IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 12, 2017, accepted September 3, 2017, date of publication September 8, 2017, date of current version September 27, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2750399

An Approach to Improve the Performance of Simulated Annealing Algorithm Utilizing the Variable Universe Adaptive Fuzzy Logic System

LISHU QIN[1,2], JIANWEI WANG[2], HONGXING LI[1], YI SUN[3], AND SHUYANG LI[4]
[1]School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, China
[2]College of Mechanical Engineering, Dalian University, Dalian 116622, China
[3]Nanjing Golden Dragon Bus Company, Nanjing 211215, China
[4]Department of Basic Science, Dalian Naval Academy, Dalian 116013, China

Corresponding author: Lishu Qin (qinlishu@dlu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 51275061 and Grant 61672121.

ABSTRACT This paper focuses on improving the standard form of the classical simulated annealing algorithm (CSAA). A novel method of improving the performance of CSAA by the variable universe adaptive fuzzy logic system (VUAFLS) is studied. We develop the VUAFLS to adjust the annealing temperature, which is a very important parameter governing the performance of CSAA, and this algorithm is named VUAFLS-CSAA. The main innovations of VUAFLS-CSAA lie in the application of VUAFLS containing the fast cooling mechanism and reheating mechanism relative to the characteristic of the sustained temperature fall of CSAA. Compared with the conventional method for controlling annealing temperature, VUAFLS-CSAA can control the annealing temperature more effectively, leading to the high efficiency of CSAA. The performance of the proposed method is evaluated and compared with CSAA through two examples. One is the image restoration problem, and the other is the traveling salesman problem (TSP). The experimental result indicates that the new method proposed in this paper can improve the efficiency of CSAA by tremendously shortening the iteration optimization process. And at the same time, the successful application of the new method for tackling two different problems demonstrates the generality of this method. In addition, techniques that can further improve the performance of CSAA are discussed.

INDEX TERMS Fuzzy logic system, variable universe, simulated annealing algorithm.

I. INTRODUCTION

Nowadays, optimization is an important technique in many areas. Due to its performance in applications to the complex optimization problem, metaheuristics has recently received a great deal of attention. CSAA is one of the most popular metaheuristics. CSAA was introduced to the combinatorial optimization fields by Kirkpatrick *et al.* in 1983 [1]. It has been successfully applied in many areas, such as path planning [2], transportation and logistics [3], medical diagnostics [4], design optimization of control systems [5], and facilities location [6]. CSAA originated from statistical mechanics and is based on a Monte Carlo model presented by Metropolis *et al.* in 1953 [7] to simulate a collection of atoms in equilibrium at a given temperature.

In essence, CSAA is a type of greedy algorithm. However, CSAA extends the local search algorithm. And the main advantage of CSAA is that stochastic factors are introduced to ensure the possibility of reaching the global optimum. The key concept of CSAA can be simply described as the following. CSAA is a search procedure that uses local hill climbing but in a modified manner. Given an initial solution, CSAA will continue searching for a better solution. Meanwhile, by accepting worse solutions with a certain probability, the algorithm has the chance to climb out of the local minima and find the global optimum. The criterion is based on the Metropolis criterion. To be an effective, nonlinear combinational optimization algorithm, CSAA has been proven strictly in theory [8], [9].

CSAA uses the Boltzmann probability distribution, $P(\Delta C) \propto exp(-\Delta C/k_b T_c)$, where ΔC and T_c show the energy and temperature of the system, respectively, and k_b is the Boltzmann constant. For a single system with a certain temperature, the Boltzmann distribution gives the probability that the system is in the specific state [10]. In practice, temperature like parameter T_c is used as the key controlling parameter like temperature in the annealing process.

The optimization process begins with an initial configuration S_i with C_i for the objective function. The new candidate configuration S_{i+1} is generated by perturbing S_i then, the new objective function C_{i+1} can be obtained. The acceptance criterion for the new configuration is defined as below [8].

$$P = \begin{cases} 1, & \text{if } C_i \geq C_{i+1} \\ exp((C_i - C_{i+1})/T_c), & \text{otherwise} \end{cases} \quad (1)$$

Then, a number is randomly taken from the uniform probability distribution in the range [0, 1]. If P is larger than the chosen number, the new configuration will be accepted to replace the former one. The acceptance of the configuration with the larger objective function value allows the algorithm a chance to escape from sinking into the local minimum.

As fig.1 shows, there are five solutions, $S_i(i = A, B, C, D, E)$, representing the solutions in connection with some questions. And $C_i(i = A, B, C, D, E)$ are the objective functions corresponding to S_i. It is obvious that the local minimum objective function is C_B brought by the solution S_B, and the global minimum objective function is C_E corresponding to S_E. Assuming S_A is the initial solution, S_B should be the better solution, but it is merely a local minimum solution. According to the key concept of CSAA, when S_B is searched, there exists a certain probability that CSAA will reach worse solutions, which is jumping out of the local minima, S_C or S_D relatively to the solution S_B. Therefore, as the search process continues, it is eventually possible to achieve the global optimum solution S_E.

In many practical applications, the search results for the optimal solutions of CSAA are occasionally far from satisfactory. In recent decades, many studies have sought to improve the performance of CSAA. Usually, hybrid algorithms can overcome the disadvantages and combine the advantages of individual algorithms. The idea of fusing the CSAA with other intelligent algorithms is one of the means by which CSAA has been improved. For each optimization algorithm combined with the CSAA, its popularity is due to excellent characteristics such as easy implementation and good optimization performance. Genetic algorithm (GA) [11], [12], artificial neural network (ANN) [13], [14], particle swarm optimization (PSO) [15], [16], and so forth are the most popular algorithms of recent years. However, there are certain drawbacks to the algorithms mixed with CSAA. For example, the complexity of the algorithm's implementation is often large with respect to GA. And the choice of many parameters, such as crossover and mutation rates, which greatly affect the quality of the solutions, lacks distinct theoretical instruction and mainly depends on the experience of the algorithm designer [17]. With respect to PSO, due to a few adjustable parameters, this algorithm is simple in construction and can be easily applied to many issues. But the existence of premature convergence is a disturbing problem and, therefore, tends to be trapped in local optima [18], [19]. With respect to ANN, there are also several problems that need to be studied further. For instance, when constructing

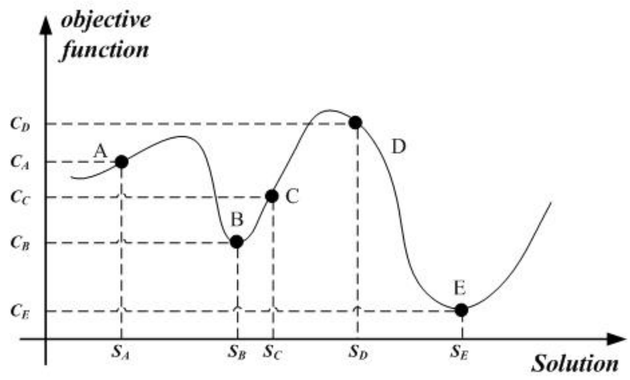

FIGURE 1. The sketch map of the core concept of CSAA.

an ANN, the number of hidden layers that should be chosen also lacks distinct theoretical instruction. And, the training errors of each step's training will be fed back to the ANN for the next step's training. Accumulation and propagation of the errors will greatly degrade the performance of the ANN [20]. Great effort has been made to improve the properties of ANN. However, the needed computation is typically increased considerably. Meanwhile, for the distinguishing characteristics of these evolutionary algorithms, it is always difficult for users to establish the relationship between evolutionary algorithms and the specific questions. To some degree, none of these evolutionary algorithmic techniques is easily understood by the related technician, and this situation hinders convenience in wide utilization.

