IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 24, 2019, accepted October 28, 2019, date of publication November 1, 2019,
date of current version November 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2950918

Real-Time FPGA-Based Hardware Neural Network for Fault Detection and Isolation in More Electric Aircraft

QIN LIU[1,2], (Student Member, IEEE), TIAN LIANG[2], (Student Member, IEEE), ZHEN HUANG[2], (Student Member, IEEE), AND VENKATA DINAVAHI[2], (Senior Member, IEEE)

[1]School of Mechanical, Electronic, and Control Engineering, Beijing Jiaotong University, Beijing 100044, China
[2]Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada

Corresponding authors: Qin Liu (qin9@ualberta.ca) and Tian Liang (tian.liang@ualberta.ca)

This work was supported by the Natural Science and Engineering Research Council of Canada (NSERC). The work of Q. Liu and Z. Huang was supported by the China Scholarship Council (CSC).

ABSTRACT More Electric Aircraft (MEA) has a great development prospect in the aviation industry thanks to the progressive power electronics technology and digital systems. Nevertheless, any fault occurring in the power electric system of MEA could be a fatal risk for the safety of the aircraft. To deal with the real-time fault detection and isolation (FDI) on the MEA, we put forward an FPGA-based neural network method which includes two stages: off-line construction using TensorFlow and real-time monitoring on the FPGA. Long Short-Term Memory (LSTM) network is applied because of its capability of learning from the long-term historical information in time series. A comprehensive MEA model based on the Boeing 787 power system created in PSCAD/EMTDC® and a commercial electric aircraft model based on Airbus E-Fan in Simscape are simulated to validate the effectiveness and generality of our proposed method. Adequate contrast experiments are conducted to acquire the most applicable architecture and configurations of the network. The results illustrate that the evaluation of the real-time condition can achieve accuracy over 99.5% within one sampling time on FPGA and reasonable hardware resource utilization.

INDEX TERMS Fault detection and isolation (FDI), field-programmable gate array (FPGA), gated recurrent unit (GRU), long short-term memory (LSTM), more electric aircraft (MEA), neural networks, real-time systems.

I. INTRODUCTION

Since human beings fulfilled the dream of flying in the sky, researchers have always endeavored to make aircraft more energy efficient, environmental friendly, reliable and safer as well as less heavy and maintenance cost. More Electric Aircraft (MEA) could be a promising answer. The design of conventional aircraft leads to the sacrifice of energy efficiency in balancing various types of energy sources consisting of mechanical, hydraulic, pneumatic, and electrical [1]. On the other hand, MEA, a fly-by-wire (FBW) aircraft, is able to overcome the above problems with advanced power electronics and digital systems. The most recent commercial transport aircraft, such as the Boeing 787 and the Airbus A380, are largely equipped with electrical systems [2].

However, the MEA still cannot completely avoid the accidental fault and damage during a flight because of harsh environmental conditions. When the severity of the fault reaches a certain level, it would be a disaster. Therefore, the Fault Detection and Isolation (FDI), which timely estimates whether an abnormal event occurs, accurately distinguishes the location and type of the failure [3], is crucial for the safe flight of the aircraft. The reconfiguration actions could be taken as soon as possible if only the failure is diagnosed at the early stage. Due to high complexity of the MEA system, it is a challenging task to detect and isolate faults correctly and rapidly. Many strategies researched can be summarized into two categories, namely model-based and data-driven methods.

Model-based methods build up a mathematical model to estimate states of the system, then the difference between actual outputs and estimated ones is taken as the criterion

The associate editor coordinating the review of this manuscript and approving it for publication was Yanzheng Zhu.

to diagnose the fault. In order to diagnose the fault in sensors and actuators of the aircraft, sliding mode and adaptive observers are adopted to estimate their states. [4], [5]. The faults of auto-transformer unit and transformer rectifier unit in aircraft electrical power system has been detected by particle filtering state estimation and smoothed residual in [6]. The parity space-based approach has solved the fault detection for discrete-time switched systems under dwell-time switching. Multiple-model based fault detectors for aircraft are introduced in [7], where each fault corresponds to one independent model and the "right" model can be automatically activated with the type of present fault. Another widely discussed approach is multi-agent based method that detects and isolates a fault by the vote of a majority of agents [8]. The excellent performance of these model-based methods relies on the expertise of designers who have gained years of experience in practice. If the complete knowledge of the system including its inputs and outputs as well as interactions with environment can be obtained, the model would be well-defined to detect multiple faults [9]. Unfortunately, as systems get more complex as well as highly integrated, nonlinear characteristics and uncertainties that are hardly captured and expressed mathematically prevail among modern electrical systems like MEA. Moreover, it will become too computationally expensive to apply in the real world even if a more accurate model can be established.

When using data-driven approaches, it is not necessary to have a priori knowledge of the system model, but to have access to massive historical data that is easy to obtain in this era of big data. This type of methods can be further classified into signal processing methods, statistical methods and machine learning. Signal processing methods depend on the analysis of time-frequency domain information, e.g. amplitude-frequency and phase-frequency characteristics, from the original sampled data. FFT transform [10], Vold-Kalman filter [11], ensemble empirical mode decomposition [12], Hilbert-Huang transform [13], wavelet transform [14] are applied in the aircraft industry. Statistical methods make use of the fact that the probability distribution of the parameters in a system should be normally stable, but deviate from the normal values under faulty conditions. Bayesian network [15], hidden Markov models [16], and Hotelling's T^2 statistic [17] have been used to control the safety and risk of the electrical system of aircraft. These methods are sensitive to the anomaly of the system, but also to the signal noise, i.e. the raw data sampled from the systems usually cannot be used directly but specific signal processing technologies should be developed to eliminate noises of the signals.

Machine learning is a significant research topic in recent years having been applied widely in many engineering fields [18]–[21]. We focus on neural networks that have broad applications in fault detection of aviation industry [22], [23]. Artificial neural network (ANN) and support vector machine (SVM) are traditional shallow learning models, which are not effective to learn the non-linearity and uncertainty of the complex system [24]. Another network is the convolutional neural network (CNN) usually used in image identification which demands lower in real-time. Consequently, the Long Short-Term Memory (LSTM) network, a significant branch of Recurrent Neural Network (RNN), is employed to the FDI of MEA in this work since it is able to learn from the long-term historical information of the previous context for problems in time series as our raw sampled data. Some efforts, [25] for the engines, [26] for the rolling bearing and [27] for potential cyber-attacks, have been carried out, but scarcely any of them are able to monitor the whole MEA system during a flight. In this paper, we establish a comprehensive model of a MEA based on the Boeing 787 with the help of PSCAD/EMTDC® and implement the real-time fault detection and isolation method on the field-programmable gate array (FPGA).

The remainder of this paper is structured as follows. Section II briefly describes the high potential failure on the MEA as well as the model exploited and the conditions simulated in this work. Section III illustrates the proposed method to deal with the problem of FDI, including the basic theory of LSTM, the procedure of the off-line construction and the real-time monitoring. Section IV presents the scenarios and data sampling approach, and then compares and summarizes the results of different configurations of the neural network implementation, followed by the conclusions in Section V.

II. PROBLEM DESCRIPTION

The most recent commercial transport aircraft, such as the Boeing 787 and the Airbus A380, are largely equipped with electrical systems. A potential failure may occur on any location of an MEA with a variety of components. First, the fault tree for the power system of the MEA is analyzed to figure out the failure that is more likely to cause the loss of aircraft. Then specific operating conditions of the MEA are defined in accordance to the fault tree and simulated by a comprehensive model based on the Boeing 787 with the help of PSCAD/EMTDC®. The experimental conditions include both normal and fault scenarios, and the sampled datasets from the model are then used to verify our proposed method.

