



ARTICLE

A User-Transformer Relation Identification Method Based on QPSO and Kernel Fuzzy Clustering

Yong Xiao¹, Xin Jin¹, Jingfeng Yang², Yanhua Shen^{3,*} and Quansheng Guan⁴

¹China Southern Power Grid Research Institute Co., Ltd., Guangzhou, 510663, China

²China Southern Power Grid Co., Ltd., Guangzhou, 510663, China

³School of Materials Science and Engineering, South China University of Technology, Guangzhou, 510641, China

⁴School of Electronics and Information, South China University of Technology, Guangzhou, 510641, China

*Corresponding Author: Yanhua Shen. Email: shenyh@scut.edu.cn

Received: 04 July 2020 Accepted: 20 August 2020

ABSTRACT

User-transformer relations are significant to electric power marketing, power supply safety, and line loss calculations. To get accurate user-transformer relations, this paper proposes an identification method for user-transformer relations based on improved quantum particle swarm optimization (QPSO) and Fuzzy C-Means Clustering. The main idea is: as energy meters at different transformer areas exhibit different zero-crossing shift features, we classify the zero-crossing shift data from energy meters through Fuzzy C-Means Clustering and compare it with that at the transformer end to identify user-transformer relations. The proposed method contributes in three main ways. First, based on the fuzzy C-means clustering algorithm (FCM), the quantum particle swarm optimization (PSO) is introduced to optimize the FCM clustering center and kernel parameters. The optimized FCM algorithm can improve clustering accuracy and efficiency. Since easily falls into a local optimum, an improved PSO optimization algorithm (IQPSO) is proposed. Secondly, considering that traditional FCM cannot solve the linear inseparability problem, this article uses a FCM (KFCM) that introduces kernel functions. Combined with the IQPSO optimization algorithm used in the previous step, the IQPSO-KFCM algorithm is proposed. Simulation experiments verify the superiority of the proposed method. Finally, the proposed method is applied to transformer detection. The proposed method determines the class members of transformers and meters in the actual transformer area, and obtains results consistent with actual user-transformer relations. This fully shows that the proposed method has practical application value.

KEYWORDS

User-transformer relation identification; zero-crossing shift; fuzzy C-means clustering; quantum particle swarm optimization; attractor multiple update strategy; dynamic crossover strategy; perturbation strategy of potential-well characteristic length



1 Introduction

User-transformer relation refers to the power supply membership between electricity users and the transformer that serves them [1]. Unclear user-transformer relations would cause significant errors between data from area line losses and split-phase line losses, and irrational load increases based on business expansion, affecting the load balance and lowering the success rate of remote cost controls and remote recharge systems. While influencing fundamental business, it would also curb implementing further business applications. When a power failure occurs and an immediate repair is needed, wrong loads and membership data could impair the practicality and safety of the repair strategy and timeliness of the repair project [2]. Due to changing user information, replacing malfunctioning meters, upgrading station areas, and more, user-transformer relations change frequently. In the meanwhile, under the “null wire shared” and coupling conditions, cross-area communications and readings are attainable in low voltage power line carrier and micropower wireless communications. All of the above make it very challenging to resolve user-transformer relations [3].

User-transformer relation identification is a new field that has just recently emerged in China. With small populations and simply-structured low-voltage station areas, it is not a problem for developed Western countries; nor is it a problem for ordinary developing countries. Despite big populations, they have fewer grid terminal equipment, and thus can manually resolve user-transformer relations. On the contrary, most parts of China are densely-populated. With growing power demand and an increasing number of terminal equipment, it is urgent for user-transformer relations to be accurately resolved. At present, this is usually done by patrol persons during their routine on-site patrols and thus is time-consuming and laborious. In addition, energy meters placed in intersecting station areas can't be distinguished unless the power is cut off. The operations are not only difficult but bound to cause huge economic losses. Since 2016, many organizations in the power system have tried to apply big data to energy meter currents, power, power frequency zero-crossing shifts, and power frequency zero-crossing distortions to identify user-transformer relations. However, affected by the equipment sampling accuracy, data synchronization, and high distinguishing costs, the big data method has not proved to be effective [4].

User-transformer relations identification is a typical classification problem. The commonly-used clustering analysis methods primarily include: Mean clustering [5], kernel subtraction clustering [6], hierarchical clustering [7], and graph theory clustering [8]. The clustering algorithms avoid solutions for complex gradients and adjoint equations and have no strict requirements on model expression and continuity. They can effectively extract system features for classifications and are easy to implement. Hajek et al. [9] proposed a clustering method based on semidefinite programming, which implemented a binary symmetric stochastic block model to deal with stochastic graph classification and solved the problem of semidefinite programming relaxation with a maximum likelihood estimation to obtain accurate cluster boundaries. Wade et al. [10] proposed a clustering method based on Bayesian analysis, where appropriate point estimates and confidence intervals were set to determine the clustering structure based on the decision-making and information theory and techniques. Javadi et al. [11] implemented the K-means clustering method to cluster underground aquifer vulnerabilities, where vulnerability maps were created based on on-site data features to substitute for conventional fixed indicators, such as weights and ratios and accurately classify groundwater contamination. Combining principal component analysis and hierarchical clustering analysis, Granato et al. [12] presented relevant classifications for multiple compounds in food based on data characteristics.

However, conventional methods are not applicable to all data structures and are unable to solve the linear inseparability problem. By introducing kernel functions into the standard kernel mean clustering algorithm, kernel fuzzy C-Means clustering (KFCM) [13] maps the original problem to a high-dimensional space for clustering analysis. It can effectively achieve clustering for various data structures and solve the linear inseparability problem, thus widely applicable to engineering practices. However, a lack of theoretical guidance in selecting kernel functions, overly sensitive to the initial cluster center and kernel parameters, and long in computation time [14], standard KFCM needs to be further improved. To solve the image segmentation problem, Chen and his team proposed the kernel fuzzy C-means clustering algorithm, established a kernel function library, and adaptively selected the optimal kernel function through pixel feature information [15]. Ding et al. [16] used the Genetic Algorithm to optimize KFCM kernel parameters and increased clustering accuracy and efficiency. Likewise, Liu et al. [17] introduced the good-point set theory and speed weight to improve the Bat Algorithm (BA) and optimize KFCM. Combining the knowledge and theory of Quantum Mechanics and Particle Swarm Optimization, and Quantum Particle Swarm Optimization (QPSO) absorbed the strengths of both. It has advantages, such as few optimization parameters, fast convergence, strong generalization, a simple evolution and global convergence. QPSO is thus introduced in this paper to optimize KFCM cluster center and kernel parameters. To summarize, to identify the user-transformer relations in the grid, this paper proposes an improved KFCM algorithm, where the clustering analysis of zero-crossing shift data inherent in voltage signals in low-voltage station areas are made to identify user-transformer relations. To improve clustering accuracy and efficiency, we adopt QPSO to optimize the cluster center and kernel parameters. Meanwhile, we also define the intra-class distance and inter-class distance to improve the fitness function and introduce the Attractor Multiple Update Strategy to solve the problem of “prematurity”. Through simulation tests and application verifications, the method has proved to be effective in identifying real user-transformer relationships in low-voltage station areas.

2 Problem Descriptions

For the actual low-voltage distribution network shown in Fig. 1, the AC signals are transmitted to each low-voltage station area along the medium-voltage line through a 10 KV/380 V transformer. When the electrical signals are passing through the transformer, or when the capacitive or inductive loads are being connected to or disconnected from the power line, the phase of the AC electrical signal will exhibit voltage zero-crossing shift, phase distortion, and other characteristics. The specific shift is shown in Fig. 2.

