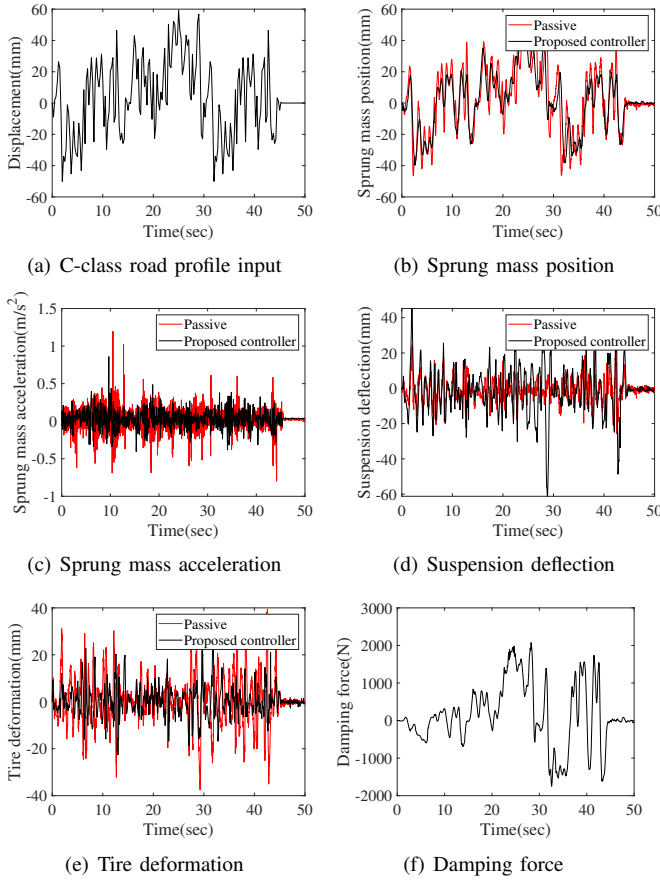


Fig. 5. Results of HIL test

TABLE III
COMPARISON OF PEAK VALUES IN HIL TEST

Evaluation indices	Passive Suspension	SEDCPID	Comparison
Sprung mass acceleration (m/s^2)	1.1961	0.8583	reduced by 28.15%
Suspension deflection (mm)	27.8939	61.2127	increased by 119.45%
Tire deformation (mm)	39.3169	29.4294	reduced by 25.15%

is carried out on the QCTR so as to test the proposed controller in a practical manner. The HIL test is implemented on an ISO profile for C-class road excitation which is commonly used criterion of evaluating the car, as shown in Fig.5(a). The tire suffers from a rough road with parametric uncertainty and external disturbances. The C-class road represents a poor road condition of soil and gravel with the time-varying operating points in the test process. Here the passive suspension is applied for comparison. Fig.5(b) shows the sprung mass position under the proposed SEDCPID control system and passive suspension, respectively. The vibration of the suspension is suppressed quickly under the bumps of the road surface. Figs.5(c)-5(e), Tables III and IV depict that both the sprung mass acceleration and tire deformation are decreased in the proposed control system compared with the passive suspension. Fig. 5(f) shows the control input in the operation process. The results illustrate that the ride comfort and road holding ability are improved by changing the damping force when passing the uneven road. In Fig.5(f), the damping force as control input is generated automatically along with the change

of the road profile. At the beginning of the testing process, the damping force is relative small, and then becomes more acute with higher amplitude since more bumps in the predefined road profile occur. When the road becomes smooth from 45s, the road displacement is zero, while the damping force is also zero. This means that the proposed controller is stable and the divergence of damping force has never happened. Therefore, the HIL test depicts that the proposed SEDCPID controller can be applied to the practical semi-active suspension with good vibration control performance.