A. FAULT TREE

In the MEA power system, the potential failure mechanisms are intricate, so the fault tree is used as a basic analysis method. The failure rates of typical components that may encounter problems on an MEA are displayed in Fig. 1. The probability of total function loss of aircraft must less than 10^{-9}/flight hour [28]. Electrical actuators failing will directly lead to the loss of aircraft, and the failure rate of one actuator is estimated as 3.2×10^{-4}/flight hour. A number of failures may result in this event, such as power supply failure, signal failure and mechanical/electrical failure [29]. The probability of the loss of power supply can derive from the failure of generator, converter or busbar. Sensor failure is considered as the cause of signal loss.

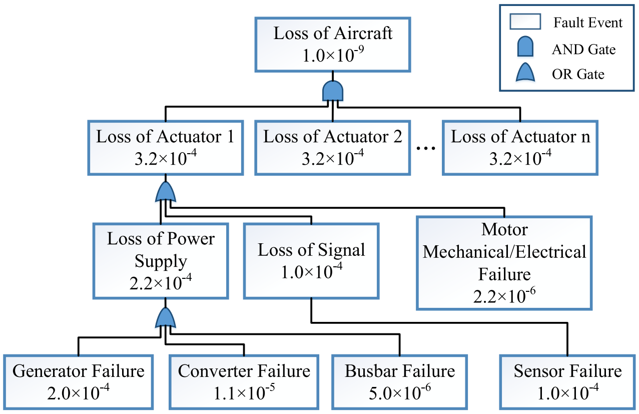

Fault Event
AND Gate
OR Gate
Loss of Aircraft 1.0×10^{-9}
Loss of Actuator 1 3.2×10^{-4}
Loss of Actuator 2 3.2×10^{-4}
...
Loss of Actuator n 3.2×10^{-4}
Loss of Power Supply 2.2×10^{-4}
Loss of Signal 1.0×10^{-4}
Motor Mechanical/Electrical Failure 2.2×10^{-6}
Generator Failure 2.0×10^{-4}
Converter Failure 1.1×10^{-5}
Busbar Failure 5.0×10^{-6}
Sensor Failure 1.0×10^{-4}

FIGURE 1. A fault tree diagram built for the MEA system [29].

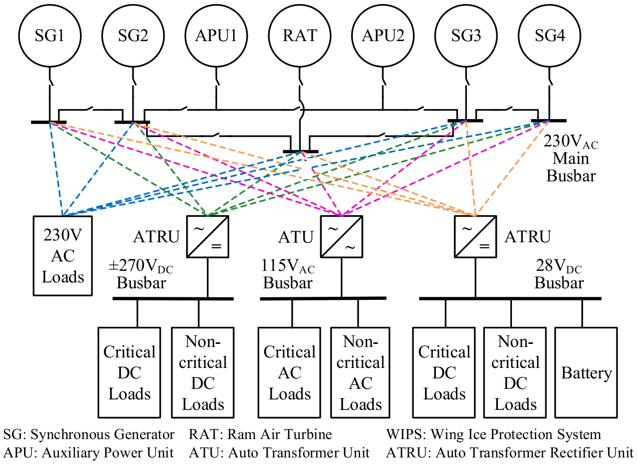


FIGURE 2. Structure of Boeing 787 power system.

B. MODEL DESCRIPTION

A simulation model of the Boeing 787 power system similar to the one developed and verified in [30] is used for this work, as shown in Fig. 2. The power supply system consists of four 250kVA variable frequency starter generators (SG1-SG4), the auxiliary power unit composed of two 225kVA variable frequency generators (APU1 and APU2) and one 50kVA generator driven by the ram air turbine system (RAT). The main busbar of the power supply system is 230VAC/400Hz, and the secondary buses with different voltages (i.e. ±270V VDC, 115VAC/400Hz and 28VDC) converted by auto transformers are applied to multiple loads.

In the constant voltage variable frequency bus power system, the main bus should solely be connected to one generator at any moment in time, the secondary bus should only be connected to one main bus simultaneously [31]. Therefore, one generator with its auxiliaries and supplied loads can be separated as an independent unit. We simplify the model which comprises one synchronous generator, the Wing Ice Protection System (WIPS) that is a 230VAC load, a sychronous motor as an electrical actuator, two resistive loads respectively on the ±270VDC bus and 28VDC bus, an energy storage system charged or discharged on 28VDC bus as well as converters needed in the system, as displayed in Fig. 3.

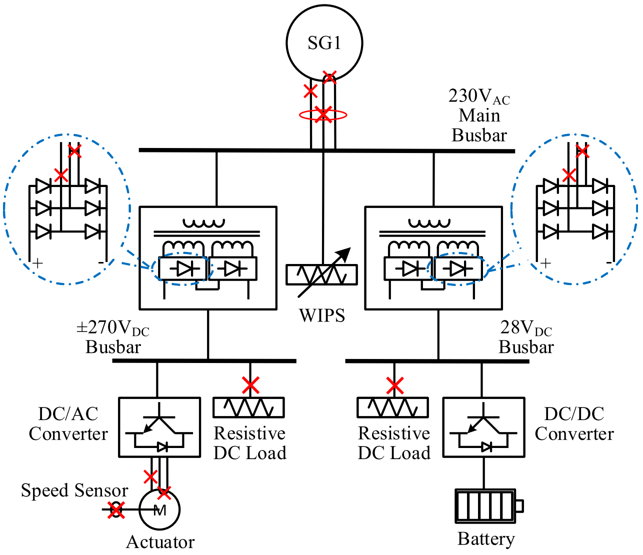


FIGURE 3. Fault locations on the model of Boeing 787 power system.

TABLE 1. Conditions modeled in the MEA system.

Locations	Descriptions	Conditions
No fault	Load change of WIPS Speed change of motor	Normal Normal
Generator	Phase-to-phase short circuit Open circuit on one phase Open circuit on three phases	Fault1 Fault2 Fault3
Converter	Short/open circuit on ±270VDC line Short/open circuit on 28VDC line	Fault4 Fault5
DC load	Open circuit of load on ±270VDC line Open circuit of load on 28VDC line	Fault6 Fault7
Motor	Phase-to-phase short circuit Open circuit on one phase	Fault8 Fault9
Speed encoder	Omission Bias	Fault10 Fault11

C. SELECTED CONDITIONS OF THE MEA POWER SYSTEM

Twelve operating conditions consisting of 1 normal (including 2 scenarios) and 11 faults in total are selected in our simulated MEA system and summarized in Table 1. The fault locations are shown in Fig. 3 with red marks. Generator failure includes phase-to-phase short circuit fault and open circuit fault on one phase or three phases inside the generator. Converter failure contains phase-to-phase short circuit fault or open circuit fault on one phase in two converters of two DC branch lines (i.e. ±270VDC and 28VDC). Sensor failure comprises omission and bias in speed encoder of the actuator. Motor electrical failure includes phase-to-phase short circuit fault and open circuit fault on one phase. DC load (one resistance loaded on each one of DC branch lines respectively) failures are also introduced in our simulation.