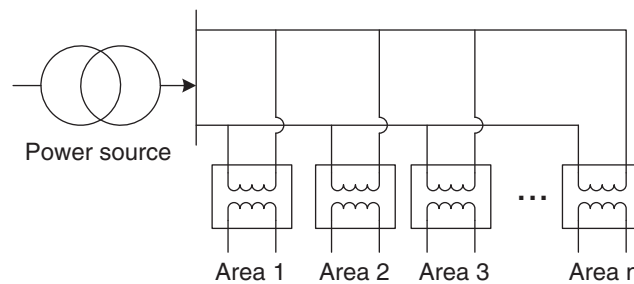


Figure 1: Topological graph for low-voltage area network

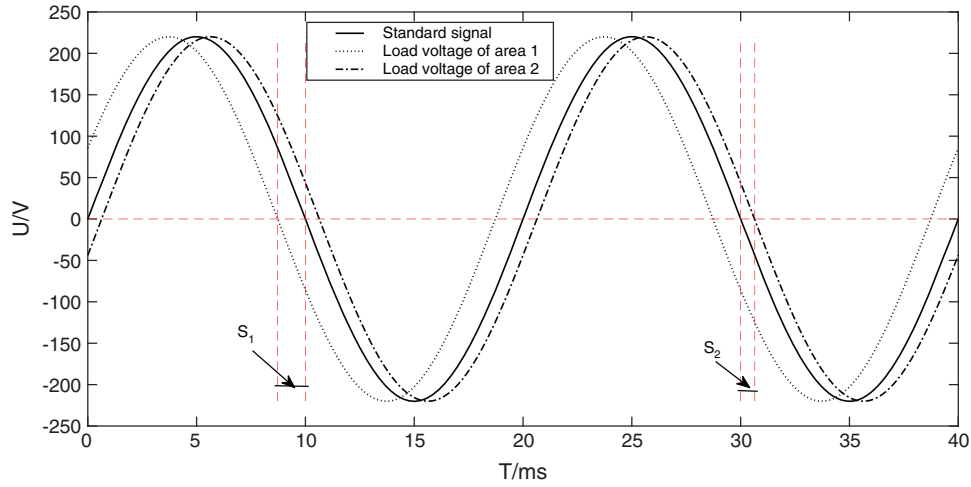


Figure 2: Sketch map for phase shift

Zero-crossing shift refers to the time or phase deviation of the AC sine wave signal at the point where the amplitude changes positively or negatively (that is, the zero-crossing point) with the standard frequency signal. Less affected by the load, the voltage signal is often used for analysis. In Fig. 2, the solid line represents the power frequency voltage signal (standard signal) with an effective value of 220 V and a frequency of 50 HZ. The dotted line and the dash line are the load voltages of station area 1 and station area 2 respectively. S_1 and S_2 are the zero-crossing shift data of the voltage signals in the two station areas respectively.

In order to verify the influences of different inductances and capacitances on the zero-crossing shift, a simulation circuit, as shown in Fig. 3, is constructed. Module 1 is used to generate a 220 V power frequency power supply and simulate a transformer; Module 2 is used to generate a zero-crossing signal on the transformer side; Module 3 is used to simulate the impedance environment of the power network. The impedance range is simulated between 2Ω and 50Ω . The transmission line resistor is 1Ω ; Module 4 is used to generate a zero-crossing signal on the energy meter side. Let $R_3 = R_4 = 1 \Omega$, adjust differences in L_1 and L_2 , and record the zero-crossing shifts in Tab. 1. It can be observed that inductance and zero-crossing shift are positively correlated.

In an actual scenario affected by the inductance of the medium-voltage line, the capacitance to the ground, the primary-side equipment, and the transformer, the zero-crossing shift characteristics vary between different station areas. The phases in different station areas vary as well. In addition, the distributions of transformers and transmission lines are inherently inductive and capacitive, hence further highlighting the characteristics of each station area.

Assume that there are M low-voltage station areas (corresponding to M transformers) in the target power grid, and a total of N energy meters. The zero-crossing shift of the voltage signal corresponding to the j th energy meter is:

$$x_j = T_j - T, \quad j \in [1, N], \quad (1)$$

where T_j and T are the time between two adjacent zero crossings of the j th energy meter and the standard signal respectively.

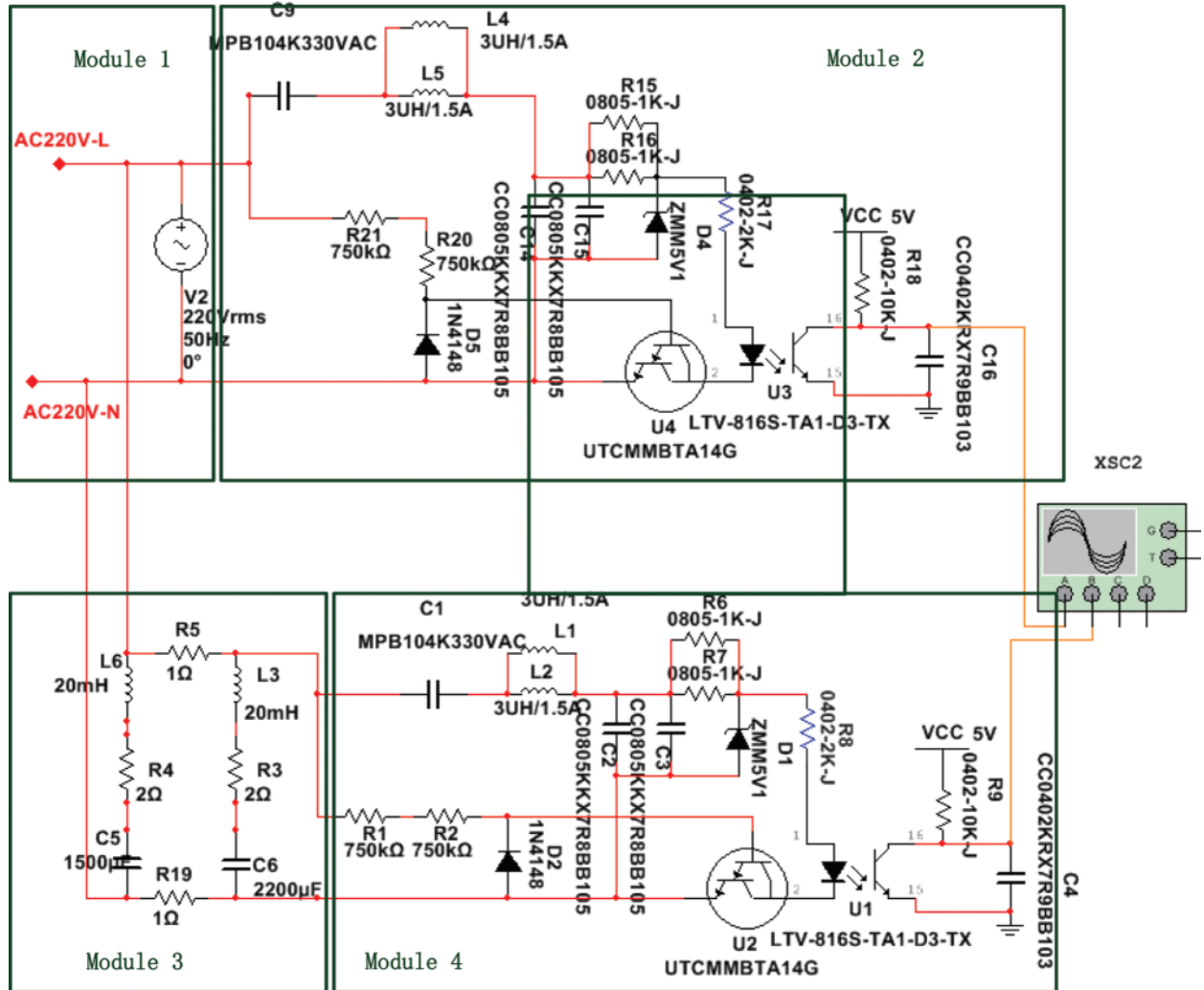


Figure 3: Artificial circuit of zero-crossing shift

Table 1: Zero-crossing shift under different inductance values

Line inductance $L_1 = L_2$ (mH)	Zero-crossing shift (μs)
1	1.86
10	4.90
20	12.16
50	34.43
100	54.70

Similarly, the zero-crossing shift of the i th transformer is:

$$\tilde{X}_i = \tilde{T}_i - T, \quad i \in [1, M], \quad (2)$$

where $\tilde{T}_i$ represents the time between two adjacent zero crossings of the i th transformer.