In such situations, intelligent techniques such as fuzzy logic system (FLS) can be useful. Fuzzy sets theory was proposed by Zadeh [21]. FLS has been proven to be a competitive intelligent system in many fields, such as stock market prediction [22], path tracking [23], and detection and recognition [24]. Many works have simulated human reasoning in the face of uncertainty using approximate information to generate proper decisions. FLS is used in these studies to handle uncertainty and provide systems whose behavior could be easily understood and analyzed by users.

To further enhance FLS performance, the idea of VUAFLS was proposed by Li [25]–[28] in 1995. VUAFLS is regarded as a promising FLS due to its out-standing performance. And the excellent properties of VUAFLS has been proven extensively in the application of quadruple inverted pendulum control [29], chaotic systems [30], on-line identification [31], controller optimization [32] and so on.

Because FLS can be easily used to express knowledge and simulate human thinking, the experience of experts in enhancing the performance of CSAA can be beneficial in the use of FLS. Unfortunately, there are few FLS combined methods incorporating CSAA to improve CSAA. So, the main goal of this paper is to introduce new thinking into FLS to improve the performance of CSAA. In this paper, we have developed an algorithm that can greatly improve the performance of the CSAA by introducing single-input single-output (SISO) FLS to modify the annealing temperature of CSAA. Furthermore, SISO VUAFLS is designed for annealing temperature tuning. These new methods can be treated as the standard strategy regardless of the specific application areas in the concrete application of CSAA. The image restoration problem and the TSP are used to evaluate the performance of the proposed methods.

This paper is structured as follows: Section 2 provides preliminaries, including CSAA, FLS and VUAFLS. Section 3 introduces the details for improving CSAA through FLS and VUAFLS techniques. The experimental results are presented in section 4. Finally, the conclusion is provided in section 5.

II. PRELIMINARIES

A. CSAA

The inspiration for CSAA comes from the annealing process in metallurgy. The theoretical background is fully described in Aarts and Korst [8].

The relationship between the annealing process in metallurgy and the solving process in the optimization problem is shown in TABLE 1.

TABLE 1. The correspondence between the annealing process in metallurgy and the solving process in optimization problem.

Annealing process	Solving process
metallic substances	optimization problem
states of the substances	solutions
states of the substances contained the lowest energy	optimum solution
energy	objective function
temperature	control parameter

The controlled cooling process during which the crystal size of a material increases is often called "Annealing". Transition probability is a standard feature that allows a non-improving move to be made. At each iteration, the system is perturbed, and the change in energy is calculated. CSAA is a global optimization algorithm only because it obeys the Metropolis acceptance criterion. The search starts from an initial feasible solution. Each solution has a specific cost value. During algorithm execution, the temperature is decreased, and as a result, worse solutions are less likely to be accepted. Fig.2 shows the basic steps of CSAA. In the figure, $P = e^{-\Delta C/T_c}$, where $\Delta C = C_i - C_{i+1}$ and P_{ran} is a random number in the range [0, 1].

There are two loops contained inside CSAA, the inner loop and outer loop. The inner loop is controlled by the perturbation time during which new solutions are generated. Meanwhile, the acceptance of new solutions in accordance with the Metropolis criterion is also done in the inner loop. And temperature renewal and the stop condition of CSAA are considered in the outer loop. Therefore, the outer loop is controlled by the iterations. The iterations are kept running until a stop criterion is met.

In addtion, according to the CSAA, the random number P_{ran} plays a vital role during the annealing process. Therefore, in order to ensure that CSAA is properly implemented, the problem of whether the random number P_{ran} is a $\mathbb{REAL}$ random number should be considered.

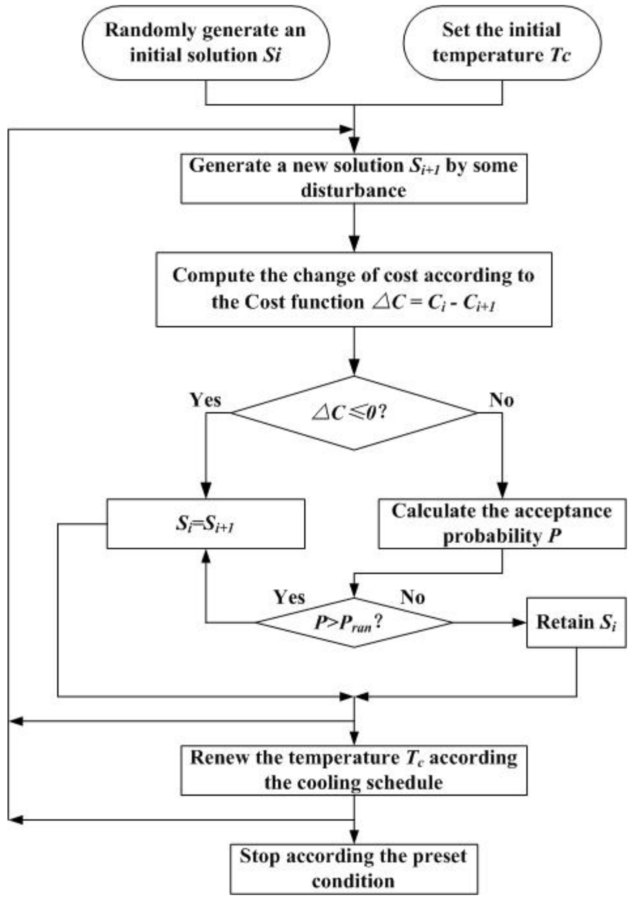

FIGURE 2. Flow chart of CSAA.

B. FLS

FLS is a system that mimics the way a human brain thinks and solves problems. It is also a paradigm with which human subjectivity is introduced into objective science and a method to model and use human knowledge and senses as they are. The distinctive characteristic of FLS is to approximate human decision making using natural language terms instead of quantitative terms. It enables computerized devices to reason more like humans. Using FLS, designers can generally realize lower development costs, superior features, and better application effects. The general structure of FLS is shown in Fig.3.

1) INPUT AND OUTPUT

Let $X_i = [-E_i, E_i](i = 1, 2, \cdots, n)$ be the universe of the input variable $x_i(i = 1, 2, \cdots, n)$ and $Y = [-U, U]$ be the universe of output variable y.

