A New Particle Swarm Optimization Algorithm for Outlier Detection: Industrial Data Clustering in Wire Arc Additive Manufacturing

Jingzhong Fang, Zidong Wang, Weibo Liu, Stanislao Lauria, Nianyin Zeng, Camilo Prieto, Fredrik Sikström, and Xiaohui Liu

***Abstract*—In this paper, a novel outlier detection method is proposed for industrial data analysis based on the fuzzy C-means (FCM) algorithm. An adaptive switching randomly perturbed particle swarm optimization algorithm (ASRPPSO) is put forward to optimize the initial cluster centroids of the FCM algorithm. The superiority of the proposed ASRPPSO is demonstrated over five existing PSO algorithms on a series of benchmark functions. To illustrate its application potential, the proposed ASRPPSO-based FCM algorithm is exploited in the outlier detection problem for analyzing the real-world industrial data collected from a wire arc additive manufacturing pilot line in Sweden. Experimental results demonstrate that the proposed ASRPPSO-based FCM algorithm outperforms the standard FCM algorithm in detecting outliers of real-world industrial data.**

***Note to Practitioners*—Electric arc (which is governed by the current and arc voltage) plays a significant role in monitoring the operating status of the wire arc additive manufacturing (WAAM) process. The nominal periodic current and voltage may occasionally change abruptly due to anomalies (such as arc instability, unstable metal transfer, geometrical deviations, and surface contaminations), which would affect the quality of the fabricated component. This paper focuses on detecting possible anomalies by analyzing the current and voltage during the WAAM process. A novel clustering-based outlier detection method is proposed for anomaly detection where abnormal and normal instances are categorized into two separate clusters. A new particle swarm optimization algorithm is put forward to optimize the initial cluster centroid so as to improve the detection accuracy. The proposed outlier detection method is applied to real-world data collected from a WAAM pilot line for detecting abnormal instances. Experimental results demonstrate the effectiveness of the proposed outlier detection method. The proposed outlier detection method can be applied to other industrial applications including electrical engineering, mechanical engineering and medical engineering. In the future, we aim to develop an online outlier detection system based on the proposed method for real-time for anomaly detection and defect prediction.**

***Index Terms*—Industrial data analysis, outlier detection, fuzzy C-means, particle swarm optimization, wire arc additive manufacturing.**

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under Grant 820776 (INTEGRADDE), the National Natural Science Foundation of China under Grants 61933007 and 62073271, the Fundamental Research Funds for the Central Universities of China under Grant 20720220076, the Royal Society of the UK, and the Alexander von Humboldt Foundation of Germany. (*Corresponding author: Nianyin Zeng.*)

Jingzhong Fang, Zidong Wang, Weibo Liu, Stanislao Lauria and Xiaohui Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom. (email: Zidong.Wang@brunel.ac.uk)

Nianyin Zeng is with the Department of Instrumental and Electrical Engineering, Xiamen University, Fujian 361005, China. (email: zny@xmu.edu.cn)

Camilo Prieto is with the AIMEN Technology Centre, E36418 - O Porriño – Pontevedra - Spain. (email: camilo.prieto@aimen.es)

Fredrik Sikström is with the Department of Engineering Science, University West, 461-32 Trollhättan, Sweden. (email: fredrik.sikstrom@hv.se)

I. Introduction

Additive manufacturing (AM) is a disruptive technology in industrial manufacturing, which has attracted an ever-increasing research interest during the past few years. During the AM process, the metal material required by the production specification is deposited to the substrate layer by layer under the control of a computer [8]. Compared with some traditional subtractive manufacturing technologies, the AM technology exhibits better design flexibility and produces less waste. Thanks to its strong abilities in fabricating components with complex geometries, the AM technology has been successfully applied to a variety of fields such as electrical engineering, healthcare and transportation [9], [14].

To meet the requirement of fabricating components with complex structures at fine resolutions, a large number of AM methods have been developed, e.g., selective laser sintering, direct energy deposition (DED), liquid binding in three-dimensional printing, contour crafting and laminated object manufacturing [26]. Among existing AM methods, the DED method is a competitive one which uses high-power energy sources (including laser beam, electron beam, and electric arc) to deposit the metal powder or feedstock wire into the substrate layer by layer without the requirement of a strict seal structure [8].

Wire arc AM (WAAM) is a wire-based DED method with relatively high deposition efficiency. Compared with other AM methods, the WAAM has demonstrated significant advantages in material loss and cost savings [31]. In WAAM, the quality of the fabricated component is highly dependent on the operating status (manipulator- and feedstock feeding accuracy, shielding gas flow, metal surface contaminations, heat- and metal transfer, heat accumulation, and part distortion). The electric arc used as a heat source is primarily governed by the total current and arc voltage. These quantities are straight forward to monitor, and they directly convey vital information about the operating status. During processing, the nominal periodic current and voltage may occasionally change abruptly because of anomalies in the conditions such as, arc instability, unstable metal transfer, geometrical deviations, and surface contaminations. All such indications should be detected and possibly classified since they could be directly influencing the quality of the fabricated component.

Reaching a sufficient detection performance is of high industrial relevance since it will make quality control more efficient. If the data is accessible as a post process batch, it can be used to guide and optimize post process inspection. If the process change point detection can be implemented in real time, it can be used to initiate either a controlled stop of the process to directly enable some rectifying actions or, even better, enable an automatic corrective action (closed loop control).

Serving as a popular data analysis method, outlier detection plays an important role in identifying abnormal instances [10]. In the past few decades, a lot of outlier detection methods have been proposed [1], [13], [40], [45], [56]. For example, machine-learning-based outlier detection methods have been employed in [40] to analyze the semiconductor manufacturing etching data. A new outlier detection method has been developed in [45] for process monitoring based on the Gaussian process method. In this context, it becomes natural to apply outlier detection methods to analyze the current and voltage of the welding equipment in WAAM, where the instances with sudden change are treated as outliers.

As a powerful family of outlier detection methods, the clustering-based outlier detection methods are utilized to identify abnormal instances according to the corresponding clustering results. Compared with existing clustering algorithms including the density-based spatial clustering of applications with noise (DBSCAN) algorithm and the K-means algorithm, the fuzzy C-means (FCM) algorithm has the advantages of easy implementation and high efficiency, which has been successfully applied to a large number of real-world applications [44], [46], [47]. Nevertheless, as a distance-based clustering algorithm, the clustering performance of the FCM algorithm is highly dependent on the initial location of the cluster centroid [43]. Selecting an optimal set of initial cluster centroids seems to be an effective way to guarantee the performance of the FCM algorithm.

It is worth mentioning that evolutionary computation (EC) has been widely used to solve various optimization problems [39], [50]–[52]. Among the EC algorithms, the particle swarm optimization (PSO) algorithm is a population-based one, which is inspired by the mimics of social interactions, e.g., birds flocking and fish schooling [55], [57]. In comparison with some existing EC algorithms, the PSO algorithm has the following three advantages: 1) the number of parameters required to be adjusted is relatively small; 2) the convergence rate of the PSO algorithm is relatively fast; and 3) the implementation of the PSO algorithm is simple [5], [41]. Owing to the technical merits of the PSO algorithm, a seemingly reasonable idea is to adopt the PSO algorithm to optimize the initial locations of the cluster centroids with the purpose of improving the clustering performance of the FCM algorithm.

Despite their wide applicability, most existing population-based EC algorithms suffer from premature convergence, and this is particularly true when dealing with complex and large-scale optimization problems [19]. During the past few decades, a great many PSO variants have been proposed to improve the convergence rate and the search ability of the optimizer, which can be roughly categorized into four groups: 1) adjusting control parameters; 2) developing novel velocity updating strategies; 3) designing new topological structures; and 4) hybridizing with other EC algorithms [6], [15], [21], [32], [37], [38], [42], [53], [54]. For instance, the PSO algorithm with a linear decreasing inertia weight (PSO-LIDIW) has been proposed in [37], [38]. A PSO algorithm with time-varying acceleration coefficients (PSO-TVAC) has been introduced in [32]. An adaptive weighted PSO algorithm has been put forward in [21], where an adaptive weighting strategy (AWU) has been designed to control the acceleration coefficients, which could significantly enhance the convergence rate of the PSO algorithm.

A switching PSO (SPSO) algorithm has been put forward in [42] by employing a switching strategy to update the velocity of the particles, where the switching strategy divides the evolutionary process into four evolutionary states (i.e., convergence, exploitation, exploration and jumping-out). By using the switching strategy, the SPSO algorithm has shown relatively fast convergence rate. Nevertheless, the deployment of the switching strategy as well as the AWU could not solve the premature convergence problem.

In the literature, some popular PSO variants have been proposed by embedding random perturbations (e.g., time delays and noises) in the PSO algorithm with hope to alleviate the premature convergence problem [21], [54]. In [54], a switching delayed PSO (SDPSO) algorithm has been developed, where time delays are embedded in the velocity updating equation of the SPSO algorithm. Compared with the SPSO algorithm, the SDPSO algorithm exhibits stronger search ability especially for multimodal optimization problems. It should be noticed that the employment of the perturbation has proven to be another effective way to help particles jump out of the local optima [2], [16]. In this situation, it is reasonable to adaptively embed the random perturbation into the PSO algorithm based on the switching strategy to alter the dynamical behavior of the particles and expand the search range of the optimizer.

Motivated by the above discussions, the purpose of this paper is to develop an adaptive switching randomly perturbed particle swarm optimization (ASRPPSO) algorithm so as to automatically choose an optimal set of initial locations of the cluster centroids of the FCM algorithm. Specifically, a distance-based weighting strategy is designed to adaptively control the acceleration coefficients. The switching strategy is employed to accelerate the searching process and balance the global and local searches. In addition, Gaussian white noises are embedded in the velocity updating equation to randomly alter the system dynamics of the optimizer, which could expand the search space and alleviate the premature convergence problem. The proposed ASRPPSO-based FCM algorithm is applied to detect outliers in real-world data.

