The Scientific Temper (2023) Vol. 14 (2): 366-370
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.2.19
https://scientifictemper.com/

RESEARCH ARTICLE

Analog Circuits Based Fault Diagnosis using ANN and SVM

Archana Dhamotharan*, Kanthalakshmi Srinivasan

Abstract

In this study, we provide a technique for identifying analog errors using a neural network and an SVM (SVM). The study's major objective is to produce a trustworthy diagnostic based on a technique that reduces testing durations by resolving the problem of component tolerances.The suggested strategy uses an artificial neural network and a backward propagation mechanism. The impact of methods like Principal Component Analysis on feature extraction is discussed in this work. The simulation results show that the technique is effective and efficient for fault identification in tolerant mixed-signal circuits.

Keywords: Artificial Neural Networks, Kernel Principal Component Analysis, Support Vector Machine.

Introduction

Over a few decades, electronic technology advancements increased and became a significant research issue. Reducing the size of the devices in terms of reducing the size of the circuits becomes laborious when considering the idea that "little things are beautiful." Analog, digital, and mixed-signal circuits are the three types of electronic circuits. Research on analog defect diagnosis has been ongoing, with enough work done at the circuit, chip, and system levels (Liu, 2012; Sun, 1989). Automated fault diagnosis and classification procedures can increase effectiveness and reliability by quickly identifying and isolating system flaws.

Pattern recognition, signal processing, image analysis, and other applications have all made use of artificial neural networks (ANNs) (Spina & Upadhyaya, 1992; He *et al.*, 1998). The advantages of ANNs include online computing, robust adaptive training, parallel storage, and large-scale parallel processing. They work wonderfully for fault detection in sensitive analogue circuits (He *et al.*, 1998).

The Support Vector Machine is an effective method when dealing with non-linear classification and function estimation problems. The non-linear classification can be employed to distinguish between the response vectors of healthy and unhealthy circuits. Because prior knowledge of the responses from the Circuit under Test is not required, less time and memory are used. Nevertheless, it entails a quadratic programming assignment, which raises the computing difficulty.

The data set needed for testing in this study is gathered using Pspice software from the Monte-Carlo analysis. Kernel Principal Component Analysis (KPCA) is applied as a preprocessing step to the assembled dataset to generate the most useful features for training and verifying neural networks.These ideal features are automatically normalized before being fed to the network, which improves the features in the data set and increases the training efficiency of the neural network and SVM.

This paper's contents are organized as follows. Section 2 of this article provides a concise introduction to the KPCA. Section III introduces and discusses artificial neural networks (ANNs), specifically back propagation neural networks (BPNN).Section IV discusses the algorithm used in support vector machines. The circuit examples are discussed in great depth in Section V. Section VI provides the findings from our simulations and studies of the illustrated circuits. In Section VI, a conclusion was made.

KPCA

The principal component analysis (PCA) technique is useful for de-noising high-dimensional data sets (Bishop, 1995). On the other hand, PCA is a linear technique, hence it can't reveal any non-linear patterns in the data. "As a result, non-linear modifications have been proposed, such as kernel principal component analysis (KPCA), which use kernel theory to compute the principal components of a data

Department of EEE, PSG College of Technology, Coimbatore, India

***Corresponding Author:** Archana Dhamotharan, Department of EEE, PSG College of Technology, Coimbatore, India, E-Mail: ece04kpr@gmail.com

How to cite this article: Dhamotharan, A., Srinivasan, K. (2023). Analog Circuits Based Fault Diagnosis using ann and svm. The Scientific Temper, **14**(2):366-370.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.2.19

Source of support: Nil

Conflict of interest: None.

Received: 15/03/2023 **Accepted:** 29/04/2023 **Published**: 30/06/2023

set after its non-linear translation into a high-dimensional feature space. Since KPCA involves the implicit translation of sample data from an input space to a high-dimensional feature space, it is most successfully performed by means of kernel techniques and may be viewed as an eigenvalue issue of its kernel matrix (Scholkopf *et al.*, 1998; Scholkopf & Smola, 2002). Here is how KPCA is currently understood:

Given a setof'l ' training samples and a non-linear mapping
$\Phi : R^d \to F, x \to \Phi(x)$, For construct a generic covariance matrix in the feature space F, we may do the following:

$\hat{R} = \frac{1}{l}\sum_{j=1}^{l} (\Phi(x_j) - m^{\Phi})(\Phi(x_j) - m^{\Phi})^T$ (1)Where, $m = \frac{1}{l}\sum_{j=1}^{l} \Phi(x_j)$, Where, the assumption is $\frac{1}{l}\sum_{j=1}^{l} \Phi(x_j) = 0$,then the modified covariance matrix R has thefollowing form

$$R = \frac{1}{l}\sum_{j=1}^{l} \Phi(x_j)\,\Phi(x_j)^T \qquad (2)$$

We find that a linear expression for each eigenvector of R, v may be written as

$$v = \sum_{i=1}^{l} \alpha_i\ \Phi(x_i) \qquad (3)$$

Where $\alpha_i = 1, \ldots, l$are the expansion coefficients.
$LetQ = [\Phi(x_i), \ldots \Phi(x_l)]$, $1_l = (1/l)_{l\,x\,l}$ and ,$\tilde{K} = Q^T Q$, the matrixK is defined as

$$K = \tilde{K} - 1_l\tilde{K} - \tilde{K}1_l + 1_l\tilde{K}1_l \qquad (4)$$

Where $\tilde{K}_{ij} = \Phi(x_j)^T\Phi(x_j) = k(x_i, x_j)$ and k (· , ·) is the kernel function corresponding to a given non-linear mapping Φ. Calculating the eigen vectors $\gamma_1 \ldots \gamma_l$ and eigen values $\lambda_1 \ldots \lambda_l$ of K, the orthonormal eigen vectors obtained are $v_1 \ldots v_l$ corresponding to the m largest eigen values $\lambda_1 \ldots \lambda_p$ of R as follows:

$$v_j = \frac{1}{\sqrt{\lambda_j}} Q\gamma_j \ , \ j = 1, \ldots, p \qquad (5)$$

Principal components are extracted using KPCA, which involves calculating projections on eigen vectors,v^k.=1,...,p in feature space F. For this reason, we'll assume that there's a test point x with an associated image Φ,x. in F, then

$$y_j = v_j^T\Phi(x) = \frac{1}{\sqrt{\lambda_j}}\gamma_j^T Q^T\Phi(x) = \frac{1}{\sqrt{\lambda_j}}\gamma_j^T[k(x_1, x), \ldots k(x_l, x)],$$

$j = 1, \ldots, p$ (6)

Where y^j is the jth extracted principal component.

Artificial Neural Networks

Data mining, chemical processes, digital circuits, control systems, and other areas have effectively used ANNs because they offer adaptive pattern classification mechanisms. In recent years, ANNs have attracted a great deal of attention from numerous scientific disciplines (Yuan *et al.*, 2006; Yadav & Swetapadma, 2014; Aminian *et al.*, 2002) . Regardless of conditions, they are still capable of performing robust categorization. It can even be overstated how critical it is to select the appropriate ANN architecture for a production deployment. One of the most popular forms of artificial neural networks is the backward propagation neural network (BPNN). When used for fault diagnosis and testing, neural networks provide important advantages such as quick online diagnosis after network training. In comparison to conventional classifiers, ANN classifiers also need less defect characteristics. Additionally, neural networks are capable of categorizing faults at various levels of hierarchy.

Based on how they are trained, artificial neural networks (ANNs) are divided into two types: supervised and unsupervised. In the case of the BPNN, we have a supervised network. Two or three levels of linking weights are used in conventional BPNNs. The typical two-layer network is shown in Figure 1. Every hidden layer node is linked to some input node; likewise, every hidden layer node is connected to some output node. The BPNN will create a network topology with no dead ends. Patterns of input data must be learned in order to be communicated from the input nodes to the output nodes. When the outputs are compared, a discrepancy arises between the actual and predicted values. Following that, the weights are adjusted so that the error is within a predetermined tolerance. Thus, this is a supervised learning method because the target values are known in advance.