2) FUZZIFIER

performs a mapping from a crisp input x_i to a fuzzy set A_{x_i} in X_i, where A_{x_i} is the label of the fuzzy set such as "small", "medium", "large", etc. In fuzzy set theory, a fuzzy set A_{x_i} of

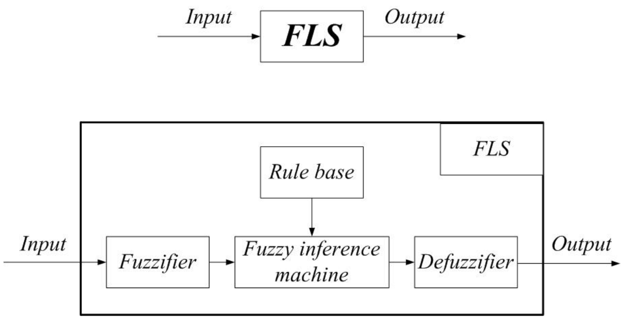

FIGURE 3. The structure of FLS.

a universe of discourse X_i is characterized by a membership function $\mu_{A_{x_i}}(x_i)$, which assigns to each element $x_i \in X_i$ a number $\mu_{A_{x_i}}(x_i)$ in the interval 0 to 1 that represents the grade of membership in A_{x_i}, i.e., $A_{x_i} = ((x_i, \mu_{A_{x_i}}(x_i))|x_i \in X_i)$.

3) RULE BASE

consists of a collection of fuzzy IF-THEN rules. Assume that there are M rules, and the lth rule is rule l: if x_1 is $A^l_{x_1}$ and x_2 is $A^l_{x_2}$ and $\cdots$ and x_n is $A^l_{x_n}$ then y is B^l, $l = 1, 2, \cdots, M$ where $x_i(i = 1, 2, \cdots, n)$ and y are the crisp input and output of the fuzzy system, respectively, and $A^l_{x_n}$ and B^l are labels of fuzzy sets in X_i and y, respectively.

4) FUZZY INFERENCE MACHINE

performs a mapping from fuzzy sets in X_i to fuzzy sets in Y based on the IF-THEN rules in the rule base.

5) DEFUZZIFIER

maps fuzzy sets in Y to a crisp value in Y. Here, we use the sum-product inference and the center-average defuzzifier. So, the FLS can be expressed as

$$y(x) = \frac{\sum_{l=1}^{M} y^l \prod_{i=1}^{n} \mu^l_{A_{x_i}}(x_i)}{\sum_{l=1}^{M} \prod_{i=1}^{n} \mu^l_{A_{x_i}}(x_i)}$$

where $y(x)$ is the crisp output of the fuzzy system, $\mu^l_{A_{x_i}}(x_i)$ is the membership degree of input x_i to fuzzy set $A^l_{x_i}$, and y^l is the point at which the membership function of fuzzy set B^l achieves its maximum value.

C. VUAFLS

After the concept of VUAFLS was presented by Li in 1995, many studies were carried out to determine its properties and constructions. The most representative examples of VUAFLS are the success of the simulation model control experiment of a quadruple inverted pendulum [29] and the real quadruple inverted pendulum control experiment [33]. As controlling a quadruple inverted pendulum is an enormous challenge that we noted have demonstrated VUAFLS's good performance.

As seen below, Fig.4, (b) illustrates the initial universe of discourse and is partitioned by the fuzzy sets, which are triangular in shape. (a) and (c) show the expansion and contraction, respectively, of the discourse of the universe. The variable universe means that universes $X = [-E, E]$ can change according to the changing of variables $x \in X$. So, the universes are denoted by,

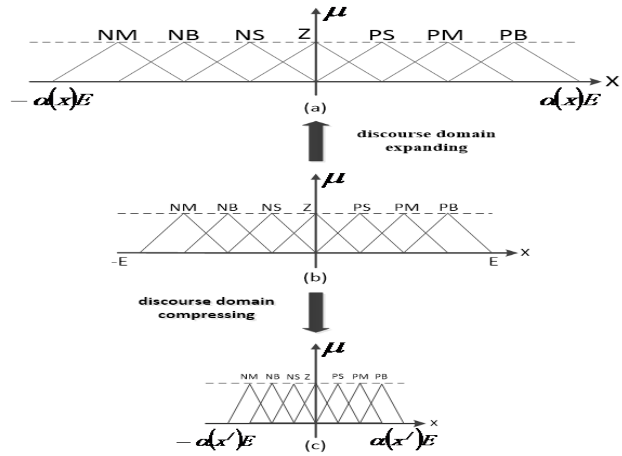

FIGURE 4. (a) The expansion of the discourse domain,(b) the original discourse domain,(c) the contraction of the discourse domain.

$$X(x) = [-\alpha(x)E, \alpha(x)E]$$

where $\alpha(x)$ is called the contraction-expansion factor. The design of $\alpha(x)$ should follow the principle of duality, monotonicity, coordination, normality and zero avoidance. Always $\alpha(x)$ can be selected as

$$\alpha(x) = 1 - Kexp(-\delta x^2)$$

where, K and δ are the design parameters. More details can be found in reference [28].

Because FLS can be regarded as a type of interpolation algorithm [27], the performance of FLS mainly depends on the number of effective fuzzy rules in the validity of the discourse universe. Thus, when the discourse universe contracts, the amount of membership functions increases relatively. Based on this concept, the design requirements of the membership functions can be greatly relaxed. For simplicity of application, the normal, consistent, and complete fuzzy sets with the triangular membership functions are often considered [25], [27], [34], although other shapes such as the bell, Gaussian and trapezoid which are used to describe the membership functions, can also be selected. Meanwhile, the performance of FLS can be enhanced despite the limited number of fuzzy rules in the rule base.

III. THE DESIGN FOR THE IMPROVEMENT OF CSAA

As mentioned before, the annealing temperature T_c is an important factor. First, the fast cooling process may lose better solution candidates, whereas slow cooling may require excessive computation time.

Second, according to (1), the function $exp((C_i - C_{i+1})/T_c)$ clearly explains that a higher temperature allows a higher chance of transition to a worse solution. Conversely, the chance of uphill transition will reduce. Therefore, the possibility of jumping out of the local minima will become lower when accompanied by monotonically decreasing temperature. If the local minima are encountered at a temperature that is low enough that the search is stopped, it will lead to deterioration in the performance of CSAA.

Therefore, improvement can be made based on the concept that, when the neighboring solution exhibits an upward trend, S_i, the heating process should be implemented in order to provide more possibility of getting out of the local minima. When the neighboring solution has a downhill trend, S_j, meaning the value of the objective function is improved, then decrease the temperature for better convergence to the optimum, as shown in Fig.5.

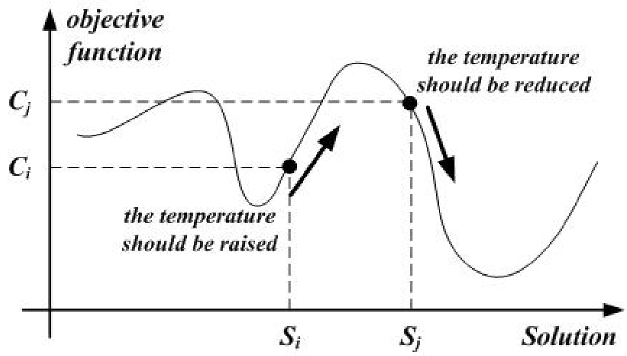

FIGURE 5. The dynamic adjustment of temperature.