The main contributions of this paper can be summarized in the following three aspects:

1) a novel ASRPPSO is proposed where an AWU strategy is designed to adaptively adjust the acceleration coef-

ficients, and the Gaussian white noises are adaptively embedded into the velocity updating equation based on the switching strategy;
2) an ASRPPSO-based FCM algorithm is developed where the ASRPPSO is employed for selecting the optimal locations of the initial cluster centroids of the FCM algorithm; and
3) the developed ASRPPSO-based FCM algorithm is applied to outlier detection of the real-world industrial data collected from a WAAM pilot line. Experimental results demonstrate the effectiveness of the proposed outlier detection method.

The remaining parts of this paper are organized as follows. The background of AM and outlier detection are discussed in Section II, where the description of the utilized data sets is also presented. In Section III, the proposed ASRPPSO and the ASRPPSO-based FCM algorithm are introduced. Experimental results of the ASRPPSO and the ASRPPSO-based FCM algorithm are presented in Section IV. Finally, conclusions and discussions on relevant future work are presented in Section V.

II. Background

A. Additive Manufacturing

AM, also known as 3D printing, has been recognized as a breakthrough technology which shows great application potentials in industrial machinery, assembly processes, and supply chains [7]. An AM system consists of three parts: a motion system, heat source and feedstock. Due to the advantages of high deposition rate, high material utilization, low cost and environmental friendliness, the WAAM has become a popular AM method, which uses the electric arc as the heat source and the wire as the feedstock. Depending on the heat source, the heat and metal transfer in WAAM can be generally divided into three categories of equipment including the Gas Tungsten Arc Welding, the Gas Metal Arc Welding, and the Plasma Arc Welding [48].

In fact, there are several key factors affecting the performance of WAAM, e.g., the programming strategies, manipulator- and feedstock feeding accuracy, shielding gas flow, metal surface contaminations, heat- and metal transfer, heat accumulation, and part distortion [49]. In recent years, a variety of advanced techniques (which focuses on the aforementioned factors) have been put forward to improve the performance of the WAAM. For instance, in [28], an improved heat transfer and fluid flow model has been introduced in WAAM, which could determine the parameters that influence the microstructure, properties, and defect formation of the component. In [24], a WAAM modelling strategy has been developed based on a novel heat source model, which has shown high accuracy in measuring distortions.

Note that the quality of AM products is highly dependent on the welding equipment used. To monitor the WAAM process, outlier detection techniques are widely adopted to analyze the sensor data (e.g., current and voltage) of the welding machine used with the purpose of detecting abnormal points. In general, the abnormal points indicate the sudden change of the process, which could bring negative influence on the manufacturing process. In this case, it is of practical importance to use outlier detection techniques for online monitoring of the WAAM process.

So far, a large number of outlier detection methods have been introduced for data analysis in WAAM [4], [12], [17], [33]. For instance, a non-contact in-situ 3D laser profilometer inspection system has been presented in [12] to monitor the visual surface defects. A modular anomaly detector has been put forward in [33] for analyzing multivariate time-series data in the WAAM process. In [4], a convolutional neural network-based method has been proposed for real-time anomaly detection in WAAM. In [17], an image-based approach has been presented for defect detection in WAAM.

B. Description of the Data Sets

The utilized data is acquired through the process in a WAAM pilot line deployed in Sweden. In total, there are five data sets where each data set contains 98000 instances. In this paper, each data set represents an individual test. In the collected data sets, there are four variables which are “X_Value”, “WeldCurrent”, “WeldVoltage”, and “ComputerTime”. “X_Value” represents the time stamp. “WeldCurrent” and “WeldVoltage” denote the processing current and voltage, respectively. “Computer time” is the total time of the process. The details of the variables are summarized in Table I. The details of data pre-processing are presented in Section IV.

III. Methodology

The FCM algorithm is a competitive clustering algorithm, whose initial location of the cluster centroid is a decisive factor affecting the clustering results of the FCM algorithm. Clearly, it is highly desirable to seek the best location of the initial centroid for the best clustering performance, which gives rise to a rather challenging optimization problem that has received little research attention so far except the preliminary efforts made in [27], [35]. In fact, as one of the powerful optimization algorithms, the PSO algorithm is ideally suited in optimizing the initial location of the cluster centroid, see [27], [35] for more details.

In the PSO algorithm, the control parameters (i.e., the inertia weight and acceleration coefficients) are utilized to balance the global and local search. In recent years, some variant PSO algorithms (which modify the control parameters) have been proposed to balance the global and local search [32], [37], [38]. The PSO-LIDIW algorithm and the PSO-TVAC algorithm both demonstrate competitive performance in maintaining the balance between the global and local search compared with the original PSO algorithm.

It should be noticed that the solution accuracy of many PSO variants is improved with the sacrifice of the convergence rate [3]. With the purpose of adequately improving the convergence rate and the capability of finding the global best solution, the AWPSO algorithm has been put forward in [21], where a sigmoid-function-based AWU strategy has been proposed to adjust the acceleration coefficients. Specifically, the AWU strategy makes full use of the distance from each particle

TABLE I
DATA SETS DESCRIPTION

Variable	Description	Data Type	Unit
X_Value	The time stamp	Numerical	s
WeldCurrent	The current of the equipment	Numerical	A
WeldVoltage	The voltage of the equipment	Numerical	V
ComputerTime	The total time of the process	Numerical	s

towards its personal best (pbest) and global best position (gbest) at each step, which significantly improves the convergence rate of the optimizer. Unfortunately, the search ability of the AWPSO algorithm is not satisfactory when dealing with complex optimization problems.

Recently, some PSO algorithms have been proposed to alleviate premature convergence based on different switching strategies [21], [42], [54]. By using the switching strategy, the evolution process is divided into four evolutionary states (including the exploration, exploitation, convergence, and jumping-out states). The velocity is updated based on different updating strategies at each state, which offers the opportunity of improving the search ability and guaranteeing the convergence rate of the optimizer at the same time. More recently, adding noises to perturb the particle's movement has been proven to be an effective way to improve the search ability of each individual particle. In [21], the intensity-adjustable Gaussian white noise has been added in the velocity updating equation to randomly alter the acceleration constants in order to explore the search space thoroughly. Motivated by the above discussions, it becomes natural to embed the random noises to the PSO algorithm according to the evolutionary states to further alleviate the premature convergence problem and improve the search ability of the optimizer.

In this paper, a novel ASRPPSO is put forward where an AWU strategy is designed to adjust the acceleration coefficients. In the ASRPPSO, the switching strategy is utilized to improve the particle's search ability, and the Gaussian white noises are embedded in the velocity updating equation based on the switching strategy to help the particles escape from the local optima. In addition, an ASRPPSO-based FCM algorithm is proposed for outlier detection, where the ASRPPSO is adopted to choose an optimal set of initial locations of the cluster centroids in the FCM algorithm.

A. The ASRPPSO

1) Framework of the ASRPPSO: The updating equations of the proposed ASRPPSO in terms of velocity and position of the ith particle at the $(k+1)$th iteration are given as follows:

$$\begin{aligned} v_{i,k+1} &= w_k v_{i,k} + c_{1,k} r_1 \left(pbest_{i,k} - x_{i,k}\right) \\ &\quad + c_{2,k} r_2 \left(gbest_k - x_{i,k}\right) \\ &\quad + \alpha_{1,\xi_k} \delta_1 \left(pbest_{i,k} - x_{i,k}\right) \\ &\quad + \alpha_{2,\xi_k} \delta_2 \left(gbest_k - x_{i,k}\right) \\ x_{i,k+1} &= x_{i,k} + v_{i,k+1} \end{aligned} \tag{1}$$

where k represents the current iteration number; w_k denotes the inertia weight at the kth iteration; $c_{1,k}$ and $c_{2,k}$ are the acceleration coefficients; r_1 and r_2 represent random numbers selected within $[0, 1]$; $pbest_{i,k}$ represents the personal best position found by the ith particle itself at the kth iteration; $gbest_k$ represents the global best position of the entire swarm at the kth iteration; α_{1,ξ_k} and α_{2,ξ_k} are parameters which are used to adjust the Gaussian white noises according to the evolutionary state; and δ_1 and δ_2 represent two independent Gaussian white noises.

The procedure of the proposed ASRPPSO is presented in Algorithm 1.

Algorithm 1 The Procedure of the ASRPPSO

1. Initialize the parameters of the ASRPPSO including the population size P, inertia weight w_1, acceleration coefficients $c_{1,1}$, $c_{2,1}$, and the maximum velocity $V_{\max}$.
2. Set a swarm that has P particles.
3. Initialize the position $x_{i,1}$, the velocity $v_{i,1}$, and $pbest_{i,1}$ of each particle $(i = 1, 2, \ldots, P)$; and initialize $gbest_1$ of the swarm.
4. Calculate each particle's fitness value.
5. Update the $pbest_{i,k}$ of each particle and $gbest_k$ of the swarm.
6. Calculate E_f according to Eq. (5) and Eq. (6), and confirm the evolutionary state according to Eq. (7).
7. Update w_k, $c_{1,k}$ and $c_{2,k}$ of each particle based on Eq. (2), Eq. (3) and Eq. (4).
8. Update the velocity $v_{i,k}$ and the position $x_{i,k}$ of each particle based on Eq. (1).
9. Terminate the algorithm if the maximum iteration is reached or the fitness value reaches the threshold. If not, repeat Steps 4-8.

2) Inertia Weight: The inertia weight is an important parameter which is designed to adequately balance the global and local search. The inertia weight shows the ability of particles to inherit their previous velocities. In order to balance the global search and the local search, the inertia weight is adaptively altered by a linear decreasing inertia weight strategy which has been introduced in [37], [38]. In the proposed ASRPPSO, the updating equation of the inertia weight is shown as follows:

$$w = w_{\max} - (w_{\max} - w_{\min}) \times \frac{k}{K} \tag{2}$$

where $w_{\max}$ and $w_{\min}$ denote the maximal and minimal value of the inertia weight, respectively; k and K represent the current iteration number and the maximum iteration number, respectively; and the maximum and minimum inertia weights are set to be 0.9 and 0.4, respectively.

