Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/he

# Real-time data-driven fault diagnosis of proton exchange membrane fuel cell system based on binary encoding convolutional neural network

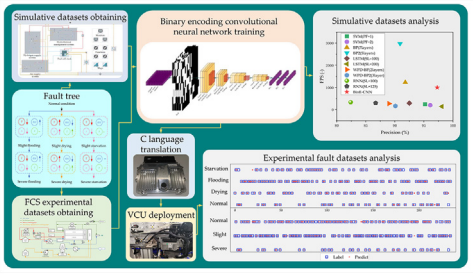
Su Zhou [a], Yanda Lu [a], Datong Bao [a,*], Keyong Wang [a,b], Jing Shan [b], Zhongjun Hou [b]

[a] School of Automotive Studies, Tongji University, Shanghai 201804, China
[b] Shanghai Hydrogen Propulsion Technology Co., Ltd., Shanghai 201804, China

## HIGHLIGHTS

- Binary encoding (BinE) is used as data preprocessing to extract abstract details.
- A multi-classification multi-level fault diagnosis algorithm: BinE-CNN is designed.
- AVL CRUISE M is used to generate simulation fault datasets.
- BinE-CNN is translated into C language and deployed on a VCU (Infineon TC275).
- An over-120kW FCS to used verify the real-time and precision of BinE-CNN.

## GRAPHICAL ABSTRACT




## ARTICLE INFO

*Article history:*
Received 8 December 2021
Received in revised form
14 January 2022
Accepted 18 January 2022
Available online 10 February 2022

*Keywords:*
PEMFC system
Data-driven
Deep learning
Simulation datasets
Embedded deployment

## ABSTRACT

The performance of proton exchange Membrane fuel cell (PEMFC) fault diagnosis system plays an important role in normal operation of PEMFC. Therefore, a new fault diagnosis algorithm based on binary matrix encoding neural network called BinE-CNN is proposed. In BinE-CNN, high-dimensional features are extracted through binary encoding, and the feature maps are transferred to a convolutional neural network (CNN) to realize seven-category fault classification. For development of BinE-CNN, a PEMFC model is modeled to generate simulative datasets. Simulative test precision and Frames per second (FPS) of BinE-CNN have reached respectively **0.973** and **999.8** (better than support vector machines (SVM), long short-term memory neural network (LSTM), etc.). In experimental verification section, fault datasets are collected during bench test. After that, BinE-CNN is deployed on vehicle control unit (VCU) to verify its engineering value (real-time and precision). The result meet both requirements, with time cost of **96.15** ms and precision of **0.931**.



* *Corresponding author.*
E-mail address: p953678595@outlook.com (D. Bao).
https://doi.org/10.1016/j.ijhydene.2022.01.145

## Introduction

With the increasingly significant energy crisis and environmental pollution, finding alternative energy sources has become a core issue. In recent years, the field of clean, efficient and sustainable fuel cells has developed rapidly [1,2], in which proton exchange membrane fuel cell (PEMFC) also has broad application prospects because of its high power density, zero emission and low operating temperature. Although it has the advantages above, some faults still occur inevitably during working, which could affect the performance and even lifespan of fuel cell system, owing to its complex working environment and changeable operating conditions. In order to improve the reliability of PEMFC, system fault diagnosis [3], fault location [4] and fault handling [5] are three urgent problems to be considered. As the basic research of the latter two, the reasonable selection of diagnosis methods and the reasonable deployment of diagnosis algorithms are of great significance to the safe operation of PEMFC system.

So far, the fault diagnosis methods of PEMFC can be roughly divided into data-driven diagnosis methods and model-based diagnosis methods [6]. In model-based methods, a fault mathematical PEMFC model should be established to predict and simulate the system performance, and judge whether the fault occurs based on the residual of the model and system results. At present, the mainstream models are divided into three categories: the white box model which is based on electrochemical mechanism, empirical formula and spatial differential equation [7,8], the black box model which is trained by a large number of experimental and simulated input-output relations [9–11], and the gray box model which combines black box and white box [12,13]. However, due to the strong nonlinearity of PEMFC system and the complexity of its electrochemical performance, it is very difficult to establish a reliable simulation model including the possibility of each fault. Therefore, the data-driven fault diagnosis method has been more widely used in PEMFC fault diagnosis.

In researches of data-driven fault diagnosis, the mainstream method is to obtain the sensor and actuator fault datasets of PEMFC system by model method or measurement, and then classify the fault by using different feature extraction methods and pattern recognition algorithms, which is also divided mainly into two categories: fault diagnosis based on traditional machine learning [14–17] and fault diagnosis based on deep learning. In Ref. [18], reference [19] and reference [20], the PEMFC simulation datasets are classified by support vector machines (SVM) and Gaussian mixture model (GMM) are used. In Ref. [21], voltage consistency is predicted through operating parameters with a integrated regression method based on Gradient Boosting Decision Tree (GBDT). In Ref. [22], the water flooding fault is real-time monitored based on the measurable experimental datasets. After identified by k-Nearest Neighbor (kNN) the cathode inlet pressure will be adjusted to the corresponding target value to avoid the occurrence of the fault. In Ref. [23], fault identifications are performed using KNN to diagnose the condition scenarios of a dual PEMFC stack system. In Ref. [24], four layers of BP neural network are used for fault diagnosis of tram fuel cell system, in which principal component analysis (PCA) is used to reduce the dimension during data preprocessing. In Ref. [25], BP neural network is used to extract high-dimensional abstract features from the original measurement information and arrange the data information into square feature maps. Then the feature maps are transferred to a proposed Convolutional Neural Network (CNN) based on InceptionNet to realize fault classification. In Ref. [26], fuel cell system operation conditions are monitored online based on multi-sensor signals and PCA. And, a simplified statistic index for fault diagnosis is deduced based on PCA as well. In Ref. [27], T-distributed stochastic neighbor embedding (t-SNE) is used to reduce the dimension of nonlinear datasets, and bidirectional long short-term memory neural network (Bi-LSTM) is used to realize fault detection of time series data. In Ref. [28], long short-term memory neural network (LSTM) is used to monitor water flooding fault of the stacks. In Ref. [29], BPNN is used to diagnose fault components and locations in air supply systems. In Ref. [30], the extreme learning machine theory (ELM) is used under the framework of BP neural network, but the weight is not updated by gradient descent method, but the inverse matrix is obtained to optimize the operation speed of fault classification.

Briefly, these above-mentioned fault diagnosis methods based on traditional machine learning or deep learning show the broad application prospect of artificial intelligence methods in the field of fuel cell fault diagnosis and have obtained good phased results. Compared with traditional machine learning, neural network has stronger non-linear adaptive ability, which represents stronger spatial mapping and fitting ability of complex problems. Hence, neural network has stronger and more flexible ability to deal with fitting problems with a large amount of data, which can better complete forward-looking research such as fault location refined to actuator and even component level or transfer learning between model and real PEMFC. However, Neural network is an algorithm with considerable time complexity algorithm, which leads to a longer forward propagation process and calculation time. And these aspects are rarely concerned in the algorithm models above. They are often too complex to meet the real-time requirements of the controller and ignore the actual engineering application requirements.

Combining all the reasons above, the purpose of this paper is to improve this situation, so Binary-encoded based convolutional neural network (BinE-CNN) is proposed, which aims at optimizing the diagnosis precision and reducing the calculation time. The algorithm is originally developed and optimized based on simulation datasets, and in order to simulate the situation when the experimental measured values are damaged by noise, Gaussian noise [31] is embedded in the simulation fault datasets, which makes the extraction of geometric features and pattern recognition more challenging. In this algorithm, the PEMFC sensor signal datasets will be firstly transformed into information matrix datasets through binary encoding, and the classification process is completed through multilayer CNN, activation function and fully connected layers. Then, several groups of faults embedding experiments have been carried out to obtain experimental datasets, which are used to verify the generalization ability, real-time performance and practical application value of this algorithm. In these works, other classical data-driven

fault diagnosis algorithms of different scales are also tested on the same experimental and simulative datasets, in order to compare the FPS, storage space requirement, precision and recall. The test results of BinE-CNN show excellent performance, which proves that BinE-CNN has great robustness, real-time performance and classification ability. Furthermore, the algorithm proposed in this paper is also regarded to have the capability to be applied to analyze other classification problems in fields of PEMFC fault diagnosis and even in fields of other industrial applications.

The research route of this paper is shown as Fig. 1. This paper is organized as follows: Section Data source introduces the data source of this paper and describes the details of fault types, datasets signal composition and fault embedding process. The details of BinE-CNN is described in Section Description of diagnosis methodologies. Section Comparison and discussion includes specific simulative and experimental results and detailed comparative discussion. Finally, the conclusion is in Section Conclusion.