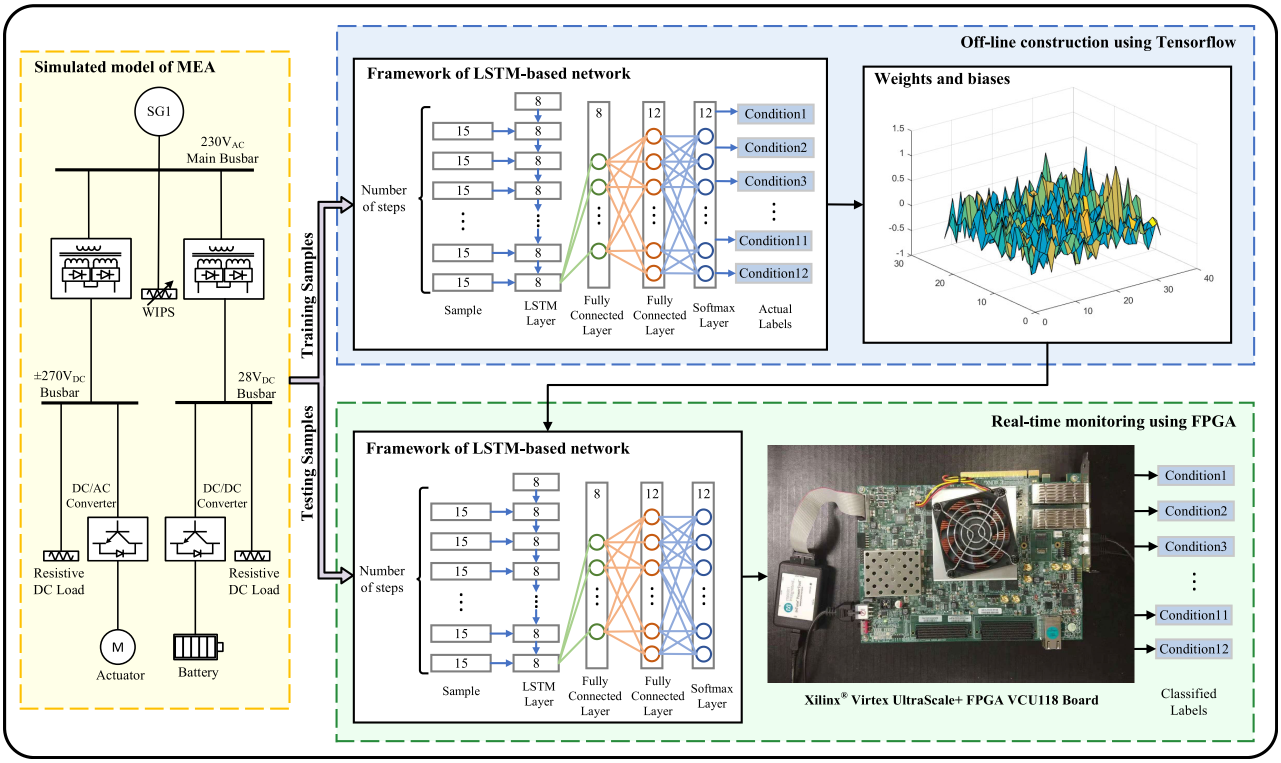


FIGURE 4. General procedure of the proposed FDI method.

III. GENERAL PROCEDURE OF THE PROPOSED METHOD

In this section, the general procedure of our proposed method for the FDI of MEA is presented in Fig. 4, which includes two stages: off-line construction and real-time monitoring. Both training and testing samples are obtained from the transient simulation by PSCAD. In the off-line construction stage, Tensorflow library is applied to build up LSTM-based networks to train a model that is capable of classifying various operating conditions of the MEA with respect to 15 features collected in simulation. The architecture of the model with optimal weights and biases is implemented on the FPGA in the real-time monitoring stage to estimate the condition in accordance to testing samples.

A. LONG SHORT-TERM MEMORY ALGORITHM

The RNN is a network with loops in itself, persisting the past information, so it is an effective tool for classification and regression problems with datasets in a sequential manner. Unfortunately, when the gap of connection between the previous information and the present task becomes very large, RNN is confronted with vanishing or exploding gradient problem called long-term dependencies [25] and is unable to learn the connections any more in practice. Thanks to the novel concept of LSTM network [32], the problem is successfully overcome by controlling the rate of data flow using "gate units" as it will be explicitly illustrated in this subsection. There have been numerous variants of LSTM. The standard LSTM is expressed as below:

$$\begin{aligned} f_t &= \sigma\left(W_f \cdot [x_t, h_{t-1}] + b_f\right) \\ i_t &= \sigma\left(W_i \cdot [x_t, h_{t-1}] + b_i\right) \\ o_t &= \sigma\left(W_o \cdot [x_t, h_{t-1}] + b_o\right) \\ \tilde{c}_t &= \tanh\left(W_c \cdot [x_t, h_{t-1}] + b_c\right) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \\ h_t &= o_t \circ \tanh\left(c_t\right) \end{aligned} \tag{1}$$

where the input is x_t and the output hidden state is h_t. The concept cell state c_t is crucial in LSTM that will reasonably accumulate the historical information being protected and controlled by gate layers, namely forget, input and output gate layer (f_t, i_t and o_t respectively in equation (1)) which are made by sigmoid layers. The gates are in charge of removing or adding information to the cell state and the output of them ranges from zero to one. A value of zero means "completely forget it", while a value of one means "completely keep it". $\tilde{c}_t$ is new memory content. Weights W and biases b are calculated during the process of network learning. "$\cdot$" is matrix product and "$\circ$" is hadamard product that means the point-wise multiplication of two matrices of the same dimension.

Some concepts can be summarized as following. 1) The information needed by the present step is from the current input x_t and the hidden state of last step h_{t-1}. 2) The sources for the present cell state c_t are itself in the last step c_{t-1} and the new memory content $\tilde{c}_t$, and they are independently controlled by forget and input gate. 3) Finally, the present output hidden state h_t derives from non-linearized cell state c_t filtered by output gate o_t.

One popular variant of LSTM is to add "peephole connections" by letting the gate layers also look at its internal cells [33]. Comparing with (1), the difference is that there is cell state addition in gate units in (2):

$$\begin{aligned}
f_t &= \sigma\left(W_f \cdot [x_t, h_{t-1}, c_{t-1}] + b_f\right)\\
i_t &= \sigma\left(W_i \cdot [x_t, h_{t-1}, c_{t-1}] + b_i\right)\\
o_t &= \sigma\left(W_o \cdot [x_t, h_{t-1}, c_t] + b_o\right)\\
\tilde{c}_t &= \tanh\left(W_c \cdot [x_t, h_{t-1}] + b_c\right)\\
c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t\\
h_t &= o_t \circ \tanh\left(c_t\right)
\end{aligned} \tag{2}$$

Another increasingly popular LSTM variant is the Gated Recurrent Unit (GRU) [34]. It simplifies the standard LSTM by combining the forget and input gates into a single update gate z_t as well as abandoning the cell state and merging it with the hidden state, as shown below:

$$\begin{aligned}
z_t &= \sigma\left(W_z \cdot [x_t, h_{t-1}] + b_z\right)\\
r_t &= \sigma\left(W_r \cdot [x_t, h_{t-1}] + b_r\right)\\
\tilde{h}_t &= \tanh\left(W_h \cdot [x_t, r_t \circ h_{t-1}] + b_h\right)\\
h_t &= z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h}_t
\end{aligned} \tag{3}$$

We compare the performances of the FDI using all three types of LSTM in terms of accuracy and the consumption of time and resources on the FPGA.

B. OFF-LINE CONSTRUCTION USING TENSORFLOW

TensorFlow [35] was released by Google as an open-source library that is widely used in large-scale machine learning. It provides plentiful Application Programming Interfaces (APIs) for users to express and execute machine learning algorithms. Furthermore, it is flexible enough for researchers who want to experiment with different model architectures and optimization methods. The core TensorFlow library is implemented in C++ and its Python-based high-level APIs are adequate for building models.