3) Acceleration Coefficients: According to [21], using an AWU function to demonstrate the relationship between the acceleration coefficients and the distances from the particle to its pbest and gbest is a good way to adaptively adjust acceleration coefficients, which can significantly improve the convergence rate. In this proposed ASRPPSO, a tanh-function-based AWU strategy is introduced to accelerate the movement of particles, which aims to improve the convergence rate.

Inspired by [21], the AWU function need to be monotonically increasing and bounding. The tanh function is a typical activation function of the neural networks, which perfectly fits the requirements mentioned above. In addition, the tanh function is differentiable, which can iteratively reflect the characteristics of the weight updating process. Hence, the tanh function seems to be an appropriate choice. The updating equations of the acceleration coefficients are demonstrated as follows:

$$c_{1,k} = \frac{-2b}{1+\exp\left(2a\left(pbest_{i,k} - x_{i,k} - m\right)\right)} + n \tag{3}$$

$$c_{2,k} = \frac{-2b}{1+\exp\left(2a\left(gbest_{k} - x_{i,k} - m\right)\right)} + n \tag{4}$$

where a and b are two parameters used to describe the curve, which represent the steepness and the peak value, respectively; m denotes the offset of the central point; n is a negative value; and $\exp(\cdot)$ is the natural exponential function.

It is worth mentioning that, in Eq. (3) and Eq. (4), a, b, m, and n are all constant values. Appropriate values of the parameters would effectively enhance the performance of the optimization algorithm. In the proposed ASRPPSO, the parameters are set by $a = -0.035$, $b = -0.275$, $m = 0$, and $n = 1.2$ based on the experimental experience.

4) Evolutionary States: In the ASRPPSO, the particle's velocity as well as position are adjusted based on the evolutionary states, which are identified by the evolutionary factor (calculated based on the mean distance from each particle i to other particles). The equation of mean distance d_i is shown as follows:

$$d_i = \frac{1}{S-1}\sum_{j=1}^{S}\sqrt{\sum_{r=1}^{D}\left(x_{i,r} - x_{j,r}\right)^2} \tag{5}$$

where S and D represent the swarm size and the dimension of the particle, respectively.

Denote d_g as the global best particle of d_i, the evolutionary factor E_f can thus be calculated by:

$$E_f = \frac{d_g - d_{\min}}{d_{\max} - d_{\min}} \tag{6}$$

where $d_{\max}$ and $d_{\min}$ are the maximum and minimum of d_i, respectively. It is worth mentioning that E_f belongs to $[0,1]$.

Based on the evolutionary factor, the four states can be classified as follows:

$$\xi_k = \begin{cases} 1, & 0.00 \le E_f \le 0.25 \\ 2, & 0.25 < E_f \le 0.50 \\ 3, & 0.50 < E_f \le 0.75 \\ 4, & 0.75 < E_f \le 1.00 \end{cases} \tag{7}$$

where $\xi_k = 1$ denotes the convergence state, $\xi_k = 2$ represents the exploitation state, $\xi_k = 3$ denotes the exploration state, and $\xi_k = 4$ represents the jumping-out state.

5) An Adaptive Weighted Velocity Updating Strategy: The novel velocity updating strategy can be explained based on four evolutionary states, which is summarized in Table II. The decision factors α_{1,ξ_k} and α_{2,ξ_k} are used for determining the value of the Gaussian white noise, which are dependent on evolutionary states; and k denotes the current iteration number.

TABLE II
PARAMETERS IN THE VELOCITY UPDATING STRATEGY OF THE ASRPPSO

Evolutionary State	ξ_k	α_{1,ξ_k}	α_{2,ξ_k}
Convergence	1	0	0
Exploitation	2	1	0.3
Exploration	3	0.3	1
Jumping-out	4	1	1

6) Gaussian White Noise: Inspired by [21], to improve the search ability of the optimizer by altering the system dynamics of the PSO algorithm, the Gaussian white noises δ_1 and δ_2 are added into the velocity updating equation to randomly perturb the movement of the particles. Note that the mean value and variance of δ_1 and δ_2 remain the same for the four evolutionary states.

B. The ASRPPSO-based FCM Algorithm

Due to the fact that the FCM algorithm's performance is highly dependent on the initial cluster centroids, the ASRPPSO is employed for optimally selecting the initial cluster centroids. As discussed before, the Gaussian white noise is embedded in the velocity updating model so that the particle's ability of getting rid of local optima is enhanced, which indicates that the probability of getting better cluster centroids would be improved. The procedure of the ASRPPSO-based clustering algorithm is described in Algorithm 2.

IV. EXPERIMENTAL RESULTS

A. The ASRPPSO

The performance of the developed ASRPPSO is evaluated on a series of benchmark functions. Here, 13 selected CEC basic benchmark functions are chosen for performance evaluation. The details of selected benchmark functions are presented in Table III and Table IV. The experimental results of the ASRPPSO are compared with the experimental results of some existing PSO algorithms. In this experiment, the swarm size and the dimension of the selected benchmark functions are set to be 30. For all chosen algorithms, the maximum iteration is set to be 10000. In order to avoid contingency, each experiment is repeated 30 times independently. In the experiment, the mean and variance of the Gaussian white noises are set to be 0.5 and 1, respectively. The CPU used in the experiment is Intel Core i7-10700K. The programming platform used in the experiment is MATLAB R2021a.

TABLE III
DETAILS OF SELECTED BENCHMARK FUNCTIONS

Function number	Function name	Search Range	Minimum	Threshold
$F_1(x)$	Sphere Function	$[-100, 100]$	0	0.1
$F_2(x)$	Schwefel 1.2 Function	$[-100, 100]$	0	0.1
$F_3(x)$	Schwefel 2.21 Function	$[-100, 100]$	0	0.1
$F_4(x)$	Schwefel 2.22 Function	$[-10, 10]$	0	0.1
$F_5(x)$	Rosenbrock Function	$[-30, 30]$	0	100
$F_6(x)$	Step Function	$[-100, 100]$	0	0.1
$F_7(x)$	Bent Cigar Function	$[-10, 10]$	0	100
$F_8(x)$	Rastrigin Function	$[-5.12, 5.12]$	0	50
$F_9(x)$	Zakharov Function	$[-5, 5]$	0	0.1
$F_{10}(x)$	Levy Function	$[-10, 10]$	0	0.1
$F_{11}(x)$	Ackley Function	$[-32, 32]$	0	0.1
$F_{12}(x)$	Griewank Function	$[-100, 100]$	0	0.1
$F_{13}(x)$	Sum of Different Powers Function	$[-100, 100]$	0	0.1

TABLE IV
DETAILS OF THE SELECTED BENCHMARK FUNCTIONS

Function number	Function name	Mathematical formula
$F_1(x)$	Sphere Function	$F_1(x) = \sum_{i=1}^{D} x_i^2$
$F_2(x)$	Schwefel 1.2 Function	$F_2(x) = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j\right)^2$
$F_3(x)$	Schwefel 2.21 Function	$F_3(x) = \max_i\{\lvert x_i\rvert, 1 \le i \le D\}$
$F_4(x)$	Schwefel 2.22 Function	$F_4(x) = \sum_{i=1}^{D} \lvert x_i\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$
$F_5(x)$	Rosenbrock Function	$F_5(x) = \sum_{i=1}^{D-1} \left(100(x_{i+1} - {x_i}^2)^2 + (x_i - 1)^2\right)$
$F_6(x)$	Step Function	$F_6(x) = \sum_{i=1}^{D} \left(\lfloor x_i + 0.5 \rfloor\right)^2$
$F_7(x)$	Bent Cigar Function	$F_7(x) = x_1^2 + 10^6 \sum_{i=2}^{D} x_i^2$
$F_8(x)$	Rastrigin Function	$F_8(x) = \sum_{i=1}^{D} \left(x_i^2 - 10\cos 2\pi x_i + 10\right)$
$F_9(x)$	Zakharov Function	$F_9(x) = \sum_{i=1}^{D} x_i^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^4$
$F_{10}(x)$	Levy Function	$F_{10}(x) = \sin^2 \pi w_1 + \sum_{i=1}^{D-1} (w_i - 1)^2 \left(1 + 10\sin^2(\pi w_i + 1)\right) + (w_D - 1)^2 \left(1 + \sin^2 2\pi w_D\right), \quad w_i = 1 + \frac{x_i - 1}{4}$
$F_{11}(x)$	Ackley Function	$F_{11}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi x_i\right) + 20 + e$
$F_{12}(x)$	Griewank Function	$F_{13}(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\frac{x_i}{\sqrt{i}} + 1$
$F_{13}(x)$	Sum of Different Powers Function	$F_{14}(x) = \sum_{i=1}^{D} \lvert x_i\rvert^{i+1}$

In the experiment, five PSO algorithms (including the basic PSO algorithm, the PSO-LIDIW algorithm [37], [38], the SPSO algorithm [42], the SDPSO algorithm [54], and the AWPSO algorithm [21] are selected for comparison. Experimental results of the adopted algorithms via thirteen selected benchmark functions are summarized. The convergence plots of the algorithms on each benchmark function are displayed in Figs. 1-13. The vertical coordinate and the horizontal coordinate of the convergence plots indicate the logarithm value of the mean fitness value and the generation number, respectively.

The statistical results (including minimum, mean and standard deviation) of the fitness values of the utilized algorithms on each benchmark function are listed in Table V and Table VI for performance evaluation. It is worth mentioning that the results of the PSO algorithms on selected unimodal functions are listed in Table V, and the results of the PSO algorithms on selected multimodal functions are listed in Table VI. The success ratio and the iteration number when the algorithm converges are listed in Table V and Table VI as well.