## Data source

Data source is composed of two parts: simulative and experimental datasets.

### *Description of simulative fault datasets*

In this study, PEMFC model is modeled in AVL CRUISE M. The core principle of AVL CRUISE M PEMFC model is that it superimposes 1D numerical model for a gas-flow onto a 2D analytic solution for concentration and velocity distribution in the plane perpendicular to the gas-flow, resulting in 3D information on species concentration and velocity in the PEMFC. This approach described above yields a hybrid 3D analytic-numerical model with computational times characteristic of 1D models. Such features make the model highly predictive and real-time capable, thereby it also has advantages in static/dynamic simulation of fault conditions.

By using AVL CRUISE M, a complete fuel cell system and its subsystems has been modeled, including the complete air supply system, hydrogen supply system and hydrothermal management system, and a corresponding control algorithm has been deployed to make sure the model can run normally under various working conditions. Considering that this paper focuses mainly on algorithm research, the description of the model will not be expanded too much except the description of important hypothesis. The simulation model is shown in Fig. 2.

During the model development, some assumptions of PEMFC stack system are made for the sake of simplification as below.

1. Heat transport and membrane hydration/dehydration are addressed as transient phenomena. All other phenomena are assumed to be in a steady-state.
2. Gases are assumed to be ideal.
3. Constant pressure is assumed in the gas equations pertaining to individual slice.
4. The diffusion constant used in the GDL is the diffusion constant for empty space divided by the tortuosity of the GDL.
5. The flow in the channels is laminar due to low velocities and small channel cross-sectional area.
6. At a given point in the membrane the mobilities of both species transported are linearly proportional to the membrane water content at that point.
7. Species are transported across the membrane only along the direction perpendicular.

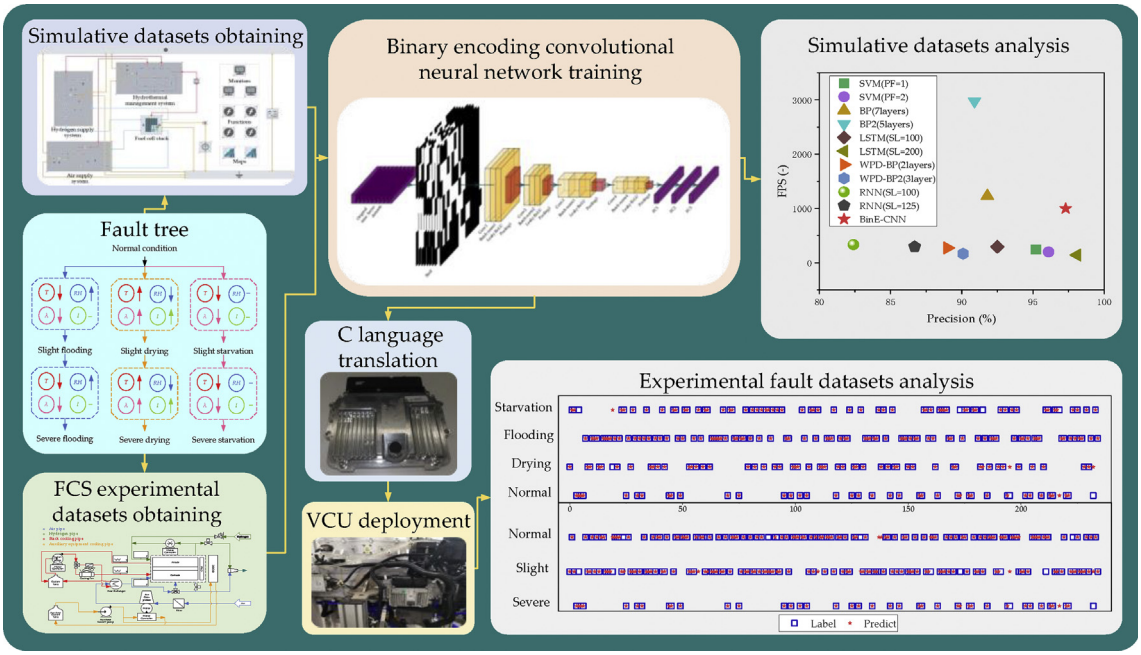



**Fig. 1 – Research route.**

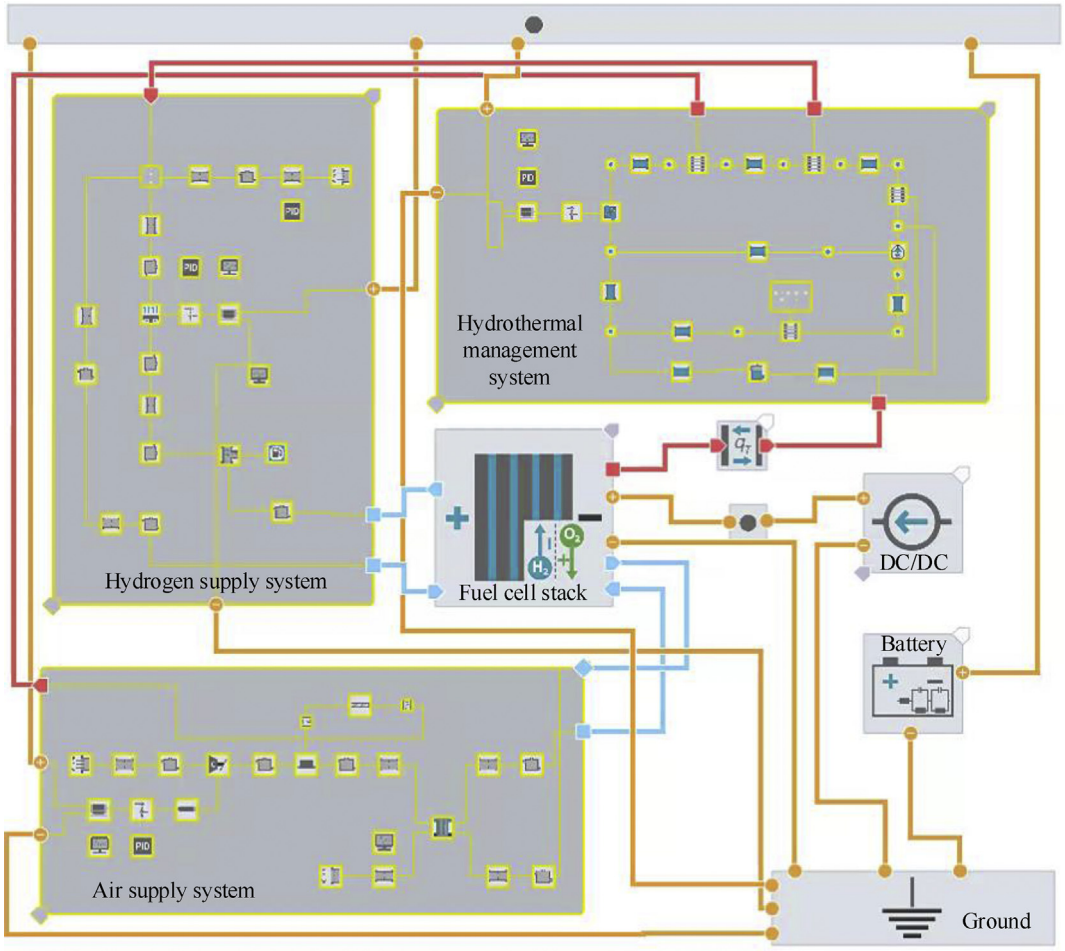



**Fig. 2 – AVL CRUISE M simulation model.**

**Table 1 – Example data.**

| Diagnostic Variable | N1 | D1 | D2 | S1 | S2 | F1 | F2 |
|---|---|---|---|---|---|---|---|
| Sampling time (s) | 512.34 | 515.13 | 519.82 | 525.32 | 517.85 | 514.37 | 517.96 |
| Compressor speed (rpm) | 990.7151 | 988.8159 | 991.7928 | 989.0842 | 980.625 | 1078.789 | 1158.483 |
| Air Stoichiometric ratio (–) | 3.387014 | 4.036353 | 4.754416 | 3.115078 | 2.33685 | 6.4551 | 6.507351 |
| Stack voltage (V) | 276.0336 | 270.8664 | 267.1963 | 276.0524 | 276.1923 | 281.9683 | 282.0132 |
| Stack current (A) | 47.58868 | 50.06478 | 45.40736 | 48.0934 | 45.61398 | 49.27722 | 48.2128 |
| Air inlet pressure (Pa) | 150001.7 | 149998.2 | 150004.3 | 149995.2 | 150000.6 | 160008.5 | 169999.2 |
| Hydrogen inlet Pressure (Pa) | 150009.4 | 150005.1 | 150002.0 | 149999.1 | 149996.9 | 160010.6 | 169996.8 |
| Back pressure valve opening ratio (–) | 0.372185 | 0.418003 | 0.307531 | 0.660566 | 0.203246 | 0.383528 | 0.254677 |