As for the LSTM algorithms studied in this work, three types of LSTM layers with various parameter configurations supported by TensorFlow are trained and compared to find the optimal network architecture for the FDI of MEA. The LSTM variants include standard LSTM, LSTM with peepholes and GRU. The framework of LSTM-based network is shown in Fig. 4. Each sample is normalized and reshaped to a two-dimensional matrix. The calculation of hidden states for each time step of the sample called one-time-step process (i.e. mathematically expressed as (1), (2) or (3)) is iterated step-by-step in the LSTM layer. Then, two fully connected layers are directly followed by the last step of LSTM, and the former layer is filtered by Rectified Linear Unit (ReLU) activation function. Finally, the learned deep features are fed into a softmax layer for condition classification. The number (15, 8 or 12) on each layer block represents the number of neurons for the layer. The algorithm used to train the network is Adam [36].

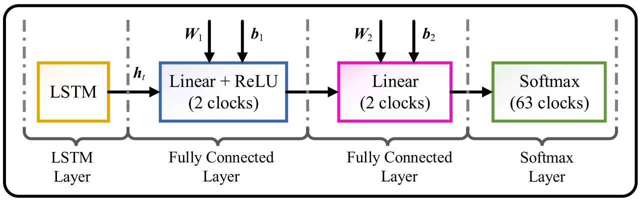


FIGURE 5. Architecture of the LSTM-based network hardware module with respect to latency.

C. REAL-TIME MONITORING ON FPGA

In order to achieve the goal of monitoring the condition of the MEA in a real-time manner, the LSTM-based hardware implementation on FPGA is proposed; the algorithm's execution time has been slashed thanks to the highly parallelism of FPGA. The detailed implementation of the algorithm on FPGA will be elaborated as follows.

The platform used in this work is Xilinx Virtex UltraScale+ FPGA VCU118 Evaluation Kit [37], shown in Fig. 4, which provides a hardware environment for designs targeting the UltraScale+ XCVU9P-L2FLGA2104E FPGA with 2586K programmable logic cells, 345.9Mb Block RAM, and 6840 DSP Slices.

Fig. 5 shows the architecture of the LSTM-based network hardware module with respect to latency. The LSTM layer is an iteration process of one-time-step algorithm. The latency for the LSTM layer depends on the type of the LSTM applied in the network and the number of time steps of the sample. The explicit structure of one-time-step for each variant is presented in Fig. 6. The vertically aligned operations are executed in parallel.

It can be seen from Fig. 6 (a) and Fig. 6 (b) that the latency of the two algorithms (namely standard LSTM and LSTM with peepholes) are almost the same for hardware implementation although more hadamard product operations are needed for the latter. On the other hand, the GRU algorithm (Fig. 6 (c)) which simplifies the standard LSTM by merging gates and states also reduce the latency on hardware to some degree.

In addition, the fixed-point format is set by trial and error to minimize the consumption of time and resources in the FPGA implementation within sufficient accuracy. Finally, we take the 24-bit word length with 19 fractional bits fixed-point format in this study.

Operation units appearing in the diagrams are summarized. Concat unit is used to merge two arrays:

$$concat\left(\boldsymbol{x}, \boldsymbol{h}\right) = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{h} \end{bmatrix}. \tag{4}$$

Linear unit is to calculate the linear combination:

$$linear\left(\boldsymbol{x}\right) = \boldsymbol{W} \cdot \boldsymbol{x} + \boldsymbol{b}. \tag{5}$$

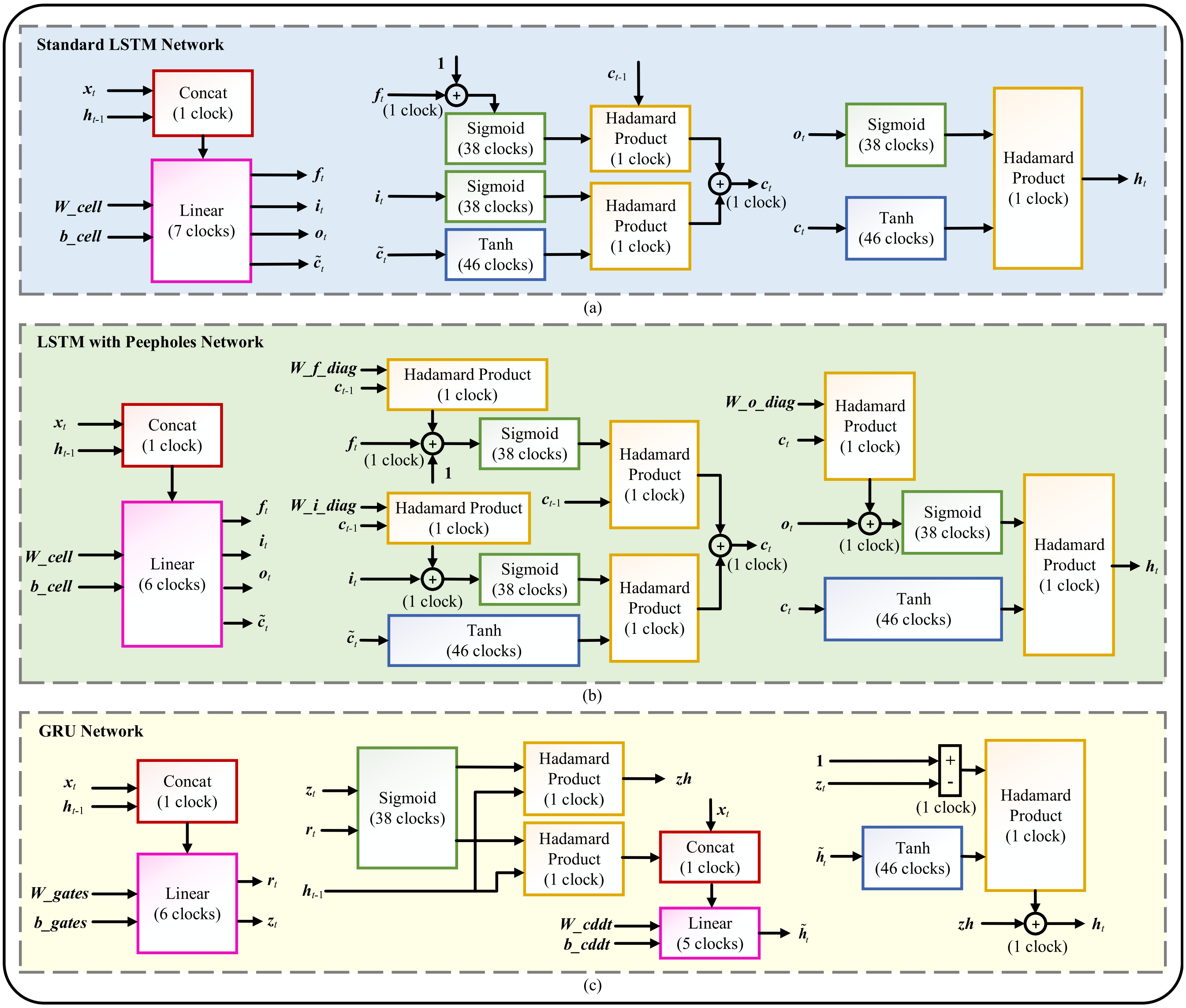


FIGURE 6. Framework of LSTM networks computation over one-time-step.