TABLE IV
COMPARISON OF RMSE ERRORS IN HIL TEST

Evaluation indices	Passive	SEDCPID	Comparison
Sprung mass acceleration (m/s^2)	0.1398	0.1243	reduced by 11.13%
Suspension deflection (mm)	6.6357	12.9773	increased by 95.57%
Tire deformation (mm)	10.2834	6.4238	reduced by 37.53%

VI. CONCLUSION

In this paper, a novel self-evolving data cloud PID-like control framework is proposed to model the direct controller with the ability to self-adapt both structure and parameters online. By employing data clouds, the traditional way of defining membership functions for each input variable in an explicit manner is omitted and the learning process is entirely data-driven. The consequence in this case is given in the form of PID-like sub-controller. It has been shown that the proposed SEDCPID controller can work simultaneously together with the self-evolving algorithm without any prior information about the data clouds. The consequence parameters are updated online with a stable adaptation mechanism and the stability is proven through Lyapunov theorem which ensures the convergence of the whole system concurrently. The simulation results illustrate that good tracking performance can be achieved in the essence of external disturbance. The experimental result in a semi-active suspension system further verifies the feasibility in practical applications.

REFERENCES

- [1] N. Wang and M. J. Er, "Direct adaptive fuzzy tracking control of marine vehicles with fully unknown parametric dynamics and uncertainties," *IEEE Transactions on Control System Technology*, vol. 24, no. 5, pp. 1845–1852, 2016.
- [2] A. A. Kalat, "A robust direct adaptive fuzzy control for a class of uncertain nonlinear mimo systems," *Soft Computing*, 2018.
- [3] X. Jin, Z. Yu, G. Yin, and J. Wang, "Improving vehicle handling stability based on combined afs and dyc system via robust takagi-sugeno fuzzy control," *IEEE Transactions on Intelligent Transportation System*, vol. 19, no. 8, pp. 2696–2707, 2018.
- [4] Y. Wang, H. R. Karimi, H.-K. Lam, and H. Shen, "An improved result on exponential stabilization of sampled-data fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 6, pp. 3875–3883, 2018.
- [5] X. Zhao, X. Wang, G. Zong, and H. Li, "Fuzzy-approximation-based adaptive output-feedback control for uncertain nonsmooth nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 6, pp. 3847–3859, 2018.
- [6] Y.-J. Liu, M. Gong, S. Tong, C. L. P. Chen, and D.-J. Li, "Adaptive fuzzy output feedback control for a class of nonlinear systems with full state constraints," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 2607–2617, 2018.