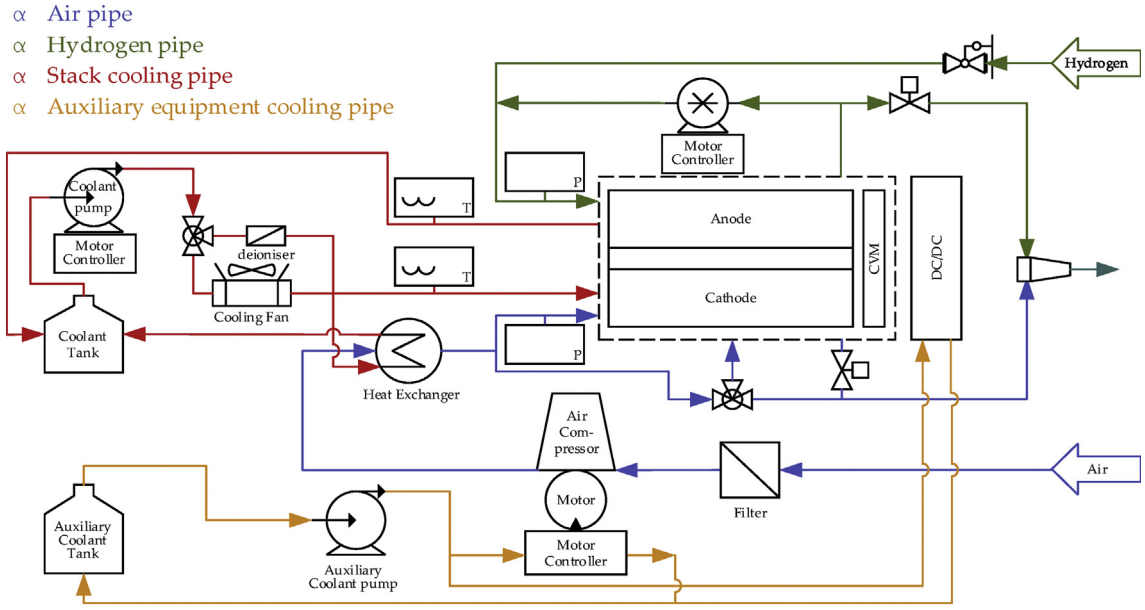
8. In the simulation model, a dynamic current condition is needed for data simulation.

This paper mainly focuses on the fault embedding of three typical fuel cell stack faults. Each typical fault is set to two levels: slight and severe, and a nickname is given for each class to facilitate subsequent description. Specifically, N stands for normal class, D1 for fault class of slight drying, D2 for fault class of severe drying, S1 for fault class of slight starvation, S2 for fault class of severe starvation, F1 for fault class of slight flooding, F2 for fault class of severe flooding.

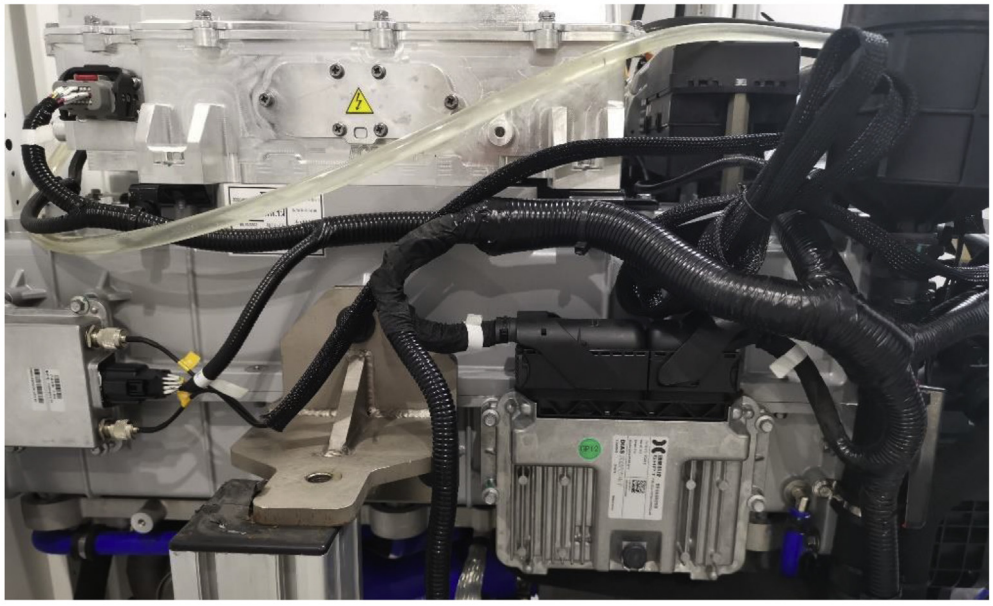
In order to screen the composition of the datasets, the Pearson product-moment correlation coefficient (PPMCC) is used to measure the correlation between the fault categories and 14 selected simulated signals. The absolute values of these PPMCC of each signal are arranged from large to small as follow: Back pressure valve opening (–0.1382), Air flow resistance (–0.1208), Air stoichiometry (–0.1068), Hydrogen flow resistance (0.0775), Stack current (0.00005), Air compressor speed (–0.0305), Air outlet pressure (0.0289), Stack voltage (–0.0221), Hydrogen outlet pressure (0.0085), Hydrogen inlet pressure (0.0082), Air inlet pressure (0.0074), Flow rate of hydrogen (–0.036) Cooling pump speed (0.0035), Flow rate of

**Table 2 – Fault embedding cases.**

| Fault case | Coolant temperature (°C) | Back pressure channel friction coefficiency (–) | Inlet relative air humidity (–) | Air Stoichiometric ratio (–) |
|---|---|---|---|---|
| N | 65 | 1 | 0.5 | 1 |
| D1 | 85 | 1 | 0.5 | 1 |
| D2 | 98 | 1 | 0.2 | 1 |
| S1 | 65 | 1 | 0.5 | 0.925 |
| S2 | 65 | 1 | 0.5 | 0.85 |
| F1 | 60 | 0.8 | 0.8 | 1 |
| F2 | 60 | 0.6 | 0.8 | 1 |




(a)



(b)

**Fig. 3 – (a) The topology of the experiment, (b) FCS for fault embedding experiment.**

air (0.0006). Considering that some of these signals are difficult to be measured in the on-board stack and some signals is significant for diagnosis seven signals are chosen that are measurable and have relatively large PPMCC. Therefore, seven signals are selected to form the datasets of fault diagnosis in this study, and one sample data of each fault category is shown in Table 1 as an example.

According to the fault type and inducement of fuel cell stack [16], the embedding of membrane drying fault is simulated by increasing the coolant temperature and the relative inlet air humidity, the fuel starvation fault is simulated by reducing the air stoichiometric ratio, and the flooding fault is realized by reducing the coolant temperature, back pressure channel friction coefficiency and increasing the inlet relative air humidity at the same time. In addition, fault embedding is realized by setting boundary conditions as shown in Table 2.

The fault datasets are composed of the sensor and actuator signals shown in Table 1 Example data. It is obtained by setting the current and different boundary conditions for the simulation model to realize fault embedding. In order to compare the robustness of the algorithm, Gaussian noise with mean value of 0 and standard deviation of 1 is added to the current and each sensor signal respectively.

### *Description of experimental fault datasets*

The PEMFC stack used in this paper is a high-power Fuel Cell Stack (over 120 kW) produced by Shanghai Hydrogen Propulsion Technology Co., Ltd. (abbreviated as SHPT), using which seven classification fault embedding experiments have been carried out. The experimental topology and experiment device are shown respectively in Fig. 3 (a) and (b).

In the process of fault embedding, a set of calibrated normal boundary conditions (the calibration value of both working temperature and Air stoichiometry of the stack under different current densities) which are applicable to this target stack are used as reference conditions ("normal" conditions). And other fault boundary conditions are set based on corresponding trend in fault tree of PEMFC, as shown in the cyan block in Fig. 1. In order to embed faults and meanwhile avoid irreversible damage to the PEMFC system caused by the fault embedding experiment, the cooling water temperature is increased by 3 and 5° Celsius to simulate slight and severe drying respectively, the cooling water temperature is reduced by 5 and 10° Celsius to simulate slight and severe flooding respectively, and the air stoichiometric ratio is reduced by 0.1 and 0.2 to simulate slight and severe starvation respectively. The boundary condition settings of fault embedding are shown in Table 3 and Table 4. After that, experimental datasets consisting of the same signal type as the simulation datasets signal are collected.

