

Research Article

A Novel Online Self-Structuring Fuzzy Control Algorithm and Its Application

Siyuan Feng, Hong Bao, and Xuechao Duan

Key Laboratory of Electronic Equipment Structure Design of Ministry of Education, Xidian University, 2 Taibai Road, Xi'an 710071, China

Correspondence should be addressed to Hong Bao; hbao@xidian.edu.cn

Received 22 December 2014; Revised 16 March 2015; Accepted 10 April 2015

Academic Editor: Farhang Daneshmand

Copyright © 2015 Siyuan Feng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a novel self-structuring algorithm for the online adaptive fuzzy controller (SA-OAFC). The SA-OAFC capable of adding and deleting inference rules autonomously can start operating with an empty set of fuzzy rules based on the desired output and actual output of the system to avoid conventional differential operation. It also takes advantage of the auxiliary fuzzy system to evaluate the approximated results with little information of the plant. The SA-OAFC is characterized by its good engineering approachability, robustness for all kinds of perturbations of the plant, and the ability to perform variable selection among a group of candidate input variables. Moreover, it manages to remarkably reduce the amount of computation and decrease the complexity of the system. This paper demonstrates the capabilities of SA-OAFC by a simulation example and then hardware-in-the-loop (HIL) experiment.

1. Introduction

With the popularization of artificial intelligence in control field, the fuzzy control has been widely used in recent years [1–3]. However, according to thorough investigation of state of the art of current fuzzy control methods, most of the fuzzy controllers are of fixed structures, except that a few of them have real-time adaptive parameters [4–6]. Under these circumstances, one needs to not only define the rule base and membership functions (MFs) for these fuzzy controllers in advance but also redesign the whole system in case of the changes of the plant or the working condition.

Aiming at this shortcoming, some researchers have proposed the self-structuring fuzzy controller [7–12] where the MFs and rules of the fuzzy system can be increased and adjusted to improve the system structure based on some methods. Park et al. [7] used a method with a fixed width triangular MF, whose MFs and rules increase based on the extended input space. So the input space is divided evenly by the MFs, and there are the issues of the irrational distribution of MFs and the unlimited growth of rules. Cara et al. [11] used the auxiliary fuzzy system to determine the approximate effect of controllers and the positions where there is a need

to increase the MFs to achieve a reasonable distribution. But for a complex plant, the unlimited growth of rules still remains, and it is prone to oscillate while the new rules are initializing. To overcome this drawback, Chen et al. [12] adopted the “pseudo fuzzy output” method to determine the initial consequents of the new rules, successfully resolving the oscillation problem and according to the contribution of the rules, they deleted rules to limit the number of rules, but they only removed the redundant rules, ignoring the redundant MFs.

Based on the characteristics of the methods described above, this paper proposes a novel self-structuring algorithm for the online adaptive fuzzy controller (SA-OAFC). In the case of the mathematical model of the plant unknown, this controller starts working online with empty rules and expands the coverage of MFs dynamically depending on the desired output and the actual output of the system, which has the advantages of a small amount of computation, good robustness, and a reasonable distribution of MFs. The main contribution made in this work is designing the SA-OAFC for intelligent control system and testing and analyzing this algorithm by a simulation example and a HIL experiment under disturbance condition.

The rest of this paper is organized as follows. Section 2 presents the control problem to be solved and the structure of the fuzzy controller employed. In Section 3, the proposed SA-OAFC is described in detail. Then, Section 4 shows the simulation results illustrating the main features of the SA-OAFC. Finally, several conclusions are drawn in Section 5.

2. Problem Description

Without loss of generality, we consider the following dynamic plant [13]:

$$\begin{aligned} y(k+1) \\ = f(y(k), \dots, y(k-p), u(k), \dots, u(k-q)), \end{aligned} \quad (1)$$

where f is an unknown continuously differentiable function, $y(k)$ is the output of the plant at time k , u is the control input, and p and q are unknown constants related to the plant.

The purpose of the controller is to ensure that the output of the plant is able to track a desired signal $r(k)$. As for the general controllers, the structure shown in Figure 1 often requires the differential operation for taking the error e and the rate of error e' as the inputs. But in practical engineering applications, the differential signal is often susceptible to noise which might cause large errors and that is the reason why the proportional-integral (PI) control is applied more widespread than the proportional-derivative (PD) control. Thus, it is assumed that there exists a function F such that the control input u is given by

$$u(k) = F(\bar{x}(k)), \quad (2)$$

where $\bar{x}(k) = (r(k), y(k), \dots, y(k-p), u(k-1), \dots, u(k-q))$ is employed to avoid differential input signal and make the output of the plant in the next time satisfy $y(k+1) = r(k)$. To approximate the function F , this paper employs the Takagi-Sugeno (T-S) fuzzy system. The i th fuzzy rule is as follows [14]:

$$\begin{aligned} \text{Rule } i: \text{ IF } x_1 \text{ is } A_1^{i_1} \text{ AND } x_2 \text{ is } A_2^{i_2} \text{ AND} \\ \dots x_N \text{ is } A_N^{i_N}, \text{ THEN } y = \alpha_i, \end{aligned} \quad (3)$$

where $x = (x_1, x_2, \dots, x_N)^T$ and y are the crisp input and output of the fuzzy system, $A_v^{i_v} \in \{A_v^1, A_v^2, \dots, A_v^{n_v}\}$ are the MFs of the input x_v , n_v is the number of the MFs for that input, and α_i is the consequent of the i th rule. Thus, the output of the fuzzy controller is given by [14]

$$u(k) = \widehat{F}(\bar{x}(k); \Theta) = \frac{\sum_{i=1}^{N_{\text{rules}}} \alpha_i \cdot \mu_i(\bar{x}(k))}{\sum_{i=1}^{N_{\text{rules}}} \mu_i(\bar{x}(k))}, \quad (4)$$

where Θ represents the parameters of the fuzzy controller, N_{rules} is the number of the fuzzy rules, and μ_i is the activation degree of the i th rule.