By performing cluster analysis on the zero-crossing shifts of the energy meter x_j and comparing them with the zero-crossing shifts of the transformer in the station area $\tilde{X}_i$, we can identify the class membership between the N th energy meter and the M th low-voltage station area. Therefore, in this paper, an improved KFCM algorithm is proposed, where the quantum idea, optimization and clustering algorithms are combined, and the QPSO algorithm is adopted to optimize the KFCM algorithm cluster center and kernel parameters. It enables us to make accurate and fast classifications of zero-crossing shifts and hence obtain authentic user-transformer relations.

3 KFCM Based on Improved QPSO

3.1 KFCM

Mapping the original problem into a high-dimensional space, KFCM achieves clustering analysis based on the fuzzy theory and distance objective function. With this method, the phasors composed of N sample points, namely x_j , $j = 1, \dots, N$, can be classified into M classes. For a general classification problem, suppose the center of the i th cluster is v_i , $i = 1, 2, \dots, M$, the membership degree that the j th sample belongs to for the i th class is μ_{ij} , $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$, then the following calculation Formula can be obtained [18]:

$$J = \sum_{i=1}^M \sum_{j=1}^N (\mu_{ij})^m \|\varphi(x_j) - \varphi(v_i)\|^2. \quad (3)$$

The given data needs to be mapped into a high-dimensional space for processing. Selecting the Gaussian function here as the kernel function will achieve:

$$\begin{aligned} J &= \sum_{i=1}^M \sum_{j=1}^N (\mu_{ij})^m (K(x_j, x_j) - 2K(x_j, v_i) + K(v_i, v_i)) \\ &= 2 \sum_{i=1}^M \sum_{j=1}^N (\mu_{ij})^m (1 - K(x_j, v_i)), \end{aligned} \quad (4)$$

$$\text{where } K(x_j, v_i) = e^{-\frac{\|x_j - v_i\|^2}{2\sigma_i^2}}.$$

Membership degree:

$$\mu_{ij} = \frac{1 - K(x_j, v_i)^{-1/m-1}}{\sum_{i=1}^M (1 - K(x_j, v_i))^{-1/m-1}}, \quad (5)$$

where the membership degree satisfies the constraint of $\sum_{i=1}^M \mu_{ij} = 1$, $\mu_{ij} \in [0, 1]$.

Cluster center:

$$v_i = \left(\sum_{j=1}^N \mu_{ij}^m K(x_j, v_i) x_j \right) / \left(\sum_{j=1}^N \mu_{ij}^m K(x_j, v_i) \right). \quad (6)$$

The clustering process is [18]:

- (1) Parameter settings: Ambiguity as m , Error Threshold of Objective Function as ε , Maximum Number of Iterations as $I_{\max}$;
- (2) Initialization: Randomly select M data points in the feasible domain as the initial cluster center;
- (3) For the t th iteration, compute Objective Function $J(t)$. Determine whether $|J(t) - J(t-1)| < \varepsilon$ is true. If it is true, stop the clustering. Otherwise, update the membership matrix and cluster center according to Formulae (5) and (6) and continue with clustering until the optimal cluster center is obtained;
- (4) Defuzzy to get the final clustering result.

KFCM is very sensitive to cluster center settings and kernel parameters. However, in standard KFCM, the initial cluster center is randomly determined, and the kernel parameters are set according to experience. This greatly affects clustering accuracy and efficiency. Thus QPSO is introduced in this paper to seek the optimal cluster centers and kernel parameters.

3.2 Improved QPSO

3.2.1 QPSO

Quantum Particle Swarm Optimization (QPSO) is a new intelligent optimization algorithm proposed by Sun and other scholars on the basis of particle swarm optimization and quantum mechanics [19]. Particle swarm optimization updates positions according to particle flight speeds. Based on the δ potential well theory, QPSO obtains the probability density functions for particles appearing at certain points by solving the Schrödinger equation and performs inverse-transformation alongside it to obtain particle positions. Compared with conventional algorithms, QPSO can search for the optimal solution in the entire feasible domain, with fast convergence and robustness [20,21].

In QPSO, assume the position vector of particle k ($k = 1, 2, \dots, W$, W represents the number of particles) in the t th generation is $X_{k,t} = \{x_{k,t}^1, x_{k,t}^2, \dots, x_{k,t}^N\}$, where N is the dimension of the target problem, $P_{\text{best}_{k,t}}$ represents the historically best position of Particle K and G_{best_t} is the optimal position of the swarm. As known from Chen et al. [22], as particles evolve, a particle would take $P_{k,t}$, the weighted average position of its historically best position and the optimal position of its swarm, as the attraction point and gravitate towards that point. The specific weighted average position (or attractor) is:

$$P_{k,t} = \alpha \cdot P_{\text{best}_{k,t}} + (1 - \alpha) G_{\text{best}_t}, \quad (7)$$

where α is a weighting factor, subject to uniform distribution, i.e., $\alpha \sim U(0, 1)$.

The mean value of the historically best positions of all particles (average optimal position) is:

$$m_{\text{best}_t} = \frac{1}{W} \sum_{k=1}^W P_{\text{best}_{k,t}}. \quad (8)$$

To update particle positions, the particles need to be collapsed from the quantum state to the classic state, and the Monte Carlo random simulation is used to obtain the next-generation particle position.

$$X_{k,t+1} = P_{k,t} + \beta \cdot |m_{best_t} - X_{k,t}| \cdot \ln(u), \quad (9)$$

where $\beta = \begin{cases} 0.5 + 0.5 \cdot \frac{\tilde{t}_{\max} - t}{\tilde{t}_{\max}}, & u > 0.5, \\ 0.5 - 0.5 \cdot \frac{\tilde{t}_{\max} - t}{\tilde{t}_{\max}}, & u \leq 0.5. \end{cases}$, represents the contraction-expansion coefficient, $\tilde{t}_{\max}$ is the maximum number of iterations, u is the random number, $u \sim U(0, 1)$ [23].

The Schrodinger equation [24] informs us that the state of the system cannot be determined with specific values but needs to be described with a wave function. Particle K moves in potential well δ centered on position $P_{k,t}$ in the j th dimension, with a wave function ψ that can be described as:

$$\psi(X_{k,t+1}^j) = \frac{1}{\sqrt{L_{k,t}^j}} \exp\left(-\frac{|X_{k,t+1}^j - P_{k,t}^j|}{L_{k,t}^j}\right), \quad (10)$$

where $L_{k,t}^j$ represents the characteristic length of potential well δ in the j th dimension and satisfies

$$L_{k,t}^j = 2\beta |m_{best_t}^j - X_{k,t}^j|. \quad (11)$$

The probability density function of Particle K in the j th dimension is

$$Q(X_{k,t+1}^j) = \frac{1}{L_{k,t}^j} \exp\left(-\frac{2|X_{k,t+1}^j - P_{k,t}^j|}{L_{k,t}^j}\right). \quad (12)$$

Through iterative solutions from Formulae (7) to (9), the optimal values can be obtained. In the later iteration period, with less space for the possible distributions of particle attraction points and insufficient swarm diversity, the aforementioned QPSO algorithm is difficult to jump out of local extrema and hence trapped in the “prematurity” problem. Focusing on the QPSO problems in the optimization solution, this paper proposes an Improved Quantum Particle Swarm Optimization algorithm (IQPSO), striving to improve upon the QPSO in three aspects and better the algorithm searching performance.

