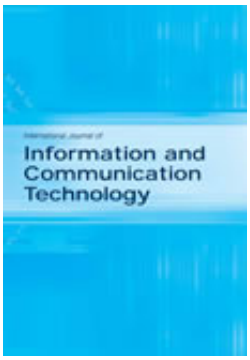
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

A radial basis function neural network algorithm based on quantum controlled NOT gate and orthogonal least squares theory

Wei Peng, Guoqing Hu, Jiahang Li, Chengzhi Lyu

DOI: 10.1504/IJICT.2025.10075004

Article History:

Received:	16 October 2025
Last revised:	29 October 2025
Accepted:	31 October 2025
Published online:	05 January 2026

A radial basis function neural network algorithm based on quantum controlled NOT gate and orthogonal least squares theory

Wei Peng and Guoqing Hu*

School of Mechanical and Automotive Engineering,
South China University of Technology,
Guangzhou, Guangdong, 510640, China
Email: pengwei8788788@163.com
Email: gqhu168@163.com
*Corresponding author

Ji Li

School of Aeronautical Manufacturing and Mechanical Engineering,
Nanchang Hangkong University,
Nanchang, Jiangxi, 330063, China
Email: lj@nchu.edu.cn

Chengzhi Lyu

School of Mechanical and Automotive Engineering,
South China University of Technology,
Guangzhou, Guangdong, 510640, China
Email: lyuchengzhi@163.com

Abstract: Temperature compensation is crucial for improving sensor accuracy and stability in high-precision measurement. Although radial basis function (RBF) neural networks perform well in nonlinear modelling, they face slow convergence, long training time, and limited accuracy. To address these issues, this paper proposes an improved RBF algorithm (QOLS-RBF) by combining quantum controlled-NOT (C-NOT) gates with orthogonal least squares (OLS) theory. The method quantises input data and applies quantum superposition, entanglement, and interference to enhance feature extraction and centre aggregation. It further integrates OLS screening with the maximum error compression ratio, using C-NOT gate evolution to reduce hidden layer nodes and accelerate convergence. Experiments with 85 training and 170 testing sensor datasets show that QOLS-RBF outperforms RBF, OLS-RBF, K-means RBF, and FCM-RBF in convergence speed, training time, error accuracy, and network compactness. This approach enables efficient temperature compensation and offers a promising tool for modelling complex nonlinear systems.

Keywords: neural network algorithm; orthogonal least squares; OLS; sensors.

Reference to this paper should be made as follows: Peng, W., Hu, G., Li, J. and Lyu, C. (2025) 'A radial basis function neural network algorithm based on quantum controlled NOT gate and orthogonal least squares theory', *Int. J. Information and Communication Technology*, Vol. 26, No. 48, pp.80–102.

Biographical notes: Wei Peng is a Doctor candidate. He will graduate from South China University of Technology in 2025. He works at the School of Mechanical and Automotive Engineering, South China University of Technology. His research interests include mechatronics.

Guoqing Hu is a Professor and has a PhD graduated from SiChuan University in 1993. He worked in School of Mechanical and Automotive Engineering, South China University of Technology. His research interests include mechatronics.

Ji Li is a Lecturer and a Doctor graduated from Xiamen University in 2016. He works in the School of Aeronautical Manufacturing and Mechanical Engineering, Nanchang Hangkong University. His research interests include intelligent mechanical equipment design.

Chengzhi Lyu is an Associate Professor and a Doctor graduated from South China University of Technology in 2018. He works in the School of Intelligent Connected Vehicles, Hubei University of Automotive Technology. His research interests include machine vision.

1 Introduction

Artificial neural networks solve nonlinear problems in complex systems by simulating their operational processes. Due to its excellent nonlinear generalisation (Bock et al., 1992), self-organisation, and self-learning abilities (Pramanik et al., 2006), artificial neural networks have gradually been widely applied in new fields of engineering, such as signal processing and pattern recognition, where they have strong predictive capabilities (Li et al., 2013).

However, most of the current popular feedforward multi-layer networks are based on backpropagation and have many drawbacks, such as the BP algorithm, which is prone to getting stuck in local optima and has slow convergence speed. Although some algorithms can effectively avoid local optima, they generally require a large amount of computation, which causes a lot of inconvenience in practical applications (Zhang et al., 2007).

The theory of radial basis function (RBF) networks provides a novel and effective means for learning multi-layer feedforward networks (Wang et al., 2022). It can not only avoid the tedious calculations in backpropagation networks and improve learning speed, but also overcome the local minima problem of gradient descent. It has wide applications in speech recognition (Yang and Chen, 2010), data classification (Ye et al., 2019), function approximation (Liang et al., 2006), time series prediction (Cao and Tay et al., 2003), graphics processing (Zhai et al., 2022), adaptive channel equalisation (Patra et al., 2008) and other fields.

Its main function is to use RBFs as the 'basis' of the hidden unit to form the space of the hidden layer (Zar et al., 2023). The hidden layer changes the input vector into a low dimensional pattern, and the input data is transformed into a high-dimensional space, making it linearly separable in high-dimensional space instead of being linearly inseparable in low dimensional space. It applies the principle of deep data mining (Song et al., 2020) to identify the fundamental differences in the data. This is a local approximation network that requires only a small number of weights and thresholds to be adjusted for each training sample. It has the characteristics of fast learning speed, good convergence, and strong real-time performance.

The basic RBF algorithm includes four algorithms: random selection of fixed centres, self-organising selection of centres, supervised selection of centres, and orthogonal least squares (OLS) method (Huang et al., 2024). However, due to the limitations of its own data structure and fixed centre values (Wan et al., 2015), traditional RBF algorithms are already at a slower convergence speed. Therefore, in order to develop a more efficient RBF neural network algorithm (Park and Sandberg, 1991), many experts have proposed some improved algorithms, which are classified as follows:

1 K-means selects the centre point: The K-means clustering method (Shen et al., 2011) is a typical objective function clustering method, which takes a certain distance function between data points as the objective function. The distance between data points and prototypes is used as the optimisation objective function, and the adjustment rules for iterative operation are obtained by finding the extremum of the function. Mainly using Euclidean distance as a similarity measure, find the optimal classification corresponding to a certain initial cluster centre vector, so as to minimise the evaluation index. The clustering points generated by the K-means clustering algorithm as the centre points of the RBF can effectively reduce the number of nodes and better achieve improved network structure (Whitehead and Choate, 2002).

2 FCM selects the centre point: The FCM algorithm is a partition-based clustering algorithm (Halim et al., 2016), which aims to maximise the similarity between objects classified into the same cluster and minimise the similarity between different clusters. The fuzzy C-means algorithm is an improvement of the ordinary C-means algorithm, which is rigid for data partitioning, while FCM is a flexible fuzzy partitioning (Geng et al., 2016). The clustering points generated by applying fuzzy clustering algorithm as the centre points of RBFs, similar to K-means method, can improve the convergence speed of the network.

3 OLSs method: It is achieved by minimising the sum of squared errors to find the optimal function match for the data, which can easily obtain unknown data and minimise the sum of squared errors between these obtained data and actual data. Applying the OLSs method to select the centre of the RBF as the training mode, one sample is selected at a time, and the orthogonal matrix is used to select each component to find the regression operator with the highest error compression ratio. Based on this, the centre point is continuously searched downwards to ultimately find a suitable network structure.

Through the above analysis, we can find that RBF optimisation can be approached from two perspectives. On the one hand, K-means or FCM algorithms can be used to cluster and optimise existing data points, ultimately obtaining the desired hidden layer

benchmark points. On the other hand, the OLS algorithm continuously filters principles to find the most suitable benchmark point. But in fact, neither of these two types of ideas has found the fundamental characteristics of data from the perspective of the most fundamental data structure, and the search speed can be further improved.

The quantum gate in quantum communication theory is a theory that applies quantum theory to deeply explore the internal relationships of data. The most commonly used controlled NOT gate (C-NOT) can vertically increase the dimensionality of the RBF, further improving the search speed of the algorithm (Wu and Byrd, 2009; Harsij and Mirza, 1995; Kak, 1995; Purushothaman and Karayiannis, 1997; Li and Zhao, 2018).

