C. R. Mecanique 342 (2014) 229–233

Contents lists available at ScienceDirect

Comptes Rendus Mecanique

www.sciencedirect.com

# The application of Shuffled Frog Leaping Algorithm to Wavelet Neural Networks for acoustic emission source location

Xinmin Cheng [a,*], Xiaodan Zhang [b], Li Zhao [b], Aideng Deng [b], Yongqiang Bao [b], Yong Liu [c], Yunliang Jiang [a]

[a] School of Information and Engineering, Huzhou Teachers College, Huzhou, Zhejiang, 313000, China
[b] School of Information Science and Engineering, Southeast University, Nanjing, Jiangsu, 210096, China
[c] Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou, Zhejiang, 310027, China

ARTICLE INFO

*Article history:*
Received 15 April 2013
Accepted 23 December 2013
Available online 27 January 2014

*Keywords:*
Acoustic emission
Location
Wavelet Neural Network
Shuffled Frog Leaping Algorithm

ABSTRACT

When using acoustic emission to locate the friction fault source of rotating machinery, the effects of strong noise and waveform distortion make accurate locating difficult. Applying neural network for acoustic emission source location could be helpful. In the BP Wavelet Neural Network, BP is a local search algorithm, which falls into local minimum easily. The probability of successful search is low. We used Shuffled Frog Leaping Algorithm (SFLA) to optimize the parameters of the Wavelet Neural Network, and the optimized Wavelet Neural Network to locate the source. After having performed the experiments of friction acoustic emission's source location on the rotor friction test machine, the results show that the calculation of SFLA is simple and effective, and that locating is accurate with proper structure of the network and input parameters.



## 1. Introduction

The Time Difference Of Arrival (TDOA) location method is a usual approach to locate the position of a fault using the acoustic emission technology [1–3] in the fault location. The TDOA location method detects the time differences of arrival at different sensors from homologous acoustic emission signals, and calculates the source location in accordance using the space array between the sensors.

When we use the signals measured above the default threshold of the acquisition system to calculate the arrival time, the latter not only depends on the parameters of acoustic emission instrumentations, e.g. the position of transducers, frequency dispersion, attenuation, noise interference, and other factors, but also depends on the experience of the operator. In order to decrease the impact of man-made factors, it is important to design a smart algorithm for source locating [4–6].

In the intelligent algorithms, the Neural Network is a usual and effective method, and it is with the characteristics of self-organizing, self-adaptive, self-learning, and better robustness. With the rational structure of the network, right input samples, and enough training samples, this method can provide the precise activity of the acoustic emission [7–9].

This paper introduces the Wavelet Neural Network module, uses the Shuffled Frog Leaping Algorithm (SFLA) [10–12] instead of the traditional decreasing gradient algorithm, optimizes the parameters of the network, and implements the acoustic emission source location method. The experimental results show that its accuracy and efficiency are much higher than those of traditional positioning methods.

* Corresponding author.
*E-mail address:* chengxinmin@yahoo.cn (X. Cheng).

1631-0721/$ – see front matter 
http://dx.doi.org/10.1016/j.crme.2013.12.006

## 2. Shuffled Frog Leaping Algorithm

In a $d$-dimensional target searching space, $D$ frogs (solutions) are randomly generated to compose initial population. The $i$-th frog represents a potential solution of the problem $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$. Frogs are arranged from good to bad according to their fitness values to divide the whole population into $N$ sub-populations. Among them, the frog ranked 1st is assigned into the 1st sub-population, the one ranked 2nd into the 2nd sub-population, the one ranked $N$th into the $N$th sub-population, the one ranked $N + 1$ into the 1st sub-population, the one ranked $N + 2$nd into the 2nd sub-population, and the sequence continues until all frogs have been assigned.