3.2.2 Betterment Strategy

(1) Attractor Multiple Update Strategy

The QPSO algorithm obtains the retained good positions from the historically best positions of individuals and optimal positions of the swarm through the attractor shown in Formula (7), and gradually gravitates towards the global optimal position. However, as the iteration progresses, it causes a fast decline in swarm diversity and a degradation in the searching performance for the multi-extrema optimization problem, thus becoming stuck in the local optimal solution [21]. To solve this problem, the Attractor Multiple Update Strategy was introduced. Suppose the neighborhood optimal position of Particle

j is $\xi P_{\text{best}_{k,t}}$, and ξ represents the neighborhood weighing factor, replace the updated Formula with:

$$P_{k,t} = \alpha \cdot P_{\text{best}_{k,t}} + (1 - \alpha) G_{\text{best}_t}, \quad (13)$$

$$P_{k,t} = \alpha \cdot P_{\text{best}_{k,t}} + (1 - \alpha) \xi P_{\text{best}_{k,t}}, \quad (14)$$

$$P_{k,t} = \alpha \cdot P_{\text{best}_{k,t}} + (1 - \alpha) (G_{\text{best}_t} - \xi P_{\text{best}_{k,t}}). \quad (15)$$

Formula (13) is the original attractor update Formula that can select and retain the optimal solutions from the swarm's best position. Formula (14) selects the better solution from the neighborhood optimal position. Formula (15) selects and retains the better solution from the global optimal position and neighborhood best position. In the Attractor Multiple Update Strategy, the update is conducted based on comprehensive consideration for the swarm's optimal positions, the individual historically best positions and neighborhood best position. It helps increase swarm diversity effectively and enables searching for the global optimal solution. In practice, an auxiliary random weight $\omega \in (0, 1)$ is introduced. If $\omega < \xi$, the attractor will be updated according to Formula (14). If $\xi \leq \omega < 1 - \xi$, the attractor will be updated according to Formula (13). Otherwise, Formula (15) will be utilized. In practice, $\xi \in (0, \frac{1}{3})$ is often selected.

- (2) Perturbation Strategy of Potential-Well Characteristic Length
Integrating Formulae (9) and (11), we will get:

$$X_{k,t+1} = P_{k,t} + \frac{L_{k,t}^j}{2} \cdot \ln(u). \quad (16)$$

It shows that the Potential-Well characteristic length $L_{k,t}^j$ is directly related to the algorithm search speed and convergence performance. The QPSO algorithm adopts the swarm's average position to update the position, which fails to present search information for the entire group. When the swarm's best position G_{best} is only the local best, almost all particles only conduct local searches and are hardpressed to jump out of local extrema. Therefore, considering the information from different individuals, this paper introduces the Perturbation Strategy. For the swarm composed of W particles, we select θW particles randomly proportional to $\theta \in (0.1, 0.3)$ and perform perturbation to the Potential-Well characteristic length according to the Formula:

$$L_{k,t}^j = 2\beta \left| \frac{P_{\text{best}_{k,t}}^{j,\lambda_1} - P_{\text{best}_{k,t}}^{j,\lambda_2}}{2} \right|, \quad (17)$$

where λ_1 and λ_2 represent two randomly-selected particles respectively.

- (3) Dynamic Crossover Strategy

To further enhance the global swarm search ability, we introduce the crossover operation of the genetic algorithm into QPSO. During the algorithm iteration, for the t th iteration, firstly, we obtain $X_{k,t+1}$, the particle position in the $(t+1)$ th iteration according to Formula (8) and Formulae (13) to (17). Then, we perform crossover operations on $P_{\text{best}_{k,t}}$ and the individual's historically best position $P_{\text{best}_{k,t}}$, and generate the new individual position

$Z_{k,t} = \{z_{k,t+1}^1, z_{k,t+1}^2, \dots, z_{k,t+1}^N\}$. To perform crossover operations on the particle position according to the following Formula, we will get:

$$z_{k,t+1}^j = \begin{cases} X_{k,t+1}^j, & \eta^j < \gamma_c, \\ P_{\text{best}_{k,t}}^j, & \text{else,} \end{cases} \quad (18)$$

where η^j is the random number, satisfying $\eta^j \sim U(0, 1)$, and γ_c is the crossover probability. The crossover operation of the particle's historically best position is:

$$P_{\text{best}_{k,t+1}}^j = \begin{cases} z_{k,t+1}^j, & f(z_{k,t+1}^j) < f(P_{\text{best}_{k,t}}^j), \\ P_{\text{best}_{k,t}}^j, & \text{else,} \end{cases} \quad (19)$$

where $f(\cdot)$ is the fitness function.

Large crossover probability can fully retain the individual's empirical knowledge and speed up algorithm convergence. While retaining individual information, small crossover probability increases swarm diversity and enhances the global search ability. Therefore, this paper introduces it in the middle and late iteration periods. The dynamic crossover strategy is defined as:

$$\gamma_c(t) = \gamma_{\max} - (\gamma_{\max} - \gamma_{\min}) \ln[1 + (e - 1)t/I_{\max}], \quad (20)$$

where $\gamma_c \in [\gamma_{\min}, \gamma_{\max}]$, $\gamma_{\min}$ and $\gamma_{\max}$ represent respectively the given minimum and maximum crossover probabilities, and e is the natural base.

3.3 IQPSO-KFCM

The QPSO algorithm is improved according to the strategy stated in Section 3.3.3 and is used to optimize the cluster center and kernel parameters of fuzzy c-means clustering. This is the main idea of the IQPSO-KFCM algorithm proposed in this paper. As is shown in Formula (4), generally, KFCM determines the cluster center by minimizing the intra-class distance. The inter-class distance is thus ignored, and clustering accuracy is reduced. Therefore, this paper comprehensively considers the intra-class distance and inter-class distance and improves the fitness function.

3.3.1 Fitness Function Construction

According to Formula (2), the inter-class distances of different samples in KFCM are defined as:

$$\begin{aligned} D_i &= \sum_{\zeta, i=1}^M (\mu_i)^m \|\varphi(v_i) - \varphi(v_\zeta)\|^2 \\ &= 2 \sum_{\zeta, i=1}^M (\mu_i)^m (1 - K(v_i, v_\zeta)). \end{aligned} \quad (21)$$

The objective function is:

$$E_i = \frac{J_i}{D_i}, \quad (22)$$

where D_i represents the sum of the inter-class distances and describes the relation between relevant classes; J_i is the sum of intra-class distances and the objection function of the original clustering algorithm. In the objective function, a smaller J_i leads to a greater D_i and a smaller E_i will be. It indicates that a closer intra-class distance leads to a more distant inter-class distance and better clustering. Therefore, the fitness function is constructed as:

$$fit_i = \frac{1}{1 + E_i}. \quad (23)$$

Thus, a bigger fit_i leads to better clustering.