2 RBF neural network

RBF neural network is a network with a simple structure, fast convergence speed, and the ability to approximate any nonlinear function. In 1988, Broomhead and Lowe introduced RBFs into neural networks based on the principle that biological neurons have local correspondences. They were proven to have consistent approximation performance for nonlinear networks and have gradually been widely applied in different industries and fields.

The RBF is denoted as $\Phi(x, y) = \phi(\|x - y\|)$, where $\|x\|$ refers to the Euclidean norm. Treating the problem of neural networks as a regularisation problem, the solution can be given by the following equation:

$$F(x) = \sum_{i=1}^{N} w_i G(x, x_i) \tag{1}$$

Among them G is the green function, which w_i is the weight. Similar to BP neural networks, regularised RBFs also have a three-layer structure. The first m layer consists of input nodes, with the number of input nodes equal to the **x** dimension of the input vector. The second layer belongs to the hidden layer, consisting of nodes directly connected to the input nodes. Each hidden node corresponds to one training data point, so its number is the same as the number of training data points i. The output $X_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$ of the hidden node is $\phi(\|\mathbf{X} - \mathbf{X_i}\|)$, is the basis function, and is the centre of the basis function. The output layer consists of several linear units, each connected to all hidden nodes, and its final output is the linear weighted sum of each hidden layer node. The algorithm structure is shown in Figure 1.

Let the actual output be $Y_k = [y_{k1}, y_{k2}, \ldots, y_{kj}, y_{kJ}]$, is the J number of output units, and represents the k output generated by the th input vector. So outputting the training samples $\mathbf{X_k}$, the j result obtained by the network's first output neuron is

$$y_{kj} = \sum_{i=1}^{N} w_{ij} \phi(X_k, X_i),\ j = 1, 2, \ldots, J \tag{2}$$

If the basis function is a Gaussian function, then the $\phi(\mathbf{X_k}, \mathbf{X_i}) = G(\mathbf{X_k}, \mathbf{X_i}) = G(\|\mathbf{X_k} - \mathbf{X_i}\|) = \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{X_k} - \mathbf{X_i}\|\right)$ basic algorithm principle of RBF is as follows:

1 As long as there are enough hidden layer nodes, it can approximate any multivariate continuous function with arbitrary accuracy.

2 Given a nonlinear function at a given location, a set of coefficients can always be selected to make the approximation of the network optimal.

The algorithm flow is shown in Figure 2.

1 Initialisation: Determine the weight coefficients.

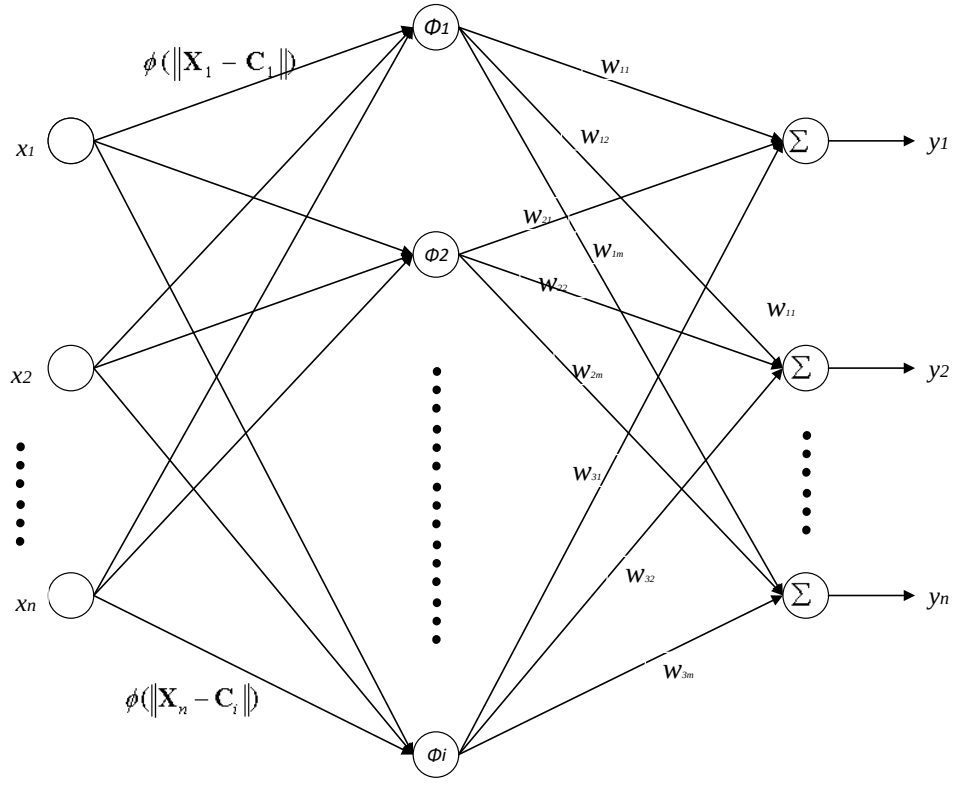
2 Select a set of inputs, the centre point of the most hidden layer, and calculate the radial basis.

3 According to the weight corresponding to each output.

4 If the final error is smaller than the target error, end the process; otherwise, continue with the above steps until the error requirements are met.

5 Select another set of data according to certain criteria and repeat the process in the second step.

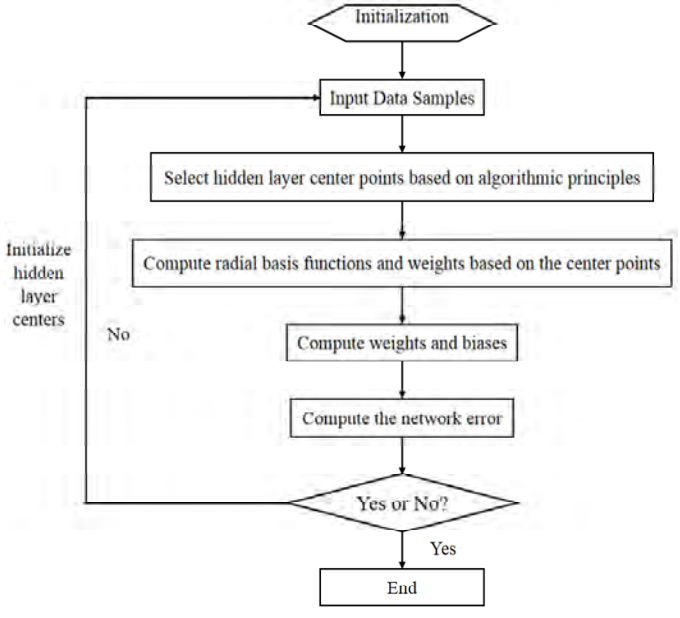
Figure 1 RBF algorithm structure diagram



From this, it can be seen that the network structure in the RBF algorithm is simple and highly efficient, and the principle of adding the most important new centre point directly affects the convergence speed of the RBF network structure. However, it did not

introduce the search for patterns from the fundamental perspective of data, which can be deeply explored.

Figure 2 RBF algorithm flowchart



3 QOLS theory

3.1 Quantum controlled NOT gate

In quantum mechanics, when the state of a microscopic particle at a certain moment is known, the state of the particle at subsequent moments is determined by the Schrödinger equation, that is:

$$i\hbar\frac{\partial}{\partial t}\psi(\mathbf{r},t)=\left(-\frac{\hbar}{2m}\nabla^2+V\right)\psi(\mathbf{r},t) \tag{3}$$

Among them, ψ is the wave function, m is the particle mass, V is the potential energy of the particle in the force field, $\nabla^2\hbar$ is the Planck constant, is the Laplace operator, which is defined as $\nabla^2=\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}$.