According to the above boundary conditions, a dynamic current is loaded to the stack. At the same time, the seven classification fault datasets are collected at a sampling frequency of 10 Hz, with each type of data of about 950 s. In addition, in order to confirm the success of fault embedding, high frequency impedance (HFR) and air flow are also collected as shown in Fig. 4.

**Table 3 – The Normal state boundary conditions.**

| Current density ($mA/cm^2$) | Air stoichiometry (–) | Cooling water temperature (°C) |
|---|---|---|
| 80 | 2.9 | 69 |
| 100 | 2.8 | 69 |
| 200 | 2.6 | 70 |
| 300 | 2.5 | 73 |
| 400 | 2.3 | 77 |
| 500 | 2.2 | 78 |
| 600 | 2.2 | 80 |
| 700 | 2.1 | 82 |
| 800 | 1.9 | 85 |
| 900 | 1.9 | 86 |
| 1000 | 1.9 | 86 |
| 1100 | 1.8 | 87 |

**Table 4 – The boundary conditions of experiment.**

| State | Boundary conditions | |
|---|---|---|
| | Air stoichiometry (–) | Cooling water temperature (°C) |
| Normal | – | – |
| Slight drying | – | +3 |
| Severe drying | – | +5 |
| Slight flooding | – | −5 |
| Severe flooding | – | −10 |
| Slight starvation | −0.1 | – |
| Severe starvation | −0.2 | – |

## Description of diagnosis methodologies

Through modeling and simulation, seven-category PEMFC dynamic working condition fault datasets were constructed, with about 102,000 samples. To distinguish these datasets, an algorithm as shown in Fig. 5 is designed. This algorithm mainly consists of two parts: the binary encoding (BinE) part and the convolutional neural network (CNN) part.

### *Binary encoding (BinE)*

For the binary encoding part, its main function is to expand and encode the original sensor signal, which is further convenient for constructing the information matrix (with 7 rows and 18 columns). Before constructing the information matrix, the Z-score standardization algorithm is used to standardize the data, and its calculation process is shown in formula (1) and (2) respectively.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{1}$$

$$z = \frac{x - \mu}{\sigma} \tag{2}$$

where $\sigma$ represents the standard deviation of the input data, $\mu$ represents the mean of the input data, z represents the standardized result.

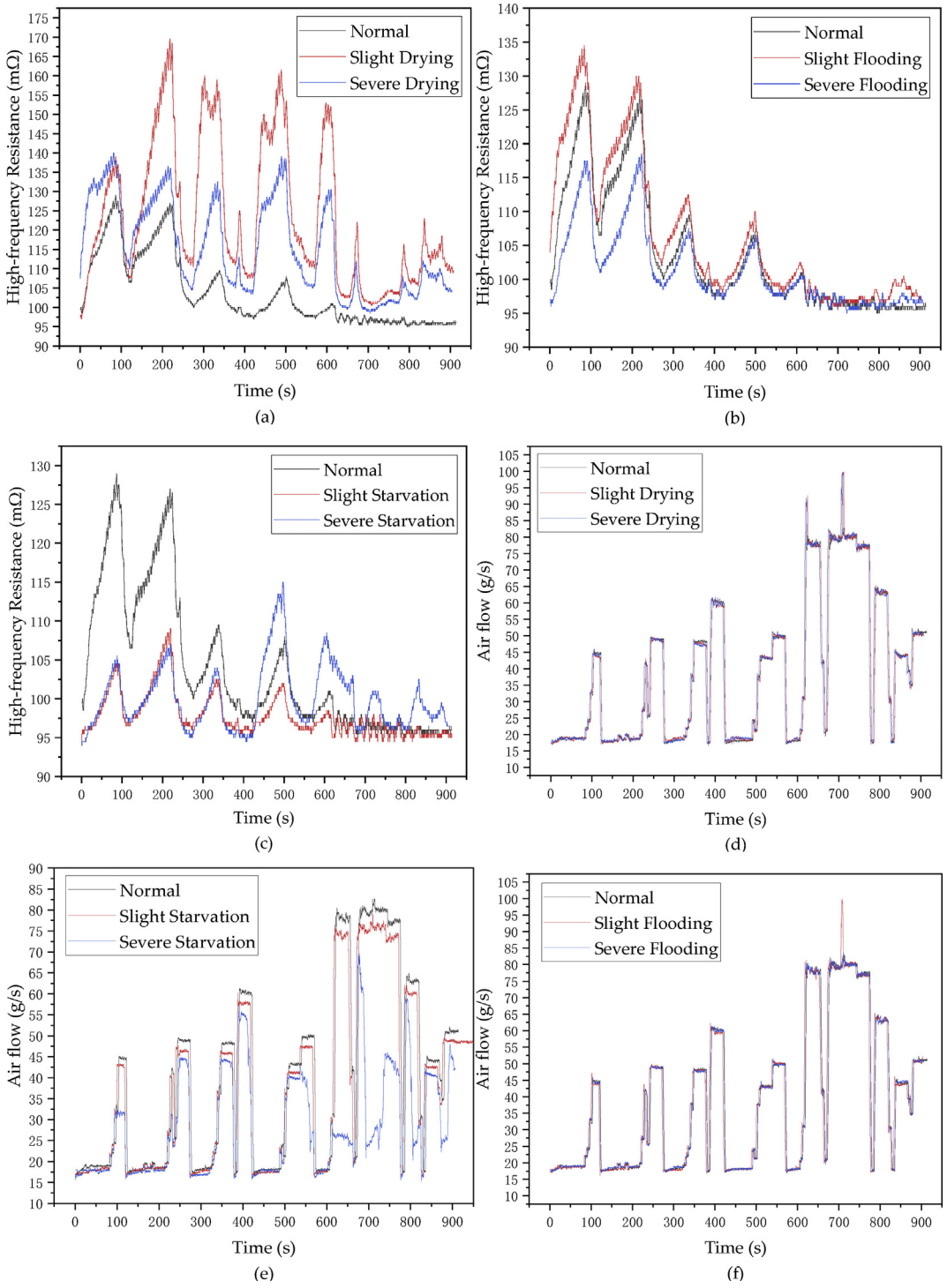



**Fig. 4 – (a) Comparison of HFR between Normal and Drying, (b) Comparison of HFR between Normal and Starvation, (c) Comparison of HFR between Normal and Flooding, (d) Comparison of Air flow between Normal and Drying, (e) Comparison of Air flow between Normal and Starvation, (f) Comparison of Air flow between Normal and Flooding.**

The construction process of information matrix is shown in Fig. 6. For the integer part of each signal, the 2nd to 8th digits in each row of the information matrix are used to hold their binary converted digits. In order not to make the information matrix too wide in the row direction, the fractional part of each signal is simplified, so that the 9th to 18th digits of each row can be used to store their binary converted numbers. The first digit of each line represents the positive or negative of the sensor signal (0 represents negative, 1 represents positive).