3.3.2 Algorithm Steps

The basic process of IQPSO-KFCM is: In improved quantum particle swarm optimization, the particle positions represent the feasible solutions to the optimization problem, and each position is composed of a group of cluster centers. The IQPSO and KFCM algorithms are executed in an alternating fashion to obtain the kernel parameters of the optimal cluster center and Gaussian kernel function. The detailed steps are:

Step 1: Parameter settings: Determines the number of clusters as M , variable dimension as N , error threshold as ε , maximum number of iterations as $I_{\max}$, and maximum and minimum crossover probability as $\gamma_{\min}$ and $\gamma_{\max}$ respectively;

Step 2: Initialization: Swarm size is W , particle position is $X_{k,0}$, cluster center is v_i , and the particle historically best position and swarm best position is 0;

Step 3: Calculate the fitness function according to Formulae (4) and (21) to (23), sort the historically best fitness functions in descending order (find the maximum value) and update the particle historically optimal position $P_{\text{best}_{k,t}}$ and swarm best position G_{best_t} according to the calculation result;

Step 4: Execute Steps 5 to 9 on each particle in swarm $k, 1 \leq k \leq W$;

Step 5: Determine the neighborhood optimal position of each particle according to the fitness function $\xi P_{\text{best}_{k,t}}$, execute the attractor multiple update strategy according to Formulae (13) to (15) to obtain local attractor $P_{k,t}$;

Step 6: Execute the perturbation strategy of the Potential-Well characteristic length and update the particle position $X_{k,t+1}$ according to Formulae (16) and (17);

Step 7: Execute the dynamic crossover strategy and update the particle positions according to Formula (18);

Step 8: Update the particle historically best position $P_{\text{best}_{k,t}}^j$ according to Formula (19);

Step 9: Compare the fitness functions and update the swarm best position G_{best_t} ;

Step 10: Update the crossover probability according to Formula (20);

Step 11: Keep repeating Steps 4 to 10 until the two fitness functions of two adjacent iterations meet the error threshold, or the maximum number of iterations is reached. Stop the iterations and the final result $G_{\text{best}_{t+1}}$ will be the optimal solution obtained by IQPSO-KFCM and $f(G_{\text{best}_{t+1}})$ the corresponding best indicator.

3.4 Algorithm Validation

The IQPSO-KFCM algorithm proposed in this paper consists of two parts, the improved quantum particle swarm optimization and IQPSP-based kernel fuzzy C-means clustering. The two parts will be validated respectively. All validation tests and simulations are based on Windows 10 Pro Edition (×64), configured with an i7-8550U CPU, 1.80 GHz CPU Frequency, 8 GB RAM and Matlab R2016b.

3.4.1 Algorithm Validation of IQPSO

To illustrate the improvements in the IQPSO algorithm in its ability to search for optimal solutions, this paper introduces 6 commonly-used benchmark functions [25,26], as shown in Tab. 2, uses QPSO and IQPSO for optimization and conducts tests with the maximum number of iterations, and the error threshold of the objective function as the termination condition respectively.

Table 2: Benchmark function set

Function	Formula	Feasible region	Optimal value
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	$[-100, 100]$	0
Schwefel	$f_2(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	$[-5, 5]$	0
Rosenbrock	$f_3(x) = \sum_{i=1}^{n-1} \left[100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2 \right]$	$[-100, 100]$	0
Griewank	$f_5(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$	0
Ackley	$f_4(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{30} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{30} \sum_{i=1}^n \cos 2\pi x_i\right) + 20 + e$	$[-32, 32]$	0
Schaffer	$f_6(x) = \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2} + 0.5$	$[-100, 100]$	0

Take the number of iterations as the only termination conditions. Set the maximum number of iterations as 1000 and the swarm size as 50. Employ QPSO and IQPSO and run them 50 times each. Take the statistical mean of the mean and standard deviation of the objective function as the performance evaluation metrics for the algorithms. Parameter settings: The feasible region of variable initialization is shown in Tab. 2, where the crossover probability $\gamma_{\min} = 0.7$, $\gamma_{\max} = 0.9$. The optimization problem for different variable dimensions is solved, and the optimization results are recorded in Tab. 3. By contrast, it can be found that, whether it is mean or standard deviation, the results obtained from IQPSO optimization are smaller than that from QPSO optimization, and IQPSO archives strike improvements in optimization accuracy. For the solution to the low-dimensional optimization problem, all test functions achieve good optimization effects. For the solution to the high-dimensional problem, all functions, except the Rosenbrock function can obtain the ideal optimal solution. For the Rosenbrock function, although the improved method in this paper can further approximate the optimal solution, there are still some errors. Hence, the IQPSO algorithm proposed in this paper is good enough to solve the low-dimensional optimization problem but slightly deficient for solving the high-dimensional problem. To demonstrate the specific search process for the optimal solution, one result of the 50-dimension solutions is

randomly chosen and graphed for analysis, as shown in Figs. 4 to 9. By comparison, it is found that, whether in mean or standard deviation, the IQPSO algorithm proposed in this paper achieves better accuracy than QPSO. In the later iteration period, the QPSO algorithm is stuck in the local extrema. On the contrary, thanks to the introduction of the improvement strategy, IQPSO enjoys better swarm diversity, hence it is able to jump out of the local extrema, search, and obtain the global best solution.

Table 3: Statistical analysis of optimization results

Function	Experiment No.	Problem dimension	QPSO		IQPSO	
			Mean value	Standard deviation	Mean value	Standard deviation
Sphere	50	50	3.51×10^{-5}	4.32×10^{-6}	1.75×10^{-11}	5.67×10^{-12}
		25	2.08×10^{-12}	1.57×10^{-13}	5.18×10^{-18}	1.52×10^{-18}
Schwefel	50	50	1.08×10^{-2}	3.05×10^{-2}	1.12×10^{-7}	8.67×10^{-8}
		25	8.14×10^{-10}	3.08×10^{-10}	6.19×10^{-15}	4.35×10^{-16}
Rosenbrock	50	50	$2.84 \times 10^{+1}$	5.75×10^{-1}	7.90×10^{-1}	2.15×10^{-1}
		25	1.17×10^{-8}	5.01×10^{-9}	2.44×10^{-13}	3.12×10^{-13}
Griewank	50	50	1.99×10^{-6}	4.06×10^{-6}	1.13×10^{-12}	2.13×10^{-13}
		25	6.19×10^{-14}	1.62×10^{-14}	8.65×10^{-19}	2.14×10^{-20}
Ackley	50	50	4.42×10^{-3}	1.32×10^{-3}	2.83×10^{-6}	1.02×10^{-6}
		25	1.49×10^{-10}	2.88×10^{-11}	3.71×10^{-18}	9.80×10^{-20}
Schaffer	50	2	4.39×10^{-12}	5.11×10^{-11}	9.01×10^{-19}	1.67×10^{-19}

Take the error threshold of the objective functions as the only termination condition and set the specific parameters as above. Run QPSO and IQPSO independently 100 times, and record the average number of iterations and success rates for each solution in Tab. 4, where the success rate ϕ is calculated by Formula (24).

$$\phi = \frac{l_y}{l_{total}}, \quad (24)$$

where l_y is the number of successful tests, and l_{total} is the total number of tests.

By comparing the data in Tab. 4 it can be seen that based on the premise of meeting target accuracy, the average number of iterations for the IQPSO algorithm proposed in this paper is significantly less than that of the QPSO algorithm. This is because the QPSO algorithm lacks a corresponding strategy to ensure swarm diversity in the late iterations of the algorithm. As a result, the algorithm tends to fall into local extrema, and thus often fails to find the optimal solution under the conditions of a given accuracy and satisfy accuracy threshold conditions. By contrast, the IQPSO algorithm can obtain the optimal solution properly. Therefore, the IQPSO algorithm proposed in this paper has a better convergence speed and accuracy.

3.4.2 Algorithm Validation of IQPSO-KFCM

To validate the effect of IQPSO-KFCM in clustering applications, the UCI Dataset, including Iris, Wine, Blance-archive, Vote, Heart, and Australia, are introduced to conduct clustering tests on the algorithm [27]. The parameters of the dataset are shown in Tab. 5.