ReLU, sigmoid and tanh units that are popular activation functions used in neural networks can be expressed as:

$$Relu\,(x) = \begin{cases} x & if\ x > 0 \\ 0 & if\ x \leq 0, \end{cases} \tag{6}$$

$$sigmoid\,(x) = \frac{1}{1+e^{-x}}, \tag{7}$$

$$\tanh\,(x) = \frac{2}{1+e^{-2x}} - 1. \tag{8}$$

For two matrices $\boldsymbol{A}$, $\boldsymbol{B}$ of the same dimension $m \times n$, the elements of the hadamard product $\boldsymbol{A} \circ \boldsymbol{B}$ can be calculated by:

$$(\boldsymbol{A} \circ \boldsymbol{B})_{i,j} = (\boldsymbol{A})_{i,j}(\boldsymbol{B})_{i,j}. \tag{9}$$

For a vector $\boldsymbol{V}$, the ith element of the vector $\boldsymbol{V}_i$ after the softmax operation is given by:

$$softmax\,(\boldsymbol{V}_i) = \frac{e^{V_i}}{\sum_j e^{V_j}}. \tag{10}$$

IV. EXPERIMENT RESULTS AND DISCUSSION

To verify the effectiveness of our proposed method, the experiments of 11 fault scenarios and 2 normal ones are carried out, as mentioned in Section II C. In this section, the details of data sampling from our simulated model will be introduced. The simulation is conducted in PSCAD/EMTDC®. Then we compare and analyze the effect of various LSTM networks and their parameters on the performance of the FDI on MEA. The off-line construction and real-time monitoring stages are implemented on CPU with Tensorflow and FPGA respectively. For this study, a total of three trials are carried out for each different configuration of the LSTM network and 100-epoch iterations are processed.

A. SCENARIOS AND DATA SAMPLING

There are 12 operating conditions of the MEA chosen as our study cases. One normal condition contains 2 scenarios:

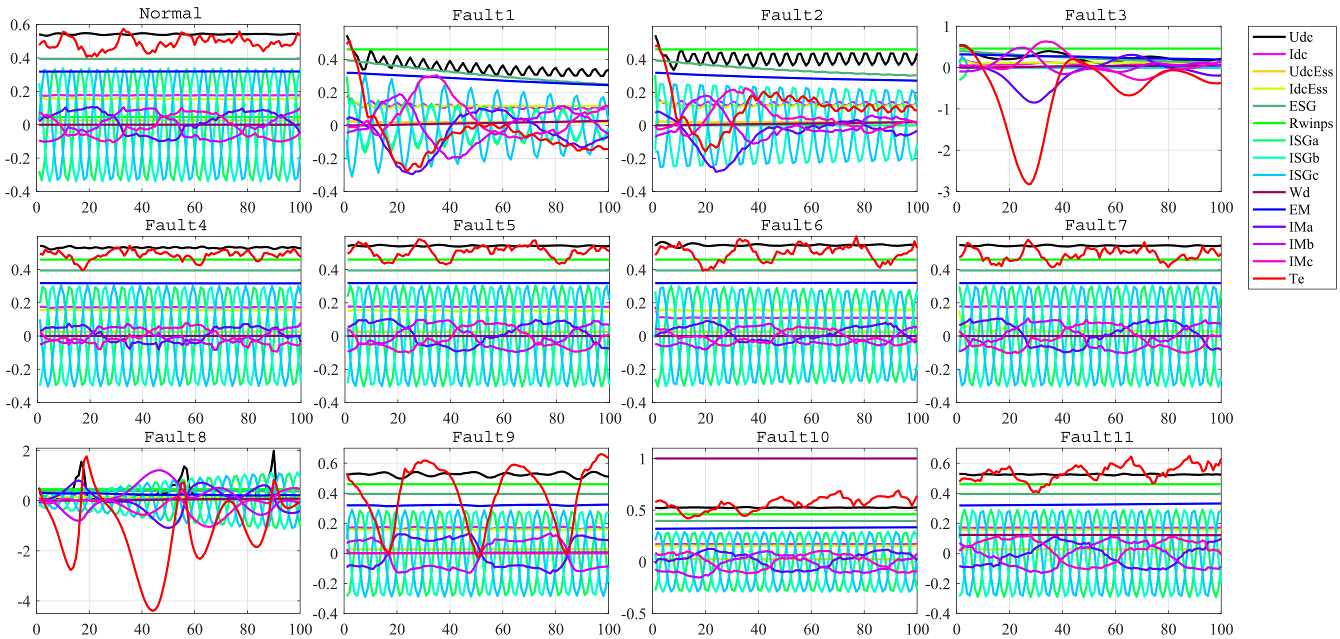


FIGURE 7. Examples of the time series data for 12 different conditions.

the change of 230VAC load and the change of actuator speed. The other 11 conditions are 11 different fault scenarios located on 5 major components in the MEA power system. All of these scenarios are simulated in PSCAD/ EMTDC®.

Faults are injected in the model after the system reaches steady-state (that is 6s after starting simulation in our model). For training samples, the exact start time of a fault injection is configured at evenly distributed points in one period of the system (the frequency of the system is 400Hz) considering there are some differences when the fault happens at the different points of an alternating current period. In the case of testing samples, the start time is set randomly during the same period as training.

As mentioned above, we take 2 scenarios into account as the normal condition. One is the change of 230VAC load (WIPS) on the main bus, and the other is the change of actuator speed under control. The start time for training and testing is similar to other fault conditions. While the setting reference of the load or the speed will change to a random value in a limited scope for each sample.

TABLE 2. Features for each sample collected from the MEA system.

Locations	Signals	Features
Main line	Three-phase RMS voltage Current on phase A Current on phase B Current on phase C	ESG ISGa ISGb ISGc
±270VDC line	Voltage Current	Udc Idc
28VDC line	Voltage Current	UdcEss IdcEss
WIPS	Resistance	Rwips
Motor	Three-phase RMS voltage Current on phase A Current on phase B Current on phase C	EM IMa IMb IMc
Sensor	Speed difference between reference and measurement Torque	Wd Te

Table 2 represents the signals including voltages, currents, resistance, speed and torque that are able to be measured from the system model. Each channel of the signals acts as a feature for samples, and the tabs are listed in the last column. The principles of the selected features are: the accessibility of measurement, the variability in different conditions as well as the ability of covering all branches and components. The range of the time we collect data is from the start time of a fault or change to a specific time point (6.2s in our case). The length of each sample is not the same due to the different start points. The signals under all operating conditions are collected at a sampling frequency of 4 kHz, so that the time steps of one sample amounts to a random number between 700 to 800.

In summary, we obtain 101 samples for training and 300 for testing under each scenario. The number of total training samples is 1313, and testing samples is 3900. Each sample has 15 features in time series. The number of time steps for each sample can be reached between 700 and 800.

The time series of one sample for each condition are randomly picked out and displayed in Fig. 7. Each colored line corresponds to a feature. Each unit on the horizontal axis stands for one time-step and the vertical axis stands for the signals or features. Only the first 100 time steps are plotted

TABLE 3. Comparison of different LSTM networks with respect to time consumption.

LSTM Network	Time Consumption (μs)	
	FPGA	CPU
Standard LSTM	98.13	1547.59
LSTM with Peepholes	97.13	1609.48
GRU	94.14	1189.25

TABLE 4. Comparison of other neural networks with respect to accuracy and FPGA time consumption.

