BP Neural Network for the Signal Recognition of Micro-Energy Devices

Chuting. Zhang, Haopeng. Wang, Huaiwang. Zeng, Guo. Chen, Wei. Kang, Bin. Zhang

1United Microelectronics Center, Shapingba, Chongqing 400030, China
bin.zhang@cumec.cn

Abstract. The recognition of electrical energy generated by different micro-energy devices is a key step for effective energy management in the integrated microsystems. In this paper, we present an improved Backpropagation Neural Network (BPNN) model in conjunction with principle component analysis (PCA) and Levenberg-Marquadt (LM) algorithm. The open-circuit voltage curves containing the noise characteristics from three different types of micro-energy device including radio frequency energy harvester (RFEH), solar cell (SOR), vibration energy harvester (VEH) are used as input vector for classification model training and validation. The classifier of our model achieves a recognition accuracy of 100%, 91.8%, 83.1% for RFEH, SOR and VEH, respectively. These results indicate that our model is valid for energy recognition for these micro-energy devices and may further apply for hybrid energy management in smart systems such as wireless sensor networks (WSN).

1. Introduction

The recent advancements in sensor, computation and communication technologies in conjunction with the need for continuously monitoring the physical phenomena lead to the development of wireless sensor networks (WSN) and also the revolution of power supply [1]. Generally, the sensor node of the WSN may occupy in a complex or harsh environment thus is challenged for battery replacement. While energy harvesting from the external environment may allow self-sustainable WSN, the external source, such as solar energy, may not always be available. Therefore, the system should be able to harvest the energy from multi-sources, such as solar, RF energy and vibrational energy, to ensure a reliable power supply for the system. However, the energy harvested from different energy sources have different characteristics (e.g. can be either in voltage or charge forms) and different actions of the system have different energy requirement thus an intelligent hybrid energy management scheme are needed to unitize energy harvested from multi-sources and deliver optimized energy output regarding the requirement of different system actions. The first step of the intelligent energy management is to recognize the input energy source which allows to select corresponding management strategy. Therefore, one of the researches is focus to develop a reliable method for recognition of electrical energy generated by different micro-energy devices for hybrid energy supply of WSN.

Neural Networks (NN) are a set of algorithms, that are designed to realize the pattern recognition. They interpret sensory data through a kind of machine perception, labeling or clustering raw input and the patterns they recognize are numerical. Comparing with the traditional machine learning algorithm, NN algorithm has better self-learning, self-adaptive and fault-tolerant ability [2]. For the example of nonlinear separable, NN has better classification ability than that of traditional machine learning algorithm such as Support Vector Machine, Decision Tree. Among various NN approaches, the BPNN
with multilayer feed-forward neural network trained by the error is the most popular training algorithm occupying better fault-tolerant ability. It can provide effective solutions for various applications, especially for signal classification and recognition. Li et al. presented a classification technique to classify cardiology or heartbeat sound signals by BPNN and logistic regression [3]. Cerna-Vázquez et al. [4] used BPNN for PM10 contamination data prediction. Guo et al. [5] presented a BP neural network and wavelet packet time-frequency entropy method based on Levenberg-Marquadt algorithm to analyze the circuit breaker fault. Here we showed an improved BPNN model to recognize the open-circuit voltage of different micro-energy devices. Our classification results based on experiment data of 23000 entries indicate that the improved BP network model has an accuracy > 83% in the recognition of electrical energy which generated by different micro-energy devices.

The paper is organized as follows: In section 2, we propose a BP model with LM algorithm. In section 3, we evaluate the effectiveness of this model with experiment data. In section 4, we give the conclusion.

2. The improvement and training process of BPNN

2.1. The BPNN model

BPNN is well-known multi-layer feedforward neural network that trains some given feed-forward NN for a labeled classification. At the beginning of the training, the labeled data input to each node of the input layer, then it arrives the output layer through the hidden layer and the output is calculated. After that, the comparison done between the desired output and the calculated output with the error value determined by the error function. Usually, Mean Square Error Function is employed as the error function. Later, the BPNN does the backward propagation to adjust the connection weight upon the error value. As the weight is updated with a small delta step of time several iterations are required for the network to converge the learning. After each iteration, the gradient descent force updates the weights to minimize the global loss function. The iterations are continuously executing, until the error value converges into preset range and hereafter the model is ready to make predictions for unknown input data [6-7].

