Vol. 5(15) Apr. 2015, PP. 2130-2135

Contents list available at IJMEC

International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)

Journal Homepage: www.aeuso.org

PISSN: 2411-6173 - EISSN: 2305-0543

Accuracy Improvement of All-Pole Spectrum Estimation Using Particle Swarm Optimization and Its Comparison with Classic Methods

Zabihollah Ahmadpour [1] and Jahanshah Kabudian [2*]

[1] Lecturer, Computer Eng. Dept., Islamic Azad University, Ayatollah Amoli Branch, Amol, IRAN

[2*] *Corresponding Author*, Assistant Professor, Dept. of Computer Eng. and Information Technology, Kermanshah, IRAN

E-mails: *z.ahmadpour@iauamol.ac.ir* , *kabudian@razi.ac.ir*

Abstract

Spectrum estimation has many applications in digital signal processing. Parametric models for spectrum estimation include AR model, MA model, and ARMA model. Normally, classic methods such as Levinson-Durbin or Burg method are used to calculate parameters of AR model. Clearly, there is a distance (error) between spectrum estimated using these algorithms and the actual signal spectrum. In this paper, Particle Swarm Optimization (PSO) method is used to reduce this error. Results show at least 40% improvement in decreasing error of estimated all-pole spectrum in comparison with classic spectral estimation methods.

Keywords: *Spectrum Estimation, Linear Prediction, Particle Swarm Optimization, Global Optimization, Autoregressive model, Levinson-Durbin method, Burg method.*

1. Introduction

The main goal of all-pole spectrum estimation of a signal is to model the signal spectrum by using some resonators (poles) such that the number of parameters of all-pole model would be much less than the number of original signal samples. The all-pole model is also known as linear prediction model. By reducing the number of system parameters, the obtained model can also be used for the purpose of signal compression. Spectrum estimation methods are classified into two general categories of parametric and nonparametric. Nonparametric methods make use of advantages provided by Fourier Transform. However, frequency resolution of these methods is low for small-length data. In contrast, parametric methods can provide high frequency-resolution as well as appropriate computational efficiency and spectral leakage avoidance [3].

With respect to the limitation of nonparametric methods, parametric methods including AR (Autoregressive), MA (Moving Average), and ARMA (Autoregressive Moving Average) model have received much attention. In the AR model, it is assumed that observed data are produced by the system expressed by the all-pole filter equation (Eq. (1)):

$$a_0 x(n) = -\sum_{k=1}^{P} a_k x(n-k) + b_0 u(n) \tag{1}$$

Where $x(n)$ is output, and $u(n)$, named input, has Gaussian distribution (normal) with zero mean and unit variance and shows properties of white noise. b_0 is the gain coefficient of the system and coefficients a_k and P represent its parameters and order of the AR system, respectively. Model

Article History:

IJMEC DOI: 649123/10134

Received Date: Jan. 04, 2015

Accepted Date: Mar. 24, 2015

Available Online: Apr. 10, 2015

parameters are obtained by solving a set of linear equations with respect to minimizing mean squared errors across all data ([8], [6]).

MA model is expressed by Eq. (2) where b_k and Q represent model parameters and order, respectively, and $u(n)$ is assumed as in the AR model. This filter is called all-zero [8].

$$x(n) = \sum_{k=0}^{Q} b_k u(n-k) \tag{2}$$

ARMA model is obtained by combining AR and MA models and is expressed by Eq. (3). This filter employs poles and zeros of the both above models [6].

$$x(n) = -\sum_{k=1}^{P} a_k x(n-k) + \sum_{k=0}^{Q} b_k u(n-k) \tag{3}$$

There is a distance (error) between the spectrum estimated using this methods and the actual spectrum of signal. A good spectrum estimation method seeks to achieve the lowest possible error or the minimum distance between Fourier spectrum of signal and the spectrum of the model.

Some of the well-known classic methods usually used for estimation of the all-pole model parameters include autocorrelation-based methods (*e.g.* Levinson-Durbin's method) and covariance-based methods (*e.g.* Burg's method). Despite having some drawbacks, each of these methods has an appropriate computational order [5].

Many studies have been done to reduce this error. For example, in the algorithm proposed in [2], calculations of covariance method are improved, and the results show enhanced spectral leakage and bias due to covariance method.

Poles of the AR model are obtained through model parameters. In the approach presented in [7], the poles of model are obtained directly from frequency-domain samples, and model parameters are calculated using poles. The results show that no spectral leakage occurs, and the poles of the model have a better stability.

2. AR model

In the time domain expression of the AR model, the current signal sample is equal to the linear combination of P previous signal samples plus an additional error term (Eq. (1)). In the frequency domain, the spectrum of the signal is parameterized using an all-pole model (AR) of the order P and is defined as the transfer function $H(z)$ as follows:

$$H(z) = \frac{b_0}{a_0 + \sum_{k=1}^{P} a_k z^{-k}} = \frac{b_0}{\sum_{k=0}^{P} a_k z^{-k}} \tag{4}$$

Without loss of generality of the problem, in some references, a_0 is assumed equal to 1, however, such an assumption is not made in this paper.