- [7] Y. Yu, H.-K. Lam, and K. Y. Chan, "T-s fuzzy-model-based output feedback tracking control with control input saturation," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 6, pp. 3514–3523, 2018.
- [8] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.
- [9] W. Yu and X. Li, "Fuzzy identification using fuzzy neural networks with stable learning algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 3, pp. 411–420, 2004.
- [10] A. Kumar and V. Kumar, "Evolving an interval type-2 fuzzy PID controller for the redundant robotic manipulator," *Expert Systems With Applications*, vol. 73, pp. 161–177, 2017.
- [11] J. Li, Q. Xiong, K. Wang, X. Shi, and S. Liang, "A recurrent self-evolving fuzzy neural network predictive control for microwave drying process," *Drying Technology*, vol. 32, no. 12, pp. 1434–1444, 2016.
- [12] F. F. M. El-Sousy, "Adaptive hybrid control system using are current RBFN-based self-evolving fuzzy-neural-network for PMSM servo drives," *Applied Soft Computing*, vol. 21, pp. 509–532, 2014.
- [13] Y.-Y. Lin, J.-Y. Chang, N. R. Pal, and C.-T. Lin, "A mutually recurrent interval type-2 neural fuzzy system (MRIT2NFS) with self-evolving structure and parameters," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 3, pp. 492–509, 2013.
- [14] S. Blažič, D. Dovžan, and I. Škrjanc, "Robust evolving fuzzy adaptive control with input-domain clustering," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 5387–5392, 2014.
- [15] A. Zdešar, D. Dovžan, and I. Škrjanc, "Self-tuning of 2 dof control based on evolving fuzzy model," *Applied Soft Computing*, vol. 19, pp. 403–418, 2014.
- [16] S. Blažič, I. Škrjanc, and D. Matko, "A robust fuzzy adaptive law for evolving control systems," *Evolving Systems*, vol. 5, no. 1, pp. 3–10, Mar. 2014.
- [17] P. Angelov, "A fuzzy controller with evolving structure," *Information Sciences*, vol. 161, no. 1, pp. 21–35, 2004.
- [18] A. B. Cara, H. Pomares, I. Rojas, Z. Lendek, and R. Babuška, "Online self-evolving fuzzy controller with global learning capabilities," *Evolving Systems*, vol. 1, no. 4, pp. 225–239, 2010.
- [19] A. Cara, L. Herrera, H. Pomares, and I. Rojas, "New online self-evolving neuro fuzzy controller based on the TaSe-NF model," *Information Sciences*, vol. 220, pp. 226–243, 2013.
- [20] L. Daniel, P. Reinaldo, C. V. Actor, and G. Fernando, "Evolving granular fuzzy model-based control of nonlinear dynamic systems," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 4, pp. 923–938, 2015.
- [21] S.-Y. Chen and T.-S. Liu, "Intelligent tracking control of a pmlsm using self-evolving probabilistic fuzzy neural network," *IET Electric Power Applications*, vol. 11, no. 6, pp. 1043–1054, 2017.
- [22] H.-J. Rong, Z.-X. Yang, P. K. Wong, and C. M. Vong, "Adaptive self-learning fuzzy autopilot design for uncertain bank-to-turn missiles," *Journal of Dynamic Systems, Measurement, and Control*, vol. 134, no. 4, pp. 041 002–041 002–12, 2017.
- [23] H.-J. Rong, Z.-X. Yang, P. K. Wong, C. M. Vong, and G.-S. Zhao, "Self-evolving fuzzy model-based controller with online structure and parameter learning for hypersonic vehicle," *Aerospace Science and Technology*, vol. 64, no. 4, pp. 1–15, 2017.
- [24] P. Angelov, I. Škrjanc, and S. Blažič, "Robust evolving cloud-based controller for a hydraulic plant," in *2013 IEEE Conference on Evolving and Adaptive Intelligent Systems*, pp. 1–8, 2013.
- [25] S. Cetin and A. V. Akkaya, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *Nonlinear Dynamics*, vol. 61, no. 3, pp. 465–476, 2010.
- [26] Q. Chen, W. Li, and G. Chen, "Fuzzy P+ID controller for a constant tension winch in a cable laying system," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 2924–2932, 2017.
- [27] J. L. Meza, V. S. nez, R. Soto, and M. A. Llama, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 6, pp. 2709–2717, 2012.
- [28] G. Andonovski, S. Blažič, P. Angelov, and I. Škrjanc, "Robust evolving cloud-based controller in normalized data space for heat-exchanger plant," in *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–7, 2015.
- [29] G. Andonovski, B. S. J. Costa, S. Blažič, and I. Škrjanc, "Robust evolving controller for simulated surge tank and for real two-tank plant," *at-Automatisierungstechnik*, vol. 6, no. 99, pp. 725–734, 2018.