Neural Network	Accuracy	Time Consumption (μs)
RNN	99.23%	49.13
CNN1D	99.64%	122.72
LIBSVM	98.67%	1307718.02

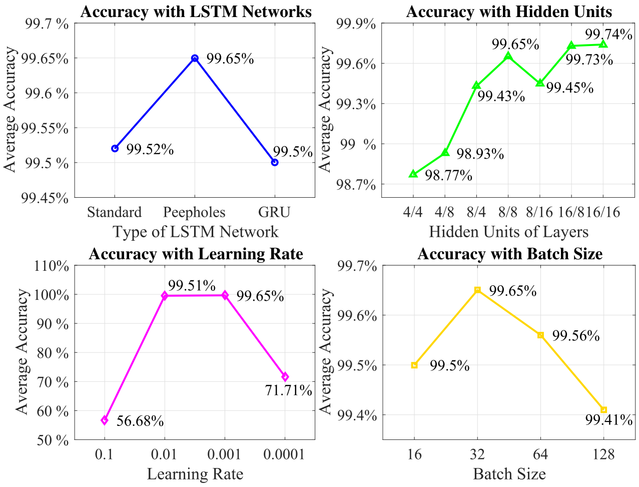


FIGURE 8. FDI performance under different type of LSTM network, hidden units of layers, learning rate and batch size.

for conciseness. It is clear that supervising all these signals and evaluating the condition of the system all the time is difficult since signals from some conditions have few differences, such as fault4, fault5, fault6 and fault7. Even though they are put together for comparison it is not easy to distinguish between them.

B. TYPE OF LSTM NETWORK

The networks applied to the system are standard LSTM, LSTM with peepholes, and GRU. The evaluation criteria of better performance of the network is less time consumption with reasonable accuracy and resource utilization. First, we fix all the other parameters except for the type of LSTM network. The input of the network is 15 features from a batch size of samples. The hidden layers are a single LSTM layer and a fully connected layer filtered by ReLU activation function both with 8 hidden units. The output layer is a fully connected layer connected to a softmax classifier having 12 classes because the conditions simulated and collected are 12. The batch size for training is 32, the optimizer is Adam with the learning rate of 0.001 and the time steps are selected as 100 for each sample.

The consumption of time and FPGA resources in Table 3 and Table 5 proves the aforementioned analysis of hardware implementation of three algorithms. GRU with less execution time and resource utilization a little bit outperforms standard LSTM and LSTM with peepholes. On the other hand, the average accuracy of the three trials of constructed models using the three LSTM variants are displayed in Fig. 8(a), which indicates that they are similar, but the LSTM with peepholes may slightly outperform the other two in terms of classification accuracy. Together, the LSTM with peepholes could be the most suitable architecture since the accuracy is more significant under the applicable consumption of time and resources in the case of the FDI on MEA.

C. HYPER-PARAMETERS

In this subsection, we analyze and compare the influence of various hyper-parameters of the LSTM network. Hidden units of hidden layers, batch size for training and learning rate are considered in our work. In order to evaluate each hyper-parameter, we follow the principle of a single variable, that is only one parameter is changeable each time. The scale of the network structure has a significant effect on the runtime and resource utilization of FPGA in real-time monitoring stage. Therefore, the single-layer LSTM and the fully connected layer both with 8 hidden units are selected since more complex architecture will not notably improve the accuracy, as illustrated in Fig. 8(b). According to Fig. 8(c) and 8(d), 0.001 and 32 are selected as learning rate and batch size respectively due to their outstanding performance. The results using only the network of the LSTM with peepholes are presented here for saving space. The curve trends of the other two networks are the same.

D. TIME STEPS OF SAMPLES

We also conduct experiments to find out how many time steps for a sample is the most proper for the data sampling. From Fig. 9, it is apparent that the runtime of the algorithm increases linearly with increase of time steps. When it comes to the accuracy, the 100 time steps stands out for its highest precision with relatively less time consumption. For the sake of simplicity, the data are based on the network of the LSTM with peepholes, and the curve trends of the other two networks are the same.

E. SOFTWARE VS. HARDWARE

The comparison of the performance between CPU and FPGA are made to prove the necessity of using FPGA for real-time application, exactly as our study case. The same C++ code runs on Visual Studio 2017 for CPU and Vivado HLS 2017.2 for FPGA. The time consumption of one sample of the three LSTM variants for both CPU and FPGA implementation is presented in Table 3. More than 1000μs are required for running the algorithms on CPU while less than 100μs on FPGA. It can be concluded that the FPGA provides a runtime speed-up of about 15x, which means whenever one sampling

TABLE 5. Comparison of different neural networks with respect to hardware resource utilization on the Xilinx XCVU9P FPGA.

NN Type	BRAM	DSP	FF	LUT
Standard LSTM	84(1%)	3195(46%)	467834(19%)	521559(44%)
LSTM with Peepholes	84(1%)	3219(47%)	467693(19%)	521618(44%)
GRU	44(1%)	2290(33%)	325738(13%)	359103(30%)
RNN	84(1%)	1626(23%)	320651(13%)	316723(26%)
CNN1D	169(3%)	826(12%)	992523(41%)	1146756(96%)
LIBSVM	8(0%)	29(0%)	1453775(61%)	1181086(99%)

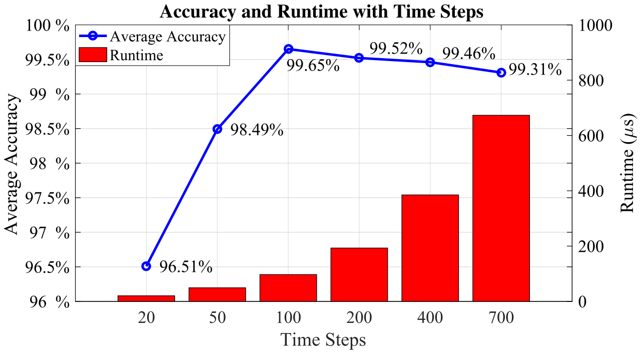


FIGURE 9. FDI performance under different time steps.

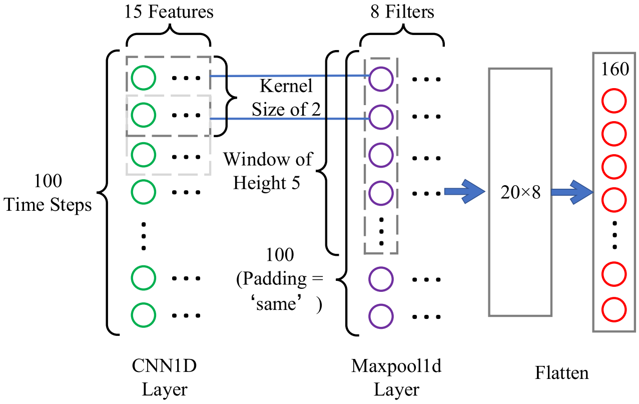
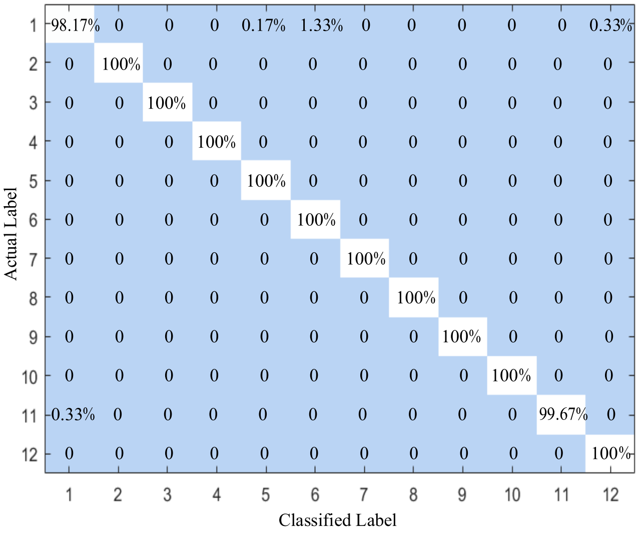


FIGURE 10. Architecture of CNN1D-based neural network.

process is completed, an evaluation can be made accordingly for the sampling frequency at 4kHz. Since the sampling frequency is the highest frequency to update the condition of the monitoring system, the speed meets the demand of the real-time fault detection and isolation application.