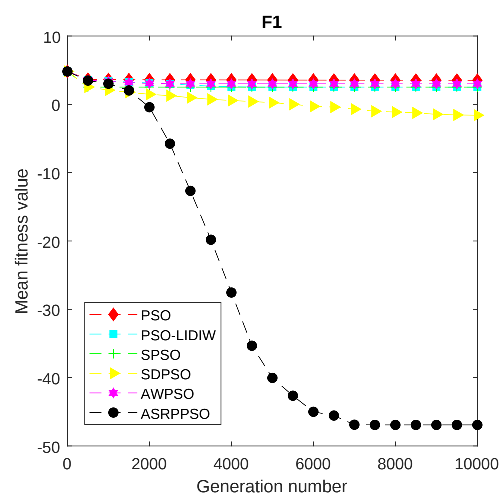


Fig. 1. Convergence plot for the Sphere Function $F_1(x)$

TABLE V
STATISTICAL RESULTS OF THE SELECTED PSO ALGORITHMS ON UNIMODAL FUNCTIONS

		PSO	PSO-LIDIW	SPSO	SDPSO	AWPSO	ASRPPSO
$F_1(x)$	Minimum	1.62×10^{3}	8.20×10^{-65}	3.42×10^{-197}	3.22×10^{-5}	1.69×10^{-79}	7.02×10^{-62}
	Mean	3.30×10^{3}	3.33×10^{2}	3.33×10^{2}	2.57×10^{-2}	1.00×10^{3}	1.20×10^{-47}
	Std. Dev.	3.05×10^{3}	1.83×10^{3}	1.83×10^{3}	7.58×10^{-2}	3.05×10^{3}	6.37×10^{-47}
	Ratio	0%	96.67%	96.67%	93.33%	90%	100%
	Con.num	10000	5142	580	6440	3770	1927
$F_3(x)$	Minimum	1.59×10^{1}	4.03×10^{-2}	1.58×10^{-6}	2.27	1.02×10^{-5}	5.09×10^{-4}
	Mean	1.92×10^{1}	1.25×10^{-1}	1.82×10^{-5}	3.63	1.42×10^{-4}	1.27×10^{-2}
	Std. Dev.	1.40	8.04×10^{-2}	2.53×10^{-5}	8.09×10^{-1}	2.34×10^{-4}	1.94×10^{-2}
	Ratio	0%	43.33%	100%	0%	100%	100%
	Con.num	10000	9853	3233	10000	5820	4153
$F_6(x)$	Minimum	1.81×10^{3}	0.00	0.00	0.00	0.00	0.00
	Mean	2.31×10^{3}	6.67×10^{2}	3.33×10^{-2}	2.33×10^{-1}	3.33×10^{2}	0.00
	Std. Dev.	2.48×10^{2}	2.54×10^{3}	1.83×10^{-1}	5.68×10^{-1}	1.83×10^{3}	0.00
	Ratio	0%	93.33%	96.67%	83.33%	96.67%	100%
	Con.num	10000	5537	1748	6906	3447	2166
$F_7(x)$	Minimum	1.66×10^{7}	2.36×10^{-4}	1.44×10^{-5}	4.42	6.06×10^{-5}	1.02×10^{-4}
	Mean	2.14×10^{7}	6.67×10^{6}	4.96×10^{1}	3.98×10^{3}	3.33×10^{6}	4.34×10^{1}
	Std. Dev.	2.06×10^{6}	2.54×10^{7}	5.05×10^{1}	1.34×10^{4}	1.83×10^{7}	5.04×10^{1}
	Ratio	0%	50%	53.33%	10%	53.33%	56.67%
	Con.num	10000	7617	4836	9749	6359	5531
$F_9(x)$	Minimum	1.61×10^{1}	9.31×10^{-15}	1.75×10^{-46}	1.64×10^{-1}	5.91×10^{-22}	2.80×10^{-18}
	Mean	5.74×10^{1}	1.05×10^{6}	1.05×10^{6}	4.82	3.92×10^{1}	2.08×10^{1}
	Std. Dev.	4.14×10^{1}	5.78×10^{6}	5.78×10^{6}	9.39	4.99×10^{1}	3.09×10^{1}
	Ratio	0%	33.33%	76.67%	0%	33.33%	60%
	Con.num	10000	8888	2988	10000	8010	5537
$F_{13}(x)$	Minimum	4.62×10^{-232}	0.00	0.00	0.00	0.00	0.00
	Mean	4.92×10^{-150}	0.00	0.00	0.00	0.00	0.00
	Std. Dev.	2.69×10^{-149}	0.00	0.00	0.00	0.00	0.00
	Ratio	100%	100%	100%	100%	100%	100%
	Con.num	3	2	2	2	2	2

Algorithm 2 The Procedure of the ASRPPSO-based FCM Algorithm

1. Initialize parameters of the ASRPPSO and the FCM clustering algorithm including the population size P, inertia weight w_1, acceleration coefficients $c_{1,1}$, $c_{2,1}$, and the maximum velocity $V_{\max}$.
2. Set a swarm that has P particles.
3. Initialize the position $x_{i,1}$, the velocity $v_{i,1}$, and $pbest_{i,1}$ of each particle i; and initialize $gbest_1$ of the swarm.
4. Getting the cluster centroids.
5. Calculate the fitness value of each particle.
6. Set the $pbest_{i,k}$ of each particle and the $gbest_k$ of the swarm.
7. Calculate E_f according to Eq. (5) and Eq. (6), and confirm the evolutionary state according to Eq. (7).
8. Update w_k, $c_{1,k}$ and $c_{2,k}$ of each particle based on Eq. (2), Eq. (3) and Eq. (4).
9. Update the velocity $v_{i,k}$ and the position $x_{i,k}$ of each particle based on Eq. (1).
10. Terminate the algorithm if the maximum iteration is reached. If not, repeat Steps 4-9.

1) Convergence Plot: It can be found from the figures that the proposed ASRPPSO shows better convergence performance than selected PSO algorithms. In Fig. 1, Figs. 5-7, Fig. 10 and Figs. 12-13, the ASRPPSO obtains the smallest mean fitness value. In Figs. 2-3, Figs. 8-9 and Fig. 11, the mean fitness value of the ASRPPSO is smaller than most of selected PSO algorithms. In Fig. 4, the differences between the mean fitness values of selected PSO algorithms are not obvious. To summarize, the proposed ASRPPSO exhibits satisfactory convergence performance.

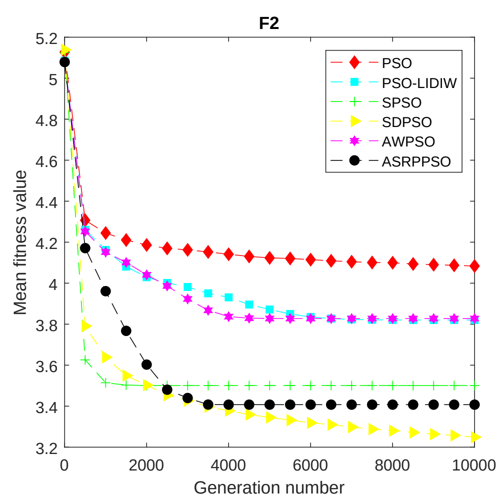


Fig. 2. Convergence plot for the Schwefel 1.2 Function $F_2(x)$

2) Statistical Analysis: The statistical results of selected PSO algorithms are presented in Table V and Table VI, which contain the minimum, the mean fitness value, and the standard deviation of the solutions. The minimum fitness value, the mean fitness value, and the standard deviation are the evaluation indices of the search ability of the algorithm. As the

TABLE VI
STATISTICAL RESULTS OF SELECTED PSO ALGORITHMS ON MULTIMODAL FUNCTIONS

		PSO	PSO-LIDIW	SPSO	SDPSO	AWPSO	ASRPPSO
$F_2(x)$	Minimum	5.24×10^{3}	2.55×10^{-3}	1.30×10^{-13}	2.21×10^{2}	5.63×10^{-12}	2.93×10^{-9}
	Mean	1.21×10^{4}	6.61×10^{3}	3.17×10^{3}	1.77×10^{3}	6.72×10^{3}	2.56×10^{3}
	Std. Dev.	5.40×10^{3}	6.95×10^{3}	5.33×10^{3}	3.17×10^{3}	8.17×10^{3}	4.28×10^{3}
	Ratio	0%	33.33%	66.67%	0%	40%	66.67%
	Con.num	10000	9770	5277	10000	8299	5987
$F_4(x)$	Minimum	1.73×10^{1}	2.34×10^{-40}	3.17×10^{-102}	9.41×10^{-4}	1.03×10^{-29}	6.70×10^{-18}
	Mean	3.80×10^{1}	2.07×10^{1}	1.30×10^{1}	6.39	2.60×10^{1}	1.60×10^{1}
	Std. Dev.	1.67×10^{1}	1.47×10^{1}	1.26×10^{1}	8.88	1.54×10^{1}	1.19×10^{1}
	Ratio	0%	13.33%	33.33%	50%	6.67%	20%
	Con.num	10000	9284	6755	8351	9527	8350
$F_5(x)$	Minimum	2.34×10^{5}	3.02×10^{-2}	3.89	2.31×10^{1}	2.48×10^{-2}	5.00×10^{-2}
	Mean	4.31×10^{5}	9.36×10^{3}	3.16×10^{3}	3.15×10^{3}	9.06×10^{3}	1.76×10^{2}
	Std. Dev.	1.10×10^{5}	2.74×10^{4}	1.64×10^{4}	1.64×10^{4}	2.74×10^{4}	5.57×10^{2}
	Ratio	0%	70%	86.67%	56.67%	80%	86.67%
	Con.num	15000	11145	2388	10800	7547	4132
$F_8(x)$	Minimum	1.78×10^{2}	1.49×10^{1}	4.97×10^{1}	3.00×10^{1}	1.59×10^{1}	8.95
	Mean	1.94×10^{2}	4.04×10^{1}	8.98×10^{1}	6.11×10^{1}	5.36×10^{1}	4.75×10^{1}
	Std. Dev.	1.08×10^{1}	2.24×10^{1}	2.55×10^{1}	2.20×10^{1}	2.58×10^{1}	2.11×10^{1}
	Ratio	0%	70%	3.33%	36.67%	46.67%	56.67%
	Con.num	10000	6446	9674	8107	6805	5372
$F_{10}(x)$	Minimum	5.29	1.50×10^{-32}	8.95×10^{-2}	1.00×10^{-4}	1.50×10^{-32}	1.50×10^{-32}
	Mean	7.22	6.65×10^{-2}	8.63	1.80	2.74×10^{-2}	8.90×10^{-13}
	Std. Dev.	8.01×10^{-1}	2.56×10^{-1}	7.89	2.95	1.50×10^{-1}	4.93×10^{-12}
	Ratio	0%	93.33%	3.33%	53.33%	96.67%	100%
	Con.num	10000	5175	9691	7667	3210	1958
$F_{11}(x)$	Minimum	9.42	6.22×10^{-15}	6.22×10^{-15}	1.78×10^{-3}	6.22×10^{-15}	6.22×10^{-15}
	Mean	1.09×10^{1}	8.82×10^{-15}	8.97×10^{-1}	3.99×10^{-2}	2.34	1.44×10^{-14}
	Std. Dev.	1.78	3.48×10^{-15}	9.60×10^{-1}	4.61×10^{-2}	5.32	4.68×10^{-15}
	Ratio	0%	100%	46.67%	90%	83.33%	100%
	Con.num	10000	5037	5487	7082	4281	2009
$F_{12}(x)$	Minimum	1.73×10^{1}	0.00	0.00	7.25×10^{-5}	0.00	0.00
	Mean	2.20×10^{1}	6.05	2.12×10^{-2}	5.83×10^{-2}	3.02	1.54×10^{-2}
	Std. Dev.	2.56	2.29×10^{1}	2.46×10^{-2}	7.20×10^{-2}	1.64×10^{1}	2.18×10^{-2}
	Ratio	0%	93.33%	96.67%	80%	96.67%	100%
	Con.num	10000	5435	756	7877	3191	1960