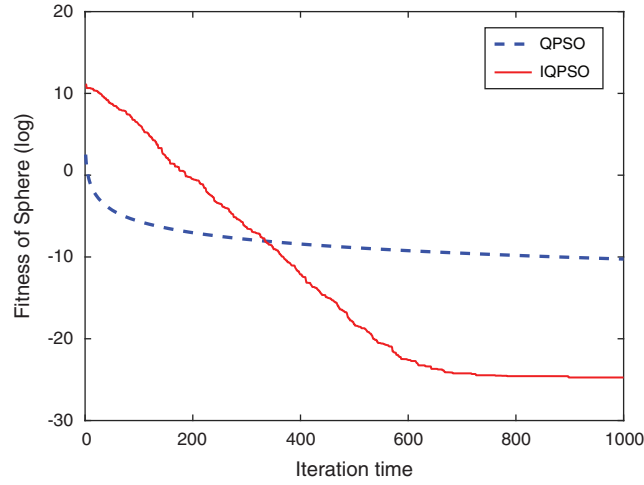


Figure 4: Fitness curve of Sphere function

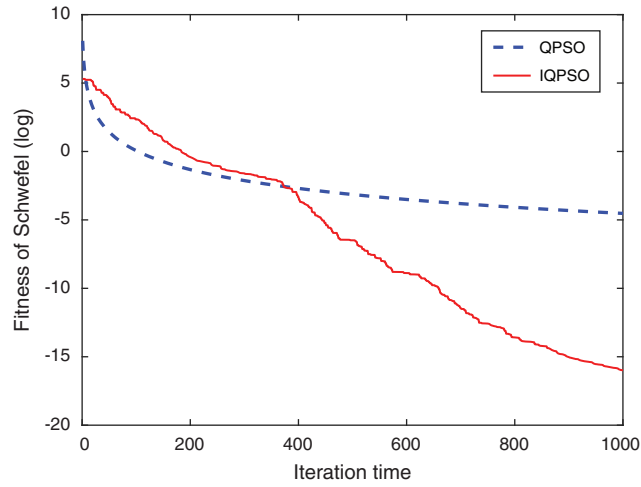


Figure 5: Fitness curve of Schwefel function

KFCM and IQPSO-KFCM are used to conduct clustering analysis on the aforementioned UCI dataset. Clustering is performed 100 times. The success rate index is defined as:

$$\tilde{\phi} = \frac{N_y}{N_{total}}, \quad (25)$$

where N_y is the number of accurate clustering samples, N_{total} is the total number of samples. The average accuracy of 100 tests is recorded in [Tab. 6](#).

The experimental data in [Tab. 6](#) shows the clustering results of 3 clustering algorithms on 6 UCI data. The proposed algorithm shows the best clustering performance on all UCI data sets. It is easy to observe that the data in columns 3 and 4 in [Tab. 6](#) prove that the KFCM clustering performance based on the optimized algorithm has been greatly improved. This shows

that the optimization algorithm can indeed find a good clustering center, thereby improving the final clustering effect from the algorithm. The data in the fourth column of Tab. 6 is slightly improved compared to the third column, which shows that IQPSO can indeed prevent QPSO from falling into local extreme values, thereby improving IQPSP-KFCM clustering performance.

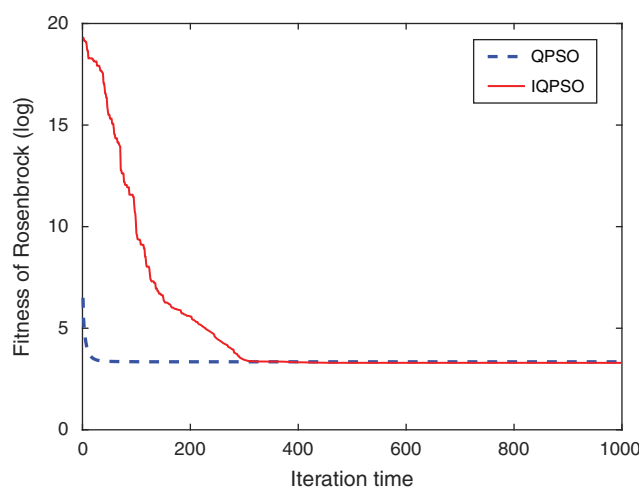


Figure 6: Fitness curve of Rosenbrock function

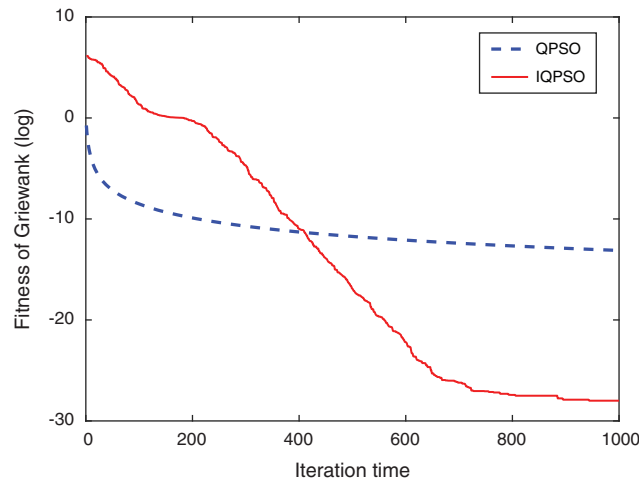


Figure 7: Fitness curve of Griewank function

4 Membership Identification of Low-Voltage Station Areas Based on IQPSO-KFCM

The IQPSO-KFCM algorithm proposed in this paper is adopted to find identification solutions to the user-transformer relations based on the zero-crossing shift data of on-site low-voltage energy meters and transformers. As shown in Fig. 10, the data is collected based on the existing equipment from the energy meters and concentrators at some station areas in some coastal city and pooled at the main station for further processing. There are four transformers connected

to 234 energy meters. The specific zero-crossing shift data at the transformer side and energy meter side are shown in [Tabs. 7 and 8](#). Thus, the problem is about how to use the IQPSOKFCM algorithm to conduct clustering analysis and classify the dataset composed of 234 energy meters into 4 classes.

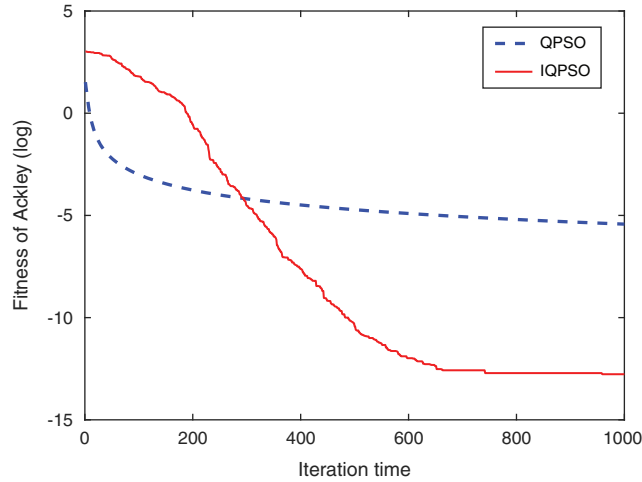


Figure 8: Fitness curve of Ackley function

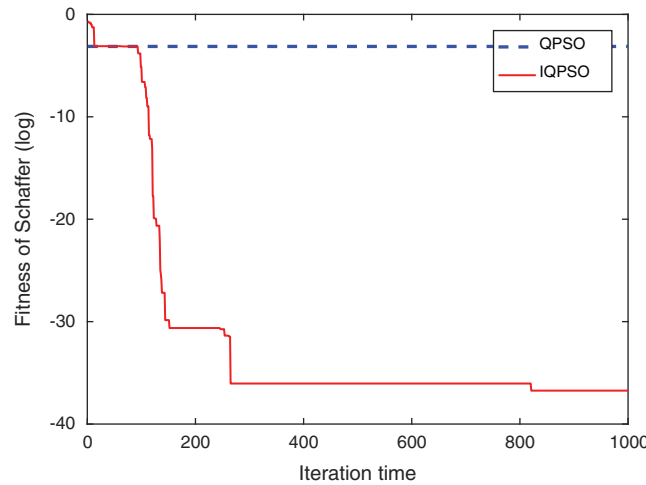


Figure 9: Fitness curve of Schaffer function

Parameter settings: Number of clustering $M = 4$, variable vector $N = 234$, error threshold $\varepsilon = 10^{-3}$, maximum number of iterations $I_{\max} = 1000$, minimum crossover probability $\gamma_{\min} = 0.7$, maximum crossover probability $\gamma_{\max} = 0.9$, Gaussian function as kernel function, swarm size $W = 300$, initial position of the particle $X_{k,0}$ and cluster center v_i are determined randomly. When the algorithm satisfies either one of the conditions, the error threshold or the maximum number of iterations, the clustering process can be stopped and the optimal solution obtained. To compare the clustering effect, KFCM is introduced to compare with the proposed algorithm. The specific clustering results are shown in [Fig. 11](#).