F. OTHER NEURAL NETWORKS

Other prevailing neural networks are introduced to compare with the performance of our proposed method. They are one shallow learning model, Support Vector Machine (SVM), and two deep learning models, Recurrent Neural Network (RNN) and 1D Convolutional Neural Network (CNN1D). For the sake of fairness, the training and testing samples are the same with 100 time steps. In addition, the architectures of deep learning models are as consistent as our proposed one. Namely, only the LSTM layer is replaced by RNN or CNN1D layer, and the configuration for the two fully connected layers and softmax layer remain unchanged. RNN layer is set to have 16 hidden units. While a kernel size of 2 is applied to each of the 8 filters for CNN1D layer which is connected to a maxpool1d layer sliding a window of height 5 across the output of CNN1D layer, as shown in Fig. 10. As for the SVM model, we bring in the LIBSVM, a popular open source library [38]. From Table 4 and Table 5, it indicates that the shallow learning model (LIBSVM), which is least accurate but consuming most runtime and resources, is not qualified for this kind of comprehensive problem. Likewise, RNN cannot reach the accuracy as LSTM does due to its inability to avoid the vanishing or exploding gradient problem in long-term sequential data. On the other hand, CNN1D may be an alternative in terms of accuracy, but we definitely have to pay the price, i.e. tremendous hardware resources.



	1	2	3	4	5	6	7	8	9	10	11	12
1	98.17%	0	0	0	0.17%	1.33%	0	0	0	0	0	0.33%
2	0	100%	0	0	0	0	0	0	0	0	0	0
3	0	0	100%	0	0	0	0	0	0	0	0	0
4	0	0	0	100%	0	0	0	0	0	0	0	0
5	0	0	0	0	100%	0	0	0	0	0	0	0
6	0	0	0	0	0	100%	0	0	0	0	0	0
7	0	0	0	0	0	0	100%	0	0	0	0	0
8	0	0	0	0	0	0	0	100%	0	0	0	0
9	0	0	0	0	0	0	0	0	100%	0	0	0
10	0	0	0	0	0	0	0	0	0	100%	0	0
11	0.33%	0	0	0	0	0	0	0	0	0	99.67%	0
12	0	0	0	0	0	0	0	0	0	0	0	100%

FIGURE 11. Confusion matrix of the LSTM with peepholes network for one trial.

Therefore, we can reach the conclusion that our LSTM-based network is competitive for this complex fault detection and isolation case.

G. EXPERIMENTAL RESULTS

Putting all the carefully selected configurations of the LSTM network into use, we are able to achieve the goal to make an evaluation of the real-time condition of the MEA during one sampling time on FPGA and to expect the performance to be more accurate than 99.5%. The confusion matrix of the proposed method for one trial is shown in Fig. 11. Each row of the matrix represents the actual label of instances, while each column represents the classified label of instances. The percentage in the matrix represents the ratio of the number of samples which are classified to the label and the total number of samples as the label they should actually belong to. For instance, there are 300 samples belong to the Label 11; 1 sample classified to Label 1 and the other 299 samples are classified correctly to Label 11. So the number in the 11th row and the 1st column of the matrix is 0.33% (1/300), and in the 11th column is 99.67% (299/300). Almost all of the fault conditions can be detected and isolated correctly. Only Fault 10 may get a little bit confused with merely 1 in 300 error rate. The vast majority of errors happen in normal condition which may be classified into Fault 4, Fault 5 and Fault 11, but only with 1, 8 and 2 in 600 error rates respectively. It is reasonable to some extent, because the curve trends of the 15 features in specific cases are similar.

Additionally, a commercial electric aircraft model based on Airbus E-Fan in Simscape of Matlab [39] is also tested to validate the feasibility and generality of the approach. This model has similar major components as the model we build, containing a combustion engine, a generator, two DC networks connected with two converters, a battery and a set of loads. On the high voltage DC network, there is a mechanical model of the aircraft acting as a load which considers atmosphere condition and motion dynamics. The mass of the fuel consumed by the engine is also included in the simulation. Because of the restriction of the model, 11 features, including voltages, currents, speed, torque and flight height, are adopted to detect and isolate 5 faulted and 1 normal scenarios. The faulted scenarios take place respectively on the five major components, i.e. generator, converter, motor, sensor and load. The architecture of the LSTM-based network and the data sampling method are completely consistent with those applied in our model. Ultimately, the average accuracy of three trials is 99.85% and the best performance is only 1 in 1800 classified to the wrong scenario. The result proves that our data-based method can still work well, although the model may not be exactly the same.

V. CONCLUSION

In this paper, we propose an FPGA-based neural network method to handle the real-time FDI on the MEA. We explain the procedure of the method which contains two stages: offline construction and real-time monitoring. In the first stage, TensorFlow library is used to construct various LSTM-based networks with different types and hyper-parameters of the network layers. In the second stage, the detailed FPGA implementation is carefully designed to maximize parallelism of the algorithms. Specific operating conditions of the MEA simulated by the comprehensive model in PSCAD/EMTDC® as well as a commercial electric aircraft model in Simscape are selected to verify the effectiveness and generality of the method proposed. The contrast experiments indicate that the netwok of LSTM with peepholes outperforms the other architectures (i.e. standard LSTM, GRU, CNN1D, RNN and SVM) in our case. After employing all the best configurations of the network layers, the results manifest that the runtime speed-up of algorithms can achieve up to 15× on FPGA than CPU, which means an evaluation can be made in one sampling time. Besides, the accuracy of the evaluation of the real-time condition can be better than 99.5% within reasonable hardware resource utilization. While the possibility of critical fault occurring simultaneously is small, in future research, we will investigate the detection and isolation of simultaneous multiple faults in aircraft systems.

REFERENCES

[1] K. Ni, Y. Liu, Z. Mei, T. Wu, Y. Hu, H. Wen, and Y. Wang, "Electrical and electronic technologies in more-electric aircraft: A review," *IEEE Access*, vol. 7, pp. 76145–76166, 2019.

[2] B. Sarlioglu and C. T. Morris, "More electric aircraft: Review, challenges, and opportunities for commercial transport aircraft," *IEEE Trans. Transp. Electrification*, vol. 1, no. 1, pp. 54–64, Jun. 2015.

[3] X. Xin-Li, J. Zhen, and F. Bing-Xiao, "Research on the fault diagnosis of flight control system," in *Proc. Int. Conf. Mechatronic Sci., Electr. Eng. Comput. (MEC)*, Aug. 2011, pp. 1427–1431.

[4] F. Chen, J. Niu, and G. Jiang, "Nonlinear fault-tolerant control for hypersonic flight vehicle with multi-sensor faults," *IEEE Access*, vol. 6, pp. 25427–25436, 2018.

[5] B. Guo and Y. Chen, "Output integral sliding mode fault tolerant control for nonlinear systems with actuator fault and mismatched disturbance," *IEEE Access*, vol. 6, pp. 59383–59393, 2018.

[6] L. Hongliang, H. Yannian, Q. Jianglei, and S. Haocheng, "Fault diagnosis using particle filter for MEA typical components," *J. Eng.*, vol. 2018, no. 13, pp. 603–606, 2018.

[7] Y. Ji and J. Bals, "Multi-model based fault detection for the power system of more electric aircraft," in *Proc. 7th Asian Control Conf.*, Aug. 2009, pp. 93–98.