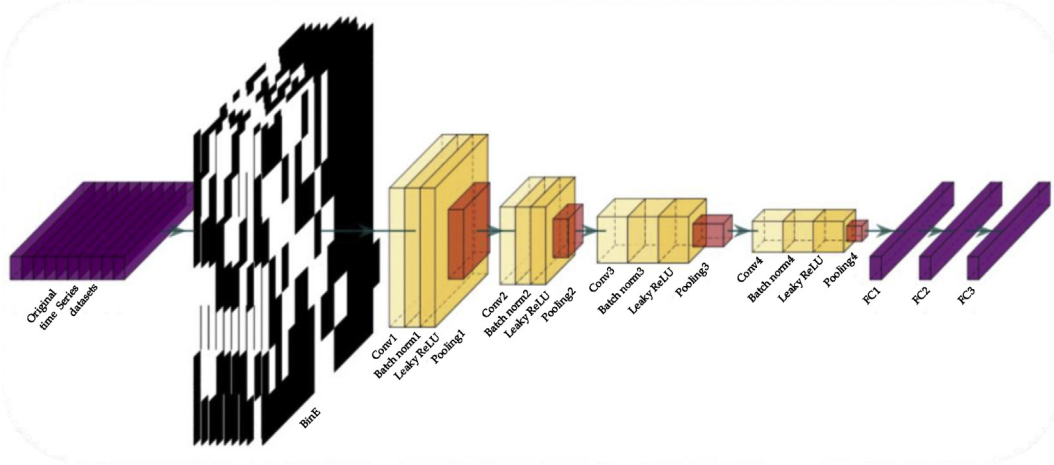



**Fig. 5 – BinE-CNN algorithm model.**

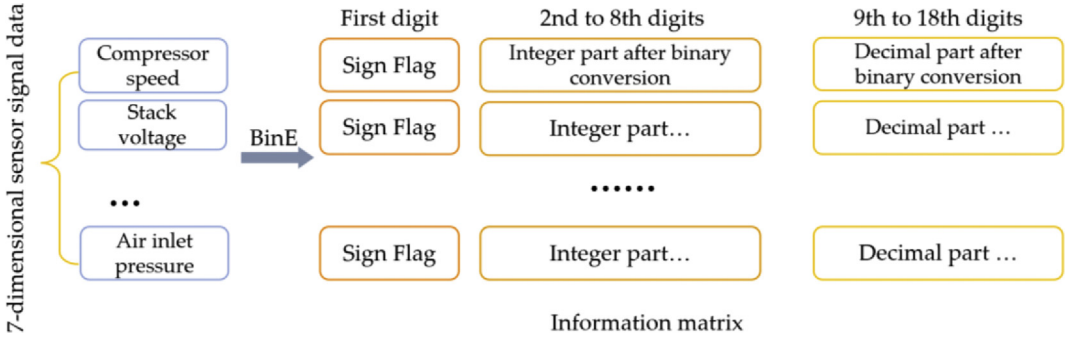



**Fig. 6 – Binary encoding part.**

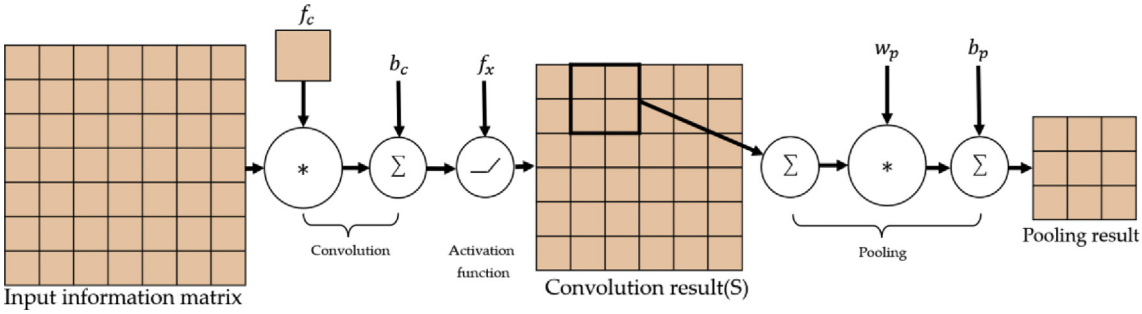



**Fig. 7 – Convolution and pooling process.**

### *Convolutional neural network*

After constructing the information matrix through BinE, a CNN model with convolutional layer, pooling layer and fully connected layer is designed to diagnose faults [32].

*Convolutional layer*

The convolutional layer is a set of parallel feature maps (feature maps), which are composed by sliding different convolution kernels on the input matrix and running certain operations [33]. In addition, at each sliding position, the convolution kernel and the input image will run an element-corresponding product and summation operation to project the information in the receptive field to an element in the feature map. All elements in a feature map are calculated by a convolution kernel, which means that a feature map shares the same weight and bias terms.

As shown in Fig. 7, the convolution process includes: deconvolving an input matrix with a trainable filter $f_c$, and then adding a bias $b_c$ to obtain the convolutional layer result.

*Pooling layer*

Pooling is actually a non-linear form of down sampling. There are many different forms of non-linear pooling functions, of which Max pooling is the most common. As shown in Eq. (3), it divides the input matrix into several rectangular areas, and outputs the maximum value for each sub-area.

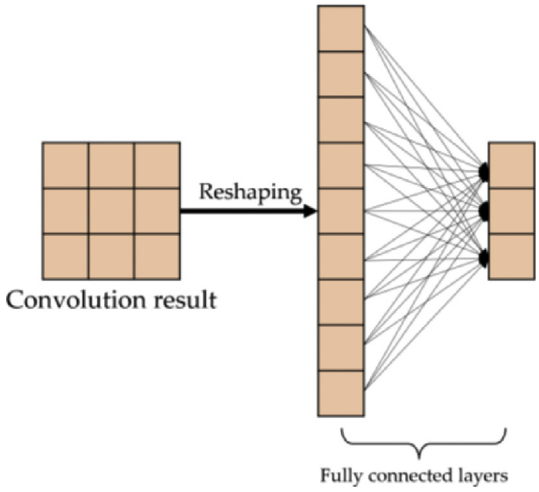



**Fig. 8 – Fully connected layers.**

**Table 5 – The structure of BinE-CNN model.**

| Layer | Convolution kernel size (–) | Feature map (–) |
|---|---|---|
| BinE | – | 10 × 18 × 18 |
| Conv 1 | 3 × 3, Padding = 1 | 10 × 18 × 18 |
| Batch norm 1 | – | 10 × 18 × 18 |
| Leaky ReLU | – | 10 × 18 × 18 |
| Pooling 1 | 2 × 2, Stride = 2 | 10 × 9 × 9 |
| Conv 2 | 2 × 2, Padding = 1 | 10 × 10 × 10 |
| Batch norm 2 | – | 10 × 10 × 10 |
| Leaky ReLU | – | 10 × 10 × 10 |
| Pooling 2 | – | 10 × 5 × 5 |
| Conv 3 | 2 × 2, Padding = 1 | 20 × 6 × 6 |
| Batch norm 3 | – | 20 × 6 × 6 |
| Leaky ReLU | – | 20 × 6 × 6 |
| Pooling 3 | 2 × 2, Stride = 2 | 20 × 3 × 3 |
| Conv 4 | 2 × 2, Padding = 1 | 20 × 4 × 4 |
| Batch norm 4 | – | 20 × 4 × 4 |
| Leaky ReLU | – | 20 × 4 × 4 |
| Pooling 4 | 2 × 2, Stride = 2 | 20 × 2 × 2 |
| FC1 | 80 | 80 |
| FC2 | 20 | 20 |
| FC3 | 7 | 7 |

**Table 6 – The training environment.**

| Item | |
|---|---|
| CPU | AMD Ryzen R7-5800 |
| RAM | 32 GB |
| Cuda | 10.2 |
| Cudnn | 9.0 |
| Programming language | Python 3.8.11 |
| Deep learning lib | Pytorch 1.9 |
| GPU | Nvidia Quadro P5000 |
| System | Windows 10 |

$$f_p(S(x,y)) = max_{a,b=0}^{1} S_{x+a,y+b} \tag{3}$$

where $S$ represents the feature map, $a, b$ represents the size of the convolution kernel of the pooling layer.

**Table 7 – The hyperparameters of BinE-CNN model.**

| Hyperparameter | |
|---|---|
| Batch size | 4192 |
| Epochs | 500 |
| L2 Regulation | 5e-4 |
| Negative slope | 0.01 |
| Initial learning rate | 8e-4 |

**Table 8 – The results of preprocessing comparison.**

| | Precision | Recall | FPS |
|---|---|---|---|
| WPD-CNN | 0.612 | 0.577 | 125.2 |
| BP-CNN | 0.978 | 0.978 | 625.1 |
| **BinE-CNN** | **0.973** | **0.973** | **999.8**[3] |

*Batch normalization*

Batch normalization (BN) is a way to improve the speed and stability of network training by normalizing the middle layer of the CNN model. The main way it works is to make the weight of the network initialization closer to the distribution of the result after the previous layer of calculation. Its calculation process is as shown in Eqs (4)–(7), respectively [34].

$$\mu_c = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{4}$$

$$\sigma_c^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_c)^2 \tag{5}$$

$$x_i^{new} = \frac{x_i - \mu_c}{\sqrt{\sigma_c^2 + \varepsilon}} \tag{6}$$

$$y_i^{new} = \gamma x_i^{new} + \beta \tag{7}$$

where $x_i$ represents the input value over a batch, $\gamma, \beta$ represents the parameters to be learned, $\mu_c$ is the mean value of $x_i$, $\sigma_c^2$ is the variance of the batch, $\varepsilon$ is the minimum value that prevents division by zero.