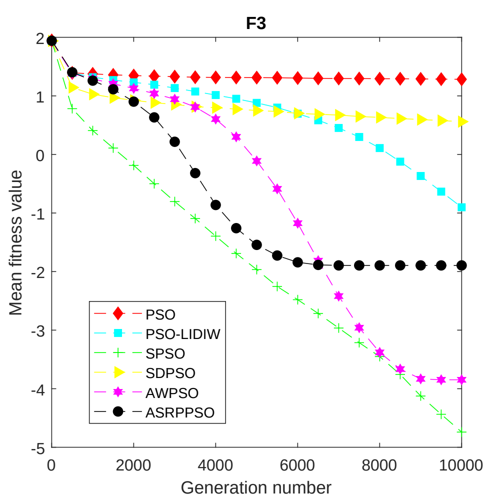


Fig. 3. Convergence plot for the Schwefel 2.21 Function $F_3(x)$

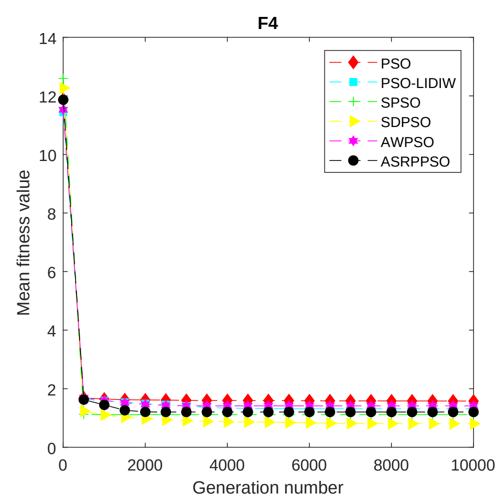


Fig. 4. Convergence plot for the Schwefel 2.22 Function $F_4(x)$

selected benchmark functions are all minimization problems, so the smaller the fitness value, the better the solution found by the particles.

In Table V and Table VI, the minimum fitness value of the ASRPPSO is the smallest among all that of the selected PSO algorithms on $F_6(x)$, $F_8(x)$ and $F_{10}(x)$-$F_{13}(x)$. On the rest of the benchmark functions, the ASRPPSO also shows competitive performance in terms of the minimum fitness value. Compared with other selected PSO algorithms, the ASRPPSO obtains the smallest mean fitness value on $F_1(x)$, $F_5(x)$-$F_7(x)$, $F_{10}(x)$ and $F_{12}(x)$-$F_{13}(x)$. Considering the standard deviation of the fitness value, the ASRPPSO obtains