3. Online Self-Structuring Fuzzy Controller

A novel online self-structuring fuzzy controller employing triangular MFs is designed to approximate the function F .

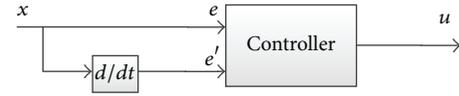


FIGURE 1: Structure of the common controller.

Its main operational processes are divided into the following three stages.

- Parameter adaption stage: in this stage, the controller expands the input range dynamically and adopts the reward or punishment policy to adjust the consequents of fuzzy rules based on the system error.
- Self-structuring stage: when the performance of the controller fails to be improved by the adjustment of the consequents, the proposed SA-OAFC will get this stage involved. Then there will be a new MF added to one of the input variables of the controller and a new corresponding rule will be generated to perfect the structure of the system.
- Deletion stage: in order to reduce the amount of computation, it is of necessity to delete the redundant rules and ineffective MFs according to the contribution rate of the rules.

The detailed description of each stage will be introduced in the following sections.

3.1. Parameter Adaption Stage. Since the characteristics of the plant are unknown, the inputs of the system are likely to exceed the coverage of the existing MFs with the operation of the controller. To solve this problem, we propose a method of expanding the input range dynamically. As shown in Figure 2, when the input $x_i(t)$ exceeds the coverage of the current MFs, the endpoint of the MF closest to $x_i(t)$ will be chosen as $x_i(t) + W$ to expand the existing coverage, but no MF or rule is added in at the moment. W is a predefined value and, in practice, W is chosen based on the designer's prediction of the lowest degree of nonlinearity.

Based on the reward or punishment policy, the consequent of the i th rule is modified according to the following expression [11]:

$$\Delta\alpha_i(k) = C \cdot \mu_i(k-1) \cdot (r(k-1) - y(k)), \quad (5)$$

where $\mu_i(k-1)$ and $r(k-1)$ are the activation degree of the i th rule and the desired output of the system at instant $k-1$, $y(k)$ is the current output of the system, and C is a constant with the same sign as the monotonicity of the plant with respect to u . Since the SA-OAFC is to give priority to the modification of the consequents, the value of C is of particular importance to affect the speed of adjustment directly. The greater the value of C , the shorter the adjustment time of the controller, but it is also more likely to cause oscillation at the same time. In [11], the absolute value of C can be set as $|C| = \Delta u / \Delta r$, where Δu is the drive range of the controller and Δr is the range in which the desired signal varies.

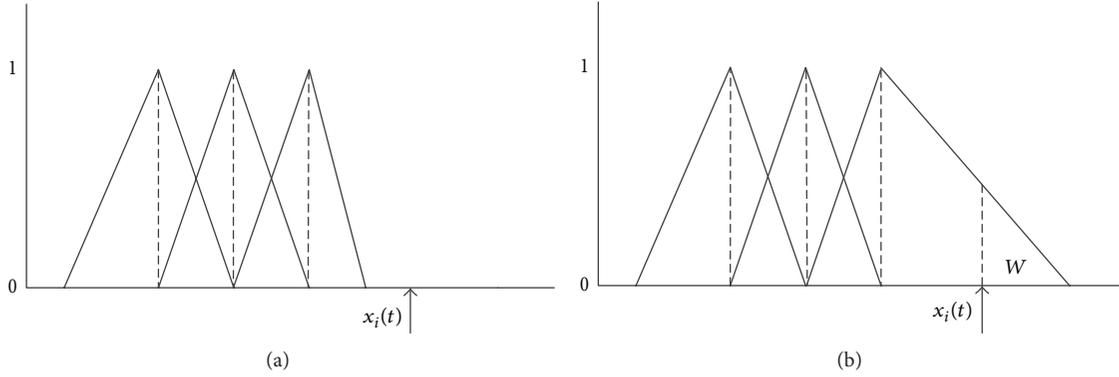


FIGURE 2: Extension of the input range.

3.2. Self-Structuring Stage. Once the modification of the parameters of rules no longer improves the quality of the control, we consider that it is of necessity to make a structural change. In this paper, the mean square error (MSE), given by (6), is used to judge whether the performance of the controller is improved or not

$$\text{MSE} = \frac{1}{N_s} \sum_{k=1}^{N_s} (r(k) - y(k+1))^2, \quad (6)$$

where N_s is the number of the data used to calculate the MSE. However, when the plant is already under control, the MSE stops improving, and the decreasing of the improvement rate of the MSE does not always mean that the controller should make a structural change. To avoid this situation, we define a critical threshold MSE_{\min} to represent the minimum value of the control error. When the MSE stops improving and its value is larger than MSE_{\min} , it is indicating that the system error no longer meets the accuracy requirements and needs adding in more MFs. Therefore, if the value of MSE_{\min} is too large, it will be difficult for the controller to make structural changes and will lead to the increase of the adaptive time for the system.

In [15, 16], each new set of input data will impact on the structure of the controller. For the same data, the different sequences will lead to different structures of the controller. To avoid this phenomenon, we store all the data in the memory M with equal weight in which the N_s data are made use of to judge the performance of the fuzzy controller so that the controller can be more robust. The selection of N_s depends on how variable the signal is. If the variation is relatively small, a smaller value will be selected, and on the contrary a larger value will be chosen.