*Activation function*

The main purpose of adding the activation function to the neural network is to improve the nonlinear fitting ability of the network, so that the network can approximate any nonlinear model. The Leaky ReLu function is used in BinE-CNN as the activation function, which is shown in Eq. (8).

$$f(x) = \begin{cases} x & x \geq 0 \\ ax & x < 0 \end{cases} \tag{8}$$

*Fully connected layers*

Fully connected layers (FC) function as a classifier in the entire convolutional neural network, as shown in Fig. 8. In the network, if operations such as the convolutional layer, pooling layer, and activation function layer mapping the original data to the hidden layer feature space, the fully connected layer plays the role of mapping the learned distributed feature representation to the sample label space [35].

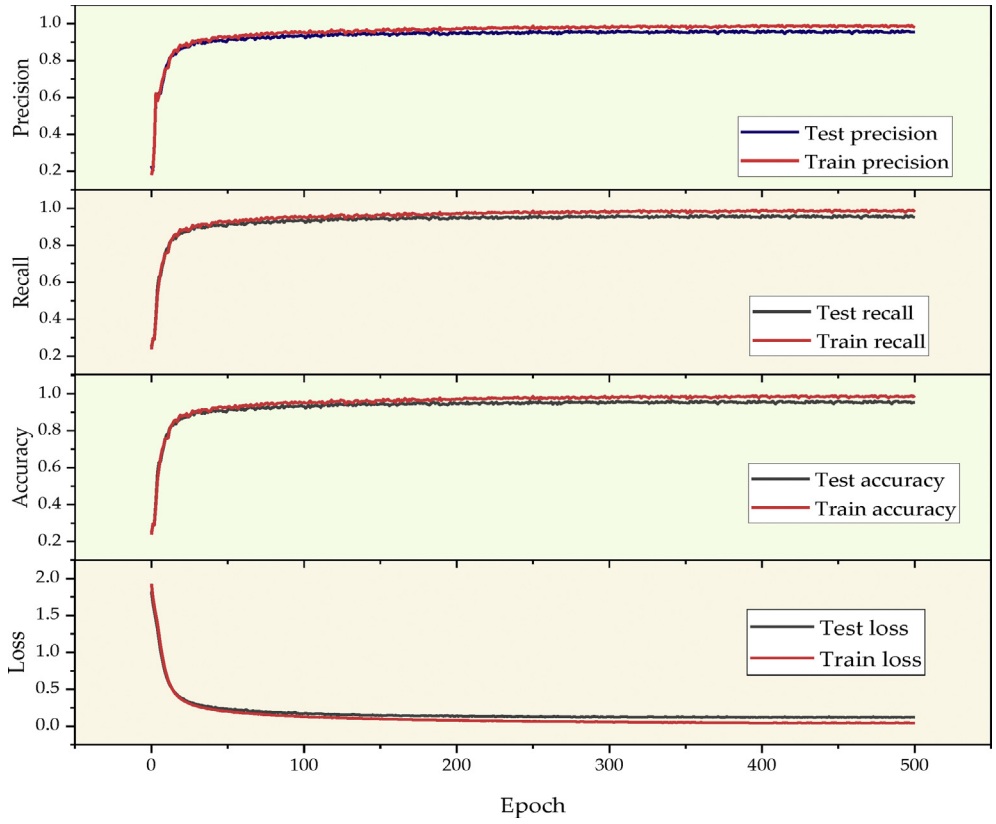



**Fig. 9 – Evaluation index and learning rate during training.**

Based on the simulation datasets, we designed the BinE-CNN model, whose structure is shown in Table 5.

### *Training environment*

Considering that this is a multi-classification problem, the Cross-Entropy function is selected as the loss function. 80% of the samples in the simulation data are used as the training set, and 20% are used as the test set. Training environment is shown in Table 6. The hyperparameters used in the training are shown in Table 7.

## Comparison and discussion

### *Algorithm comparison results on the simulation datasets*

The simulative algorithm discussion is divided into three parts. The first part is the comparison of BinE and other data preprocessing methods, the second part is the training process discussion, and the third part is comparison of BinE-CNN and other classical algorithms.

In this paper, the main task of data preprocessing is to transform the original data into a timing information matrix in size of 18 × 18, and then input it into CNN to realize fault classification. And, in order to show the characteristics of BinE preprocessing, we compared it with WPD, BP neural network in terms of precision, recall and FPS on the premise that the post-processing is the CNN with same structure and parameter setting. The results are shown in Table 8. It can be seen that BinE has achieved great precision and the highest FPS. Compared with BP-CNN, BinE-CNN can ensure similar precision, however, FPS is 1.599 times of it. Through calculation, the floating-point operations per second (FLOPS) of BP-CNN is about 430 k, while FLOPS of BinE-CNN is about 400 k, which improves the performance of its real-time deployment on VCU.

The evaluation index during the training process are shown in Fig. 9. It can be seen from Fig. 9 that the training process has entered a slow growth stage after 200 epochs, and there is a slight overfitting, so the result on the test set will be used as the result for contrast.

To verify the effectiveness of the algorithm, LSTM neural network algorithm [36,37], BP neural network algorithm [38], SVM multi-classification algorithm [39] and WPD algorithm [40] are used to compare with BinE-CNN on the same datasets. The results of the comparison are shown in Table 9.

It can be seen from Table 9 that although the SVM multi-classification algorithm has achieved high precision and recall on the simulation datasets, its storage space requirement for storing hyperplane parameters is too large, which has exceeded the size of ROM of most VCUs [41], and this storage space requirement will further increase along with the expansion of datasets. This is a challenge to the engineering application of SVM multi-classification algorithm. BP neural network algorithm is a fast but low precision algorithm, which also requires considerable storage space. The LSTM neural network algorithm has the highest precision rate in this comparison, but its time complexity is relatively large, and some VCUs may not be able to afford

**Table 9 – The results of the comparison.**

| | Parameters | Precision | Recall | FPS[d] | Storage (kB)[a] |
|---|---|---|---|---|---|
| SVM | Decision function shape = OVR<br>Penalty factor = 1<br>Kernel = RBF | 0.952 | 0.963 | 249.4 | 10.5 (MB) |
| | Decision function shape = OVR<br>Penalty factor = 2<br>Kernel = RBF | 0.961 | 0.944 | 199.9 | 12.0 (MB) |
| BP | Input node = 7<br>Hidden layer node: (16, 64)<br>(64, 256)<br>(256, 256)<br>(256, 256)<br>(256, 256)<br>(256, 32)<br>(32, 7) | 0.918 | 0.906 | 1230.6 | 878 |
| | Input node = 7<br>Hidden layer node: (16, 64)<br>(64, 256)<br>(256, 256)<br>(256, 32)<br>(32, 7) | 0.909 | 0.909 | 2976.2 | 363 |
| LSTM | Input layer = 7<br>Hidden layer = 16<br>Sequence length = 100 | 0.925 | 0.927 | 294.0 | 22.1 |
| | Input node = 7<br>Hidden layer node = 16<br>Sequence length = 200 | 0.981 | 0.981 | 142.8 | 36.3 |
| WPD-BP | Two-layer wavelet packet decomposition tree | 0.890 | 0.880 | 273[b] | 7.5 + 5.2[b] |
| | Three-layer wavelet packet decomposition tree | 0.901 | 0.892 | 167[b] | 7.6 + 11.2[b] |
| RNN | Input layer = 7<br>Hidden size = 64<br>Hidden layer = 8<br>Sequence length = 100 | 0.824 | 0.826 | 334.2 | 163 |
| | Input layer = 7<br>Hidden size = 64<br>Hidden layer = 12<br>Sequence length = 125 | 0.867 | 0.864 | 296.7 | 165 |
| **BinE-CNN** | - | **0.973** | **0.973** | **999.8**[c] | **104.5** |

[a] Storage is mainly the parameters, weights and other data necessary for the operation of the algorithm.
[b] The FPS and storage of WPD-BP consists of two parts: WPD itself and the subsequent BP neural network.
[c] The FPS of BinE-CNN has considered two parts of calculating time, which is Bin-encoding and convolutional neural network forward propagation.
[d] Each result is calculated on the CPU.

the calculation requirements of the LSTM neural network algorithm. The WPD algorithm uses Fourier transform in the calculation, which makes its calculation speed slower, and at the same time it performs poorly in terms of classification precision. For our BinE-CNN algorithm, due to the use of the timing information in the data, it can achieve a higher precision and recall rate. And as one of its features, sharing the weights in the calculation results in less storage space required for the algorithm weights. At the same time, the use of matrix operations in the process of forward reasoning also improves the speed of operations. Overall, BinE-CNN has achieved a relatively balanced and excellent performance on the simulation datasets.