In this model, $x(n)$ signal is equal to a linear weighted combination (linear prediction) of previous samples of the same signal and an input signal $u(n)$, which is presented by the following difference equations:

$$a_0 x(n) + \sum_{k=1}^{P} a_k x(n-k) = b_0 \; u(n) \tag{5}$$

$$a_0 x(n) = -\sum_{k=1}^{P} a_k x(n-k) + b_0 \; u(n) \tag{6}$$

$$x(n) = \frac{-\sum_{k=1}^{P} a_k x(n-k)}{a_0} + \frac{b_0 \; u(n)}{a_0} \tag{7}$$

$$x(n) = \hat{x}(n) + \frac{b_0 \ u(n)}{a_0} \tag{8}$$

$$\hat{x}(n) = \frac{-\sum_{k=1}^{P} a_k x(n-k)}{a_0} \tag{9}$$

In the above equations, $\hat{x}(\mathrm{n})$ is an approximation of the *x(n)* signal obtained using Linear Prediction method.

Linear Prediction error, *E*, is defined as follows:

$$E = \sum_{n=0}^{N-1} \left(x(n) - \hat{x}(n) \right)^2 \tag{10}$$

$$E = \sum_{n=0}^{N-1} \left(x(n) + \frac{\sum_{k=1}^{P} a_k x(n-k)}{a_0} \right)^2 \tag{11}$$

$$E \cong \sum_{n=P}^{N-1} \left(x(n) + \frac{\sum_{k=1}^{P} a_k x(n-k)}{a_0} \right)^2 , \quad N \gg P \tag{12}$$

Indeed, gain coefficient, b_0, and linear prediction coefficient, a_k, should be estimated such that the error, *E*, is minimized. Finally, the error function can be calculated relatively by the following relation:

$$E_r \cong \frac{\sum_{n=P}^{N-1} \left(x(n) + \frac{\sum_{k=1}^{P} a_k x(n-k)}{a_0} \right)^2}{\sum_{n=P}^{N-1} x^2(n)} \tag{13}$$

3. Particle Swarm Optimization

Particle Swarm Optimization (PSO), similar to the other population-based global optimization algorithms (metaheuristic algorithms), is initialized by creating a population of random solutions. Each particle moves around in the solution space with a velocity which is tuned up based on the particle and population experiences. The location and velocity of a particle is calculated using the following rules:

$$v_i^{t+1} = w.v_i^t + c_1 u_1^t \left(pb_i^t - x_i^t \right) + c_2 u_2^t \left(gb^t - x_i^t \right) \tag{14}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{15}$$

In the equations (14) and (15), v_i^t is the velocity of the *i*-th particle in iteration *t*, coefficient *w* is the inertial weight (its initial value was set to 1 in this paper), x_i^t is the particle location in iteration *t*, pb_i^t is best location of the *i*-th particle up to iteration *t* (personal best position), and gb^t is the best location of all particles up to iteration *t*. Coefficients c_1 and c_2 are the constants of the problem, which specify the relative impact of personal-best particle and the impact of global best particle, and their

value is assumed equal to 2 in this paper. Also, u_1 and u_2 are two random numbers with a uniform distribution in [0,1] interval.
Particle swarm optimization algorithm works as follows:

Initial values for velocity and location of particles are assigned randomly.
Repeat the following steps until the desired solution is achieved:

Calculate the objective function for each particle; if the objective function value of a particle is better than its *pb* value, then *pb* is set to the current value of the particle; if *pb* value of that particle is better than the global best (*gb*), then the global best is set to *pb*.

Update the velocity and location of each particle.

Decrease inertial weight.

The value of objective function in this problem is, in fact, equal to the negative relative distance between 'Fourier Spectrum of signal' and 'All-pole spectrum of signal', and the Location of particle is also equal to the parameters set of the linear prediction model (*i.e.* a_k*'s* and b_0).

4. Implementation and Experiment Results

In this paper, Particle Swarm Optimization (PSO) algorithm is used to estimate all-pole model-based parametric spectrum so that the spectrum estimation error is minimized. In order to generate more accurate results, one file of a male voice and one file of a female voice with the length and sampling frequency of about 10 seconds and 16000 Hz, respectively, were selected from FarsDat Persian speech database. Each file was divided into overlapped frames of 32 ms frame length and 128 frmae shift. Each frame was considered to have 512 sample length. Then, the parametric AR all-pole spectrum of each frame was estimated using Levinson-Durbin's method of order 12, and also Burg's method of order 12 (using MATLAB functions). Also, AR model parameters were further re-estimated using PSO optimization algorithm in each method, such that the spectrum estimation error was decreased. Finally, the spectrum obtained by the classic methods was compared with that obtained using PSO optimization.