In this work, we propose a three-layers BPNN with the bias node to recognize the open-circuit voltages of different micro-energy devices. The topology of model is shown in Fig.1, and it contains an input layer $n$, an output layer $k$ and a single hidden layer $m$. As the zero offset issue is always arising from the linear calculation ($y = ax + b$, where $x$ is the weight value), during the model training, we add a bias node to the input of hidden layer and output layer to solve the offset problem. Moreover, we use the principal component analysis (PCA) to reduce the dimension of input data by projecting each data point onto only the first few principal components to obtain lower-dimensional data while preserving as much of the data's variation as possible. Further, we can effectively avoid a large number of collinear sets to reduce feature redundancy and speed up the training process [8-9].

![Figure 1](image-url)
The training process of the model is as follows. First, initial weights are applied to all the neurons. For the forward propagation, the inputs from a training set are passed through the neural network and an output is calculated. The equations of forward propagation are showed in follows:

\[ O_m = f(\text{net}_m) = \frac{1}{1 + e^{-\text{net}_m}} \]  
(1)

\[ \text{net}_m = \sum_m W_m o_n + \theta_{mn} \]  
(2)

\[ O_k = f(\text{net}_k) = \frac{1}{1 + e^{-\text{net}_k}} \]  
(3)

\[ \text{net}_k = \sum_k W_{mk} o_m + \theta_{mk} \]  
(4)

Where \(O_m\) and \(O_n\) are the output of the hidden layer and the output layer, respectively, \(W_{mn}\) and \(W_{mk}\) are the connected weight in the model, \(\theta_{mn}\) and \(\theta_{mk}\) are the bias among layers. Especially, \(k\) represents the output layer.

After the forward propagation, we determined the error between the desired output and the calculated output using Mean Square Error Function as shown in equation (5):

\[ \text{error} = \frac{1}{2} (\text{output}_{\text{desired}} - \text{output}_{\text{calculated}})^2 \]  
(5)

Then, this error turns to the backward propagation with the updated connection weights, in order to minimize the error. This training process is measured by equations as follows:

\[ w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t+1) \]  
(6)

\[ \Delta w_{ij}(t+1) = \eta \delta_j o_j + \alpha \Delta w_{ij}(t) \]  
(7)

\[ \delta_k = o_k(1 - o_k)(d_k - o_k) \]  
(8)

\[ \delta_m = o_m(1 - o_m) \sum_k \delta_m w_{km} \]  
(9)

Where \(w_{ij}\) (\(i = k, m, j = m, n\)) and \(\Delta w_{ij}\) are the connected weight and weight adjustment, respectively; \(\eta\) and \(\alpha\) are static parameter to adjust the velocity of weight update. \(\delta_i\) represents the error of the output layer, \(o_i\) is the calculated output of the output layer, \(d_i\) is the desired output of the output layer. Especially, \(\delta_m\) is the error of the hidden layer which is measured by equation 9, \(o_m\) is the calculated output of the hidden layer.

After the amount of iterations, weights are achieved at the optimal values according to the results of the backpropagation algorithm and training stops.

2.2. Levenberg–Marquadt Algorithm

The Levenberg-Marquardt algorithm combines two minimization methods: the gradient descent method and the Gauss-Newton method. In the gradient descent method, the sum of the squared errors is reduced by updating the parameters in the steepest-descent direction. In the Gauss-Newton method, the sum of the squared errors is reduced by assuming the least squares function is locally quadratic, and finding the minimum of the quadratic. The Levenberg-Marquardt method acts more like a gradient-descent method when the parameters are far from their optimal value, and acts more like the Gauss-Newton method when the parameters are close to their optimal value [10]. In order to avoid the our improved BPNN model falling into the local optimal solution, in the backward propagation, we use LM algorithm as optimization strategy to optimize the updated weight and threshold of model by finding the extremum value of the error function. The updated weight and threshold is optimized by following equations:
Weight optimization:

\[ w(k + 1) = w(k) - [J^T J + \mu I]^{-1} J^T e \]  

Threshold optimization:

\[ b(k + 1) = b(k) - [J^T J + \mu I]^{-1} J^T e \]  

Where \( e \) represents the error function, \( J \) is Jacobian Matrix, \( \mu \) is damping coefficient, \( I \) is identity matrix.