Every sub-population is used for local area deep searching, that is for each sub-population, the worst individual $X_{\mathrm{w}}$, the best one $X_{\mathrm{b}}$, and the global best one $X_{\mathrm{g}}$ of the sub-population in each iteration are determined first. The update operation is applied only to the current worst individual $X_{\mathrm{w}}$, which is described as:

$$\Omega_i = \mathrm{rand}() * (X_{\mathrm{b}} - X_{\mathrm{w}}) \quad (-\Omega_{\max} \leqslant \Omega_i \leqslant \Omega_{\max}) \tag{1}$$

$$\text{new } X_{\mathrm{w}} = X_{\mathrm{w}} + \Omega_i \tag{2}$$

where rand() represents random number uniformly distributed between 0 and 1, $\Omega_{\max}$ represents the maximum of update steps allowed. If the fitness value of new $X_{\mathrm{w}}$ is better, $X_{\mathrm{w}}$ will be replaced. If it is not improved, $X_{\mathrm{b}}$ in Eq. (1) and Eq. (2) is replaced with $X_{\mathrm{g}}$. Then, the new update strategy is:

$$\Omega_i = \mathrm{rand}() * (X_{\mathrm{g}} - X_{\mathrm{w}}) \quad (-\Omega_{\max} \leqslant \Omega_i \leqslant \Omega_{\max}) \tag{3}$$

$$\text{new } X_{\mathrm{w}} = X_{\mathrm{w}} + \Omega_i \tag{4}$$

If the fitness value of new $X_{\mathrm{w}}$ is still not improved, then a new $X_{\mathrm{w}}$ will be generated randomly. We then repeat this update operation until the update number is satisfied. After the local area deep-searching of all sub-populations has been finished, all frogs in the whole sub-population are mixed and reordered into sub-population and the worst individual in each sub-population is replaced. The fitness values of all individuals are changed as a consequence. Therefore, we may reorder the population according to the new fitness values from the highest value to the lowest one, and the corresponding individual frogs can be assigned into $N$ sub-groups in the same way as the first time. Then, the local area deep-searching are processed until satisfying the number of mixed iterations.

## 3. Wavelet Neural Network

The main idea of Wavelet Neural Networks (WNN) [13–15] consists in using the wavelet function as a neuronal activation function and relating the wavelet to the BP network. Since wavelet transform has a good local time-frequency feature and multi-resolution analysis capability, Wavelet Neural Network performs well for identification and approximation to any functions.

The wavelet transform should satisfy $\psi(t) \in L^2(R)$, where $L^2(R)$ is the square integrable space of real numbers, and it is an energy-limited signal space. $\Psi(\omega)$ denotes the Fourier transform of $\psi(t)$, and it should satisfy:

$$c_\psi = \int_R \frac{|\Psi(\omega)|}{|\omega|} \, \mathrm{d}\omega < \infty \tag{5}$$

where $\psi(t)$ is the basic wavelet or mother wavelet. After stretching and shifting, we can get a wavelet sequence:

$$\psi_{a,b}(t) = |a|^{-1/2} \psi\left(\frac{t-b}{a}\right), \quad a, b \in R; \; a \neq 0 \tag{6}$$

where $a$ is the stretching factor, and $b$ is the shift factor. If the function $f(t) \in L^2(R)$, the wavelet transform of $f(t)$ is defined as:

$$W_f(a,b) = \langle f, \psi_{a,b} \rangle = |a|^{-1/2} \int_R f(t) \psi\left(\frac{t-b}{a}\right) \mathrm{d}t \tag{7}$$

Its inverse transform is defined as:

$$f(t) = \frac{1}{C_\psi} \int_R \frac{1}{a^2} W_f(a,b) \psi\left(\frac{t-b}{a}\right) \mathrm{d}a \, \mathrm{d}b \tag{8}$$

The discrete form of $f(t)$ is:

$$f(t) = \sum_{i=1}^{k} \omega_i \psi\left(\frac{t-b_i}{a_i}\right) \tag{9}$$

where $k$ is the number of wavelets.