Therefore, to improve CSAA performance, considering the aforementioned tracks, this paper proposes a novel method containing a cooling and reheating mechanism for the dynamic adjustment of the annealing temperature according to neighboring solutions conditioned relative to the current solution using a SISO FLS scheme. Furthermore, the SISO VUAFLS will also be used to further enhance the performance of CSAA.

A. THE DESIGN OF FLS FOR THE TUNING OF THE ANNEALING TEMPERATURE OF CSAA

The *Input* of the FLS, ΔC, is the change rate of the cost function and is defined directly as

$$\Delta C = \frac{C_i - C_{i+1}}{C_i}$$

ΔC reflects the state of the solution. There are three cases where a comparison is made between the current cost function C_i and the neighboring cost function C_{i+1}. $\Delta C < 0$ means the current solution S_i is "better" than the neighboring solution S_{i+1}. $\Delta C = 0$ means the current solution S_i shows "no change" relative to the neighboring solution S_{i+1}. Similarly, $\Delta C > 0$ means the current solution S_i is "worse" than the neighboring solution S_{i+1}.

So, we define three fuzzy sets for ΔC, which are taken as the triangle waves shown in Fig.6.

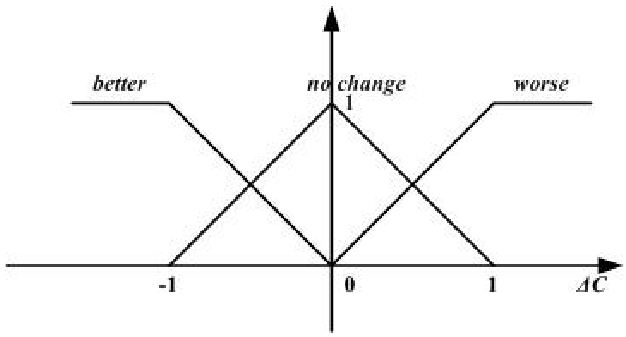

FIGURE 6. The three fuzzy sets of ΔC.

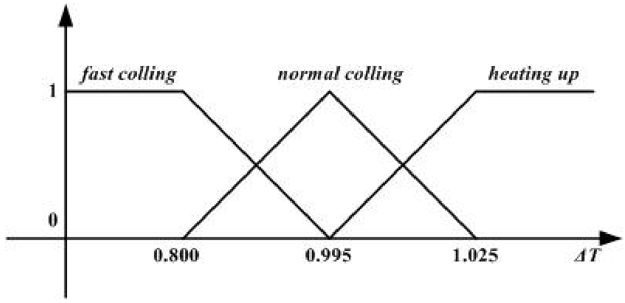

FIGURE 7. The three fuzzy sets of ΔT.

Expressions are as follows:

$$better(\Delta C) = \begin{cases} 1, & \Delta C \leq -1 \\ -\Delta C, & -1 < \Delta C < 0 \\ 0, & others \end{cases}$$

$$no\ change(\Delta C) = \begin{cases} 1 + \Delta C, & -1 < \Delta C \leq 0 \\ 1 - \Delta C, & 0 < \Delta C \leq 1 \\ 0, & others \end{cases}$$

$$worse(\Delta C) = \begin{cases} \Delta C, & 0 < \Delta C \leq 1 \\ 1, & 1 < \Delta C \\ 0, & others \end{cases}$$

The *Output* of the FLS is ΔT, ΔT will be used to weight the current annealing temperature T_c. Here, we also define three fuzzy sets for ΔT, which are taken as triangle waves. The three fuzzy sets are assigned the meanings "fast cooling", "normal cooling" and "heating up" as shown in Fig.7.

Expressions are as follows:

$$fast\ colling(\Delta T) = \begin{cases} 1, & \Delta T \leq 0.800 \\ -\Delta T, & 0.800 < \Delta T < 0.995 \\ 0, & others \end{cases}$$

$$normal\ colling(\Delta T) = \begin{cases} 1 + \Delta T, & 0.800 < \Delta T \leq 0.995 \\ 1 - \Delta T, & 0.995 < \Delta T \leq 1.025 \\ 0, & others \end{cases}$$

$$heating\ up(\Delta T) = \begin{cases} \Delta T, & 0.995 < \Delta T \leq 1.025 \\ 1, & 1.025 < \Delta T \\ 0, & others \end{cases}$$

where, 0.8, 0.995 and 1.025 are the peak values corresponding to the three fuzzy sets, "fast cooling", "normal cooling" and "heating up", respectively.

The *fuzzyrules* for the construction of the FLS are listed as follows:

*Rule*1 : if ΔC is "better", then ΔT should be "fast cooling".

*Rule*2 : if ΔC is "no change", then ΔT should be "normal cooling ".

*Rule*3 : if ΔC is "worse", then ΔT should be "heating up".

*Rule*1 means that a better solution has been discovered or that the good trend in the optimization process is likely to continue; so, the annealing process can be carried out rapidly to accelerate the solution search process. This "fast cooling" can improve algorithm efficiency, as shown by point *D* in Fig.5.

*Rule*2 "no change", means that the solution just searched may be near the optimum solution, so the annealing temperature should be slowly decreasing comparatively because the search for the neighbor solutions of the present solution should be enhanced.

*Rule*3 embodies the very essence of CSAA: that is, although the worse solution has been searched, the optimization process may be in an "uphill" move. Accordingly, the search process requires more energy to preserve the "uphill" trend by the specific operation for increasing T_c, as shown by point *C* shown in Fig.5.

Here, the Mamdani fuzzy inference system is used for the inference system. In this approach, according to [27] and [28], only the peaks of the output fuzzy sets can be used. That is, the inference process is irrelevant to the shapes of the output fuzzy sets. So, each rule's consequent parameter is specified by a fuzzy singleton. Using the Mamdani inference mechanism, the output of the FLS can be written as follows:

$$\Delta T = \frac{\sum_i \mu_i(\Delta C) Peak_i}{\sum_i \mu_i(\Delta C)}$$

where i is the index for the input and output set; here, i should be three. To input, $i = 1, 2, 3$ indicate the three fuzzy sets "better", "no change" and "worse", respectively. Meanwhile, to output, $i = 1, 2, 3$ stands for the three fuzzy sets "fast cooling", "normal cooling" and "heating up", respectively. $\mu_i(\Delta C)$ is the firing strength of the related fuzzy set, and $Peak_i$ are the consequent parameters of the three fuzzy sets "fast cooling", "normal cooling" and "heating up", 0.800, 0.995 and 1.025, respectively.

B. THE DESIGN OF VUAFLS FOR THE TUNING OF THE ANNEALING TEMPERATURE OF CSAA

As mentioned previously, VUAFLS can open the way to improving FLS performance. As the annealing process continues, the range of the valid discourse domain of ΔC is in a constant process of change. So, in order to ensure the performance of FLS, VUAFLS can dynamically adjust the range of the valid discourse domain according to the change in ΔC. In this sense, for example, if the solution by which FLS-CSAA searches becomes better and better approximated by the optimum solution, the value of ΔC will become smaller and smaller. And the valid discourse domain of ΔC becomes smaller and smaller. With the domain of constant discourse contraction, although there is no change in the absolute number of fuzzy rules, the number of fuzzy rules increases in a relatively small range of the discourse domain. Obviously, the performance of FLS can be significantly improved.