In Figure 1, input $x = [x_0, x_1, \ldots . x_{m-1}]$ and output $y = [y_0, y_1, \ldots . y_{n-1}]$. one common layer definition includes both the hidden layer and the output layer. The relation of output $O_i^{(1)}$ and input $O_j^{(l-1)}$ of layer j is defined as:

$$O_i^{(l)} = f_s[I_i^{(l)}] \qquad (7)$$

$$I_i^{(l)} = \sum W_{ij}^{l}\, O_i^{(l-1)} \qquad (8)$$

Equation (1) can be transformed into

F_s (I)=1 /(1 +exp (-I)) (3) (9)

We assume that the initial weight values are uniformly distributed between -0.5 and 0.5, and we define the weight between the jth neuron of the (k-1)th layer and the ith neuron of the kth layer as $W_{i,j,k}$. In order to get the equation for weight adaptation, we have

$$W_{i,j,k}(t_n) = \frac{\left(W_{i,j,k}(t_{n-1}) - \alpha\, E\,(t_n)\right)}{W_{i,j,k}(t_{n-1}) + \eta.\Delta W_{i,j,k}(t_{n-1})} \qquad (10)$$

The following are some of the many benefits of using BP neural networks for fault detection with tolerance:

The BP classifier is effective even when there is background noise present, allowing for accurate class recognition.

A BPNN's recall capacity and ability to put together the full fault characteristics from partial ones are two of its most useful capabilities.

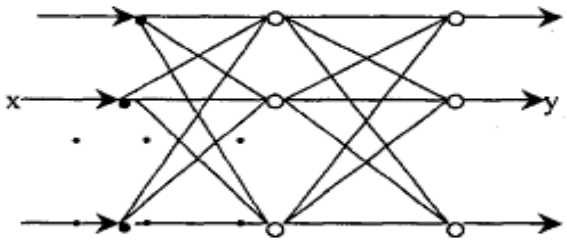

Figure 1: BPNN architecture

Support Vector Machine

Support Vector Machines (SVMs), an innovative kind of machine learning, are responsible for this classification job. Finding a hyperplane that exactly separates the two d-dimensional sets of data is the major goal. When example data is not linearly separable, SVMs map it onto a higher-dimensional space where it is separable using the concept of a kernel-induced feature space". Overfitting and computing issues would likely arise if casting were done in such a space.

The idea that dealing with higher-dimensional space indirectly is the main insight behind SVMs. SVMs choose the best separating hyperplane to divide data points into distinct groups in classification issues. In this context, the term "optimal" refers to the separating hyperplane's superior statistical learning theory generalization capability for the unobserved data points. By resolving the fundamental optimization problem, this ideal separation hyperplane is produced. A non-linear kernel map's implicit definition of a highly non-linear separation hyperplane, which SVMs create, allows them to distinguish between different complex data patterns. This feature makes SVMs useful to a wide range of significant real-world issues, including breast cancer diagnosis and prognosis, face detection, DNA microarray analysis, bankruptcy prognosis, and bankruptcy prognosis.

Support vector machine-based intelligent defect diagnosis methods can be divided into two phases:

Training stage

- The training sample set should be sorted from most to least frequent according to class size..
- Make a training dataset for SVMs; if a k-support vector machine is being taught, the dataset is $X = \{(x_i, y_i) \mid y_i \geq m, y_i \in (1, k), m \in (1, k-1)\}$

Where k is the total number of class.

- Having a collection of support vectors in hand is useful before extracting them.
- To conduct the training of all k-1 SVM, start over from the start.

ii. Fault identification phase

- During the learning phase, the support vector machine (SVM) is loaded with the training sample dataset set (x, y), the Lagrangian coefficient I, and the SVM itself.
- Predict the category of unidentified samples using decision-making methods.

Experimental Validation

Example Circuits and Faults

The three-stage bandpass and biquad filters are examined here as case studies. Applying inputs and then sampling the outputs of every CUT to obtain the correct frequency and temporal responses for filters are carried out in accordance with the broad guidelines indicated by Yadav & Swetapadma (2014). In this circumstance, both single and double faults are taken into consideration. Since double faults are more likely to happen in real-world applications than other multiple errors, the triple and many faults are disregarded. Due to the exceptionally high number of fault configurations and the complexity of the processing, some double faults are also investigated for simplicity purposes and as an example. These examples illustrate how to use the KPCA outlined in earlier sections to extract useful characteristics from CUTs' response outputs.

Example 1: A 3-stage bandpass filter is used as the first CUT by Aminian & Aminian (2002) and Aminian*et al.*, 2002). A 1 kHz fundamental frequency is achieved with the standard values of the components illustrated in Figure 2. Capacitors and resistors are assumed to have 5%, 10%, and 10%, 10% tolerances. Circuit responses in Figure 2 are non-fault because components C1, C2, R2, and R3 all behave normally (NF). Before doing feature selection on the sampled data, KPCA is used for preprocessing.