In classical computing, binary numbers of 0 and 1 are used to represent information, commonly referred to as bits, which can only be in two states: '0' or '1'. However $|1\rangle$, in quantum computing based on the Schrödinger equation, the $|\ \rangle$ state of a quantum bit can be linearly combined through the superposition of $|1\rangle$ two fundamental states (Dirac notation), which are commonly referred to as superposition states

$$|\varphi\rangle=\alpha|0\rangle+\beta|1\rangle \tag{4}$$

$$|\alpha|^2 + |\beta|^2 = 1 \tag{5}$$

Formulas (2) and (3) both satisfy the argument of the wave function in the Schrödinger equation, which is a complex number known as the probability amplitude of the quantum state and can be expressed as $[\alpha\ \beta]^T$. Formula (2) states that a quantum state $|\varphi\rangle$ will $|\alpha|^2$ collapse with a certain probability due to measurement $|0\rangle$, or $|\beta|^2$ collapse with a certain probability $|1\rangle$ to $|\varphi\rangle$ represent $|1\rangle$ any state that satisfies:

In quantum computing, the logic transformation function is achieved by performing some unitary transformations on the state of quantum bits, and the logic transformation is realised within a certain time interval. The quantum device that performs unitary transformations is called a quantum gate. Due to quantum phenomena such as superposition, entanglement, and interference, quantisation of data allows it to exhibit these characteristics in new dimensions. The most commonly used quantum gate is the controlled NOT gate, also known as the C-NOT gate

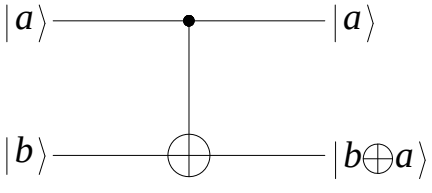
$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} : \begin{matrix} |00\rangle \to |00\rangle \\ |01\rangle \to |01\rangle \\ |10\rangle \to |11\rangle \\ |11\rangle \to |10\rangle \end{matrix} \tag{6}$$

If the control qubit of the CNOT gate is set to 0, the target qubit will remain unchanged. If the control qubit is set to 1, the target qubit will flip.

$$\begin{aligned} &|0,0\rangle \to |0,0\oplus 0\rangle = |0,0\rangle, |0,1\rangle \to |0,1\oplus 0\rangle = |0,1\rangle \\ &|1,0\rangle \to |1,0\oplus 1\rangle = |1,1\rangle, |1,1\rangle \to |1,1\oplus 1\rangle = |1,0\rangle \end{aligned} \tag{7}$$

The formal expression $\oplus$ of the equation is $|a, b\rangle \to |a, b \oplus a\rangle$, where is modulo-2 addition. The circuit of CNOT gate in quantum computing and quantum information is shown in Figure 3.

Figure 3 Quantum controlled non gate (CNOT) circuit diagram



The generalised controlled NOT gate can be derived from three general types of controlled NOT gates:

$$CNOT_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} : \begin{matrix} |00\rangle \to |01\rangle \\ |01\rangle \to |00\rangle \\ |10\rangle \to |10\rangle \\ |11\rangle \to |11\rangle \end{matrix} \tag{8}$$

$$CNOT_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} : \begin{matrix} |00\rangle \to |00\rangle \\ |01\rangle \to |11\rangle \\ |10\rangle \to |10\rangle \\ |11\rangle \to |01\rangle \end{matrix} \tag{9}$$

$$CNOT_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} : \begin{matrix} |00\rangle \to |10\rangle \\ |01\rangle \to |01\rangle \\ |10\rangle \to |00\rangle \\ |11\rangle \to |11\rangle \end{matrix} \tag{10}$$

These four types of generalised controlled NOT gates can achieve quantum data superposition, entanglement, and interference, and are widely and ingeniously applied in the field of data processing. According to research, for each different data type, generalised controlled NOT gates exhibit different improvement abilities, but overall they can discover the core internal rules of the data from another perspective.

3.2 OLSs method

The OLSs algorithm is derived from linear regression models and is used to design RBF neural networks. It has the advantages of low computational complexity, easy implementation, and the ability to determine the number of hidden nodes while learning weights. Let the training samples of the RBF neural network be $\{x_i, y_i\}$, (i = 1, 2, …, n), where n is the number of training samples and x_i is the input vector of the network. Therefore, the output of the network is:

$$y_i = \sum_{j=1}^{m} p_j(i) w_j + e_i \tag{11}$$

In the formula m, represents the number of hidden layer neurons, which can be referred to as the number of neural networks in RBF neural networks; w_j. It is a model parameter, which is the connection weight between the output layer and the hidden layer; e_t. It is a residual;

$$y = [y_1, y_2, \ldots, y_n]^T \tag{12}$$

$$\mathbf{w} = [w_1, w_2, \ldots, w_m]^T \tag{13}$$

$$\mathbf{P} = [\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_m}]^T \tag{14}$$

$$\mathbf{e} = [e_1, e_2, \ldots, e_m]^T \tag{15}$$

P for the regression matrix; **w** is a weight vector; **e** is the residual vector; **y** to expect the output response vector.

The basic idea of this algorithm is to use orthogonalisation method to anaise the contribution of each selected vector to reducing residuals, eliminate the vectors with small contributions, and leave the vectors with large contributions as the network centre.

The task of the OLS algorithm is to learn and select appropriate regression operator vectors $\mathbf{p}_j$ and their number of centres m, as well as determine the output weights w_j.

Perform **p** orthogonal triangulation on the regression matrix:

$$\mathbf{P} = \mathbf{UA} \tag{16}$$

A one of them is an $m \times m$ upper triangular matrix with diagonal elements of 1; it **HU** is a h_{jj} $n \times m$ matrix whose columns are $\mathbf{u}_j$ orthogonal, that is $\mathbf{U^TU} = \mathbf{H}$, a diagonal matrix with diagonal elements. Namely:

$$h_{jj} = u_j{}^T u_j \tag{17}$$

From the above, it can be concluded that:

$$y = UAw + e = Ug \tag{18}$$

$$g = Aw \tag{19}$$

The OLSs solution is:

$$g = \frac{u_j{}^T y}{u_j{}^T u_j} \tag{20}$$

The above orthogonalisation is achieved using the traditional Gram Schimit orthogonalisation method. This algorithm uses orthogonal decomposition and least squares method to define the u_j contribution of orthogonal vectors to reducing output errors:

$$\varepsilon_j = \frac{g_j{}^2 u_j{}^T u_j}{y^T y} (1 \le j \le m) \tag{21}$$

So as to effectively find a suitable subset of regression operators p_j and determine the centre and connection weights of RBF neurons through it. At that $1 - \sum_{j=1}^{m} \varepsilon_j = \rho$ time, the iteration ended ρ. To establish the allowable error threshold, this involves obtaining m an orthogonal matrix consisting of orthogonal vectors **U** and their corresponding triangular matrices **A**, and then calculating the weights **w**. Due to the strong correlation introduced by quantum feature expansion, the orthogonalisation capability of OLS makes it suitable for selecting representative basis functions and avoiding redundancy.

4 QOLS-RBF algorithm

The innovation of this article lies in the combination of quantum controlled-NOT (CNOT) gate theory and the OLSs method, establishing a new and efficient centre selection principle. In the proposed framework, all input data are first quantised and expanded from one-dimensional to four-dimensional representations through CNOT gate transformations, generating quantum-derived features with enhanced separability and internal correlations. Before the OLS selection process begins, these quantum-transformed data points serve directly as the initial candidate centres for the

RBF network. Unlike conventional clustering methods such as k-means, which rely on iterative partitioning and are sensitive to initialisation, the quantum pre-processing step provides a structured and well-distributed set of centres derived from the inherent entanglement and superposition properties of the data. This not only eliminates the need for an additional clustering procedure but also offers a more robust and globally informed initialisation for the network. The width of the RBF functions is then adaptively determined based on the pairwise distances among these candidate centres, ensuring sufficient coverage of the input space and stable convergence in subsequent learning.

OLS is deliberately chosen as the learning mechanism because its orthogonalisation and greedy selection process aligns well with the structure of quantum-entangled features. Specifically, OLS effectively filters redundant or weakly informative quantum basis functions and selects the subset that contributes most to error reduction, enabling compact network structures. With these quantum-enhanced data as candidate centres, radial bases are generated and then screened by OLS based on the maximum error compression ratio. This synergy between quantum feature expansion and OLS-driven selection leads to faster convergence, higher accuracy, and more representative hidden layer centres.