Besides, it is of necessity to determine an input variable to add in a new MF at first, which has been a problem in the field of online control. Some researchers suggested adding in MFs for all the input variables [17], but it will cause a substantial growth in the number of rules. Gao and Er [18] proposed to add MFs in the input variable which corresponds to the maximum membership degree in the rule with the largest weight. The purpose of this approach is essentially to find out the input variable which causes the system error mostly, but it will make the controller become sensitive to noise and

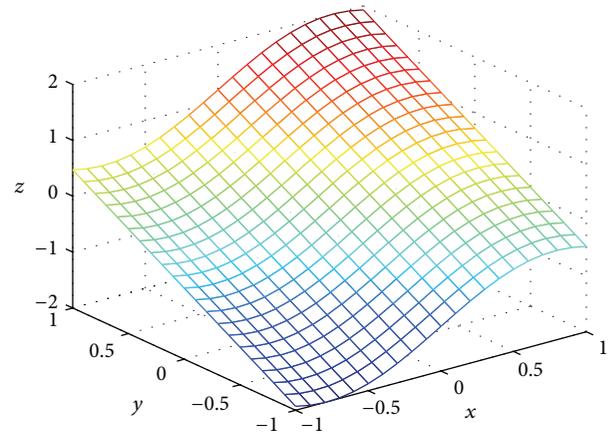


FIGURE 3: Surface of the plant function.

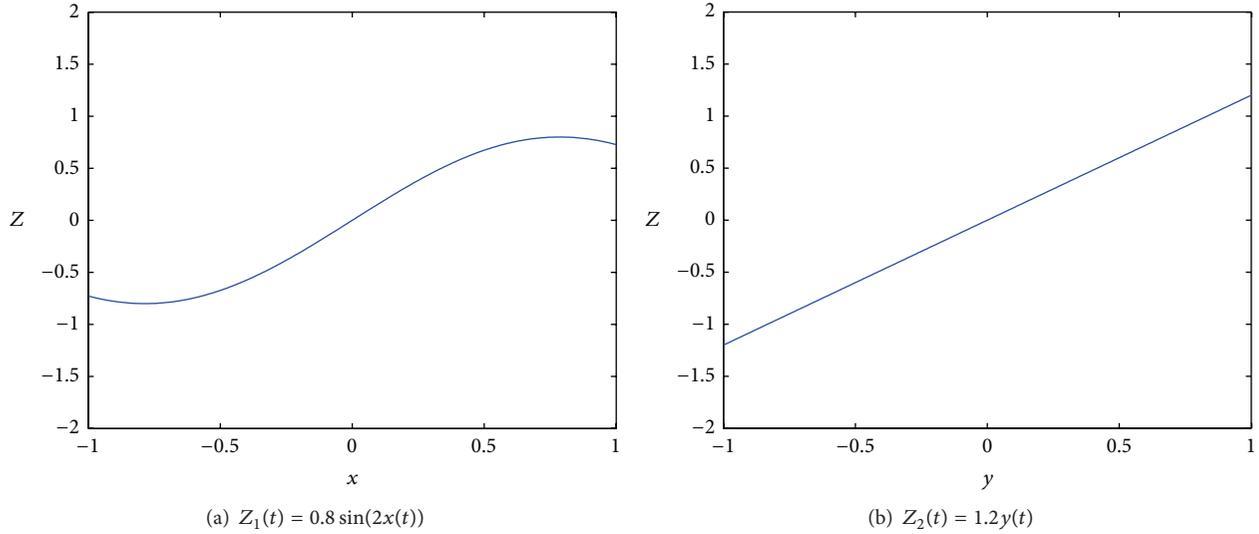
reduce the robustness. This paper utilizes the auxiliary fuzzy system [11] to evaluate the situation on the approximation of the input variables of the fuzzy controller and to find out the one with the worst approximation results to add in a new MF.

To illustrate the principle of the auxiliary fuzzy system, a plant with mathematical model in [13] is taken as an example

$$z(t) = 0.8 \sin(2x(t)) + 1.2y(t). \quad (7)$$

The fuzzy system is used to approximate this plant which is a double-input single-output system, and the surface of the function is shown in Figure 3.

When the input x is going to be approximated, the range of the input y will be divided into infinitesimal intervals with width dy , and each of them will be taken into account individually. Therefore, the situation in each of the interval dy is equivalent to make a fuzzy approximation on function $Z_1(t)$ in the one-dimensional input space, as shown in Figure 4(a). Then we synthesize the effect of approximations in each dy to get the full input x approximation. Similarly, when the input y is going to be approximated, the range of the input x will be divided into infinitesimal intervals with width dx , and the situation in each of the interval dx is equivalent to make a fuzzy approximation on function $Z_2(t)$ in the one-dimensional input space, as shown in Figure 4(b).

FIGURE 4: Functions $Z_1(t)$ and $Z_2(t)$.

In the auxiliary fuzzy system, we adopt the cover of MFs instead of the infinitesimal intervals. But if the number of the MFs is too small, the approximation accuracy will be affected. In contrast, if the number of the MFs is too large, the amount of computation will be increased greatly. Hence, the number of the MFs is given by [11]

$$N_{\Delta} = \frac{\sqrt[3]{K}}{2}, \quad (8)$$

where N is the number of input variables and K is the number of data stored in the memory M .

For each input x_v , specific procedures are as follows.

- (1) Each of the input variables except x_v is covered by N_{Δ} MFs to avoid the influence of the approximation errors of these input variables, and the input x_v is covered by the same MFs as it does in the fuzzy controller.
- (2) At this time, the i th rule of the auxiliary fuzzy system has changed into

$$\text{Rule } i: \text{ IF } x_v \text{ is } A_{v_i}^i, \quad \text{ THEN } y = \alpha_i \quad (9)$$

so that the approximation function $F_{N_{\Delta}}^v$ can be obtained.