- [30] G. Andonovski, P. Angelov, S. Blažič, and I. Škrjanc, "A practical implementation of robust evolving cloud-based controller with normalized data space for heat-exchanger plant," *Applied Soft Computing*, vol. 48, pp. 29–38, 2016.
- [31] B. Costa, I. Škrjanc, S. Blažič, and P. Angelov, "A practical implementation of self-evolving cloud-based control of a pilot plant," *Applied Soft Computing*, vol. 48, no. 2, pp. 29–38, 2016.
- [32] G. Andonovski, P. Angelov, I. Škrjanc, and S. Blažič, "Robust evolving cloud-based controller (RECCo)," in *2017 Evolving and Adaptive Intelligent Systems*, pp. 1–6, 2013.
- [33] S. Labiod and T. M. Guerra, "Adaptive fuzzy control of a class of SISO nonaffine nonlinear systems," *Fuzzy Sets and Systems*, vol. 158, no. 10, pp. 1126–1137, 2007.
- [34] I. Škrjanc, S. Blažič, and P. Angelov, "Robust evolving cloud-based PID control adjusted by gradient learning method," in *2014 IEEE Conference on Evolving and Adaptive Intelligent Systems*, pp. 1–8, 2014.
- [35] G. Andonovski, S. Blažič, P. Angelov, and I. Škrjanc, "Analysis of adaptation law of the robust evolving cloud-based controller," in *2015 IEEE International Conference on Evolving and Adaptive Intelligent Systems*, pp. 1–7, 2015.
- [36] H.-J. Rong, P. P. Angelov, X. Gu, and J.-M. Bai, "Stability of evolving fuzzy systems based on data clouds," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 2774–2784, 2018.
- [37] J. Huang, L. Dou, H. Fang, J. Chen, and Q. Yang, "Distributed backstepping-based adaptive fuzzy control of multiple high-order nonlinear dynamics," *Nonlinear Dynamics*, vol. 81, no. 1–2, pp. 63–75, 2015.
- [38] Y. Wu, X. Yu, and Z. Man, "Terminal sliding mode control design for uncertain dynamic systems," *Systems and Control Letters*, vol. 34, no. 5, pp. 281–287, 1998.
- [39] S. Li, H. Du, and X. Yu, "Discrete-time terminal sliding mode control systems based on euler's discretization," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 546–552, 2014.
- [40] Z. Galias and X. Yu., "Euler's discretization of single input sliding-mode control systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1726–1730, 2007.
- [41] S. S. Ge, J. Zhang, and T. H. Lee, "Adaptive MNN control for a class of non-affine NARMAX systems with disturbances," *Systems and Control Letters*, vol. 53, no. 1, pp. 1–12, 2004.
- [42] L. Guo, H. Wang, and A. P. Wang, "Optimal probability density function control for NARMAX stochastic systems," *Automatica*, vol. 44, no. 7, pp. 1904 – 1911, 2008.
- [43] S. S. Ge, C. C. Hang, T. H. Lee, and T. Zhang, *Stable Adaptive Neural Network Control*. Springer, Boston, MA, 2002.
- [44] P. Angelov, X. Gu, and D. Kangin, "Empirical data analytics," *International Journal of Intelligent Systems*, vol. 32, no. 12, pp. 1261–1284, 2017.
- [45] S. Schaal, *Dynamic Movement Primitives- Framework for Motor Control in Humans and Humanoid Robotics*, pp. 261–280. Tokyo: Springer Tokyo, 2006.
- [46] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal on Robotics and Automation*, vol. 2, no. 1, pp. 14–23, 1986.
- [47] N. J. Higham, *Accuracy and Stability of Numerical Algorithms. Second edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2002.
- [48] Y.-C. Hsueh, S.-F. Su, and M.-C. Chen, "Decomposed fuzzy systems and their application in direct adaptive fuzzy control," *IEEE Transactions on Cybernetics*, vol. 44, no. 10, pp. 1772–1783, 2014.
- [49] J. Morimoto, G. Endo, J. Nakanishi, and G. Cheng, "A biologically inspired biped locomotion strategy for humanoid robots: Modulation of sinusoidal patterns by a coupled oscillator model," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 185–191, 2008.
- [50] F.-J. Lin, S.-Y. Chen, and M.-S. Huang, "Tracking control of thrust active magnetic bearing system via hermite polynomial-based recurrent neural network," *IET Electric Power Applications*, vol. 4, no. 9, pp. 701–714, 2010.
- [51] X. Ma, P. K. Wong, and J. Zhao, "Practical multi-objective control for automotive semi-active suspension system with nonlinear hydraulic adjustable damper," *Mechanical Systems and Signal Processing*, vol. 117, pp. 667–688, 2019.