2.3. Model Training

The training process of the model is shown in Fig.2. We take the open-circuit voltages of three types of micro energy devices including RFEH, SOR, VEH as inputs. We label the RFEH voltage as \([1,0,0]\), the SOR voltage as \([0,1,0]\) and the VEH voltage as \([0,0,1]\). Data preprocessing is carried out to extract features and filter out redundant data. In this experiment, we set the momentum rate \( M = 0.1 \), the initial learning rate \( \eta_1 = 0.01 \) when the iteration times \( N = 1 \), and adopt the adaptive learning strategy to update the learning rate. After each iteration, we judge whether the error between the desired weight and the calculated weight becomes smaller. If so, the learning rate \( \eta_{N+1} \) is updated to \( \eta_N \times \text{error} \). Otherwise, the learning rate \( \eta_{N+1} \) remain unchanged as \( \eta_N \). Next, we compare the calculated output with the desired output. If the absolute value of error \( \varepsilon \) is out of the pre-set convergence error range (0-0.001), then the error will input to the backward propagation and will be optimized by the Levenberg-Marquardt algorithm. Otherwise, the training is completed and all parameters are saved, we label the micro energy devices as follows: for the output \([y_1, y_2, y_3]\), if \( y_1 > y_2 \) and \( y_1 > y_3 \), the device is marked as RFEH \([1,0,0]\); If \( y_2 > y_1 \) and \( y_2 > y_3 \), it’s recognized as SOR \([0,1,0]\); If \( y_3 > y_1 \) and \( y_3 > y_2 \), it’s labeled as VEH \([0,0,1]\).

![Figure.2 The training process of the BPNN model](image-url)
3. Experiments

3.1. Data Collection

The input vector for model training and validating contains a sequence of open-circuit voltage values collecting from a high-speed sampling system with sampling time of 0.0.512s. The system is showed in Fig.3(a) and is composed of an analog-to-digital converter (ADC), a microprocessor ad a wireless data transmitter. The model of ADC module is ADS8867 from Texas Instruments with a maximum sampling rate of 100kSa/s and 16-bit precision. In this work, the ADC module works at sampling rate of 10kSa/s to acquire voltage sequence continuously when the micro-energy device is accessed to input connector. After the sampling of one sequence with 14 bits is finished, the data is packaged as a data frame and sent to the computer where all sequences are assembled for further training or validating. As a result, the noise characteristics of different generators are also contained in the data sequence. These characteristics are also the key features in electrical energy generated by three types of micro-energy devices.

To obtain the actual open-circuit voltage sequences, SOR, VEH and RFEH are employed in this paper. The SOR is made of polycrystalline silicon and is illuminated by a high-pressure xenon lamp to simulate the circumstance of sunlight shown in Fig.3(b). The VEH device is assembled by multi-layer PZT-5H piezoelectric film and produce power when stimulated by vibration. The testing platform for VEH includes vibration table and accelerometer as shown in Fig.3(c). During the measurement, the VEH is mounted on the vibration table which supplies the vibration excitation and the z-direction vibration intensity is monitored by an accelerometer. The RFEH could generate power by collecting electromagnetic wave from its surrounding. The receiving antenna is ultra-wideband designed and is sensitive to the frequency range of 1.137 GHz - 6.167 GHz, covering common RF communication bands such as Wi-Fi, CDMA, GPS, etc. In this work, we use a pair of 2.4 GHz wireless network bridges to provide RF radiant for the RFEH antenna and the testing platform is shown in Fig.3(d). In particular, the output voltage of VEH and RFEH is alternating signal. Thus, the output of both devices is rectified before accessing to the sampling system. The voltage levels of all the power-generating devices are set to same value by adjusting excitation condition of the testing platforms to simulate the extreme condition that classifier works.

Figure.3 Data collector platform for micro-power generators: (a) data collecting system; (b) platform of solar cell; (c) platform of vibration energy harvester; (d) platform of radio-frequency energy harvester
3.2. Experiment Results
The Fig.4 shows the error value between the desired output and the calculated output for three types of micro-energy devices classification is decreased with the increasing iteration times in the training process. We used 19400 samples as training set and another 3600 samples as test set to evaluate the model performance. As showed in the Fig.4, this error is coverage to 0.001 preset convergence error range since the number of iteration $\geq 800$. Hence, the experiment results demonstrate an excellent recognition ability and strong generalization of our proposed model.