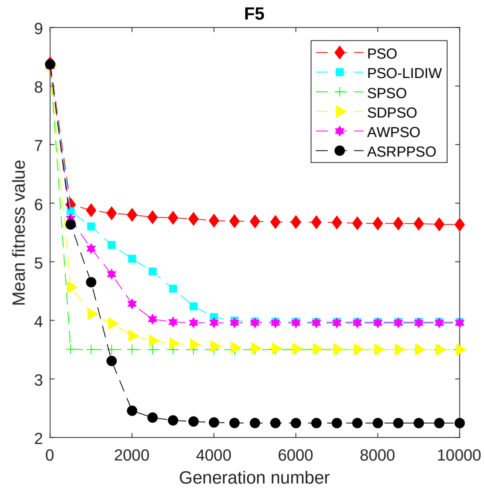


Fig. 5. Convergence plot for the Rosenbrock Function $F_5(x)$

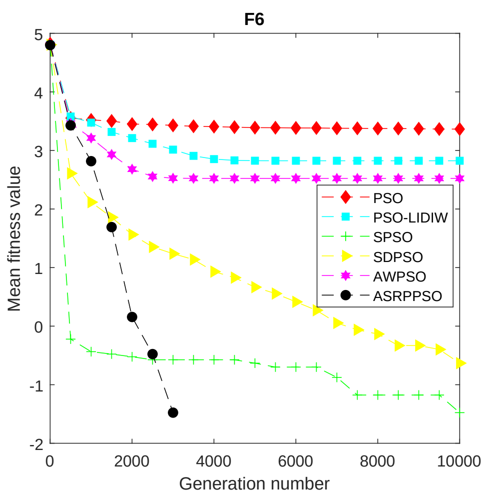


Fig. 6. Convergence plot for the Step Function $F_6(x)$

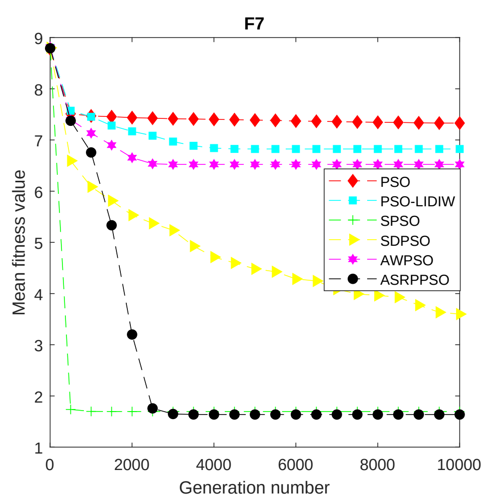


Fig. 7. Convergence plot for the Bent Cigar Function $F_7(x)$

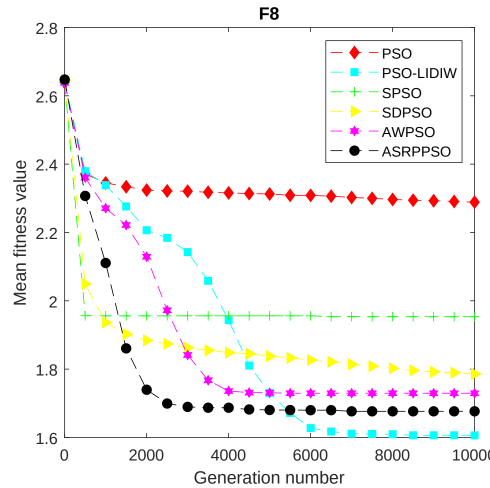


Fig. 8. Convergence plot for the Rastrigin Function $F_8(x)$

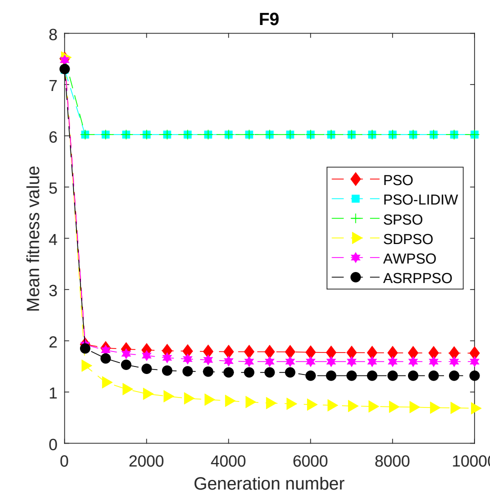


Fig. 9. Convergence plot for the Zakharov Function $F_9(x)$

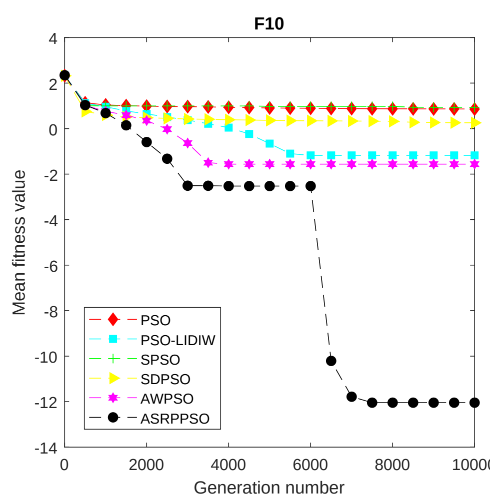


Fig. 10. Convergence plot for the Levy Function $F_{10}(x)$

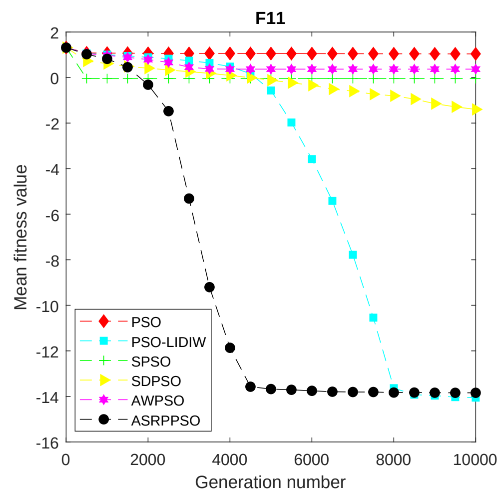


Fig. 11. Convergence plot for the Ackley Function $F_{11}(x)$

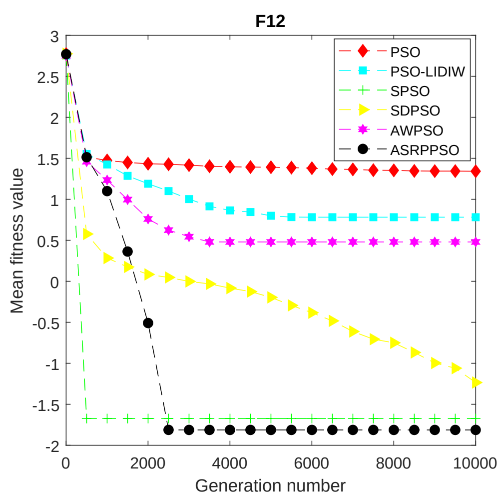


Fig. 12. Convergence plot for the Griewank Function $F_{12}(x)$

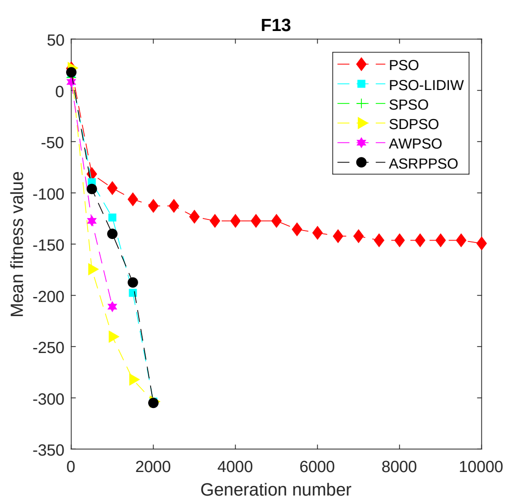


Fig. 13. Convergence plot for the Sum of Different Power Function $F_{13}(x)$

the smallest one on $F_1(x)$, $F_5(x)$-$F_7(x)$, $F_{10}(x)$ and $F_{12}(x)$-$F_{13}(x)$ compared with other chosen algorithms. To conclude, the ASRPPSO has shown competitive performance on the benchmark functions in terms of the solution quality and the search capability based on the statistical analysis.

3) Success Ratio: The success ratio is also listed in Table V and Table VI. According to the table, the ASRPPSO has the highest success ratio on $F_1(x)$-$F_3(x)$, $F_5(x)$-$F_7(x)$ and $F_{10}(x)$-$F_{13}(x)$, which means that the ASRPPSO could find the optimum with high probability. The success ratio of the ASRPPSO on $F_9(x)$ also shows competitive performance compared with other PSO algorithms. The ratio of all selected PSO algorithms on $F_4(x)$ and $F_8(x)$ is not satisfactory because the number of local optima of these functions are very large, which indicates that it is difficult to find the globally optimal solution.

4) Convergence Rate: The number of iterations when the algorithm converges is illustrated in Table V and Table VI as well. It can be seen from the table that the ASRPPSO performed well in all 13 functions compared with other selected algorithms.

To summarize, by comparing with other selected PSO algorithms, the proposed ASRPPSO demonstrates superior performance over the compared ones in terms of convergence and search ability.

B. Evaluation of the ASRPPSO-based FCM Algorithm

To verify the effectiveness of the proposed ASRPPSO-based FCM clustering algorithm, the Iris data from UCI machine learning repository is employed for performance evaluation. Experimental results are assessed by using the Rand Index (RI), the Adjusted Rand Index (ARI) and the Weighted Kappa (WK) coefficient.

The Rand Index (RI) is a similarity measure indicator between two data clustering algorithms. Notice that when the clustering results are random, RI is not a constant close to 0. As such, the ARI is introduced to overcome this shortcoming. The value of the ARI is between -1 and 1. When the clustering results are compared with labelled data, the larger the ARI, the higher the accuracy of the clustering results. In addition, the larger the ARI, the higher the performance similarity between two clustering algorithms.

The WK coefficient is a statistic tool which has the ability of measuring the inter-rater reliability of categorizing instances [23]. The value of the WK coefficient shows the agreement between two comparators. If the value is negative, the agreement between two comparators is worse than random. If the value is 0, it means the agreement between the comparators is equal to random. If the value is closer to 1, the agreement between the two comparators is higher.

The comparison evaluation of the clustering results by using the ASRPPSO-based FCM algorithm and the traditional FCM algorithm is listed in Table VII.

According to Table VII, the WK coefficient, ARI and RI of the ASRPPSO-based FCM algorithm are 0.7287, 0.7287 and 0.8797, respectively. The WK coefficient, ARI and RI of the FCM algorithm are0.7149, 0.7149 and 0.8737, respectively. It

TABLE VII
EVALUATION OF TWO CLUSTERING ALGORITHMS

Algorithm	WK	ARI	RI
ASRPPSO-based FCM	0.7287	0.7287	0.8797
FCM	0.7149	0.7149	0.8737

can be found from the results that the ASRPPSO-based FCM algorithm is an effective clustering algorithm, which exhibits better performance than the FCM algorithm in aspects of WK, ARI and RI. Therefore, the effectiveness and reliability of the developed ASRPPSO-based FCM algorithm are proven based on experimental results.

C. Outlier Detection on WAAM Data Sets

1) Data Pre-Processing: As mentioned previously in Section II, there are four variables (e.g., X_Value, Current, Voltage, and ComputerTime) in each data set. In the experiment, only the voltage and current are utilized for outlier detection. The missing value and null value in the data sets are removed for data cleaning. In order to avoid the influence between different attributes, it is necessary to make every instance in each data set have the same scale so that each feature is equally measured. Thus, the $\min$-$\max$ normalization process is carried out to pre-process the data. The $\min$-$\max$ normalization is given by:

$$X_{N_i} = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \tag{8}$$

where X_{N_i} denotes the ith normalized data of the variable X; $X_{\max}$ and $X_{\min}$ represent the maximum and the minimum value of the variable X in the data, respectively.

2) The Results of the Outlier Detection on WAAM Data Sets: During the AM process, some values of current and voltage may occasionally change abruptly, which indicates that the process is unstable. In this case, these instances are regarded as outliers. The results of the outlier detection are shown in Table VIII.

TABLE VIII
OUTLIER DETECTION RESULTS

Data Set	Cluster1	Cluster2 (Outlier)
Data Set 1	81326	16674
Data Set 2	78979	19021
Data Set 3	79954	18046
Data Set 4	83023	14977
Data Set 5	83096	14904

The clustering performance of the developed outlier detection method on the WAAM data sets is evaluated by using the silhouette coefficients, which is an effective approach to assess the unlabelled data clustering results [34]. Generally, the silhouette coefficient value is between -1 and 1. The performance of clustering is better when the value is closer to 1. In this experiment, the clustering performance of the proposed method is evaluated by using the average silhouette coefficients of all data points in each data set. Experimental results are shown in Table IX and Figs. 14-18.

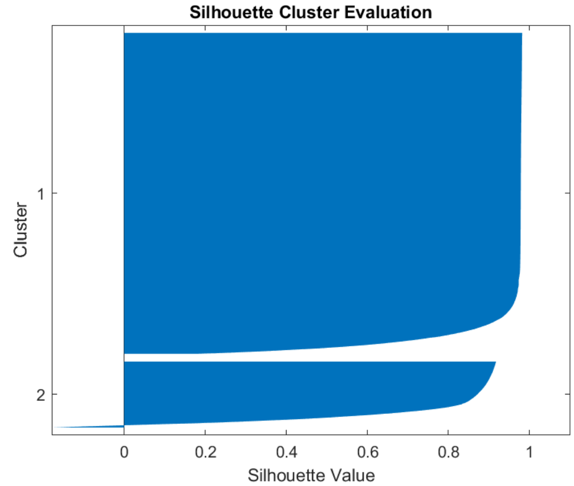


Fig. 14. The average silhouette coefficient of the clustering result on data set 1

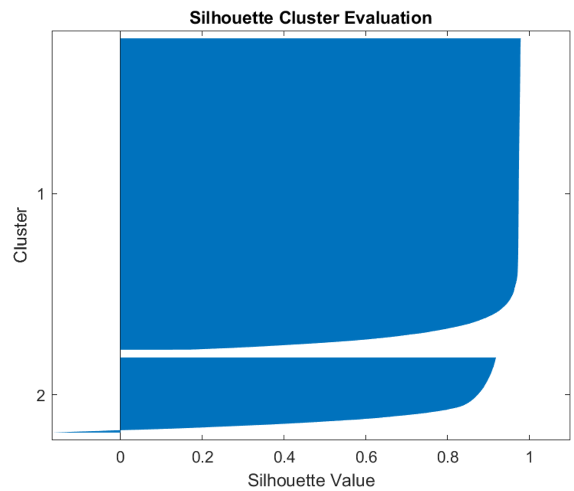


Fig. 15. The average silhouette coefficient of the clustering result on data set 2

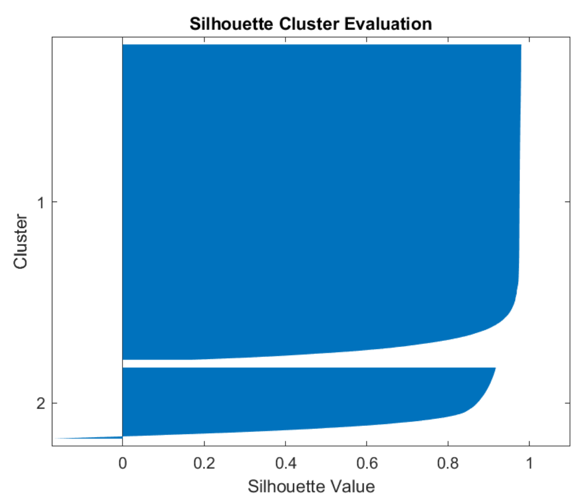


Fig. 16. The average silhouette coefficient of the clustering result on data set 3