- (3) Compute the sum of squared errors (SSE_v) [11] for input x_v as

$$\text{SSE}_v = \sum_{\vec{x}_j \in M} (\hat{u}(\vec{x}_j) - F_{N_{\Delta}}^v(\vec{x}_j))^2, \quad (10)$$

where \vec{x}_j is the j th input vector data stored in the memory M and $\hat{u}(\vec{x}_j)$ is the actual output of the fuzzy controller produced by that input.

- (4) Find the maximum value of the SSE; then the input variable corresponding to that value is the one selected to add in new MFs.

Additionally, the use of the auxiliary fuzzy system, whose essence is the universal approximation properties of fuzzy systems [19], is the key to the SA-OAFC. However, it does not have a direct effect on the controller. In fact, it merely evaluates the approximation effect and chooses the input variable to add in new MFs.

In order to enhance the robustness of the system and consider all of the error distribution, this paper first divides the input variable range into many small intervals Δx_i , then calculates the mean square error in each interval, and finally locates the new MF at the center of gravity c_i^* of the system mean square error distribution given by

$$c_i^* = \frac{\sum_{\Delta x_i} x_i^* \cdot \overline{e^2(\Delta x_i)}}{\sum_{\Delta x_i} e^2(\Delta x_i)}, \quad (11)$$

where x_i^* is the midpoint in the interval Δx_i .

For the new MF, the center point is the value of the current input, and the left point and the right point are located at the centre of the adjacent MFs, respectively. Meanwhile, the left point and the right point of the adjacent MFs are modified to the center of the new MF to form a continuous distribution of MFs. When a new MF is added to the input x_v , $\prod_{i=1, i \neq v}^N n_i$ new rules will subsequently be produced.

In order to avoid system oscillation, the ‘‘pseudo fuzzy output’’ method [12] is adopted to initialize the consequents of the new added rules to make the output of the system remain consistent whether the rules are increased or not at that time. Assume that, at some time k , there are n rules in the fuzzy system and the last h ones are just newly generated. We define y_p as follows, which acts as the ‘‘pseudo fuzzy output’’ of the original output of the fuzzy system with $n - h$ rules on the occasion that the h new rules were not generated at that time:

$$y_p = \sum_{i=1}^{n-h} \alpha_i \xi_i', \quad (12)$$

where $\xi_i' = \mu_i(\vec{x}(k)) / \sum_{i=1}^{n-h} \mu_i(\vec{x}(k))$.

After the generation of the new rules, the real output $y(k)$ is given by

$$y(k) = \alpha_{\text{new}} \sum_{i=n-h+1}^n \xi_i + \sum_{i=1}^{n-h} \alpha_i \xi_i = y_p, \quad (13)$$

where α_{new} is the initial value for the consequents of the new added rules and $\alpha_{n-h+1} = \alpha_{n-h+2} = \dots = \alpha_n = \alpha_{\text{new}}$. So the α_{new} can be obtained directly from the following expression:

$$\alpha_{\text{new}} = \frac{y_p - \sum_{i=1}^{n-h} \alpha_i \xi_i}{\sum_{i=n-h+1}^n \xi_i}. \quad (14)$$

3.3. Deletion Stage. If a self-structuring fuzzy system can only add in rules, the growth of the rule number will lead to the increase of computational load and the decline of parameter learning efficiency. Therefore, it is of necessity to delete those rules with little effect on the system to keep compact. We define the contribution C_i made by the i th rule on the output y and the significance index S_i representing a measurement of the importance of the i th rule as follows [12]:

$$C_i = \frac{|y_i|}{\sum_{i=1}^{N_{\text{rules}}} |y_i|} \quad (15)$$

$$S_i = \begin{cases} S_i' \tau & \text{if } C_i < \beta \\ S_i' & \text{if } C_i \geq \beta, \end{cases}$$

where $y_i = \alpha_i \xi_i$, S_i' is the previous value of S_i , $\tau \in (0, 1)$ is a decay factor, and $\beta \in (0, 1)$ is a threshold used to update S_i . S_i is initialized to 1, and if C_i is less than β , then S_i decreases; otherwise S_i keeps constant. An invariable S_i implies that the i th rule is still important for the system and should be remained. On the other hand, a decaying S_i implies that the i th rule is becoming less and less important and thus should be deleted. Therefore, the selection of τ and β will affect the rate of deleting the fuzzy rules. The smaller the τ is or the larger the β is, the faster the S_i decays, and thus the faster the ineffective fuzzy rules will be deleted. When S_i is below the delete threshold $\Omega \in (0, 1)$, the i th rule will be removed as a result. In practical applications, if computational load is given a higher priority, the delete threshold Ω can be selected as a larger value so that more fuzzy rules will be deleted and the amount of computation will be significantly reduced at the expense of less favorable system performance.

After the deletion of some rules, some MFs will no longer play a role in the system. So it is also of necessity to delete the MFs included in no rules and modify the endpoints of the adjacent MFs to maintain a continuous MF distribution.

Finally, the flow chart of the online self-structuring fuzzy controller is shown in Figure 5. For the reason that the mathematical model of the plant is unknown, it is difficult to give the strict formulaic proof of stability, but a qualitative analysis of this issue is provided in [11, 12].