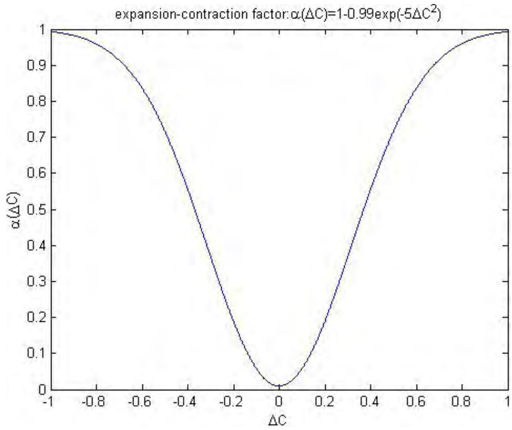

FIGURE 8. The expansion-contraction factor.

Also, by using the Mamdani inference mechanism, the output of the VUAFLS can be written as follows:

$$\Delta T = \frac{\sum_i \mu_i(\frac{\Delta C}{\alpha(\Delta C)}) Peak_i}{\sum_i \mu_i(\frac{\Delta C}{\alpha(\Delta C)})}$$

where, $\alpha(\Delta C)$ is the expansion-contraction factor. In this paper, $\alpha(\Delta C)$ can be chosen as,

$$\alpha(\Delta C) = 1 - 0.99 exp(-5\Delta C^2)$$

Fig.8 shows the mathematical properties of the expansion-contraction factor $\alpha(\Delta C)$.

From Fig.8, the mathematical properties of duality, monotonicity, coordination, normality and zero avoidance, which the expansion-contraction factor should have, can be guaranteed [28].

C. THE GENERATION OF THE $\mathbb{REAL}$ RANDOM NUMBER

It is obvious that the random number plays a very important role when CSAA is implemented. The fundamental cause of the importance of the random number lies in the production of the stochastic model and the assistance of the Metropolis criterion. Only when the random number is a $\mathbb{REAL}$ random number can the ergodicity of the solutions of the whole solution space be ensured, meaning the certainty of reaching the global optimum solution.

However, when the CSAA is implemented using a computer, almost all the programming languages can only provide the pseudo random numbers. To tackle this problem, the generation of the $\mathbb{REAL}$ random numbers based on CLOCK is adapted in this paper. It is certain that the CLOCK will continue changing with the running of CSAA. And this will lead to the constant changing of the generation conditions of the random number. So, the generation of the $\mathbb{REAL}$ random numbers can be ensured.

IV. PERFORMANCE EVALUATION

In this section, two examples, the restoration of an image problem and the TSP, are used to evaluate the performance of CSAA, FLS-CSAA and VUAFLS-CSAA. The reason for discussing the performance of FLS-CSAA here is that FLS is the basis of VUAFLS, that is to say, VUAFLS is the improvement of FLS. Therefore, for a better illustration the advantage of VUAFLS-CSAA, this paper also compares the performance of FLS-CSAA with VUAFLS-CSAA.

Because these two problems are categorized as discrete combinatorial optimization problems, the new solution S_{i+1} and the new objective function C_{i+1}, which corresponds to S_{i+1}, can be generated by the "swapping" method. Here, the "2-option" swapping method is used [35].

Two kinds of algorithms are presented here. The CSAA is implemented in accordance with **Algorithm1.** And **Algorithm2.** indicates the running processes of FLS-CSAA and VUAFLS-CSAA.

Algorithm 1

Step1. Generate an initial solution S_i arbitrarily.

Step2. Set the initial value T_c for the temperature.

Step3. Generate the new candidate solution S_{i+1} by random perturbing.

Step4. Calculate the objective function C_i and C_{i+1} corresponding to S_i and S_{i+1}.

Step5. If C_{i+1} is better than C_i, let S_i be S_{i+1}. Otherwise, let S_i be S_{i+1} with probability $exp((C_i - C_{i+1})/T_c)$.

Step6. If the given stopping condition is satisfied, **STOP**. Otherwise, let $T_c = \Delta T \times T_c$, and then go to **Step3**. ΔT is just a given constant for "normal cooling"; $\Delta T = 0.995$ here.

A. ALGORITHM EVALUATION USING IMAGE RESTORATION PROBLEM

Lenna is the name given to a standard test image widely used in the field of image processing since 1973. The original image is shown in Fig.9.

Here, the picture Lenna was separated into twenty parts. And each part is assigned a random number from 1 to 20. Fig.10. shows the twenty slices marked with the numbers 1, 2, 3, ..., 18, 19, 20 in sequence from left to right.

The goal is to restore the image by using CSAA, FLS-CSAA and VUAFLS-CSAA. Every part contains 72 × 1440 pixels. That is, there are 1440 pixels in every column and 72 pixels in every row. To finish the restoration task, our thinking is to calculate the match degree of the total 20 parts based on the individual column edge texture and edge color of every part. So, the objective function C_i should be the match degree of the total 20 parts, and the solution S_i is one group of sequences containing the numbers 1 to 20.

The experiment is implemented following the coefficients given below. For CSAA, the initial temperature T_c is 1. There is only one coefficient $\Delta T = 0.995$ for the adjustment of the annealing temperature. And, the criterion for stopping is $T_c \leq 0.1^{15}$. Hence, this is an algorithm with the fixed iterative number, 6891 times. For FLS-CSAA and VUAFLS-CSAA, the annealing temperature T_c is adjusted by using CSAA combined with FLS and VUAFLS; the initial temperature T_c is also 1. But ΔT will be calculated by FLS and VUAFLS,

Algorithm 2

Step1. Generate an initial solution S_i arbitrarily.

Step2. Set the initial value T_c for the temperature.

Step3. Generate the new candidate solution S_{i+1} by random perturbing.

Step4. Calculate the objective function C_i and C_{i+1} corresponding to S_i and S_{i+1}.

Step5. If C_{i+1} is better than C_i, let S_i be S_{i+1}. Otherwise, let S_i be S_{i+1} with probability $exp((C_i - C_{i+1})/T_c)$.

Step6. Calculate the change rate of objective function $\Delta C = (C_i - C_{i+1})/C_i)$. And then let ΔC be the input variable of the FLS or VUAFLS.

Step7. For **FLS-CSAA**, calculate the coefficient ΔT according to the value of ΔC combined with the fuzzy reference rules. For **VUAFLS-CSAA**, to get ΔT, besides the operations of **FLS-CSAA**, the expansion-contraction factor $\alpha(\Delta C) = 1 - 0.99exp(-5\Delta C^2)$ should also be considered.

Step8. If the given stopping condition is satisfied, **STOP**. Otherwise, let $T_c = \Delta T \times T_c$, and then goto **Step3**.

FIGURE 9. Lenna.