TABLE IX
THE AVERAGE SILHOUETTE COEFFICIENTS OF THE CLUSTERING RESULTS

Data Set	Average Silhouette Coefficients
Data Set 1	0.9135
Data Set 2	0.9034
Data Set 3	0.9073
Data Set 4	0.9123
Data Set 5	0.9315

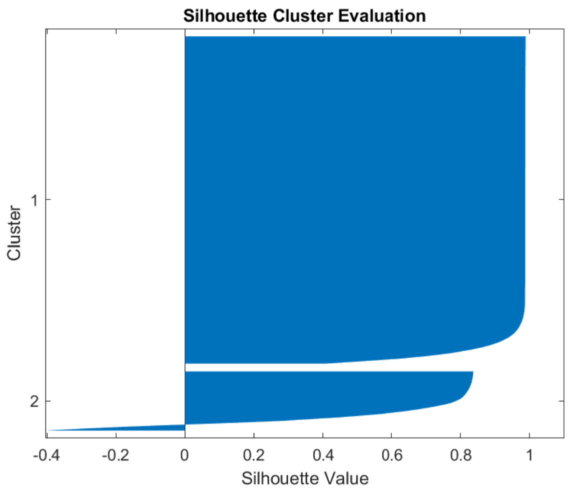


Fig. 17. The average silhouette coefficient of the clustering result on data set 4

From Figs. 14-18, most of the silhouette values are positive, which means most data points are assigned to the right clusters. At the same time, according to Table IX the average silhouette coefficients are all close to 1. Overall, the results of the ASRPPSO-based FCM clustering algorithm on the AM data are reasonable, which demonstrates the effectiveness of the developed method.

Fig. 18. The average silhouette coefficient of the clustering result on data set 5

V. CONCLUSION

In this paper, an optimized FCM-based outlier detection method has been proposed to analyze the current and voltage data collected through the process in a WAAM pilot line deployed in Sweden. Specifically, the ASRPPSO has been developed to optimize the initial locations of the cluster centroids in the FCM algorithm. An AWU strategy has been designed in the ASRPPSO to adaptively adjust the acceleration coefficients, and the Gaussian white noise has been added to the velocity updating equation based on the evolutionary states. Experimental results have shown the superiority of the ASRPPSO over some existing PSO algorithms in terms of convergence rate and solution quality. To demonstrate its application potential, the ASRPPSO-based FCM algorithm has been applied to outlier detection on the real-world WAAM data, which contributes to the online monitoring of the WAAM process with hope to fabricate qualified components.

Future work can be summarized into the following five aspects: 1) designing a parameter selection strategy to automatically choose the parameters of the adaptive weighting function; 2) adjusting the mean and variance of the random noises adaptively for different evolutionary states; 3) applying the ASRPPSO to multi-objective optimization problems; and 4) analyzing the dynamical behavior of the ASRPPSO by using the Lyapunov-like stability theory and the filtering techniques [11], [18], [20], [30]; and 5) deploying the proposed outlier detection algorithm to some other data analysis applications [22], [29].

REFERENCES

[1] I. Chatterjee, M. Zhou, A. Abusorrah, K. Sedraoui and A. Alabdulwahab, Statistics-based outlier detection and correction method for amazon customer reviews, *Entropy*, vol. 23, no. 12, pp. 1645-1668, 2021.

[2] S. Chen, I. Abdulselam, N. Yadollahpour and Y. Gonzalez-Fernandez, Particle swarm optimization with pbest perturbations, In: *Proceedings of the 2020 IEEE Congress on Evolutionary Computation*, Glasgow, UK, Sept. 2020, pp. 1-8.

[3] W. Chen, J. Zhang, Y. Lin, N. Chen, Z. Zhan H. S. Chung, Y. Li and Y. Shi, Particle swarm optimization with an aging leader and challengers, *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 2, pp. 241-258, 2013.

[4] H. W. Cho, S. J. Shin, G. J. Seo, D. B. Kim, and D. H. Lee, Real-time anomaly detection using convolutional neural network in wire arc additive manufacturing: Molybdenum material, *Journal of Materials Processing Technology*, vol. 302, art. no. 117495, 2022.

[5] Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. Hernandez and R. G. Harley, Particle swarm optimization: Basic concepts, variants and applications in power systems, *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 2, pp. 171-195, 2004.

[6] W. Dong and M. Zhou, A supervised learning and control method to improve particle swarm optimization algorithms, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1135-1148, 2017.

[7] W. Gao, Y. Zhang, D. Ramanujan, K. Ramani, Y. Chen, C. B. Williams, C. C. L. Wang, Y. C. Shin, S. Zhang and P. D. Zavattieri, The status, challenges, and future of additive manufacturing in engineering, *Computer-Aided Design*, vol. 69, pp. 65-89, 2015.

[8] I. Gibson, D. Rosen and B. Stucker, *Additive Manufacturing Technologies*, Switzerland: Springer, 2021.

[9] A. Haleem and M. Javaid, Additive manufacturing applications in industry 4.0: A review, *Journal of Industrial Integration and Management*, vol. 4, no. 4, art. no. 1930001, 2019.

[10] V. Hodge and J. Austin, A survey of outlier detection methodologies, *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85-126, 2004.

[11] Z. Hu, J. Hu, H. Tan, J. Huang and Z. Cao, Distributed resilient fusion filtering for nonlinear systems with random sensor delay under round-robin protocol, *International Journal of Systems Science*, vol. 53, no. 13, pp. 2786-2799, 2022.

[12] C. Huang, G. Wang, H. Song, R. Li and H. Zhang, Rapid surface defects detection in wire and arc additive manufacturing based on laser profilometer, *Measurement*, vol. 189, art. no. 110503, 2022.

[13] K. Huang, H. Wen, C. Yang, W. Gui and S. Hu, Outlier detection for process monitoring in industrial cyber-physical systems, *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 2487-2498, 2022.

[14] M. Javaid and A. Haleem, Additive manufacturing applications in medical cases: A literature based review, *Alexandria Journal of Medicine*, vol. 54, no. 4, pp. 411-422, 2018.

[15] M. A. Kamel, X. Yu and Y. Zhang, Real-time fault-tolerant formation control of multiple WMRs based on hybrid GA-PSO algorithm, *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1263-1276, 2021.

[16] R. Kundu, S. Das, R. Mukherjee and S. Debchoudhury, An improved particle swarm optimizer with difference mean based perturbation, *Neurocomputing*, vol. 129, pp. 315-333, 2014.

[17] C. Lee, K. Li, G. Seo, D. B. Kim, M. Kim and J. Shin, Development of defect detection AI model for wire + arc additive manufacturing using high dynamic range images, *Applied Sciences*, vol. 11, no. 16, pp. 7541-7559, 2021.

[18] M. Li, J. Liang and F. Wang, Robust set-membership filtering for two-dimensional systems with sensor saturation under the Round-Robin protocol, *International Journal of Systems Science*, vol. 53, no. 13, pp. 2773-2785, 2022.

[19] J. J. Liang, A. K. Qin, P. N. Suganthan and S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 281-295, 2006.

[20] L. Liu, L. Ma, J. Zhang and Y. Bo, Distributed non-fragile set-membership filtering for nonlinear systems under fading channels and bias injection attacks, *International Journal of Systems Science*, vol. 52, no. 6, pp. 1192-1205, 2021.

[21] W. Liu, *Novel Particle Swarm Optimization Algorithms with Applications to Healthcare Data Analysis*, Doctoral dissertation, Brunel University London, 2020.

[22] X. Luo, M. Chen, H. Wu, Z. Liu, H. Yuan and M. Zhou, Adjusting learning depth in nonnegative latent factorization of tensors for accurately modeling temporal patterns in dynamic QoS data, *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 4, pp. 2142-2155, 2021.

[23] M. L. McHugh, Interrater reliability: The kappa statistic, *Biochemia Medica*, vol. 22, no. 3, pp. 276-282, 2012.

[24] F. Montevecchi, G. Venturini, A. Scippa and G. Hui, Finite element modelling of wire-arc-additive-manufacturing process, *Procedia CIRP*, vol. 55, pp. 109-114, 2016.

[25] A. Mozaffari, S. Chenouri, E. Toyserkani and U. Ali, Functional boxplots for outlier detection in additive manufacturing, *arXiv preprint arXiv:2110.10867*, 2021.

[26] T. D. Ngo, A. Kashani, G. Imbalzano, K. T. Q. Nguyen and D. Hui, Additive manufacturing (3D printing): A review of materials, methods, applications and challenges, *Composites Part B: Engineering*, vol. 143, pp. 172-196, 2018.

[27] Q. Niu and X. Huang, An improved fuzzy C-means clustering algorithm based on PSO, *Journal of Software*, vol. 6, no. 5, pp. 873-879, 2011.

[28] W. Ou, G. L. Knapp, T. Mukherjee, Y. Wei and T. DebRoy, An improved heat transfer and fluid flow model of wire-arc additive manufacturing, *International Journal of Heat and Mass Transfer*, vol. 167, art. no. 120835, 2021.

[29] Q. Peng, Y. Xia, M. Zhou, X. Luo, S. Wang, Y. Wang, C. Wu, S. Pang and M. Lin, Reliability-aware and deadline-constrained mobile service composition over opportunistic networks, *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1012-1025, 2021.

[30] F. Qu, X. Zhao, X. Wang and E. Tian, Probabilistic-constrained distributed fusion filtering for a class of time-varying systems over sensor networks: A torus-event-triggering mechanism, *International Journal of Systems Science*, vol. 53, no. 6, pp. 1288-1297, 2022.

[31] A. Queguineur, G. Rückert, F. Cortial and J. Y. Hascoët, Evaluation of wire arc additive manufacturing for large-sized components in naval applications, *Welding in the World*, vol. 62, no. 2, pp. 259-266, 2018.