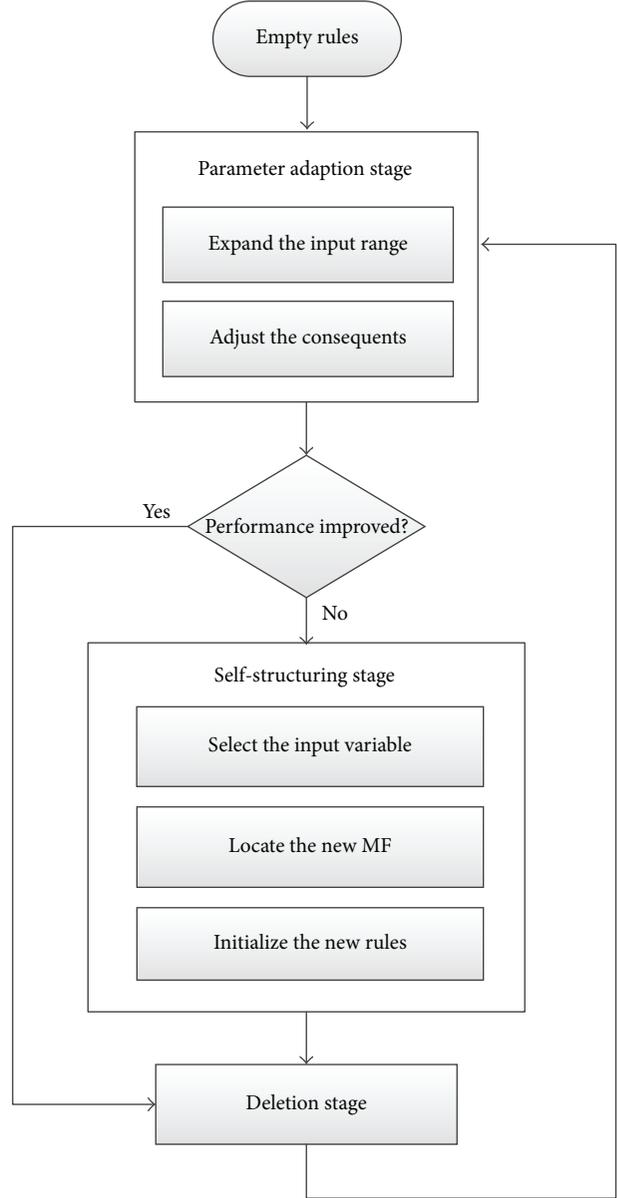


FIGURE 5: Flow chart of the controller.

4. Simulation and Experiment

4.1. Simulation. The plant is given by

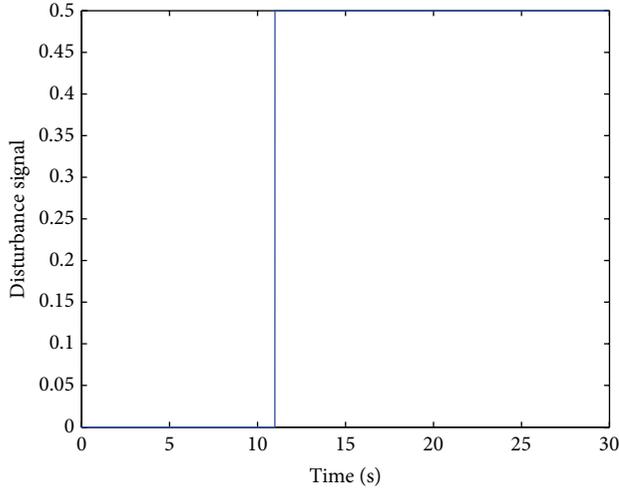
$$y(k+1) = -0.075 \sin(y(k)) \cos(y(k)) + 0.25(u + u^3) + d(k) \quad (16)$$

which shows a nonlinear behavior with respect to both the control signal u and the output y . In order to make an analysis of the robustness of the system, a disturbance signal $d(k)$, shown in Figure 6, is added to the output to simulate the abrupt changes in the plant.

We regard the desired signal $r(k)$, which is a stepwise function with the range of $[-1.5, 1.5]$, and the system output $y(k)$ as the input of the controller. The drive range of

TABLE 1: Performance comparison of different self-structuring algorithms.

Methods	DASFC [7]	DS AFC [8]	RASFC [12]	OAFSC [11]	SA-OAFC
Rule number	32	26	4	20	1
Parameter number	3	4	6	4	7
Computation time (s)	4.8362	4.1093	2.9325	5.3679	2.0492
RMSE	0.1324	0.0967	0.0895	0.0794	0.0683

FIGURE 6: Disturbance signal $d(k)$.

the controller is $[-2, 2]$, so C is set to 1.3. The simulation step is 0.01 s. The extended width for the endpoint of the MF is selected as $W = 0.2$. The number of the data used to calculate the MSE is chosen as $N_s = 100$ and the threshold for the minimum control error is chosen as $MSE_{\min} = 0.1$. The decay factor is chosen as $\tau = 0.95$. The thresholds for updating and deleting criteria are selected as $\beta = 0.01$ and $\Omega = 0.01$, respectively. These parameters are chosen through some trials to achieve favorable performance.

The initial fuzzy controller has no rules and the simulated results are shown in Figures 7 and 8. It can be observed that the controller successfully makes the plant track the desired signal $r(k)$ at first in Figure 7. When the disturbance signal $d(k)$ starts intervening after the moment 11 s, a clear deterioration in the performance of the control takes place immediately. Nevertheless, the SA-OAFC also starts counteracting this deterioration at the same time and soon the performance of the control is as good as it was, which reflects the good robustness of the system. Figure 8 shows the distribution curve of the MFs of inputs r and y at the moments 5.5 s and 30 s, reflecting the dynamic expansion of the input range and the deletion process of MFs.