The innovation of this method includes the following three points:

1 Cleverly using complex quantum gates to achieve quantum transformations, stacking, entanglement, and interference between quantum data, generating new data.

2 Using the new data generated by quantum changes as the benchmark centre is equivalent to a quantum evolution of data and networks, which can distinguish from higher dimensions and achieve higher order development space.

3 We have successfully implemented the OLSs method to improve the maximum compression ratio that already has high-speed convergence, which is different from the traditional method of using existing data points as a benchmark and accelerates the calculation process of RBFs.

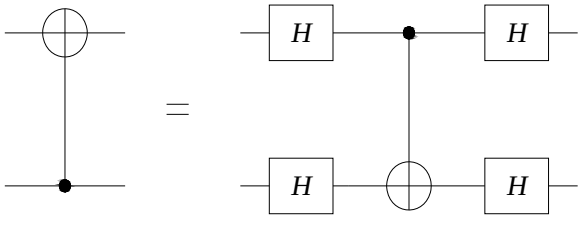
The OLSs method using quantum theory not only has the acceleration effect of quantum operations, but also has the super acceleration ability of OLSs screening for maximum compression ratio, which enables the entire RBF algorithm to undergo a new evolution in both structure and effect, reducing the time to converge to the target error and becoming a new high-speed and effective neural network algorithm.

By combining the above two algorithms, that is, organically combining the four controlled NOT gates with the OLS algorithm, it was found that the third type of generalised controlled NOT gate (CNOT3) performs the best in the process of combining the OLS algorithm. It is a combination of a standard controlled NOT gate and four Hadamard gates, as shown in Figure 4.

In the OLS algorithm, a set of quantum data with the highest error contribution is continuously stacked and used as the centre value, and continuously added upwards to find the minimum number of centre points at the fastest speed, in order to achieve high-speed and effective convergence. Importantly, because OLS is a deterministic greedy algorithm that monotonically reduces the residual error at each iteration, the convergence of the classical weight layer is theoretically guaranteed in the least-squares sense. The quantum pre-processing step introduced in this work does not alter this

convergence property; instead, it accelerates the convergence process by improving feature separability and allowing OLS to identify the most informative centres more efficiently.

Figure 4 Generalised controlled non gate decomposition circuit diagram



The algorithm steps are as follows:

1 Normalise and initialise data and programs.

2 Quantise all data, upgrade one-dimensional data to four qubit data, and perform CNOT gate transformation to achieve quantum data superposition, entanglement, and interference, completing the conversion of quantum gates.

3 Perform RBF calculations on all quantised input data for the hidden layer reference points of all CNOT gate transformations, in order to obtain the radial basis centre with the greatest impact from the perspective of quantum data.

4 Identify the column with the highest error compression ratio in the quantum data, that is, locate the centre point that has the greatest impact on the overall data representation. Extract this centre as the quantum radial basis centre point.

5 Use the selected quantum data point as the first hidden layer centre and perform RBF calculations to obtain the corresponding network parameters.

6 Compute the error of the neural network and check whether it satisfies the predefined convergence criteria. If the condition is met, terminate the algorithm; otherwise, proceed to the next iteration.

7 Update the maximum error compression ratio matrix. The column corresponding to the selected centre is orthogonalised using the Gram-Schmidt procedure, resulting in a well-conditioned upper triangular structure. This enables a stable and efficient back-substitution process to compute the linear output weights, avoiding the need for explicit matrix inversion or pseudo-inversion.

8 Return to steps 4–7, iteratively increasing the number of hidden layer centres until the network error meets the required threshold.

The algorithm flow is shown in Figure 5.

5 Experimental analysis

The QOLS-RBF algorithm is applied for temperature compensation of pressure transmitters, and the selected data are shown in Table 1.

- Experimental data source: Different differential pressure values were recorded at five temperature ranges of –20°C, 0°C, 20°C, 50°C, and 70°C. From –40 pa to 40 pa, with a gradient of 5 pa, temperature AD, pressure AD, and pressure values were recorded at 17 points. A total of 85 data points were used as training data, as shown in Table 1.

Figure 5 QOLS algorithm flowchart

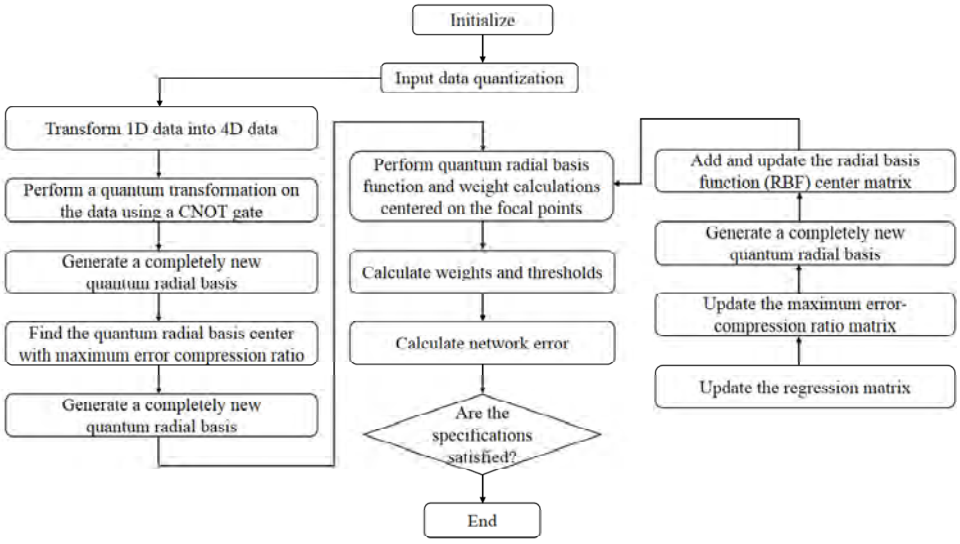


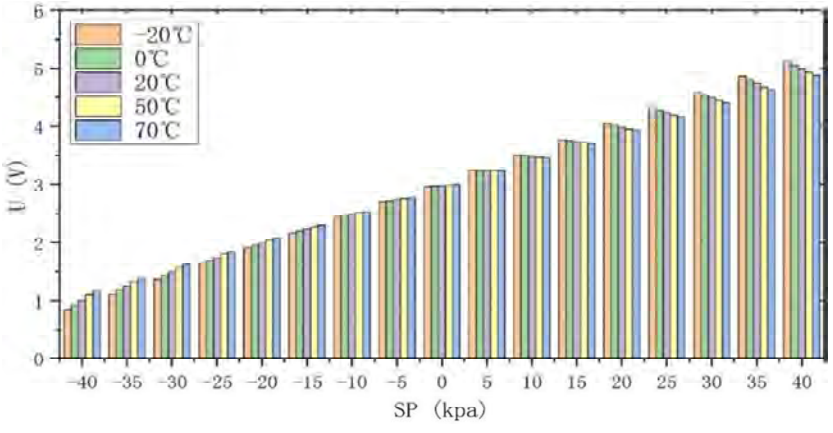
Table 1 Sensor temperature calibration points (modelling data points)

Pressure (pa)	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*
–40	46,115	225,226	20	45,268	634,383	–5	43,255	471,221
–35	46,115	259,015	25	45,268	668,039	0	43,255	502,188
–30	46,115	292,957	30	45,268	701,674	5	43,255	533,214
–25	46,115	327,038	35	45,268	735,281	10	43,255	564,373
–20	46,115	361,254	40	45,268	768,840	15	43,255	595,555
–15	46,115	395,595	–40	44,445	245,293	20	43,255	626,757
–10	46,115	430,030	–35	44,445	276,787	25	43,255	657,996
–5	46,115	464,563	–30	44,445	308,434	30	43,255	689,248
0	46,115	499,184	–25	44,445	340,227	35	43,255	720,489
5	46,115	533,847	–20	44,445	372,163	40	43,255	751,701
10	46,115	568,612	–15	44,445	404,232	–40	42,449	265,502
15	46,115	603,391	–10	44,445	436,398	–35	42,449	294,694
20	46,115	638,169	–5	44,445	468,680	–30	42,449	324,036
25	46,115	672,962	0	44,445	501,053	–25	42,449	353,526
30	46,115	707,737	5	44,445	533,475	–20	42,449	383,158
35	46,115	742,473	10	44,445	566,019	–15	42,449	412,937