It is worth noting that according to the comparison results of different algorithms based on simulation datasets, LSTM has the highest precision and BP neural network has the shortest operation time. When FPS and precision are taken

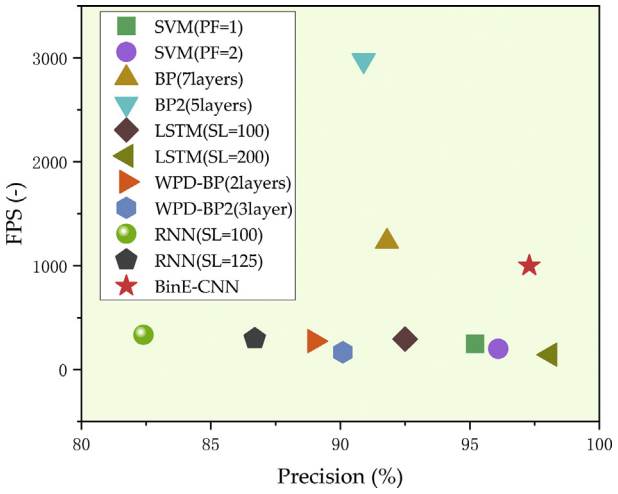



**Fig. 10 – FPS and Precision of diagnostic algorithms.**

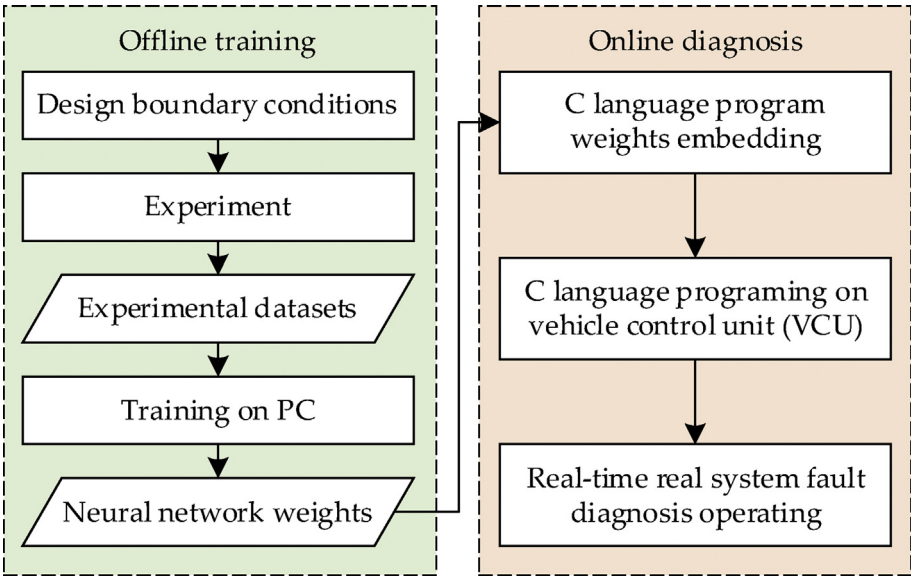



**Fig. 11 – algorithm verification flow chat.**

**Table 10 – Comparison between three typical algorithms.**

| Algorithm | Precision | Recall | Time cost | Storage |
|---|---|---|---|---|
| LSTM | 0.923 | 0.921 | 52.21 ms | 67 kB |
| BP | 0.820 | 0.820 | 51.24 μs | 5 kB |
| BinE-CNN | 0.931 | 0.929 | 96.15 ms | 110 kB |

as the core measurement criteria, the analysis results shown in Fig. 10 are obtained. The algorithm closer to the upper right corner in the figure is considered to have stronger comprehensive ability. In other words, the algorithm can give better consideration to both real-time and diagnostic precision. Obviously, the three algorithms closest to the upper right corner in the figure are LSTM, BP neural network and BinE-CNN. Therefore, they are selected to be translated into C language and programed on VCU, so as to make a comparative analysis with BinE-CNN on experimental datasets.

*Algorithm comparison results on the experimental datasets*

After training and testing on the simulation datasets, the experimental datasets are still needed to verify the classification ability and robustness of BinE-CNN algorithm.

After constructing the experimental datasets, the verification process is carried out in two parts: Offline training and Online diagnosis. The algorithm will first complete offline training on the PC, and then use the C language program to read its weights. Finally, perform online diagnosis on the controller. The verification process is shown in Fig. 11.

Infineon TC275 is used as the test controller in the verification process. During the deployment of BinE, some parameters have been adjusted slightly to meet the random-access memory (RAM) requirements of the controller. The test results are shown in Table 10. Compared with LSTM and BP neural network, BinE achieved the highest precision and recall. The consumption time required for one operation of BinE-CNN is 96.15 ms, which is less than the sensor sampling period of a

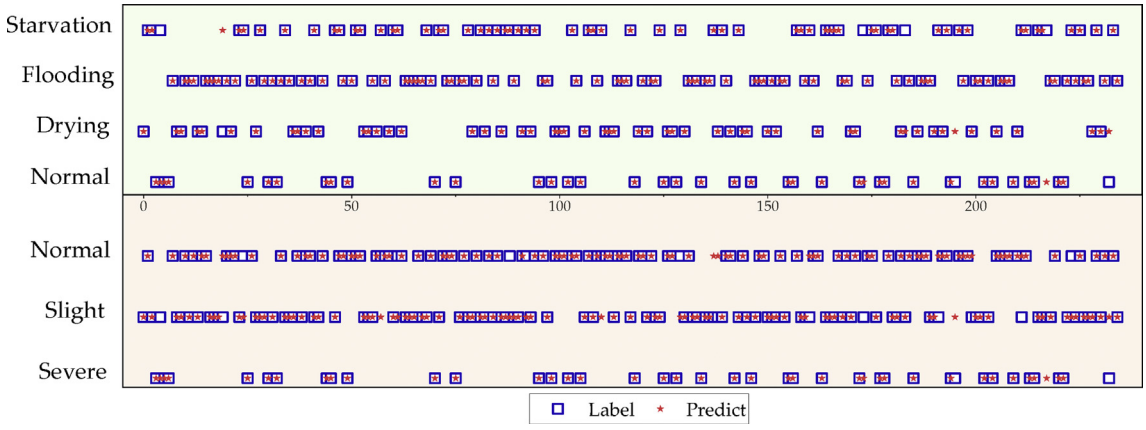



**Fig. 12 – Multi-classification and multi-level diagnosis test result.**

regular PEMFC system (100 ms), which meets the real-time requirements.

In order to further verify the test results of the algorithm, 234 groups of data were randomly selected from the test set and tested. The results are shown in Fig. 12. The diagnosis results are divided into fault type diagnosis (as shown in the upper part of Fig. 12) and fault degree diagnosis (as shown in the lower part of Fig. 12). It can be seen from the sampling test results in the figure that BinE-CNN has achieved excellent results in this test comprehensively. In terms of fault type diagnosis, drying fault and flooding fault are diagnosed relatively accurately. In contrast, the diagnosis results of normal and starvation fault are slightly worse. It is preliminarily analyzed that the reason may be caused by the slight fault boundary conditions of starvation. In terms of fault type diagnosis, BinE-CNN performs more evenly in different degrees of fault classification, which presents a high diagnostic precision.

## Conclusion

An algorithm named BinE-CNN based on binary matrix encoding and convolutional neural network is proposed in this paper. Its main purpose is to carry out a real-time multi-classification and multi-level fault diagnosis using real-time sampling signals of selected actuators and sensors of PEMFC system. In BinE-CNN algorithm, firstly, the original datasets are binary encoded and normalized through Z-score standardization. In other words, by the above processing, the original datasets are reshaped and converted into binary time series matrix with shape of 10 × 18*18. Secondly, a light-weighted deep convolutional neural network is proposed to classify the binary time series matrix sets. Finally, the 7-category fault diagnosis for the PEMFC system can be obtained.

An AVL CRUISE M simulation model is established to generate large amounts of fault simulation data in order to develop and test the algorithm effectively. The simulative test results show that BinE-CNN has the capability to detect normal state and different degrees of three typical fault states (drying, flooding and starvation) of PEMFC system. As shown in Table 9, when compared with SVM, BP neural network, LSTM neural network, WPD and RNN, the simulation-data-based test precision of BinE-CNN (reaching 0.973) is only second to the 200-timesteps version of LSTM algorithm (reaching 0.981), while FPS of it is far beyond the 200-timesteps version of LSTM.