For comparison, Table 1 shows the performance of the SA-OAFC and other self-structuring algorithms mentioned earlier after 3000 online time steps. Since only RASFC [12] and SA-OAFC contain the rules of deletion mechanisms, the number of rules and the computation time in RASFC [12] and SA-OAFC are much less than other algorithms. Performance is evaluated using the root mean square error (RMSE) and the results show that the error of the SA-OAFC is smaller than

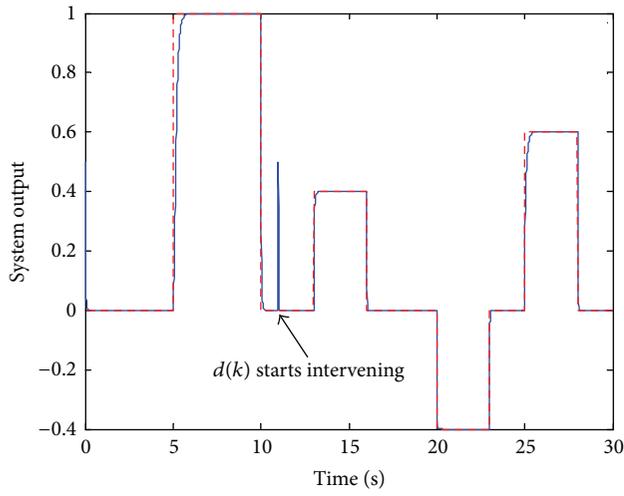
other self-structuring algorithms. However, the SA-OAFC has more design parameters for the reason that its structure is more complex.

4.2. Experiment. In this section, the plant is the Industrial Mechatronic Drives Unit (IMDU) of the Quanser Company in Canada, whose mathematical model is unknown for us, as shown in Figure 9. This device can be used to illustrate fundamentals of servo control as well as advanced topics, such as haptics and teleoperation, web winding control, backlash compensation, friction compensation, and high order coupling of complex industrial processes. The unit is equipped with four axes, two of which are the motor drive shafts and the other two shafts are free to rotate. All of the four axes contain optical encoders. Each motor is driven by a linear control amplifier with 100 Watts.

For the reason that the C-MEX file has a fast execution speed and what is more important is its ability to manipulate hardware devices in real time, we adopt the Simulink S-function in C-MEX file to achieve the SA-OAFC.

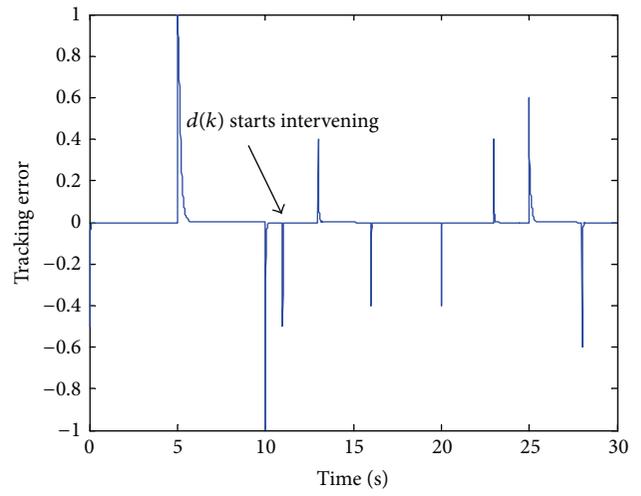
The fuzzy controller is used to control one motor drive shaft of the IMDU to track a sinusoidal signal under different load conditions. Therefore, we regard the desired signal $r(k)$, which is a sinusoidal function with the range of $[-1, 1]$, and the position output $y(k)$ of the shaft to be controlled as the inputs of the controller. For the reason that the position output $y(k)$ is measured in degrees, its unit needs to be converted to radians before $y(k)$ is sent to the controller. We set the driving interval as $[-1.35, 1.35]$, so C is set to 1.35. The extended width for the endpoint of the MF is selected as $W = 0.1$. The sampling frequency is set to 100 samples per second. The number of the data used to calculate the MSE is chosen as $N_s = 50$ and the threshold for the minimum control error is chosen as $MSE_{\min} = 0.1$. The decay factor is chosen as $\tau = 0.95$. The thresholds for updating and deleting criteria are selected as $\beta = 0.01$ and $\Omega = 0.01$, respectively. In order to test the adaption performance of the controller, the initial parameters remain unchanged whether under the load shown in Figure 10 or not. The position tracking curve is shown in Figure 11, and the number of rules is shown in Figure 12, which are both drawn based on the data collected from the real system.

For the reason that the fuzzy controller starts from empty rule, the tracking curve y has a few oscillations at first and then almost coincides with the desired signal r . It is indicated that the controller for the plant with the mathematical model unknown still can achieve an ideal performance with the parameters unchanged in different load conditions. Table 2 depicts the evolution of the self-structuring process in the inception phase when the IMDU is under load. For each



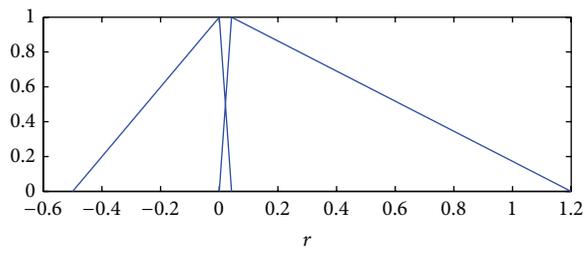
— y
- - - r

(a) Curve of the system output

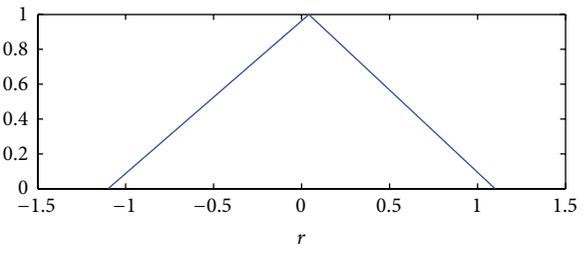


(b) Curve of the tracking error

FIGURE 7: Tracking results.



(a) 5.5 s



(b) 30 s

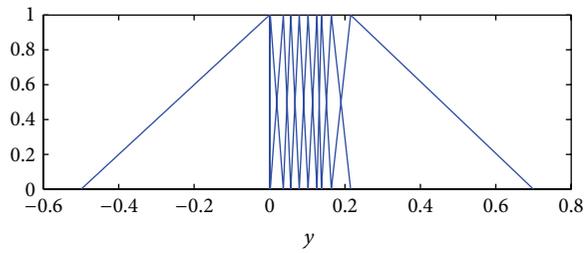


FIGURE 8: Distribution of the input MFs.