Table 1 Sensor temperature calibration points (modelling data points) (continued)

Pressure (pa)	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*
40	46,115	777,148	15	44,445	598,591	–10	42,449	442,817
–40	45,268	235,926	20	44,445	631,170	–5	42,449	472,818
–35	45,268	268,467	25	44,445	663,787	0	42,449	502,916
–30	45,268	301,166	30	44,445	696,402	5	42,449	533,077
–25	45,268	334,016	35	44,445	728,999	10	42,449	563,370
–20	45,268	367,004	40	44,445	761,560	15	42,449	593,697
–15	45,268	400,131	–40	43,255	257,738	20	42,449	624,052
–10	45,268	433,352	–35	43,255	287,807	25	42,449	654,454
–5	45,268	466,694	–30	43,255	318,035	30	42,449	684,868
0	45,268	500,112	–25	43,255	348,415	35	42,449	715,282
5	45,268	533,579	–20	43,255	378,937	40	42,449	745,668
10	45,268	567,163	–15	43,255	409,595			
15	45,268	600,769	–10	43,255	440,352			

Figure 6 Pressure output characteristics at different temperatures (see online version for colours)



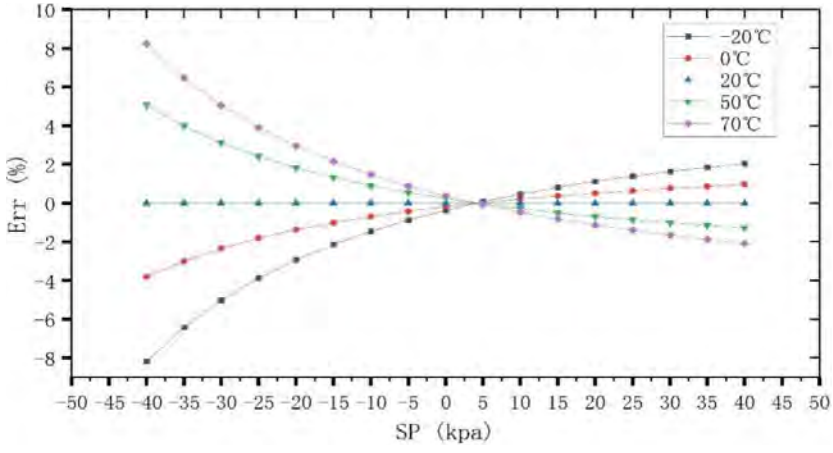
According to the analysis of Figures 6 and 7, the temperature changes of the pressure transmitter will have a direct impact on its pressure output characteristics, resulting in deviations in the obtained measurement results. At five temperatures of –20°C, 0°C, 20°C, 50°C, and 70°C, there are significant differences:

- –20°C: The sensitivity of the pressure transmitter is significantly reduced, which is affected by the cold environment, resulting in a lower output pressure value. The maximum relative error is 8.181%, which is due to factors such as material shrinkage and increased lubricant viscosity, resulting in a significant decrease in the dynamic response speed of the pressure transmitter.
- 0°C: At temperatures close to freezing, the performance of the pressure transmitter is relatively stable. Due to the still low temperature, the sensitivity and dynamic

response speed of the transmitter may still be affected to some extent. The maximum observed relative error is 3.962%, indicating a shift in performance compared to the performance at –20°C.

- 20°C: At room temperature, the performance of the pressure transmitter reaches its optimal state. At this temperature, the sensitivity and dynamic response speed of the pressure transmitter perform well, and the output pressure value is relatively accurate. Therefore, the pressure AD at 20°C is used as the standard value.
- 50°C: As the temperature increases, the pressure transmitter has experienced thermal drift, resulting in a higher output pressure value. In addition, high temperature environments may affect the performance of internal components of pressure transmitters, thereby reducing measurement accuracy. The maximum relative error value is 5.074%, and the drift is not small.
- 70°C: At this higher temperature, the thermal drift phenomenon of the pressure transmitter may be more severe, further affecting the accuracy of the output pressure value. The maximum relative error value reached 8.239%, which is quite high. The dynamic response speed of the pressure transmitter is severely affected by factors such as material thermal expansion and reduced lubricant viscosity.

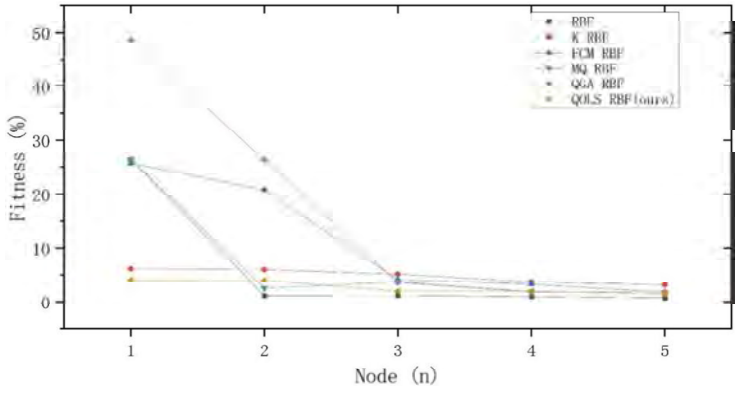
Figure 7 Relative error of 40 kPa pressure transmitter output (see online version for colours)



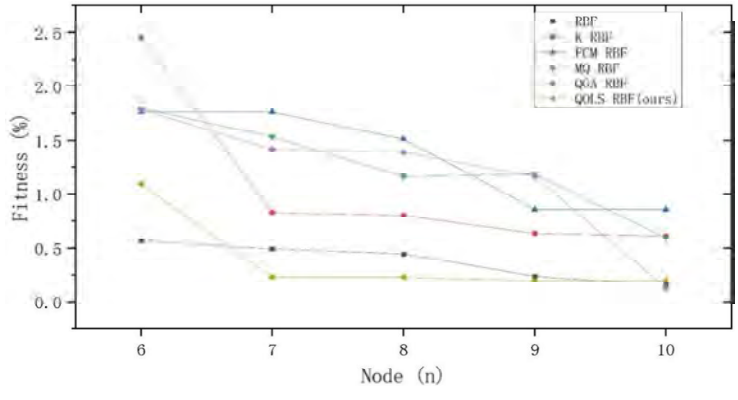
In summary, the maximum relative error occurs at 70°C. Continuing with Figure 3, it is found that as the differential pressure value increases, the relative error value also increases. The fluctuation range of –40 kPa is larger than that of 40 kPa, indicating that the negative control ability of the pressure transmitter is not as strong as the positive ability, highlighting the necessity of temperature compensation.

Using data from an additional channel, combined with Figure 8, the convergence speed of various RBF neural network compensation models in pursuing optimal adaptability is revealed, comprehensively demonstrating the dynamic characteristics of various search algorithms. This helps to understand the performance of various RBF neural network algorithms in solving temperature compensation problems in pressure transmitters. By carefully observing Figures 3–5 and anaising its results, the following conclusions can be drawn from multiple aspects.