[32] A. Ratnaweera, S. K. Halgamuge and H. C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240-255, 2004.

[33] R. Reisch, T. Hauser, B. Lutz, M. Pantano, T. Kamps and A. Knoll, Distance-based multivariate anomaly detection in wire arc additive manufacturing, In: *Proceedings of the 19th IEEE International Conference on Machine Learning and Applications*, Miami, USA, Dec. 2020, pp. 659-664.

[34] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53-65, 1987.

[35] T. A. Runkler and C. Katz, Fuzzy clustering by particle swarm optimization, In: *Proceedings of the 2006 IEEE International Conference on Fuzzy Systems*, Vancouver, Canada, Jul. 2006, pp. 601-608.

[36] Y. Shi and R. Eberhart, A modified particle swarm optimizer, In: *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation*, Anchorage, USA, May 1998, pp. 69-73.

[37] Y. Shi and R. Eberhart, Parameter selection in particle swarm optimization, In: *Proceedings of the International Conference on Evolutionary Programming*, Berlin, Germany, Mar. 1998, pp. 591-600.

[38] Y. Shi and R. Eberhart, Empirical study of particle swarm optimization, In: *Proceedings of the 1999 Congress on Evolutionary Computation*, Washington, USA, Jul. 1999, pp. 1945-1950.

[39] B. Song, H. Miao and L. Xu, Path planning for coal mine robot via improved ant colony optimization algorithm, *Systems Science & Control Engineering*, vol. 9, no. 1, pp. 283-289, 2021.

[40] G. A. Susto, M. Terzi and A. Beghi, Anomaly detection approaches for semiconductor manufacturing, *Procedia Manufacturing*, vol. 11, pp. 2018-2024, 2017.

[41] J. Tang, G. Liu and Q. Pan, A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends, *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 10, pp. 1627-1643, 2021.

[42] Y. Tang, Z. Wang and J. Fang, Parameters identification of unknown delayed genetic regulatory networks by a switching particle swarm optimization algorithm, *Expert Systems with Applications*, vol. 38, no. 3, pp. 2523-2535, 2011.

[43] T. Velmurugan and T. Santhanam, Performance evaluation of K-means and fuzzy C-means clustering algorithms for statistical distributions of input data points, *European Journal of Scientific Research*, vol. 46, no. 3, pp. 320-330, 2010.

[44] P. Verma, M. Sinha and S. Panda, Fuzzy C-means clustering-based novel threshold criteria for outlier detection in electronic nose, *IEEE Sensors Journal*, vol. 21, no. 2, pp. 1975-1981, 2021.

[45] B. Wang and Z. Mao, Outlier detection based on Gaussian process with application to industrial processes, *Applied Soft Computing*, vol. 76, pp. 505-516, 2019.

[46] C. Wang, W. Pedrycz, Z. Li and M. Zhou, Residual-driven fuzzy C-means clustering for image segmentation, *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 876-889, 2021.

[47] C. Wang, W. Pedrycz, Z. Li and M. Zhou, KullbackLeibler divergence-based fuzzy C-means clustering incorporating morphological reconstruction and wavelet frames for image segmentation, *IEEE Transactions on Cybernetics*, vol. 52, no. 8, pp. 7612-7623, 2022.

[48] Y. Wang, X. Chen and C. Su, Microstructure and mechanical properties of Inconel 625 fabricated by wire-arc additive manufacturing, *Surface and Coatings Technology*, vol. 374, pp. 116-123, 2019.

[49] B. Wu, Z. Pan, D. Ding, D. Cuiuri, H. Li, J. Xu and J. Norrish, A review of the wire arc additive manufacturing of metals: Properties, defects and quality improvement, *Journal of Manufacturing Processes*, vol. 35, pp. 127-139, 2018.

[50] Q. Wu, M. Zhou, Q. Zhu, Y. Xia and J. Wen, MOELS: Multiobjective evolutionary kist scheduling for cloud workflows, *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 166-176, 2020.

[51] L. Xu, B. Song and M. Cao, An improved particle swarm optimization algorithm with adaptive weighted delay velocity, *Systems Science & Control Engineering*, vol. 9, no. 1, pp. 188-197, 2021.

[52] L. Xu, B. Song and M. Cao, A new approach to optimal smooth path planning of mobile robots with continuous-curvature constraint, *Systems Science & Control Engineering*, vol. 9, no. 1, pp. 138-149, 2021.

[53] X. Xu, J. Li, M. Zhou, J. Xu and J. Cao, Accelerated two-stage particle swarm optimization for clustering not-well-separated data, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4212-4223, 2020.

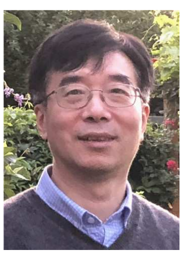
[54] N. Zeng, Z. Wang, H. Zhang and F. E. Alsaadi, A novel switching delayed PSO algorithm for estimating unknown parameters of lateral flow immunoassay, *Cognitive Computation*, vol. 8, no. 2, pp. 143-152, 2016.

[55] P. Zhang, X. Lai, Y. Wang, C. Su and M. Wu, PSO-based nonlinear model predictive planning and discrete-time sliding tracking control for uncertain planar underactuated manipulators, *International Journal of Systems Science*, vol. 53, no. 10, pp. 2075-2089, 2022.

[56] L. Zou, Z. Wang, H. Geng and X. Liu, Set-membership filtering subject to impulsive measurement outliers: A recursive algorithm, *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 377-388, 2021.

[57] R. Zou, V. Kalivarapu, E. Winer, J. Oliver and S. Bhattacharya, Particle swarm optimization-based source seeking, *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 865-875, 2015.

Jingzhong Fang received his B.Eng. degree in automation from Shandong University of Science and Technology, Qingdao, China, in 2020, and the M.Sc. degree in data science and analytics from Brunel University London, Uxbridge, U.K., in 2021. He is currently pursuing the Ph.D. degree in Computer Science at Brunel University London, Uxbridge, U.K. His research interests include data analysis and deep learning techniques.



Zidong Wang (Fellow, IEEE) received the B.Sc. degree in mathematics in 1986 from Suzhou University, Suzhou, China, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology, Nanjing, China.

He is currently Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the U.K. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published more than 700 papers in international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for *International Journal of Systems Science*, the Editor-in-Chief for *Neurocomputing*, the Editor-in-Chief for *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals including IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology, IEEE Transactions on Neural Networks, IEEE Transactions on Signal Processing, and IEEE Transactions on Systems, Man, and Cybernetics-Part C. He is a Member of the Academia Europaea, a Member of the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, a Fellow of the IEEE, a Fellow of the Royal Statistical Society and a member of program committee for many international conferences.

Weibo Liu received the B.Eng. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, U.K, in 2015, and the Ph.D. degree in artificial intelligence in 2020 from the Department of Computer Science, Brunel University London, Uxbridge, U.K. He is currently a Lecturer in the Department of Computer Science, Brunel University London, Uxbridge, U.K. His research interests include intelligent data analysis, evolutionary computation, transfer learning, machine learning and deep learning. He serves as an Associate Editor for the Journal of Ambient Intelligence and Humanized Computing and the Journal of Cognitive Computation. He is a very active reviewer for many international journals and conferences.

Stanislao Lauria received a Laurea Degree in Physics from the Unversity of Naples Federico II, Italy and a Ph.D. degree in Cybernetics from the University of Reading, U.K. He is a Lecturer at Brunel University, U.K. He has been research fellow at both the University of Reading and University of Plymouth, U.K. He is a computer scientist with an interest in robotics, HCI, robotics and social media, AI, robotics and education, image analysis, data processing.

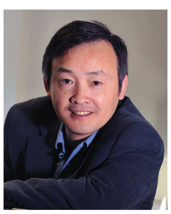
Nianyin Zeng was born in Fujian Province, China, in 1986. He received the B.Eng. degree in electrical engineering and automation in 2008 and the Ph.D. degree in electrical engineering in 2013, both from Fuzhou University. From October 2012 to March 2013, he was a RA in the Department of Electrical and Electronic Engineering, the University of Hong Kong. From September 2017 to August 2018, he as an ISEF Fellow founded by the Korea Foundation for Advance Studies and also a Visiting Professor at the Korea Advance Institute of Science and Technology.

Currently, he is an Associate Professor with the Department of Instrumental & Electrical Engineering of Xiamen University. His current research interests include intelligent data analysis, computational intelligent, time-series modeling and applications. He is the author or co-author of several technical papers and also a very active reviewer for many international journals and conferences.

Dr. Zeng is currently serving as Associate Editors for Neurocomputing, Evolutionary Intelligence, and Frontiers in Medical Technology, and also Editorial Board members for Computers in Biology and Medicine, Biomedical Engineering Online, and Mathematical Problems in Engineering.

Camilo Prieto is a Senior Researcher at AIMEN Technology Center. He holds an M.Sc. degree on Laser Physics and Technologies from University of Salamanca and an M.Eng. degree on Mechatronics by University of Vigo in Spain. He has developed his professional career in research centre and companies in the field of advanced manufacturing, being involved in R&D programs for over 15 years. He has undertaken roles as R&D application engineer and also as engineering technical leader on deployment of Laser Manufacturing Systems in production environments. His current main research and development activities include process monitoring and data analysis in the field of additive manufacturing.

Fredrik Sikström received his Ph.D. degree in electrical engineering from the Chalmers University of Technology, Sweden in 2010. In 2011 he was employed at the University West in Sweden as an assistant professor and since 2019 he holds a position as associated professor in production technology. His research activities are focused on in-process monitoring and automatic control of laser and arc welding as well as metallic directed energy deposition. He has published 35 publications indexed by ISI or SCOPUS.



Xiaohui Liu received the B.Eng. Degree in Computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in Computer Science from Heriot-Watt University, Edinburgh, UK, in 1988. He is currently a Professor of Computing at Brunel University London where he conducts research in AI and intelligent data analysis, with applications in diverse areas including biomedicine and engineering.