FIGURE 10. The disordered pieces of the image "Lenna".

which is elaborated in section 3. For the introduction to the reheating mechanism, in addition to the stopping temperature $T_c \leq 0.1^{15}$, a new stop condition named "consecutive reheating times", which we shall denote by N, should also be considered. N means that S_{i+1}, which is searched N times in succession, is always worse than the current solution S_i, so S_i can be regarded as the optimum solution. Here, set $N = 300$. For the simulated annealing algorithm, there is one type of random searching algorithm, so this feature will lead to unpredictable results of the newly generated solution S_{i+1}. Therefore, the iterative numbers of FLS-CSAA and VUAFLS-CSAA are random. For this reason, to fully test the performance of CSAA, FLS-CSAA and VUAFLS-CSAA, we have performed FLS-CSAA and VUAFLS-CSAA so many times. But due to the space limitations, only ten results are listed here for the illustration. By using CSAA, FLS-CSAA and VUAFLS-CSAA, the right sequence of the twenty pieces of the image can be obtained, as shown in TABLE 2.

TABLE 2. The correct sequence of the image.

The correct sequence of the image
5 ↔ 9 ↔ 12 ↔ 7 ↔ 10 ↔ 16 ↔ 19 ↔ 2 ↔ 11 ↔ 3 ↔ 8 ↔ 15 ↔ 4 ↔ 13 ↔ 20 ↔ 18 ↔ 1 ↔ 14 ↔ 17 ↔ 6

But the ten experimental data shown in TABLE 3 illustrate that the efficiency of CSAA, FLS-CSAA and VUAFLS-CSAA is largely different. Under the premise of the successful reassembling of the disordered pieces of the image "Lenna" to a holonomic image, only the comparison of iterations of CSAA, FLS-CSAA and VUAFLS-CSAA is listed here because the values of the objective function C_i are the same when the disordered pieces of the image "Lenna" can be restored successfully.

TABLE 3. The comparison of iterations(times).

	CSAA	FLS-CSAA	VUAFLS-CSAA
1	6891	2085	1034
2	6891	2527	1165
3	6891	1643	1169
4	6891	1877	1022
5	6891	2714	1289
6	6891	1975	896
7	6891	1700	1219
8	6891	2527	643
9	6891	1689	673
10	6891	1590	837
Average value	**6891**	**2032.7**	**994.7**

B. ALGORITHM EVALUATION USING TSP

TSP [36] is a classic combinatorial optimization problem. It belongs to the class of NP-complete problems that are difficult to solve according to computational complexity theory [37]. TSP is always used to test the performance of a newly developed optimization algorithm. A salesman departs from a city, and then he visits all cities once and only once, finally returning to the city from which he departed. TSP can be formulated as follows:

$$Minimize\ D = \sum_{m\in\Omega}\sum_{n\in\Omega} \lambda_{mn} d_{mn}$$

where Ω represents the set of cities. D is the predefined objective function C_i, which needs to be minimized, representing the total distance that the salesman travels. λ_{mn} is the decision variable, $\lambda_{mn} = \{0, 1\}, m, n \in \Omega$. $\lambda_{mn} = 1$ means that the salesman travels from city m to city n directly, or $\lambda_{mn} = 0$. To ensure the legal solutions can be obtained, the λ_{mn} is subject to qualifications such as

$$\sum_{m\in\Omega} \lambda_{mn} = 1, n \in \Omega, \quad \sum_{n\in\Omega} \lambda_{mn} = 1, m \in \Omega$$

λ_{mn} forms a Hamiltonian cycle. d_{mn} represents the distance between the cities m and n.

TABLE 4. The thirty four provincial capital cities of China.

	1	2	3	4
A	Hefei	Beijing	Chongqing	Fuzhou
B	Guangzhou	Nanning	Guiyang	Haikou
C	Harbin	Zhengzhou	Hong kong	Wuhan
D	Huhhot	Nanjing	Nanchang	Changchun
E	Macau	Yinchuan	Xining	Xian
F	Shanghai	Taiyuan	Chengdu	Taibei
G	Lhasa	Urumqi	Kunming	Hangzhou
H	Lanzhou	Shijiazhuang	Changsha	Shenyang
I	Jinan	Tianjin		

In this example, the solution S_i is one group of sequences that the salesman travels. The shortest total distance, objective function C_i, for the salesman travelling around thirty-four Chinese provincial capital cities, listed in TABLE 4, is calculated. And TABLE 5 shows the data of the thirty-four provincial capital cities represented with longitude and latitude values. The elements belonging to the two tables

TABLE 5. The longitude and latitude value of the thirty four cities.

	1	2	3	4
A	117.18,31.51	116.28,39.54	106.32,29.32	119.18,26.05
B	113.15,23.08	108.20,22.48	106.42,26.35	110.20,20.02
C	126.41,45.45	113.42,34.48	114.10,22.18	114.21,30.37
D	111.48,40.49	118.50,32.02	115.52,28.41	125.19,43.52
E	113.35,22.14	106.16,38.20	101.45,36.38	108.54,34.16
F	121.29,31.14	112.34,37.52	104.05,30.39	121.31,25.03
G	090.08,29.39	087.36,43.48	102.41,25.00	120.09,30.14
H	103.49,36.03	114.28,38.02	113.00,28.11	123.24,41.50
I	117.00,36.38	117.11,39.09		

are the positional correspondence. That is, the city name and the latitude and longitude value adhere to the correspondence between the two tables. For example, the value of the longitude and latitude of the city named "Beijing" in the "A2" position of TABLE 4 is 116.28, 39.54, which is listed in the "A2" position of TABLE 5.

In this example, the coefficients are set as follows. For CSAA, the initial temperature T_c is also 1. And assume the only coefficient ΔT to be 0.995 too for controlling the annealing process. Due to the condition of stopping criterion $T_c \leq 0.1^{20}$, this is an algorithm with a fixed iterative number of 9188 times. As in the case of the image restoration example, by using FLS-CSAA and VUAFLS-CSAA, the consecutive reheating times N should also be considered. Here, set $N = 300$. In addition, the iterative numbers of FLS-CSAA and VUAFLS-CSAA of this TSP example are random. Therefore, each time the algorithm is implemented, the results may differ. Confined to the length of this paper, the ten calculation results for the comparison of the iteration times are listed in TABLE 6.

TABLE 6. The comparison of iterations(Times) of TSP.

	CSAA	FLS-CSAA	VUAFLS-CSAA
1	9188	5959	1279
2	9188	3808	923
3	9188	6141	1394
4	9188	6469	1289
5	9188	1529	877
6	9188	1671	1272
7	9188	7091	1481
8	9188	2241	1318
9	9188	2000	1218
10	9188	3965	1763
Average value	**9188**	**4087.4**	**1281.4**

TABLE 7 shows the comparison of the objective function, that is, the total travelled distance of the salesman.

The following three figures (Fig.11.-Fig.13.) illustrate the simulated travelling paths of the salesman using CSAA, FLS-CSAA and VUAFLS-CSAA, respectively.

The experimental result of TSP indicates that, compared with CSAA, FLS-CSAA and VUAFLS-CSAA can greatly improve the efficiency of CSAA by tremendously shortening the iteration process under the precondition of achieving better objective functions.