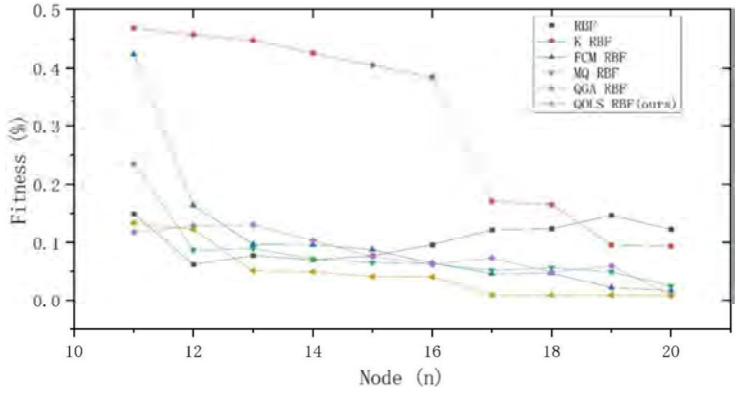
Figure 8 Changes in fitness of various algorithms, (a) algorithms for fitness changes of 1–5 nodes (b) algorithms for fitness changes of 6–10 nodes (c) change in fitness of 10–20 nodes using six algorithms (see online version for colours)



(a)



(b)



(c)

After searching for the centre point for five times, all six types of RBF neural networks entered the 4% fitness range, fully demonstrating the fast convergence characteristics of RBF neural networks. However, it is evident that QOLS-RBF and K-means RBF

converge faster. However, during the second search for the centre point, RBF quickly converged from 26.47% to 1.18%, further demonstrating the suitability of RBF neural networks for temperature compensation experiments in this stage, with QOLS-RBF having a convergence speed second only to RBF. Further analysis revealed that during the sixth iteration of searching for the centre point, QOLS-RBF had a faster convergence speed than RBF and remained ahead thereafter, while other models had their own strengths.

Due to the quantum genetic transformation of the data in the early stage, QGA's data volume increased, and its convergence speed could not be compared with QOLS-RBF and RBF before the first nine searches for the centre point. However, during the 10th search for the centre point, QGA was at the same level as QOLS-RBF, FCM-RBF, and K-means RBF, and was faster than MQ-RBF, FCM-RBF, and K-means RBF. From the performance of QOLS and QGA, quantum acceleration is effective in temperature compensation of RBF neural networks.

When entering the 0.1% range, QOLS-RBF shows a clear convergence advantage, while RBF begins to fluctuate, indicating that RBF is developing in the opposite direction of decreasing fitness. Other FCM-RBF, MQ-RBF, and QGA-RBF are relatively stable, with only MQ-RBF showing small fluctuations when searching for the centre point after 21 iterations, while others tend to stabilise.

As a clustering optimised RBF neural network, FCM-RBF has stronger convergence ability than K-means RBF. K-means RBF needs to wait until the 29th search for the centre point, while FCM-RBF needs to search for the centre point 21 times to meet the requirements.

From Figures 3–5, it can be seen that in terms of structure, QOLS-RBF is due to other RBF algorithms, and can achieve the target value after 18 iterations. However, QGA-RBF, MQ-RBF, FCM-RBF, K-means RBF, and RBF are the 21st, 23rd, 30th, 31st, and 70th iterations, respectively. This indicates that the wave motion of RBF is too strong. In the case of complex data structures for temperature compensation, the structure of RBF cannot be optimised. QOLS adopts the basic knowledge of quantum computing to optimise the RBF neural network structure, which also provides a new optimisation idea for relevant researchers. Through comparative analysis of fitness, it can be seen that QOLS-RBF has superiority over other channel data, with high algorithm accuracy, simplified algorithm structure, and fast convergence speed.

The evaluation criterion for the algorithm compensation used in this article is absolute relative error:

$$e_{rr} = \left[\frac{P_D' - P_D}{P_{FS}} \right] \tag{22}$$

Among them, e_{rr} represents absolute relative error, P_i' represents the compensated pressure value, P represents the actual pressure value, PFS represents the measurement range of the pressure transmitter, and the e_{rr} smaller it is, the higher the compensation accuracy. This project will use six pre trained RBF, K-means RBF, FCM-RBF, MQ-RBF, QGA-RBF, and QOLS-RBF models e_{rr} (max) to test the compensation effect on the test data *z* of Tables 3–9. The algorithm stop requirement is to $e_{rr} < 0.0001$ stop, and the number of nodes, testing time, error e_{rr} value, mean difference value, and e_{rr} (*mean*) maximum error value will be observed for analysis. At the same time, trend

analysis will be conducted by combining the time *x* axis, node number *y* axis, and three-dimensional response surface graph.

The test data is continuously filled with different pressure values during the heating process from –20℃ to 50℃, and the corresponding temperature AD and pressure AD are recorded as the test data. The specific data is shown in Table 2.

Table 2 Sensor temperature test point data

Pressure (pa)	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*
–40	44,363	246,247	–10	43,249	440,480	20	44,725	632,338
–35	44,367	277,604	–5	43,242	471,354	25	44,712	665,200
–30	44,372	309,122	0	43,237	502,298	30	44,698	698,028
–25	44,376	340,797	5	43,233	533,292	35	44,688	730,805
–20	44,380	372,626	10	43,228	564,401	40	44,674	763,525
–15	44,384	404,593	15	43,215	595,519	–40	44,381	246,283
–10	44,388	436,668	20	43,209	626,644	–35	44,397	277,540
–5	44,393	468,858	25	43,204	657,808	–30	44,418	308,964
0	44,398	501,160	30	43,199	688,966	–25	44,434	340,579
5	44,400	533,520	35	43,195	720,105	–20	44,454	372,386
10	44,404	566,013	40	43,189	751,212	–15	44,471	404,362
15	44,408	598,541	–40	42,931	261,177	–10	44,491	436,370
20	44,412	631,085	–35	42,921	290,949	–5	44,506	468,747
25	44,414	663,668	–30	42,916	320,829	0	44,526	501,146
30	44,417	696,255	–25	42,907	350,854	5	44,546	533,648
35	44,419	728,824	–20	42,902	381,000	10	44,557	566,320
40	44,421	761,364	–15	42,895	411,269	15	44,573	599,041
–40	44,204	248,034	–10	42,888	441,626	20	44,592	631,834
–35	44,192	279,323	–5	42,883	472,091	25	44,608	664,676
–30	44,186	310,705	0	42,876	502,630	30	44,623	697,544
–25	44,179	342,229	5	42,867	533,220	35	44,647	730,505
–20	44,173	373,869	10	42,860	563,920	40	44,665	763,434
–15	44,169	405,618	15	42,853	594,635	–40	43,515	255,479
–10	44,163	437,459	20	42,848	625,369	–35	43,466	286,241
–5	44,158	469,403	25	42,841	656,129	–30	43,447	316,802
0	44,149	501,426	30	42,837	686,892	–25	43,439	347,432
5	44,143	533,482	35	42,830	717,638	–20	43,428	378,203
10	44,136	565,641	40	42,825	748,351	–15	43,421	409,066
15	44,131	597,813	–40	45,860	228,779	–10	43,415	440,009
20	44,125	629,987	–35	45,848	262,314	–5	43,411	471,044
25	44,118	662,169	–30	45,835	295,965	0	43,403	502,177
30	44,111	694,350	–25	45,827	329,708	5	43,398	533,356
35	44,101	726,474	–20	45,810	363,585	10	43,393	564,654

Table 2 Sensor temperature test point data (continued)

Pressure (pa)	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*	*Pressure (pa)*	*Temperature AD*	*Pressure AD*
40	44,096	758,565	–15	45,793	397,554	15	43,389	595,971
–40	43,589	254,523	–10	45,779	431,552	20	43,385	627,293
–35	43,603	284,866	–5	45,768	465,629	25	43,379	658,641
–30	43,612	315,422	0	45,744	499,762	30	43,376	689,994
–25	43,618	346,161	5	45,723	533,857	35	43,371	721,335
–20	43,625	377,050	10	45,708	568,007	40	43,368	752,642
–15	43,635	408,081	15	45,693	602,135	–40	43,340	256,971
–10	43,644	439,242	20	45,678	636,222	–35	43,340	287,126
–5	43,650	470,542	25	45,657	670,259	–30	43,341	317,445
0	43,656	501,949	30	45,641	704,258	–25	43,342	347,911
5	43,661	533,425	35	45,626	738,204	–20	43,343	378,517
10	43,667	565,032	40	45,610	772,020	–15	43,344	409,263
15	43,675	596,699	–40	44,918	240,304	–10	43,346	440,106
20	43,682	628,383	–35	44,902	272,580	–5	43,348	471,065
25	43,691	660,133	–30	44,875	305,026	0	43,349	502,129
30	43,701	691,923	–25	44,859	337,473	5	43,349	533,253
35	43,708	723,692	–20	44,843	370,001	10	43,348	564,511
40	43,716	755,467	–15	44,824	402,631	15	43,349	595,795
–40	43,308	257,391	–10	44,813	435,309	20	43,349	627,099
–35	43,294	287,639	–5	44,798	468,045	25	43,350	658,434
–30	43,284	317,967	0	44,783	500,857	30	43,350	689,786
–25	43,270	348,379	5	44,769	533,673	35	43,350	721,134
–20	43,260	379,051	10	44,749	566,584	40	43,350	752,465
–15	43,254	409,711	15	44,735	599,473			