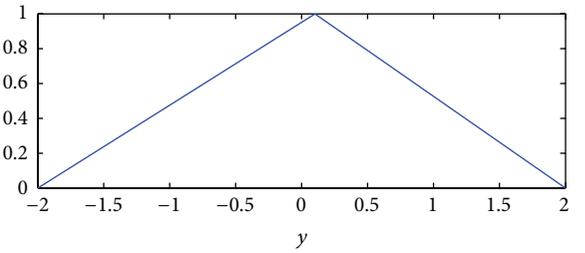


FIGURE 9: IMDU.



FIGURE 10: Load.

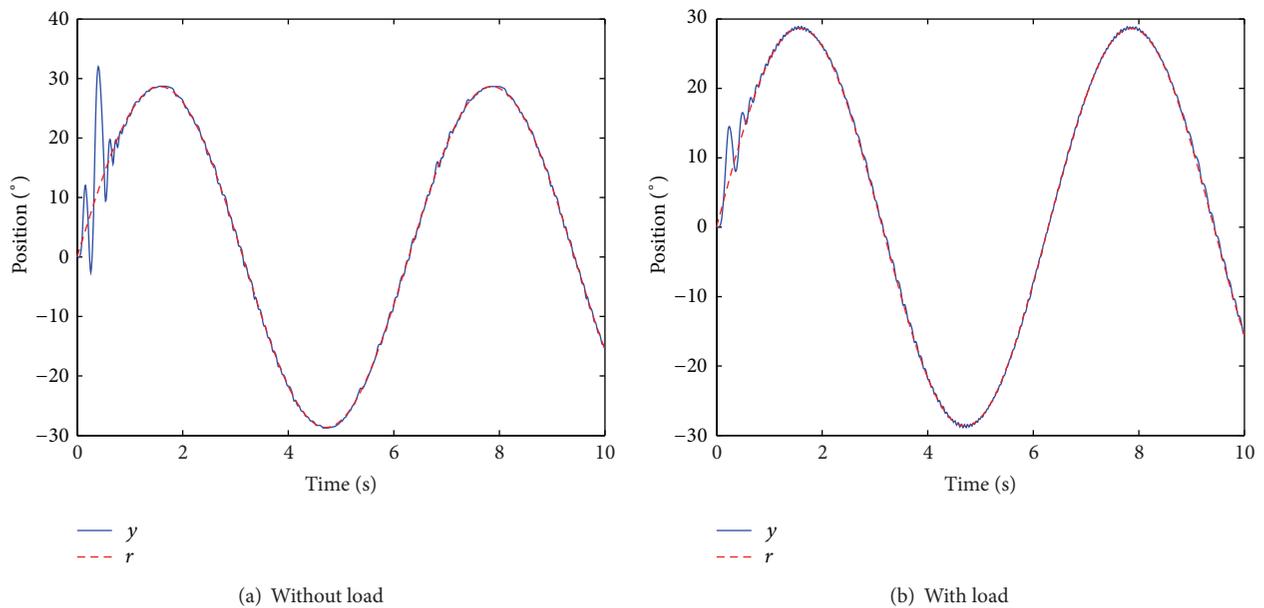


FIGURE 11: Position tracking curve with the same initial parameters.

structural change, it shows the MSE obtained after the parameter adaption stage and the values of SSE for both inputs to decide the next variable to add in new MFs. At first, the SA-OAFC always assigns new MFs to the input y , which means that the input y is more important for the system to control. When the MSE decreases and its value is less than MSE_{\min} , there is no need to add more MFs so that the self-structuring stage is skipped directly. At the same time, the rules of deletion mechanisms are on all the time and some redundant rules and MFs will be removed to keep compact when the delete condition is reached.

5. Conclusion

This paper presents a new fuzzy control algorithm, the SA-OAFC, which can adjust the structure and parameters autonomously in real time according to the desired output

TABLE 2: Evolution of the structural change under load.

MFs	MSE	SSE_r	SSE_y
1×1	0.1369	2.31	24.12
1×2	0.6315	4.69	16.79
1×3	1.4543	17.26	5.83
2×3	0.0878	—	—
1×3	0.0637	—	—
1×2	0.1218	10.37	6.26

and the actual output data of the system, in the case of absence of the mathematical model of the plant. Its main features are as follows.

- (1) Avoid introducing differential operation to improve the approachability in engineering.

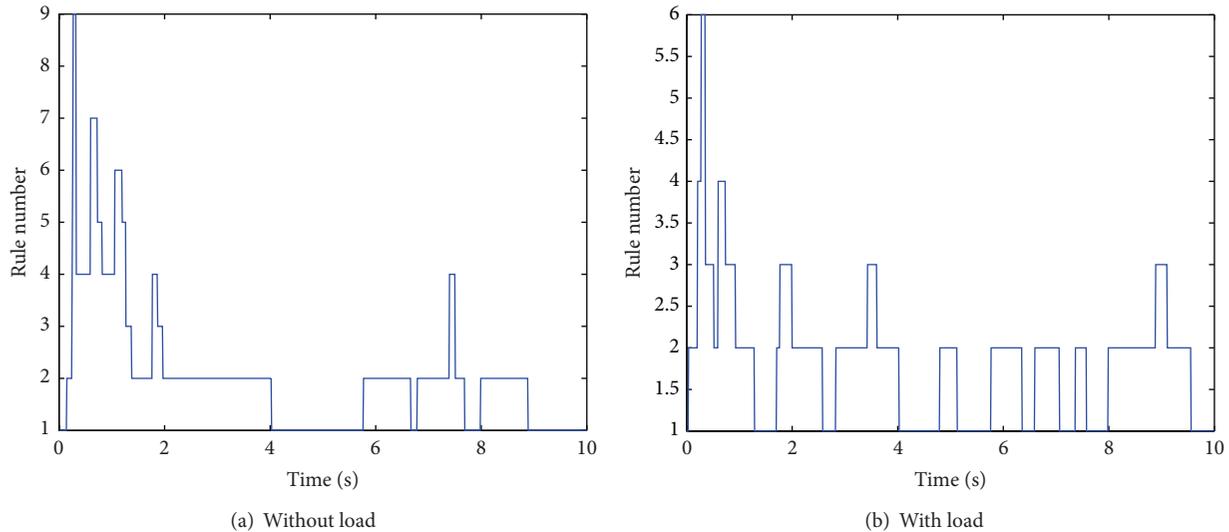


FIGURE 12: Number of rules.