In order to further verify the practical engineering value of BinE-CNN, a set of dynamic current is loaded on a commercial stack under fault boundary conditions to obtain experimental fault datasets. After that, BinE-CNN is translated into C language and deployed on a high-power Fuel Cell System (over 120 kW) with a VCU of Infineon TC275 to verify its real-time, precision and storage space requirement. As shown in Table 10, the experimental test results show that the precision of BinE-CNN reach 0.951, surpassing LSTM and BP neural network, and its time cost is 96.15 ms, which is less than sensor sampling period (100 ms), meeting the real-time requirements.

Our next consideration is to apply fp16 or int 8 quantization into the algorithm to reduce the storage space requirement and improve the computing speed. In order to optimize the utilization of data, memory mechanism is planned to be added into the utilization of time series information.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

This work is supported by Shanghai Hydrogen Propulsion Technology Co., Ltd. (abbreviated as SHPT).

## REFERENCES

[1] Jiao K, Xuan J, Du Q, et al. Designing the next generation of proton-exchange membrane fuel cells. Nature 2021;595.
[2] Online extremum seeking-based optimized energy management strategy for hybrid electric tram considering fuel cell degradation. Appl Energy 2021;285:116505–16.
[3] Lin Y, Li Y, Yin X, et al. Multisensor fault diagnosis modeling based on the evidence theory. IEEE Trans Reliab 2018;67:1–9.
[4] Li Y, Pei P, Ma Z, et al. Method for system parameter identification and controller parameter tuning for super-twisting sliding mode control in proton exchange membrane fuel cell system. Energy Convers Manag 2021;243:114370.
[5] Li Y, Pei P, Ma Z, et al. Characteristic analysis in lowering current density based on pressure drop for avoiding flooding in proton exchange membrane fuel cell. Appl Energy 2019;248(AUG.15):321–9.
[6] Mao L, Jackson L, Davies B. Investigation of PEMFC fault diagnosis with consideration of sensor reliability. Int J Hydrogen Energy 2018;43(35):16941–8.
[7] Mohammadi A, Djerdir A, Bouquain D, et al. Fault sensitive modeling and diagnosis of PEM fuel cell for automotive applications[C]//Transportation Electrification Conference & Expo. IEEE 2013.
[8] Jiashu J, Yuepeng C, Changjun X, et al. Remaining useful life prediction of PEMFC based on cycle reservoir with jump model. Int J Hydrogen Energy 2021;46(80):40001–13.
[9] Zhang X, Guo X. Fault diagnosis of proton exchange membrane fuel cell system of tram based on information fusion and deep learning. Int J Hydrogen Energy 2021;46(60):30828–40.
[10] Zeller A, RallieeRes O, ReGnier J, et al. Diagnosis of a hydrogen/air fuel cell by a statistical model-based method. IEEE; 2010.
[11] Sma B, Lst A, Myb C, et al. A data-driven digital-twin prognostics method for proton exchange membrane fuel cell remaining useful life prediction. Int J Hydrogen Energy 2021;46(2):2555–64. ScienceDirec.
[12] Forrai A, Funato H, Yanagita Y, et al. Fuel-cell parameter estimation and diagnostics. IEEE Trans Energy Convers 2005;20(3):668–75.
[13] Liu J, Li Q, Chen W, et al. Remaining useful life prediction of PEMFC based on long short-term memory recurrent neural networks. Int J Hydrogen Energy 2019;44(11):5470–80.

[14] Yj A, Gz A, Jw A, et al. Hydrogen solubility in aromatic/cyclic compounds: prediction by different machine learning techniques. Int J Hydrogen Energy 2021;46(46):23591–602.
[15] Rahnama A, Zepon G, Sridhar S. Machine learning based prediction of metal hydrides for hydrogen storage, part I: prediction of hydrogen weight percent. Int J Hydrogen Energy 2019;44(14):7337–44.
[16] A fast fault diagnosis method of the PEMFC system based on extreme learning machine and dempster-shafer evidence theory. IEEE Trans Transp Electrification 2019;5(1):271–84.
[17] Multi-objective optimization and data-driven constraint adaptive predictive control for efficient and stable operation of PEMFC system. IEEE Trans Ind Electron 2021;68(12):12418–29.
[18] Zhong ZD, Zhu XJ, Cao GY. Modeling a PEMFC by a support vector machine. J Power Sources 2006;160(1):293–8.
[19] Lim IS, Jin YP, Choi EJ, et al. Efficient fault diagnosis method of PEMFC thermal management system for various current densities. Int J Hydrogen Energy 2020;46(2):2543–54.
[20] Cheung P, Fairweather JD, Schwartz DT. Characterization of internal wetting in polymer electrolyte membrane gas diffusion layers. J Power Sources 2009;187(2):487–92.
[21] Chen H, Shan W, Liao H, et al. Online voltage consistency prediction of proton exchange membrane fuel cells using a machine learning method. Int J Hydrogen Energy 2021;46(2).
[22] Li Y, Pei P, Wu Z, et al. Approaches to avoid flooding in association with pressure drop in proton exchange membrane fuel cells. Appl Energy 2018;224:42–51.
[23] Benouioua D, Candusso D, Harel F, et al. On the issue of the PEMFC operating fault identification: generic analysis tool based on voltage pointwise singularity strengths. Int J Hydrogen Energy 2017;43(25):11606–13.
[24] Zhang X, Zhou J, Chen W. Data-driven fault diagnosis for PEMFC systems of hybrid tram based on deep learning - ScienceDirect. Int J Hydrogen Energy 2020;45(24):13483–95.
[25] Sarah J, Mao Lei, et al. Fault diagnosis of practical proton exchange membrane fuel cell system using signal-based techniques. Institut National Polytechnique De Toulouse; 2015.
[26] Zhao X, Xu L, Li J, et al. Faults diagnosis for PEM fuel cell system based on multi-sensor signals and principle component analysis method. Int J Hydrogen Energy 2017;42(29):18524–31.
[27] Liu J, Li Q, Yang H, et al. Sequence fault diagnosis for PEMFC water management subsystem using deep learning with t-SNE. IEEE Access; 2019. p. 1. 99.
[28] Gu X, Hou Z, Cai J. Data-based flooding fault diagnosis of proton exchange membrane fuel cell systems using LSTM networks - ScienceDirect. Energy and AI 2021;4.
[29] Won Jinyeon, Oh Hwanyeong, Hong Jongsup, et al. Hybrid diagnosis method for initial faults of air supply systems in proton exchange membrane fuel cells. Renew Energy 2021;180:343–52.
[30] Liu J, Li Q, Chen W, et al. A fast fault diagnosis method of the PEMFC system based on extreme learning machine and dempster–shafer evidence theory. IEEE Trans Transp Electrification 2019;5(1):271–84.
[31] Yx A, Jza D, Chao PA, et al. A novel PEM fuel cell remaining useful life prediction method based on singular spectrum analysis and deep Gaussian processes. Int J Hydrogen Energy 2020;45(55):30942–56. ScienceDirec.
[32] Liu X, Yan Z, Zhong Z. Predicting elastic modulus of porous La0.6Sr0.4Co0.2Fe0.8O3-δ cathodes from microstructures via FEM and deep learning. Int J Hydrogen Energy 2021;46.
[33] An J, Wang H, Liu B, et al. A deep learning framework for hydrogen-fueled turbulent combustion simulation. Int J Hydrogen Energy 2020;45.
[34] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. JMLR.org; 2015.
[35] Song S, Xiong X, Wu X, et al. Modeling the SOFC by BP neural network algorithm. Int J Hydrogen Energy 2021;46.
[36] Yoon H, Kim T, Park S, et al. Stable LSM/LSTM bi-layer interconnect for flat-tubular solid oxide fuel cells. Int J Hydrogen Energy 2017;43(1):363–72.
[37] Bza B, Jca B, Zz C. Degradation prediction model for proton exchange membrane fuel cells based on long short-term memory neural network and Savitzky-Golay filter. Int J Hydrogen Energy 2021;46(29):15928–37.
[38] Jian A, Gh A, Kl B, et al. Artificial neural network based chemical mechanisms for computationally efficient modeling of hydrogen/carbon monoxide/kerosene combustion. Int J Hydrogen Energy 2020;45. ScienceDirec.
[39] Hao D, Mehra RK, Luo S, et al. Experimental study of hydrogen enriched compressed natural gas (HCNG) engine and application of support vector machine (SVM) on prediction of engine performance at specific condition. Int J Hydrogen Energy 2019;45(8).
[40] Du R, Wang X, Dai H, et al. Online impedance spectrum measurement of fuel cells based on Morlet wavelet transform. Int J Hydrogen Energy 2021;46(47):24339–52.
[41] Xu L, Li J, Ouyang M, et al. Multi-mode control strategy for fuel cell electric vehicles regarding fuel economy and durability. Int J Hydrogen Energy 2014;39(5):2374–89.