Table 3 Comparison results of five RBF algorithms

Algorithm	*Nodes*	*Calculation time (ms)*	*Network error*	*Mean test error*	*Maximum testing error*
RBF	52	3,646	8.4×10^{-5}	1.119×10^{-2}	1.275×10^{-1}
K means-RBF	31	3,635	6.6×10^{-5}	2.56×10^{-4}	1.023×10^{-3}
FCM-RBF	28	3,625	8.3×10^{-5}	4.131×10^{-3}	1.875×10^{-2}
MQ-RBF	21	3,814	7.7×10^{-5}	5.189×10^{-3}	3.514×10^{-2}
QGA-RBF	22	9,760	9.8×10^{-5}	5.096×10^{-3}	3.003×10^{-2}
QOLS-RBF	17	1,880	8.2×10^{-5}	1.24×10^{-4}	5.60×10^{-4}

Through algorithm experiments using RBF, OLS-RBF, K-means RBF, FCM-RBF, and OLS-RBF, as well as comparing QOLS-RBF algorithm with these four algorithms, 85 data points were modelled and 170 dynamic data points were selected for testing. Through experiments, the number of nodes, network errors, testing errors, and

computation time of these five algorithms were observed for analysis. The results are in Table 3.

Based on the analysis of the experimental results above, the following conclusions can be drawn:

1 K-means RBF, FCM-RBF, MQ-RBF, QGA-RBF, and QOLS-RBF all have fewer nodes than RBF. Among them, QOLS-RBF has the most significant effect, reducing the number of nodes from 52 to 17. Similarly, QGA-RBF, after quantum optimisation, can also reduce the number of nodes to 22, proving that quantum optimisation theory can streamline the structure of RBF neural networks.

2 From the perspective of computation time, most RBF neural algorithms are within four seconds, except for QGA-RBF which has a computation time significantly exceeding the common range. The reason is that QGA increases the amount of data and expands the search range through genetic algorithm crossover, mutation, and mutation, resulting in a large amount of data computation work and consuming a lot of computation time, leading to a computation time of 9,760 milliseconds, which is not on the same order of magnitude as other RBF algorithms.

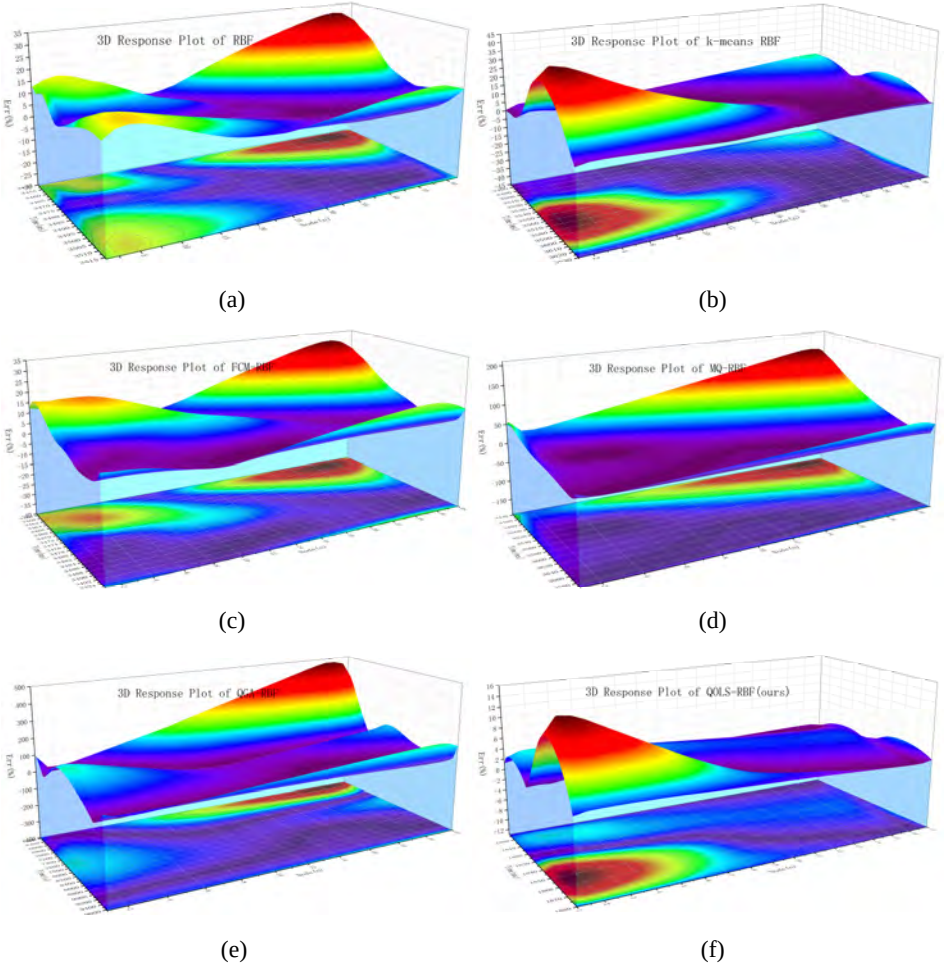
3 In the temperature compensation process of pressure transmitters, RBF, FCM-RBF, MQ-RBF, and QGA-RBF cannot control the mean e_{rr} max error to be on the order of 10^{-4}, while K means RBF is feasible. However, through analysis, it was found that the maximum relative error still cannot reach the order of 10^{-4}. Overall, the e_{rr} max sums of QOLS-RBF with significant compensation effects are 1.24×10^{-4} and 5.60×10^{-4}, respectively. On the one hand, this proves the stable performance of QOLS-RBF in temperature compensation, and on the other hand, it can also be concluded that QOLS-RBF has higher convergence accuracy, stronger optimisation ability, and can improve accuracy.

4 Under the same requirement of a relative error of 0.01%, it can be seen from the table that the final QOLS-RBF is e_{rr} only 8.2×10^{-5}, ranking third among the six types of RBF neural networks. However, it only requires 17 nodes for network establishment. Compared with RBF, K means RBF, FCM-RBF, MQ-RBF, and QGA-RBF, it saves 1,766 ms, 1,755 ms, 1,745 ms, 1,934 ms, and 7,880 ms respectively, with significant improvement, shortening 48.44%, 48.28%, 48.13%, 50.71%, and 80.73% of the time, and the effect is significant. From the perspective of simplifying the structure, QOLS-RBF synchronously reduced the structure by 67.31%, 45.16%, 39.28%, 19.04%, and 22.72%, which further proves the role of QOLS-RBF in simplifying the structure and accelerating the temperature compensation of pressure transmitters.

Figure 9 shows the response surface analysis after low pass FFT transformation. Through the study of the temperature compensation model of the RBF neural network, on the one hand, the application of RBF neural network in temperature compensation is explored, and on the other hand, the performance improvement of quantum least squares method in RBF neural network is also explored. According to Table 3 and Figures 9(a)–9(d), the RBF model is relatively stable, with RBF, K means RBF, FCM-RBF, and QOLS-RBF being able to control error fluctuations within 10% in the ranges of 51%, 75%, 68%, and 80%, respectively. However, the highest peaks of RBF, K means RBF, and FCM-RBF are 31.2%, 25.6%, and 28.5%, respectively, while the highest value of QOLS-RBF is

controlled at 11%. According to the characteristics of RBF type algorithms, the number of nodes will continue to accumulate. K-means RBF and QOLS-RBF do not need to worry about the occurrence of the maximum error value, while RBF and FCM-RBF must be aware that even if the network has been determined, the maximum error value will still occur when time is not enough.