[8] B. Nahid-Mobarakeh and M. G. Simoes, "Multi-agent based fault detection and isolation in more electric aircraft," in *Proc. IEEE Ind. Appl. Soc. Annu. Meeting*, Oct. 2013, pp. 1–8.

[9] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 293–311, 2003.

[10] J. A. Momoh, D. M. V. Kumar, A. S. Ishola-Salawu, R. Sowah, and R. Button, "Lab VIEW based implementation of remedial action for DC arcing faults in a spacecraft," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, vol. 1, Jul. 2003, pp. 491–498.

[11] Y. Li, Y. Wei, K. Feng, X. Wang, and Z. Liu, "Fault diagnosis of rolling bearing under speed fluctuation condition based on Vold-Kalman filter and RCMFE," *IEEE Access*, vol. 6, pp. 37349–37360, Jul. 2018.

[12] D. Xu and Z. Feng, "EEMD based multiple degradation feature extraction method for electronic power panel," in *Proc. Int. Conf. Sens., Diagnostics, Prognostics, Control (SDPC)*, Aug. 2017, pp. 683–687.

[13] Z. Li, Q. Li, Z. Wu, J. Yu, and R. Zheng, "A fault diagnosis method for on load tap changer of aerospace power grid based on the current detection," *IEEE Access*, vol. 6, pp. 24148–24156, 2018.

[14] J. Zhang, Y. Zhang, and Y. Guan, "Analysis of time-domain reflectometry combined with wavelet transform for fault detection in aircraft shielded cables," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4579–4586, Jun. 2016.

[15] D. L. C. Mack, G. Biswas, and X. D. Koutsoukos, "Learning Bayesian network structures to augment aircraft diagnostic reference models," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 358–369, Jan. 2017.

[16] R. Telford and S. Galloway, "Fault classification and diagnostic system for unmanned aerial vehicle electrical networks based on hidden Markov models," *IET Electr. Syst. Transp.*, vol. 5, no. 3, pp. 103–111, Sep. 2015.

[17] X. Hu, R. Subbu, P. Bonissone, H. Qiu, and N. Iyer, "Multivariate anomaly detection in real-world industrial systems," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IEEE World Congr. Comput. Intell.)*, Jun. 2008, pp. 2766–2771.

[18] S. Zhang, Y. Wang, M. Liu, and Z. Bao, "Data-based line trip fault prediction in power systems using LSTM networks and SVM," *IEEE Access*, vol. 6, pp. 7675–7686, 2018.

[19] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Al-Emadi, and M. Gabbouj, "Real-time fault detection and identification for MMC using 1-D convolutional neural networks," *IEEE Trans. Ind. Electron.*, vol. 66, no. 11, pp. 8760–8771, Nov. 2019.

[20] L. Zhang, Y. Zhu, and W. X. Zheng, "Synchronization and state estimation of a class of hierarchical hybrid neural networks with time-varying delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 2, pp. 459–470, Feb. 2016.

[21] B. Chang, R. Yang, C. Guo, S. Ge, and L. Li, "A new application of optimized random forest algorithms in intelligent fault location of rudders," *IEEE Access*, vol. 7, pp. 94276–94283, Jul. 2019.

[22] A. A. Silva, A. M. Bazzi, and S. Gupta, "Fault diagnosis in electric drives using machine learning approaches," in *Proc. IEEE Int. Electr. Mach. Drives Conf. (IEMDC)*, May 2013, pp. 722–726.

[23] S. Hussain, M. Mokhtar, and J. M. Howe, "Sensor failure detection, identification, and accommodation using fully connected cascade neural network," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1683–1692, 2015.

[24] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief network," *IEEE Trans. Ind. Electron.*, vol. 65, no. 3, pp. 2727–2736, Mar. 2018.

[25] M. Yuan, Y. Wu, and L. Lin, "Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network," in *Proc. IEEE Int. Conf. Aircraft Utility Syst.*, Oct. 2016, pp. 135–140.

[26] W. Lu, Y. Li, Y. Cheng, D. Meng, B. Liang, and P. Zhou, "Early fault detection approach with deep architectures," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 7, pp. 1679–1689, Jul. 2018.

[27] M. B. Kamal, G. J. Mendis, and J. Wei, "Intelligent soft computing-based security control for energy management architecture of hybrid emergency power system for more-electric aircrafts," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 4, pp. 806–816, Aug. 2018.

[28] J. W. Bennett, G. J. Atkinson, B. C. Mecrow, and D. J. Atkinson, "Fault-tolerant design considerations and control strategies for aerospace drives," *IEEE Trans. Ind. Electron.*, vol. 59, no. 5, pp. 2049–2058, May 2012.

[29] B. Nystrom, L. Austrin, N. Ankarback, and E. Nilsson, "Fault tree analysis of an aircraft electric power supply system to electrical actuators," in *Proc. Int. Conf. Probabilistic Methods Appl. Power Syst.*, Jun. 2006, pp. 1–7.

[30] Z. Huang and V. Dinavahi, "An efficient hierarchical zonal method for large-scale circuit simulation and its real-time application on more electric aircraft microgrid," *IEEE Trans. Ind. Electron.*, vol. 66, no. 7, pp. 5778–5786, Jul. 2019.

[31] Y. Zhang, R. Su, C. Wen, M. Y. Lee, and C. Gajanayake, "Distributed power allocation and scheduling for electrical power system in more electric aircraft," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 102–107.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] F. A. Gers and J. Schmidhuber, "Recurrent nets that time and count," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Netw. (IJCNN) Neural Comput., New Challenges Perspectives New Millennium*, vol. 3, Jul. 2000, pp. 189–194.

[34] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: https://arxiv.org/abs/1406.1078

[35] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," 2016, *arXiv:1605.08695*. [Online]. Available: https://arxiv.org/abs/1605.08695

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2015, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[37] Xilinx. (2018). *VCU118 Evaluation Board User Guide*. [Online]. Available: https://www.xilinx.com/support/documentation/boards_and_kits/vcu118/ug1224-vcu118-eval-bd.pdf

[38] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[39] Z. Wang and S. Miller. (2018). *Electric Aircraft Model in Simscape*. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/64991

QIN LIU (S'19) received the B.Eng. degree in measurement and control technology and instrument from Beijing Jiaotong University, Beijing, China, in 2013. She is currently pursuing the Ph.D. degree with the School of Mechanical, Electronic, and Control Engineering, Beijing Jiaotong University and being a Visiting Ph.D. Student in electrical and computer engineering, University of Alberta, Edmonton, Alberta, Canada. Her research interests include real-time simulation and control of power systems, hardware acceleration, and system-on-chip.

TIAN LIANG (S'16) received the B.Eng. degree in electrical engineering from Nanjing Normal University, Nanjing, Jiangsu, China, in 2011, and the M.Eng. degree in biomedical engineering from Tsinghua University, Beijing, China, in 2014. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Alberta, Edmonton, Alberta, Canada. His research interests include real-time emulation of power systems, power electronics, and system-on-chip.

ZHEN HUANG (S'18) received the B.Eng. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2012, and the M.Eng. degree in electrical engineering from Tsinghua University, Beijing, China, in 2015. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Canada. He also has two years work experience in power electronic industry. His research interests include high-performance simulation and control of power electronics.

VENKATA DINAVAHI (SM'08) received the Ph.D. degree from the University of Toronto, Canada, in 2000. He is currently a Professor in electrical and computer engineering with the University of Alberta, Edmonton, Alberta, Canada. His research interests include real-time simulation of power systems and power electronic systems, large-scale system simulation, and parallel and distributed computing.