Table 4: Statistics of average iteration number and average optimization success rate

Function	Accuracy	QPSO		IQPSO	
		Average iteration times	Success rate	Average iteration times	Success rate
Sphere	10^{-4}	596	100%	362	100%
Schwefel	10^{-4}	961	60%	675	100%
Rosenbrock	10^{-1}	347	100%	260	100%
Griewank	10^{-4}	544	100%	363	100%
Ackley	10^{-4}	882	34%	460	92%
Schaffer	10^{-4}	791	88%	88	100%

Table 5: UCI data set

Data set	Dimension	Samples	Categories
Iris	4	150	3
Wine	13	178	3
Blance-archive	4	625	3
Vote	16	435	2
Heart	14	270	2
Australia	15	690	2

Table 6: Clustering result comparisons

Data set	KFCM (%)	QPSO-KFCM (%)	IQPSO-KFCM (%)
Iris	90.10	97.44	99.52
Wine	71.88	91.88	98.69
Blance-archive	62.20	84.51	91.6
Vote	86.46	89.73	95.10
Heart	60.30	80.97	87.01
Australia	62.00	83.08	94.73

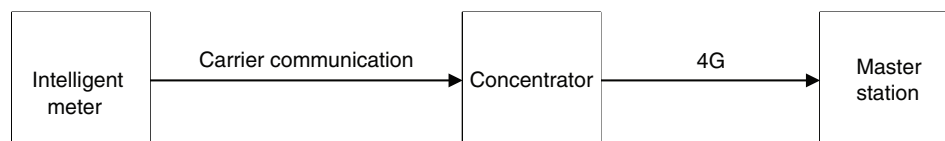
**Figure 10:** Schematic diagram of field data acquisition

Table 7: Zero-crossing shift field data on transformer side

Area number	Zero-crossing shift on transformer side (μ s)
1	320.221
2	58.894
3	73.237
4	81.054

Table 8: Zero-crossing shift field data on electric energy meter side

1 ~26	27 ~52	53 ~78	79 ~104	105 ~130	131 ~156	157 ~182	183 ~208	209 ~234
331.715	329.401	51.426	55.185	63.987	61.522	76.728	72.023	80.810
333.076	323.034	57.072	53.347	59.537	61.093	69.037	75.372	84.115
306.367	335.536	68.559	66.599	68.825	69.271	74.393	75.929	85.355
300.600	313.086	58.285	58.518	53.055	65.248	71.233	74.798	76.436
318.230	325.377	57.379	53.962	50.374	63.054	68.200	75.318	77.947
335.609	321.649	67.018	66.171	66.032	57.244	74.654	70.426	78.555
312.304	329.769	65.952	62.584	52.086	55.214	76.712	77.279	80.156
339.666	309.901	57.005	54.696	59.946	69.199	72.753	73.174	80.876
306.100	334.042	65.528	50.338	50.729	77.138	72.639	78.774	82.804
308.199	319.491	50.969	56.792	59.941	68.743	77.207	82.228	79.583
337.243	338.585	61.805	57.390	67.087	72.715	73.267	80.536	78.098
320.683	315.728	53.165	66.139	51.701	71.605	74.804	78.663	79.646
304.925	308.889	68.752	60.587	57.151	71.056	69.884	79.699	85.318
314.660	325.831	54.170	53.682	58.492	69.744	69.357	77.006	82.048
310.800	318.004	58.161	62.855	50.146	71.507	72.255	81.673	79.836
332.459	307.239	61.214	53.367	62.715	77.436	76.709	82.506	85.383
322.868	306.330	58.253	54.203	56.696	68.174	76.104	84.473	85.752
316.864	324.780	55.380	58.006	55.642	71.490	77.675	78.449	85.640
331.212	317.347	65.309	50.024	55.999	70.332	70.246	76.264	78.580
331.999	62.776	54.129	61.750	54.355	73.642	73.325	82.459	79.736
329.360	54.473	62.293	53.236	58.534	74.180	71.865	84.487	81.243
312.711	53.387	68.818	60.265	52.043	69.101	70.147	79.978	78.848
315.836	57.819	58.210	60.663	60.639	73.640	75.708	85.306	78.922
314.922	59.180	50.534	65.440	60.985	75.553	77.994	78.639	81.635
316.599	66.909	60.083	57.869	60.324	71.329	70.505	78.823	77.787
300.876	60.666	50.455	52.179	60.743	75.786	68.686	82.792	84.171

Note: 1 ~234 are the numbers of the intelligent meter, the other data is the zero-crossing shift, the unit is μ s.

In Fig. 11, the actual user-transformer relation refers to the class membership of the physical energy meters to the transformer, which can be determined by switch-in and switch-out operations. To facilitate data analysis, the results are categorized and recorded in Tab. 9. Through comparison, it can be seen that the cluster centers obtained by KFCM and IQPSOKFCM clustering are basically consistent with the zero-crossing shifts at the transformer side. By using the proposed

method, the clustering results show an accuracy of 99.15%, different from the authentic user-transformer relation of only two energy meters. Compared with that of the KFCM algorithm, the accuracy is increased by 10.26% with running time shortened by 37.35 s, an indication that the proposed method has a higher accuracy and computing efficiency. The zero-crossing shift data is fundamentally consistent in the same station area but significantly different in different station areas. To classify the user-transformer relations according to this theory, the IQPSO-KFCM algorithm proposed in this paper is very effective in identifying user-transformer relations in two station areas that are quite similar. It can be seen through analysis that in both clustering analysis methods, the energy meters in question are very close in the zero-crossing shift data. In reality, they are usually energy meters located at the junctions between two station areas. With the development of the power supply, distribution network and on-site line transformation, there are many intersections in the actual station areas, making it hard to distinguish the station areas clearly. However, the method proposed in this paper can handle this well, and thus will have an extremely wide range of application scenarios where it can actually identify the user-transformer relations.

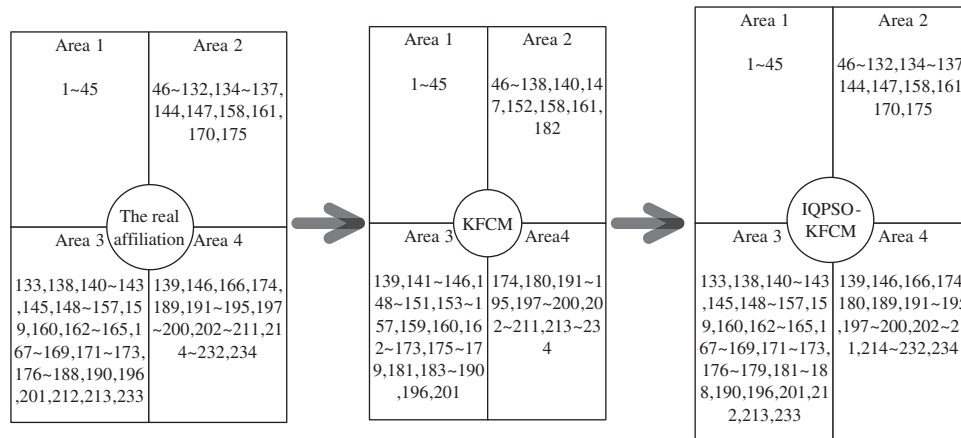


Figure 11: Result of user-transformer relation identification

Table 9: Statistics result of user-transformer relations identification

Method	KFCM	QPSO-KFCM
Cluster center	[320.007 63.426 74.663 82.116]	[320.153 59.246 73.277 80.918]
Kernel parameters	[2.377 0.988 2.046 2.523]	[1.186 1.072 1.038 1.121]
The wrong meter in 4 areas	[0 9 13 6]	[0 0 1 1]
Accuracy (%)	88.89	99.15
Running time (s)	88.77	51.42