TABLE 7. The comparison of objective function(Km) of TSP.

	CSAA	FLS-CSAA	VUAFLS-CSAA
1	16232	**16488**	16433
2	16569	16059	15743
3	17179	16869	16240
4	**16353**	16012	16102
5	16484	16836	16283
6	17215	16479	15837
7	16803	15945	15810
8	15684	15998	**16054**
9	16304	16531	16488
10	15756	16756	15808
Average value	**16457.9**	**16397**	**16079.8**

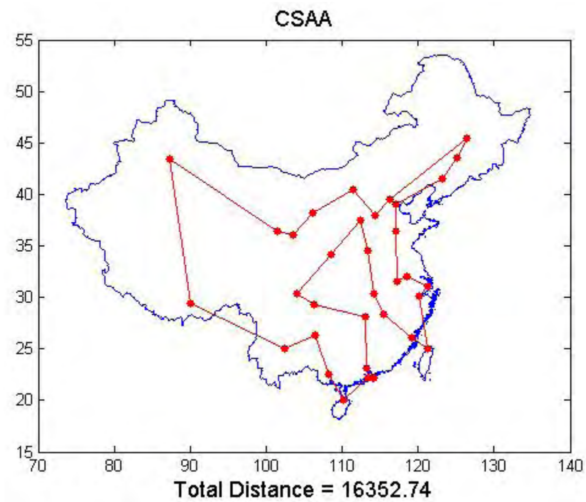

FIGURE 11. The forth experiment of CSAA(corresponding to the forth element of the first column in Tab 7).

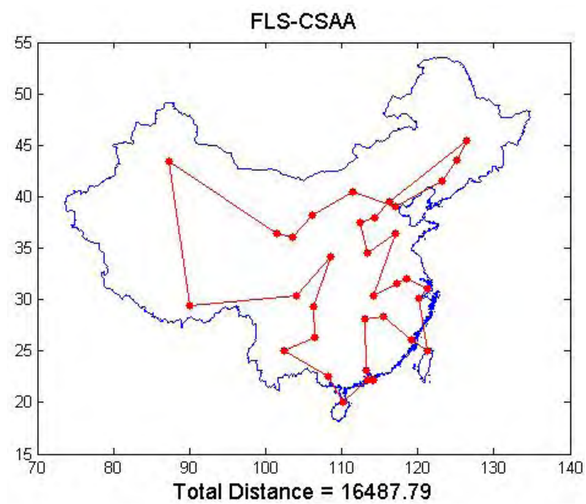

FIGURE 12. The first experiment of FLS-CSAA(corresponding to the first element of the second column in Tab 7).

When analyzing the results of the two examples, we can easily see that compared to CSAA, there is a sudden drop in the iteration count of the optimization process using FLS-CSAA and VUAFLS-CSAA. Given this, it can be

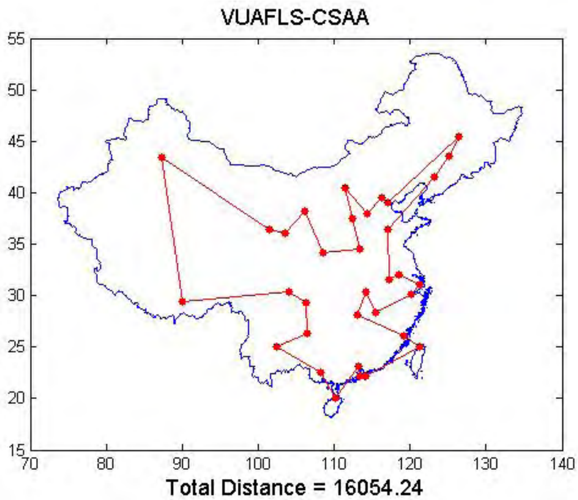

FIGURE 13. The eighth experiment of VUAFLS-CSAA(corresponding to the eighth element of the third column in Tab 7).

easily found that the CSAA search combined with FLC and VUAFLS is more effective compared to CSAA.

V. CONCLUSION

In this paper, we have investigated a novel approach to adjusting the annealing temperature of CSAA. By using FLS and VUAFLS, the reheating mechanism and fast cooling mechanism are introduced to improve the search performance of CSAA. In CSAA, there have been problems where a higher temperature allows a higher chance of transition into a worse solution. Conversely, the chance of uphill transition will decrease. To address the problem, we have developed the FLS and VUAFLS, which are applied to the annealing temperature control. By doing so, the annealing temperature's monotonical decrease is resolved, so algorithm efficiency is much improved. In order to verify the effectiveness of the novel approach, this paper examines two examples and compares the results with CSAA. As a result, the experiments reveal that FLC-CSAA and VUAFLS-CSAA show better results than CSAA. In particular, the outstanding properties of VUAFLS are shown through a comparison of the optimization iteration count. Because of the many advantages of FLS, the novel method presented in this paper can generally be applied to other applications on the simulated annealing algorithm. Consequently, compared to the CSAA, the two novel algorithms and especially VUAFLS-CSAA could significantly expand the scope of the application of the simulated annealing algorithm in the future.