Some recognition results for these types of micro energy devices are shown in the Table 1. To verify the effectiveness of model, we take 3600 samples as test set input to the model for micro-energy signal classification. In this paper, the output is classified by "maximum criterion". If $y_1$ is maximum in output $[y_1, y_2, y_3]$, the device is marked as RFEH; if $y_2$ is maximum in the output $[y_1, y_2, y_3]$, it’s recognized as SOR; else, it’s labeled as VEH. Table 1 shows a random sampling of recognition results of the micro energy devices.

| NO | Output | Recognition Result |
|----|--------|--------------------|
| 1  | 0.002424, 0.186377, 0.807617 | VEH[0,0,1] |
| 2  | 0.000519, 0.835913, 0.168556 | SOR[0,1,0] |
| 3  | 3.857655, 0.998368, 0.002307 | RFEH[1,0,0] |
| 4  | 0.001502, 0.374396, 0.621276 | VEH[0,0,1] |
| 5  | 0.000555, 0.815970, 0.188577 | SOR[0,1,0] |
| 6  | 4.100825, 0.998327, 0.002700 | RFEH[1,0,0] |

The table 2 shows the recognition accuracy of the micro energy devices. It can be seen from the table that the number of training sets and test sets are set to 6800 and 1200 respectively, in which the accuracy of RFEH recognition achieves 100%, the accuracy of SOR recognition reaches 91.8%, and the accuracy of VEH recognition gets 83.1%.

| Micro-Energy Device | Training Set | Test Set | Recognition Accuracy |
|---------------------|--------------|----------|----------------------|
| RFEH                | 6800         | 1200     | 100%                 |
| SOR                 | 6800         | 1200     | 91.8%                |
| ZHEN                | 6800         | 1200     | 83.1%                |
4. Conclusion

Based on the results and discussions presented above, the conclusions are obtained as below:

It is shown that our proposed model can recognize the input signal containing the noise characteristics from three different types of micro-energy devices, effectivity with accuracy > 83%.

(2) This method is also one of the key factors to evaluate the performance of different energy generation devices when intend to integrate them into one system, and the classification results can be used as inputs for hybrid energy management.

Acknowledgments

This work has obtained the patent certificate of invention (CN 110728360 B).

References

[1] N. R. Kumar, V. R. Nag. An overview of wireless sensor network with its applications and various challenges[J]. International Journal of Applied Engineering Research, 2012, 7(11):1970-1973.

[2] Zhang X, Wu G et al. Embedded feature-selection support vector machine for driving pattern recognition[J]. Journal of the Franklin Institute, 2015, 352(2):669-685.

[3] Li Lina, Wang Xinpei et al. Classification of heart sound signals with BP neural network and logistic regression[C]. Chinese Automation Congress, 2017, 7380-7383.

[4] Cerna-Vázquez, Daniel and Carlos et al. Backpropagation neural network for the prediction of PM10 Contamination Data[J]. Research in Computing Science, 2017, 133:41-50.

[5] Guo, Zijian and Liu, Mingliang et al. Circuit breaker fault analysis based on wavelet packet time-frequency entropy and LM algorithm to optimize BP neural network[C]. Chinese Automation Congress, 2018, 5729-5734.

[6] V. Gudise and G. Venayagamoorthy. Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks[C]. Swarm Intelligence Symposium, 2003, 110-117.

[7] Huang H X, Li J C et al. A proposed iteration optimization approach integrating backpropagation neural network with genetic algorithm[J]. Expert Systems With Applications, 2015, 42(1):146-155.

[8] Groth D, Hartmann S et al. Principal Components Analysis[J]. Methods Mol Biol, 2013, 930:527-547.

[9] Partridge M, Jabri M. Robust principal component analysis[C]. Signal Processing Society Workshop, 2000:289-298.

[10] Rubio, Jose de Jesus. Stability analysis of the modified Levenberg-Marquardt algorithm for the artificial neural network training[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 99:1-15.