5 Conclusion

Observing the intricate low-voltage power grid, messy user-transformer relations, and management difficulty in China, the paper proposes an identification method for user-transformer relations based on IQPSO and KFCM. Its main contributions includes the following parts: (1) An improved PSO optimization algorithm (IQPSO) is proposed. The Attractor Multiple Update Strategy, Perturbation Strategy of Potential-Well Characteristic Length and Dynamic Crossover

Strategies are introduced in IQPSO, which avoids PSO from falling into local optimal problems and improves global optimization of the optimization algorithm. (2) An improved FCM algorithm (KFCM) is proposed. KFCM introduced a kernel function to solve the linear inseparability problem. To make full use of data, KFCM introduces a fitness function based on the distance between classes to improve clustering performance for the clustering algorithm. (3) Adopting 6 benchmark functions and 5 UCI datasets to simulate the IQPSO algorithm and IQPSO-KFCM algorithm respectively, and fully explaining the superiority of the algorithm in global optimization and cluster analysis; (4) Adopting the IQPSO-KFCM algorithm to solve the problem of identifying the actual user-transformer relations and properly achieving the goal.

However, based on zero-crossing shift data, the proposed method can only determine the class membership of the energy meters to the transformer but can't sort the membership phases of energy meters. Next, we are going to analyze the grid voltages and current data and explore applying the fuzzy clustering algorithm in phase splitting of energy meters.

Funding Statement: This work was supported by the National Natural Science Foundation of China (61671208).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Yuan, C. (2014). *The development and manufacture of a multi-function equipment for low voltage area identified*. Dalian, China: Master. Dalian University of Technology.
2. Zhao, Y. D. (2016). An automatic determination method of smart meters transformer area distinguish. *North China Electric Power*, 5, 22–28.
3. Zhao, G., Jiang, C., Liu, J. (2019). Research on the system and control strategy of an AC–DC hybrid single-phase electric energy router. *Electronics*, 8(9), 970–989. DOI 10.3390/electronics8090970.
4. Ye, A., Gu, J., Zhang, X. (2018). Intelligent transformer room recognition technology based on temporal and spatial data correlation. *Modern Electronics Technique*, 41, 30–33.
5. Zhang, L., Lu, W., Liu, X., Pedrycz, W., Zhong, C. (2016). Fuzzy C-means clustering of incomplete data based on probabilistic information granules of missing values. *Knowledge-Based System*, 99(C), 51–70. DOI 10.1016/j.knosys.2016.01.048.
6. Liao, L., Zhou, J., Zou, Q. (2013). Weighted fuzzy kernel-clustering algorithm with adaptive differential evolution and its application on flood classification. *Natural Hazards*, 69(1), 279–293. DOI 10.1007/s11069-013-0707-x.
7. Arslan, O., Dan, P. G., Koditschek, D. E. (2017). Coordinated robot navigation via hierarchical clustering. *IEEE Transactions on Robotics*, 32(2), 352–371. DOI 10.1109/TRO.2016.2524018.
8. DePrince, A. E., III, Kennedy, M. R., Sumpter, B. G., Sherrill, C. D. (2014). Density-fitted singles and doubles coupled cluster on graphics processing units. *Molecular Physics*, 112(5–6), 844–852. DOI 10.1080/00268976.2013.874599.
9. Hajek, B., Wu, Y. H., Xu, J. M. (2014). Achieving exact cluster recovery threshold via semidefinite programming. *IEEE Transactions on Information Theory*, 62(5), 2788–2797. DOI 10.1109/TIT.2016.2546280.
10. Wade, S., Ghahramani, Z. (2018). Bayesian cluster analysis: Point estimation and credible balls (with discussion). *Statistics*, 13, 559–626.
11. Javadi, S., Hashemy, S. M., Mohammadi, K., Howard, K. W. F., Neshat, A. (2017). Classification of aquifer vulnerability using K-means cluster analysis. *Journal of Hydrology*, 549, 27–37. DOI 10.1016/j.jhydrol.2017.03.060.

12. Granato, D., Santos, J. S., Escher, G. B., Ferreira, B. L., Maggio, R. M. (2018). Use of principal component analysis (PCA) and hierarchical cluster analysis (HCA) for multivariate association between bioactive compounds and functional properties in foods: A critical perspective. *Trends in Food Science & Technology*, 72, 83–90. DOI 10.1016/j.tifs.2017.12.006.
13. Jin, L., Fu, M. Y. (2013). Images segmentation based on genetic_kernel fuzzy C-means clustering algorithm. *Pattern Recognition & Artificial Intelligence*, 26(2), 205–210.
14. Liu, Y., Liu, F., Hou, T., Zhang, X. (2016). Kernel-based fuzzy C-means clustering method based on parameter optimization. *Journal of Jilin University (Engineering Science)*, 46(1), 246–251.
15. Chen, L., Chen, C. L. P., Lu, M. (2011). A multiple-kernel fuzzy C-means algorithm for image segmentation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(5), 1263–1274. DOI 10.1109/TSMCB.2011.2124455.
16. Ding, Y., Fu, X. (2015). Kernel-based fuzzy C-means clustering algorithm based on genetic algorithm. *Neurocomputing*, 188, 233–238. DOI 10.1016/j.neucom.2015.01.106.
17. Liu, Y. L., An, J. C. (2018). Optimized kernel fuzzy C-means clustering algorithm. *Microelectronics & Computer*, 35(2), 79–83.
18. Huang, B. M. (2014). *Researches in kernel-based fuzzy C-means clustering algorithm based on GA optimization*. Wuhan, China: Master, Wuhan University of Science and Technology.
19. Sun, J., Feng, B., Xu, W. B. (2004). Particle swarm optimization with particles having quantum behavior. *Proceedings of the 2004 Congress on Evolutionary Computation*, pp. 325–331, Portland, OR, USA.
20. Zhang, Y. Y., Jin, Z. G. (2018). Quantum particle swarm optimization algorithm for high-dimensional multi-modal optimization. *Journal of Harbin Institute of Technology*, 50(11), 50–58.
21. Li, Y. C., Li, Z. P., Yang, L. Q., Wang, B. (2017). An improved quantum differential evolution algorithm for optimization and control in power systems including DGs. *Acta Automatica Sinica*, 43(7), 195–203.
22. Chen, H., Zhu, J., Ruan, Y., Liu, Z., Zhao, S. (2016). Quantum particle swarm optimization algorithm with crossover operator. *Journal of Southeast University (Natural Science Edition)*, 46(1), 23–29.
23. Zhang, L., Nie, Y. (2016). Quantum behaved particle swarm optimization algorithm merging differential evolution. *Computer Simulation*, 33(2), 313–316.
24. Duan, Q., Li, Q. (2017). Rational solutions of the general nonlinear schrödinger equation with derivative. *Mathematics in Practice and Theory*, 47(3), 224–230.
25. Xie, B., Feng, X. (2017). Toward efficient and accurate interface capturing on arbitrary hybrid unstructured grids: The THINC method with quadratic surface representation and Gaussian quadrature. *Journal of Computational Physics*, 349, 415–440. DOI 10.1016/j.jcp.2017.08.028.
26. Sun, X. Y., Chen, S. S., Gong, D. W., Zhang, Y. (2014). Weighted multi-output gaussian process-based surrogate of interactive genetic algorithm with individual 0s interval fitness. *Acta Automatica Sinica*, 40(2), 172–184.
27. Pang, N., Zhang, J. F., Qin, X. (2018). A subspace clustering algorithm of categorical data using multiple attribute weights. *Acta Automatica Sinica*, 44(3), 517–532.