REFERENCES

[1] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
[2] L. P. Behnck, D. Doering, C. E. Pereira, and A. Rettberg, "A modified simulated annealing algorithm for SUAVs path planning," *IFAC PapersOnLine*, vol. 48, no. 10, pp. 63–68, 2015.
[3] S. M. Mousavi and R. Tavakkoli-Moghaddam, "A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain," *J. Manuf. Syst.*, vol. 32, no. 2, pp. 335–347, 2013.
[4] A. Alexandridis and E. Chondrodima, "A medical diagnostic tool based on radial basis function classifiers and evolutionary simulated annealing," *J. Biomed. Inform.*, vol. 49, pp. 61–72, Jun. 2014.
[5] R.-E. Precup, R.-C. David, M. E. Petriu, M.-B. Radac, and S. Preitl, "Fuzzy control systems with reduced parametric sensitivity based on simulated annealing," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 3049–3061, Aug. 2012.
[6] A. M. Arostegui, Jr, N. S. Kadipasaoglu, and M. B. Khumawala, "An empirical comparison of tabu search, simulated annealing, and genetic algorithms for facilities location problems," *Int. J. Prod. Econ.*, vol. 103, no. 2, pp. 742–754, 2006.
[7] N. Metropolis, W. A. Rosenbluth, N. M. Rosenbluth, H. A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chem. Phys.*, vol. 21, no. 2, pp. 1087–1092, 1953.
[8] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*. Cambridge, MA, USA: MIT Press, 1988.
[9] I. O. Bohachevsky, M. E. Johnson, and M. L. Stein, "Generalized simulated annealing for function optimization," *Technometrics*, vol. 28, no. 3, pp. 209–217, 1986.
[10] L. Ingber, "Simulated annealing: Practice versus theory," *Math. Comput. Model.*, vol. 18, no. 11, pp. 29–57, 1993.
[11] H. Yu, H. Fang, P. Yao, and Y. Yuan, "A combined genetic algorithm/simulated annealing algorithm for large scale system energy integration," *Comput. Chem. Eng.*, vol. 24, no. 8, pp. 2023–2035, 2000.
[12] K. Hasani, A. S. Kravchenko, and F. Werner, "Simulated annealing and genetic algorithms for the two-machine scheduling problem with a single server," *Int. J. Prod. Res.*, vol. 52, no. 13, pp. 3778–3792, 2014.
[13] D. Sarkar and J. M. Modak, "ANNSA: A hybrid artificial neural network/simulated annealing algorithm for optimal control problems," *Chem. Eng. Sci.*, vol. 58, no. 14, pp. 3131–3142, 2003.
[14] M. Manoochehri and F. Kolahan, "Integration of artificial neural network and simulated annealing algorithm to optimize deep drawing process," *Int. J. Adv. Manuf. Technol.*, vol. 73, no. 2, pp. 241–249, 2014.
[15] A. Jamili, M. A. Shafia, and R. Tavakkoli-Moghaddam, "A hybrid algorithm based on particle swarm optimization and simulated annealing for a periodic job shop scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 54, no. 2, pp. 309–322, 2011.
[16] J. Geng, M.-W. Li, Z.-H. Dong, and Y.-S. Liao, "Port throughput forecasting by MARS-*R*SVR with chaotic simulated annealing particle swarm optimization algorithm," *Neurocomputing*, vol. 147, no. 2, pp. 239–250, 2015.
[17] M. D. Vose, *The Simple Genetic Algorithm: Foundations and Theory*. Cambridge, MA, USA: MIT Press, 1999.
[18] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1656–1671, Dec. 2013.
[19] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE ICNN*, vol. 4. Nov./Dec. 1995, pp. 1942–1948.
[20] M. H. Hassoun, "Fundamentals of artificial neural networks," *Proc. IEEE*, vol. 84, no. 6, p. 906, Jun. 1996.
[21] L. A. Zadeh, "Fuzzy sets," in *Fuzzy Sets, Fuzzy Logic, & Fuzzy Systems*. Singapore, World Scientific, 1996, pp. 394–432.
[22] S. Chakravarty and P. K. Dash, "A PSO based integrated functional link net and interval type-2 fuzzy logic system for predicting stock market indices," *Appl. Soft Comput.*, vol. 12, no. 2, pp. 931–941, 2012.
[23] G. Antonelli, S. Chiaverini, and G. Fusco, "A fuzzy-logic-based approach for mobile robot path tracking," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 2, pp. 211–221, Apr. 2007.
[24] P. Gader, J. M. Keller, and J. Cai, "A fuzzy logic system for the detection and recognition of handwritten street numbers," *IEEE Trans. Fuzzy Syst.*, vol. 3, no. 1, pp. 83–95, Feb. 1995.
[25] H. Li, "To see the success of fuzzy logic from mathematical essence of fuzzy control—On the paradoxical success of fuzzy logic," *Fuzzy Syst. Math.*, no. 4, pp. 1–14, 1995.
[26] H. X. Li, "The essence of fuzzy control and a kind of fine fuzzy controller," *Control Theory Appl.*, no. 6, pp. 868–872, 1997.
[27] H. Li, "Interpolation mechanism of fuzzy control," *Sci. China Technol. Sci.*, vol. 41, no. 2, pp. 312–320, 1998.
[28] H. Li, "Adaptive fuzzy controllers based on variable universe," *Sci. China Technol. Sci.*, vol. 42, no. 3, pp. 312–320, 1998.
[29] H. Li, M. Zhihong, and W. Jiayin, "Variable universe adaptive fuzzy control on the quadruple inverted pendulum," *Sci. China Technol. Sci.*, vol. 45, no. 2, pp. 213–224, 2002.

[30] W. Jing, and Z. Wei-Wei, "Chaos control via variable universe fuzzy theory in auto gauge control system," *Acta Phys. Sinica*, vol. 60, no. 2, p. 010511, 2011.
[31] J. J. Ibarrola, M. Pinzolas, and J. M. Cano, "A neurofuzzy scheme to on-line identification in an adaptive-predictive control," *Neural Comput. Appl.*, vol. 15, no. 2, pp. 41–48, 2006.
[32] L. Qin, J. Hu, H. Li, and W. Chen, "Fuzzy logic controllers for specialty vehicles using a combination of phase plane analysis and variable universe approach," *IEEE Access*, vol. 5, pp. 1579–1588, 2017.
[33] L. Hongxing, W. Jiayin, G. Yundong, and F. Yanbin, "Hardware implementation of the quadruple inverted pendulum with single motor," *Prog. Natural Sci. Mater. Int.*, vol. 14, no. 2, pp. 822–827, 2004.
[34] L. X. Wang, *A Course in Fuzzy Systems and Control*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
[35] E. Bonomi and J.-L. Lutton, "The N-city travelling salesman problem: Statistical mechanics and the metropolis algorithm," *SIAM Rev.*, vol. 26, no. 4, pp. 551–568, 1984.
[36] G. Reinelt, *The Traveling Salesman: Computational Solutions for TSP Applications*. Heidelberg, Germany: Springer-Verlag, 1994.
[37] C. H. Papadimitriou, "The Euclidean travelling salesman problem is NP-complete," *Theor. Comput. Sci.*, vol. 4, no. 2, pp. 237–244, 1977.

LISHU QIN received the M.S. degree in control engineering and theory from the Inner Mongolia University of Science and Technology, Baotou, China, in 2004. He is currently pursuing the Ph.D. degree in control engineering and theory from the Dalian University of Technology, Dalian, China. He is currently a Teacher with the College of Mechanical Engineering, Dalian University, Dalian. His research interests include intelligent control, robotics, and embedded systems.

JIANWEI WANG received the B.E. and M.Sc. degrees from the Department of Mechanical Engineering, Liaoning Technical University, China, in 1996 and 1999 respectively, and the Ph.D. degree from the Dalian University of Technology, Dalian, China, in 2009. Since then, he has been with the College of Mechanical Engineering, Dalian University, China, and is currently an Associate Professor. His current research interests include CAD/CAM, intelligent CAD, and conceptual design.

HONGXING LI received the degree from the Department of Mathematics, Nankai University, Tianjin, China, the degree from the Department of Mathematics, Beijing Normal University, Beijing, China, and the Ph.D. degree in engineering. He is currently a Professor and a Doctoral Tutor with the School of Control Science and Engineering, Dalian University of Technology, Dalian, China. His research interests include applied mathematics, control theory and engineering, pattern recognition, and artificial intelligence.

YI SUN received the bachelor's degree from Dalian University, China, in 2017. In college, he participated in Mathematical Modeling Contest actively and carried off many prizes. Currently, he is with Nanjing Golden Dragon Bus Company, Nanjing, China. He was involved in vehicle control unit and battery management system. His research interest areas include algorithm development, intelligent control, machine vision, and image processing.

SHUYANG LI received the M.S. degree from the Department of Mathematics, Liaoning Normal University, Liaoning, China, in 2002. She is currently pursuing the Ph.D. degree in control engineering and theory from the Dalian University of Technology, Dalian, China. She is currently a Teacher with the Department of Basic Science, Dalian Naval Academy. Her research interests include fuzzy systems and fuzzy decision making.