Zhao-Xu Yang received the B.Eng. degree in Mechanical Engineering and Automation in 2011, the M.Eng. degree in Spaceflight Engineering in 2013, and the Ph.D. degree in Aeronautical and Astronautical Science and Technology in 2018 from Xi'an Jiaotong University, Xi'an, China.

He is currently a Research Assistant in School of Aerospace Engineering, Xi'an Jiaotong University, from 2018. He visited Lancaster University as a Visiting Researcher from July to

November 2019. His research interests include neural networks, fuzzy systems, fault diagnosis and intelligent control.



Hai-Jun Rong (M'14) received the B.Eng. degree in precision instrument from Xi'an Technological University, Xi'an, China, in 2000, the M.Eng. degree in control theory and control engineering from Xi'an Jiaotong University, Xi'an, China, in 2003, and the Ph.D. degree in intelligent control from Nanyang Technological University, Singapore, in 2008.

From December 2006 to October 2008, she was a Research Associate and a Research Fellow in Nanyang Technological University. Since

then, she has been an Associate Professor in School of Aerospace Engineering, Xi'an Jiaotong University. She is an Associate Editor of the *Evolving Systems* journal (Springer). Her research interests include neural networks, fuzzy systems, pattern recognition, and intelligent control.



Pak Kin Wong received the Ph.D. degree in Mechanical Engineering from The Hong Kong Polytechnic University, Hong Kong, in 1997.

He is currently a Professor in the Department of Electromechanical Engineering, University of Macau. His research interests include automotive engineering, fluid transmission and control, artificial intelligence (AI), mechanical vibration and AI for medical diagnosis. He has published over 232 scientific papers in refereed journals, book chapters, and conference proceedings.



Plamen Angelov (M'99-SM'04-F'16) received the M.Eng. in electronics and automation from Technical University, Sofia, Bulgaria, in 1989, and the Ph.D. degree in optimal control and D.Sc. degree in informatics from the Bulgarian Academy of Science, Sofia, Bulgaria, in 1993 and 2015, respectively.

He holds a Personal Chair (full Professorship) in intelligent systems with the School of Computing and Communications, Lancaster University, Lancaster, U.K. He holds a wide portfolio of

research projects and leads the Data Science group at Lancaster.

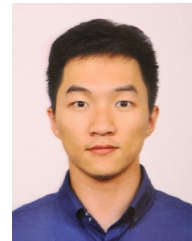
Dr. Angelov is the Vice President of the International Neural Networks Society and a member of the Board of Governors of the Systems, Man, and Cybernetics Society of the IEEE, a Distinguished Lecturer of IEEE. He is an Editor-in-Chief of the *Evolving Systems* journal (Springer) and an Associate Editor of the *IEEE Transactions on Fuzzy Systems* as well as of the *IEEE Transactions on Cybernetics* and several other journals. He received various awards and is internationally recognized pioneering results into on-line and evolving methodologies and algorithms for knowledge extraction in the form of human-intelligible fuzzy rule-based systems and autonomous machine learning.



Zhi-Xin Yang (M'14) obtained the B.Eng. in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, and the Ph.D. in industrial engineering and engineering management from the Hong Kong University of Science and Technology, Hong Kong, China.

He is currently an Associate Professor with the State Key Laboratory of Internet of Things for Smart City, the Department of Electromechanical Engineering, Faculty of Science and

Technology, and the Director of Research Service and Knowledge Transfer Office, University of Macau, Macau, China. His current research interests include fault diagnosis and prognosis, machine learning, and computer vision based robotic control.



Hang Wang received the M.Sc. degree in Automotive Mechatronics from Cranfield University, Cranfield, U.K., in 2018.

He is currently working toward the Ph.D. degree in Electromechanical Engineering, University of Macau. His research interests include robust control and fault-tolerant control for vehicle systems.