Case 1: Single faults: A faulty frequency response is produced when one component is more than 50% over or below its nominal value while the remaining components fluctuate within their tolerances. The 12 fault and non-fault classes are obtained by preprocessing these faulty impulse responses. The neural network is trained and tested using these samples.

Case 2: Double faults: Two components have abnormally high or low values, whereas the other four vary normally within their ranges.

Example 2: Biquad Filter: Figure 3 depicts the second CUT under investigation, a biquad high-pass filter with components set to their nominal values (Sun, 1989). Compared to the filter used in Example 1, this one is more complicated.

Case 1: For single faults, the 13 classes are as follows: C1↑, C1↓, C2↑, C2↓, R1↓, R1↓, R2↑, R2↓, R4↑, R4↓, and non-fault

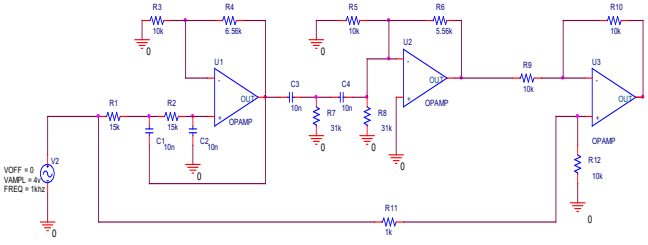

Figure 2: 3-stage bandpass filter

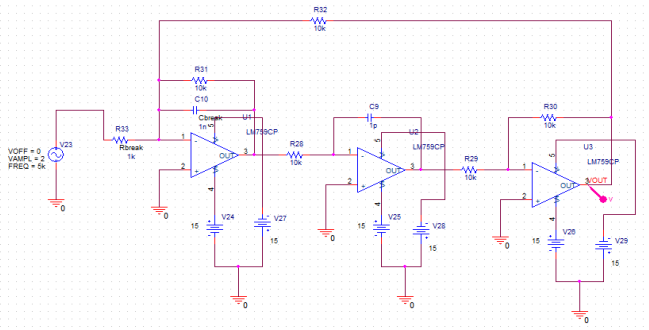

Figure 3: Biquad high-pass filter

class (NF), where and respectively denote values that are 50% higher and lower than the nominal values.

Case 2: Double faults: For the time being, we will define double faults as the simultaneous increase or decrease of two of the eight components, with the remaining two components fluctuating normally within their limits.

Feature Extraction

In this work, relevant CUTs are excited by two different types of sources in order to measure the corresponding frequency and temporal responses at the outputs of related CUTs. The incorrect frequency response is generated when a single fault exists in a circuit, such as the single fault in the two filters in Figs. 2 and 3, while the remaining components vary within their tolerance limits. Faulty frequency responses are produced for identical double faults when the values of two components are either higher or lower than all components, but the other components fluctuate within their tolerances. With the aid of the Pspice software, the dataset from the Monte-Carlo analysis is gathered, and the frequency response values are noted. The massive values from the obtained dataset require additional processing for the neural network to function. As candidate features, the first coefficients of approximation level 1 are chosen. The KPCA method from Section 2 is then used to further process the potential features in order to isolate the reduced (optimal) features that will be utilized to train the neural network. Assume that Xi is the input feature, Ti is the matching output vector, and the neural network has m output neurons. The 1-of-m coding system determines the output values. When the input feature Xi corresponds to fault class l, where m is the number of fault classes, the resulting vector Ti will have components. In these instances, the neural network always assigns input features to the defect class with the highest likelihood. In a neural network, the number of input nodes and output neurons are set in advance, depending on the nature of the job at hand. An empirical formula is used to determine how many hidden neurons are present. In order to calculate the number of hidden neurons, h, we need to assume that there are n input nodes and m output nodes.

$$\sqrt{m+n}+1 \le h \le \sqrt{m+n}+10 \quad (13)$$ (Deliyannis *et al.*, 1999)

Tanh and linear functions are always chosen as the hidden and output layers' respective activation functions in neural networks.