Figure 9 Shows the three-dimensional response surface plots of six RBF compensation models, (a) RBF compensation effect diagram (b) K-means RBF compensation effect diagram (c) FCM-RBF compensation effect diagram (d) MQ-RBF compensation effect diagram (e) QGA-RBF compensation effect diagram (f) QOLS-RBF compensation effect diagram (see online version for colours)



Compared to the other four models, the error fluctuations of MQ-RBF and QGA-RBF are not of the same order of magnitude, at 150% and 423%, respectively. This indicates that the MQ-RBF model is unstable, and the quantum binding degree of QGA-RBF cannot effectively reduce the structure, but instead increases the load on the algorithm model, resulting in longer time. Similarly, MQ-RBF and QGA-RBF can also encounter RBF and FCM-RBF problems, and sufficient training time is necessary.

Overall, QOLS-RBF has reduced the structure by 67.31%, 45.16%, 39.28%, 19.04%, and 22.72% compared to RBF, K-means RBF, FCM-RBF, MQ-RBF, and QGA-RBF in terms of simplification and convergence speed. It has also saved 48.44%, 48.28%, 48.13%, 50.71%, and 80.73% in terms of time. It can be seen that quantum optimisation theory has a significant simplification and acceleration effect in temperature compensation of RBF neural networks.

From this, it can be seen that the QOLS-RBF algorithm proposed in this article has significant advantages compared to other similar RBF algorithms. With the same 85 training data and 170 test data, using the same basis function, the best network error can be obtained in a shorter computation time with fewer centre points. The mean and maximum values of the test error reach 1.24×10^{-4} and 5.60×10^{-4}, respectively, to meet the requirements of sensor temperature compensation and significantly reduce computation time while improving algorithm accuracy.

6 Conclusions

QOLS-RBF is a novel neural network algorithm that combines quantum gate theory. This algorithm combines the comprehensive acceleration effects of quantum gates and OLS algorithms, and has significant improvements compared to existing algorithms such as RBF, OLS-RBF, K-means RBF, and FCM-RBF. Although it does not require as much computation time as the RBF algorithm, it reduces the number of hidden layer centre points and computation time while ensuring higher accuracy. Through temperature compensation data for sensors, modelling with 85 data points and testing with 170 data points, it was found that QOLS-RBF has significant advantages in network error, mean test error, and maximum test error. Therefore, it proves the superiority of the innovative RBF algorithm.

Declarations

The datasets used and analysed during the current study available from the corresponding author on reasonable request.

All authors declare that they have no conflicts of interest.

References

Bock, W.J., Porada, E. and Zaremba, M.B. (1992) ‘Neural processing-type fiber-optic strain sensor’, *IEEE Transactions on Instrumentation and Measurement*, Vol. 41, No. 6, pp.1062–1066.

Cao, L.J. and Tay, F.E.H. (2003) ‘Support vector machine with adaptive parameters in financial time series forecasting’, *IEEE Transactions on Neural Networks*, Vol. 14, No. 6, pp.1506–1518.

Geng, Z., Chen, J. and Han, Y. (2016) ‘Energy efficiency prediction based on PCA-FRBF model: a case study of ethylene industries’, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 47, No. 8, pp.1763–1773.

Halim, Z., Kalsoom, R. and Baig, A.R. (2016) 'Profiling drivers based on driver dependent vehicle driving features', *Applied Intelligence*, Vol. 44, pp.645–664.

Harsij, Z. and Mirza, B. (2014) 'Quantum correlations of helicity entangled states in non-inertial frames beyond single mode approximation', *Annals of Physics*, Vol. 351, pp.665–681.

Huang, G.B., Chen, L. and Siew, C.K. (2024) 'Universal approximation using incremental constructive feedforward networks with random hidden nodes', *IEEE Transactions on Neural Networks*, Vol. 17, No. 4, pp.879–892.

Kak, S.C. (1995) 'Quantum neural computing', *Advances in Imaging and Electron Physics*, Vol. 94, pp.259–313.

Li, J., Huang, W., Zhao, C. et al. (2013) 'A comparative study for the quantitative determination of soluble solids content, pH and firmness of pears by Vis/NIR spectroscopy', *Journal of Food Engineering*, Vol. 116, No. 2, pp.324–332.

Li, P. and Zhao, Y. (2018) 'Model and algorithm of sequence-based quantum-inspired neural networks', *Chinese Journal of Electronics*, Vol. 27, No. 1, pp.9–18.

Liang, N.Y., Huang, G.B., Saratchandran, P. et al. (2006) 'A fast and accurate online sequential learning algorithm for feedforward networks', *IEEE Transactions on Neural Networks*, Vol. 17, No. 6, pp.1411–1423.

Park, J. and Sandberg, I.W. (1991) 'Universal approximation using radial-basis-function networks', *Neural Computation*, Vol. 3, No. 2, pp.246–257.

Patra, J.C., Chin, W.C., Meher, P.K. et al. (2008) 'Legendre-FLANN-based nonlinear channel equalization in wireless communication system', in *2008 IEEE International Conference on Systems, Man and Cybernetics*, IEEE, pp.1826–1831.

Pramanik, C., Islam, T. and Saha, H. (2006) 'Temperature compensation of piezoresistive micro-machined porous silicon pressure sensor by ANN', *Microelectronics Reliability*, Vol. 46, Nos. 2–4, pp.343–351.

Purushothaman, G. and Karayiannis, N.B. (1997) 'Quantum neural networks (QNNs): inherently fuzzy feedforward neural networks', *IEEE Transactions on Neural Networks*, Vol. 8, No. 3, pp.679–693.

Shen, W., Guo, X., Wu, C. et al. (2011) 'Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm', *Knowledge-Based Systems*, Vol. 24, No. 3, pp.378–385.

Song, D., Zhen, Z., Wang, B. et al. (2020) 'A novel marine oil spillage identification scheme based on convolution neural network feature extraction from fully polarimetric SAR imagery', *IEEE Access*, Vol. 8, pp.59801–59820.

Wan, T.H., Saccoccio, M., Chen, C. et al. (2015) 'Influence of the discretization methods on the distribution of relaxation times deconvolution: implementing radial basis functions with DRTtools', *Electrochimica Acta*, Vol. 184, pp.483–499.

Wang, H., Zeng, Q., Zhang, Z. et al. (2022) 'Research on temperature compensation of multi-channel pressure scanner based on an improved cuckoo search optimizing a BP neural network', *Micromachines*, Vol. 13, No. 8, p.1351.

Whitehead, B.A. and Choate, T.D. (2002) 'Cooperative-competitive genetic evolution of radial basis function centers and widths for time series prediction', *IEEE Transactions on Neural Networks*, Vol. 7, No. 4, pp.869–880.

Wu, L.A. and Byrd, M.S. (2009) 'Self-protected quantum algorithms based on quantum state tomography', *Quantum Information Processing*, Vol. 8, pp.1–12.

Yang, Y.H. and Chen, H.H. (2010) 'Ranking-based emotion recognition for music organization and retrieval', *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 19, No. 4, pp.762–774.

Ye, X.W., Jin, T. and Yun, C.B. (2019) 'A review on deep learning-based structural health monitoring of civil infrastructures', *Smart Struct. Syst.*, Vol. 24, No. 5, pp.567–585.

Zar, A., Hussain, Z., Akbar, M. et al. (2023) 'A vibration-based approach for detecting arch dam damage using RBF neural networks and Jaya algorithms', *Smart Structures and Systems*, Vol. 32, No. 5, pp.319–338.

Zhai, G., Narazaki, Y., Wang, S. et al. (2022) 'Synthetic data augmentation for pixel-wise steel fatigue crack identification using fully convolutional networks', *Smart Struct. Syst.*, Vol. 29, No. 1, pp.237–250.

Zhang, J.R., Zhang, J., Lok, T.M. et al. (2007) 'A hybrid particle swarm optimization – back-propagation algorithm for feedforward neural network training', *Applied Mathematics and Computation*, Vol. 185, No. 2, pp.1026–1037.