- (2) Adopt the auxiliary fuzzy system to select the input variable to add in MFs effectively.
- (3) Take advantage of the “pseudo fuzzy output” method to avoid the oscillation during initializing the rule consequents.
- (4) Delete the redundant rules and MFs in real time to reduce the amount of computation to a considerable extent.

In recent years, studies on the type-2 fuzzy theory which is the extension of type-1 fuzzy theory have drawn much attention. Therefore, as the future works, the proposed control algorithm should be combined with the type-2 fuzzy theory and applied to some more complex systems in practice.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China through Grants 51490660 and 51405362, the National Key Basic Research Program of China through Grant 2015CB857100, Xinjiang Astronomical Observatory, CAS, and the Fundamental Research Funds for the Central Universities through Grant SPSY021401. Special appreciations are given to the editor and the anonymous reviewers for their careful reviews and valuable comments.

References

- [1] S. Liu, Z. Wang, P. Fei et al., “Advance and prospect of the fuzzy control,” *Mechanical Electrical Engineering Magazine*, vol. 17, no. 1, pp. 9–11, 2000.
- [2] L. Wang, *A Course in Fuzzy Systems and Control*, Tsinghua University Press, Beijing, China, 2003.
- [3] J. E. Schmitz, F. V. Silva, L. C. Neves Filho, A. M. F. Fileti, and V. Silveira, “Multivariable fuzzy control strategy for an experimental chiller system,” *Journal of Food Process Engineering*, vol. 37, no. 2, pp. 160–168, 2014.
- [4] W.-Y. Wang, Y.-H. Chien, and I.-H. Li, “An on-line robust and adaptive T-S fuzzy-neural controller for more general unknown systems,” *International Journal of Fuzzy Systems*, vol. 10, no. 1, pp. 33–49, 2008.
- [5] Y.-J. Liu and Y.-Q. Zheng, “Adaptive robust fuzzy control for a class of uncertain chaotic systems,” *Nonlinear Dynamics*, vol. 57, no. 3, pp. 431–439, 2009.
- [6] D. Bellomo, D. Naso, and R. Babuška, “Adaptive fuzzy control of a non-linear servo-drive: theory and experimental results,” *Engineering Applications of Artificial Intelligence*, vol. 21, no. 6, pp. 846–857, 2008.
- [7] J.-H. Park, G.-T. Park, S.-H. Kim, and C.-J. Moon, “Direct adaptive self-structuring fuzzy controller for nonaffine nonlinear system,” *Fuzzy Sets and Systems*, vol. 153, no. 3, pp. 429–445, 2005.
- [8] P. A. Phan and T. J. Gale, “Direct adaptive fuzzy control with a self-structuring algorithm,” *Fuzzy Sets and Systems*, vol. 159, no. 8, pp. 871–899, 2008.
- [9] R. Qi and X. Yao, “Nonlinear identification and adaptive control based on self-structuring fuzzy systems,” in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '09)*, pp. 294–299, Jeju Island, Korea, August 2009.
- [10] R. Qi and M. A. Brdys, “Indirect adaptive controller based on a self-structuring fuzzy system for nonlinear modeling and control,” *International Journal of Applied Mathematics and Computer Science*, vol. 19, no. 4, pp. 619–630, 2009.
- [11] A. B. Cara, H. Pomares, and I. Rojas, “A new methodology for the online adaptation of fuzzy self-structuring controllers,” *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 3, pp. 449–464, 2011.
- [12] P.-C. Chen, C.-H. Wang, and T.-T. Lee, “Robust adaptive self-structuring fuzzy control design for nonaffine, nonlinear

- systems,” *International Journal of Systems Science*, vol. 42, no. 1, pp. 149–169, 2011.
- [13] H. C. Andersen, A. Lotfi, and A. C. Tsoi, “A new approach to adaptive fuzzy control: the controller output error method,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 27, no. 4, pp. 686–691, 1997.
- [14] T. Terano, K. Asai, and M. Sugeno, *Fuzzy Systems Theory and Its Applications*, Academic Press, Boston, Mass, USA, 1992.
- [15] P. Angelov, “A fuzzy controller with evolving structure,” *Information Sciences*, vol. 161, no. 1-2, pp. 21–35, 2004.
- [16] C.-T. Lin, C. J. Lin, and C. S. Lee, “Fuzzy adaptive learning control network with on-line neural learning,” *Fuzzy Sets and Systems*, vol. 71, no. 1, pp. 25–45, 1995.
- [17] L. Wang and Y. Frayman, “A dynamically generated fuzzy neural network and its application to torsional vibration control of tandem cold rolling mill spindles,” *Engineering Applications of Artificial Intelligence*, vol. 15, no. 6, pp. 541–550, 2002.
- [18] Y. Gao and M. J. Er, “Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems,” *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, pp. 462–477, 2003.
- [19] J. L. Castro, “Fuzzy logic controllers are universal approximators,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25, no. 4, pp. 629–635, 1995.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