Simulation Results

Research has been into leveraging the techniques discussed in the preceding sections to diagnose problems in the analogue circuits shown in Figures 2 and 3. Each fault class has a training set of 10 circuits and a testing set of 10 circuit. To be more precise, the training set for the suggested neural network is formed by selecting 10 training samples at random from each class. The trained neural network's adaptability capacity is tested, and the performance of the classifier is assessed, using the remaining 10 testing samples. Only 97% of classifications are correctly performed by the neural network with 4 inputs, 6 hidden neurons, and 8 output neurons, resulting in 192 configurable parameters. On the other hand, our suggested neural network achieves 99% classification accuracy. In comparison to the study by Aminian *et al.* (2002), our suggested approach for diagnosing faults can significantly simplify the network structure, reduce the computational load significantly, and effectively increase fault detection performance. Furthermore, our suggested diagnostic system achieves approximately the same diagnosis performance as the approach for feature extraction in the frequency domain and uses three fixed features as inputs to neural networks (Yuan *et al.*, 2009).

Table 1 and Table 2 provide a concise comparison of the neural network and support vector machine (SVM) structures.

Table 1: Comparison between ann and svm for bandpass circuit

Parameters	*Ann*	*Svm*
Time for Training	1.5	2
Time for Testing	2.094824	1.123662
Accuracy (%)	99.5556	97.7778
Mean Square Error	0.6761	0.0259
Neural network used	Back-propagation	-
Function used	-	Linear Kernel

Table 2: Comparison between Ann and Svm for Biquad Filter

Parameters	*Ann*	*Svm*
Time for Training	1.5	2
Time for Testing	2.14632	1.4580
Accuracy (%)	98.4511	96.5555
Mean Square Error	0.5908	0.0217
Neural network used	Back-propagation	-
Function used	-	Linear Kernel

Conclusion

This work uses KPCA and normalisation as frequency and time domain preprocessors to systematically identify flaws in analogue circuits.. The study shows that the suggested preprocessing techniques significantly affect analog defect diagnostics in the choice of the ideal number of pertinent features. This results in neural network topologies that are of manageable size and have good diagnostic precision for different fault classes. The results of the neural network and the SVM are compared. According to the analysis and findings in this study, the suggested diagnostic system can obtain an acceptable classification performance if there is no substantial overlapping in these circuits. Additionally, this fault diagnosis system can accurately diagnose single and multiple faults.

References

Aminian, F., Aminian, M., & Collins, H. W. (2002). Analog fault diagnosis of actual circuits using neural networks. *IEEE Transactions on Instrumentation and Measurement*, *51*(3), 544-550.

Aminian, M., & Aminian, F. (2000). Neural-network based analog-circuit fault diagnosis using wavelet transform as preprocessor. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, *47*(2), 151-156.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Deliyannis, T., Sun, Y., & Fidler, J. K. (2019). *Continuous-time active filter design*. CRC press.\

He, Y., Luo, X., & Qiu, G. (1998). A neural-based nonlinear L 1-norm optimization algorithm for diagnosis of networks. *Journal of Electronics (China)*, *15*(4), 365-371.

Liu, R. W. (2012). *Testing and diagnosis of analog circuits and systems*. Springer Science & Business Media.

Bernhard, S., Sebastian, M., Chris, J., Burges, P. K., Müller, K. R., & Rätsch, G. (1999). Input space versus feature space in kernel-based methods. *IEEE T Neural Networ*, *10*, 1000-1017.

Schölkopf, B., & Smola, A. (2002). Learning with kernels. mit press, cambridge, ma. *Learning with kernels. MIT Press, Cambridge, MA*.

Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, *10*(5), 1299-1319.

Greene, D., Doyle, D., & Cunningham, P. (2010, August). Tracking the evolution of communities in dynamic social networks. In *2010 international conference on advances in social networks analysis and mining* (pp. 176-183). IEEE.

Yadav, A., & Swetapadma, A. (2014). Classification of readily biodegradable molecules using principal component analysis and artificial neural network. *ARTIFICIAL INTELLIGENCE*, *1*(2).

He, Y., Tan, Y., & Sun, Y. (2004, November). Fault diagnosis of analog circuits based on wavelet packets. In *2004 IEEE Region 10 Conference TENCON 2004.* (pp. 267-270). IEEE.

Yuan, L., He, Y., Huang, J., & Sun, Y. (2009). A new neural-network-based fault diagnosis approach for analog circuits by using kurtosis and entropy as a preprocessor. *IEEE Transactions on Instrumentation and Measurement*, *59*(3), 586-595.