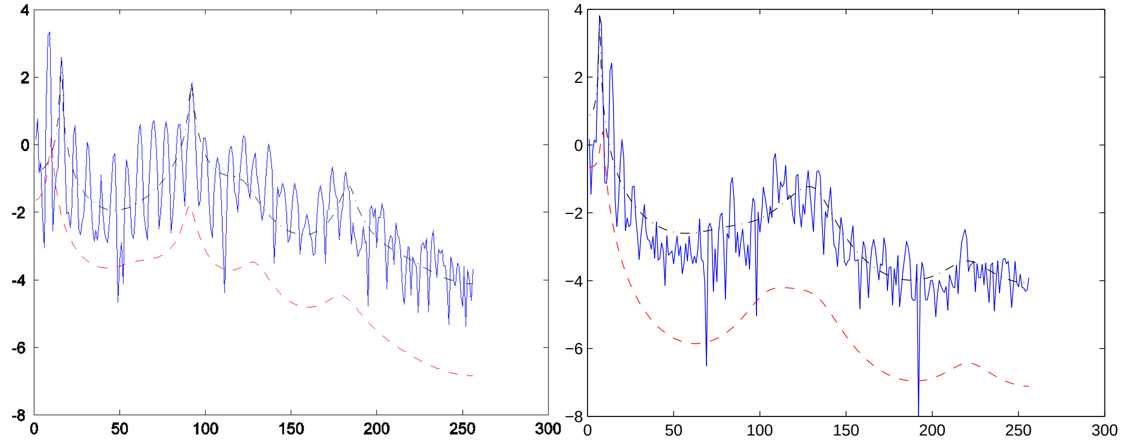
The number of particles was considered as 50 in the implementation of Particle Swarm Optimization algorithm. The initial location of one of the particles (primary particle) was assigned to the value obtained from either Levinson-Durbin's or Burg's method. Other particles locations were determined randomly in the space near the primary particle. The initial velocity of particles was assigned randomly. The number of algorithm iterations was set to 600 experimentally. The initial value of inertial weight was set to 1, which was linearly decreased in each iteration until it reached the final value of 0.4. Constant accelerators, c_1 and c_2 was set to 2.

Table 1 – Relative errors for parametric AR spectrum estimation using classic and PSO methods.

	Classic Burg's method		Classic Levinson's method	
	Female voice	Male voice	Female voice	Male voice
Mean estimation error in classic method	94.9٪	95٪	94.8٪	94.9٪
Mean estimation error in PSO method	53.6٪	56٪	53.4٪	55.7٪
Improvement of PSO method compared to classic method	43.5٪	41٪	43.7٪	41.3٪

4-1. Results

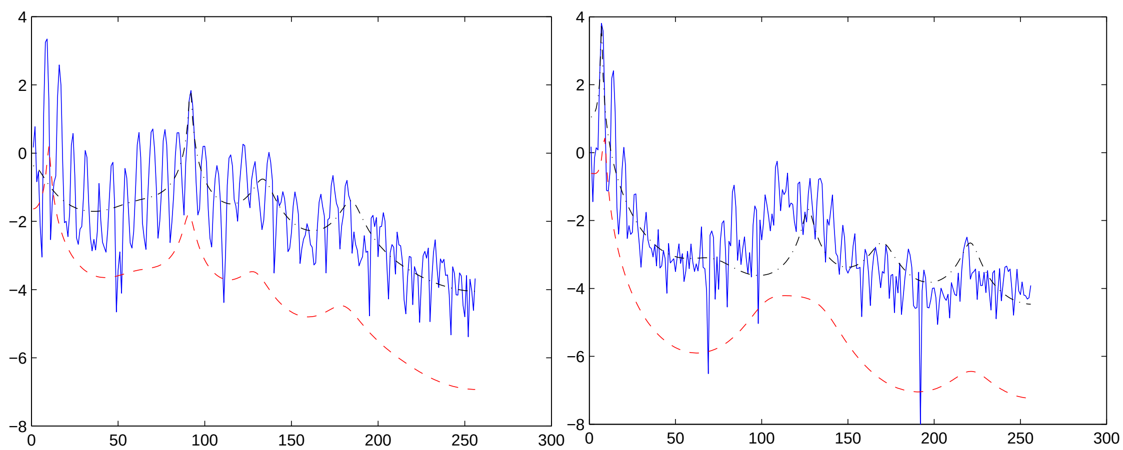
The simulation results are given in Table 1. As it can be seen, application of PSO algorithm improves estimation error at least 41%, while the computational complexity of this algorithm is more than that of the classic methods for AR model estimation.



(a) Male voice (b) Female voice

Fig. 1 – The log-spectrum of a vowel speech frame (solid line), estimated AR spectrum using Levinson's method (dashed line), and estimated AR spectrum using PSO method (dash-dotted line).

Fig. 1 shows the logarithm of the Fourier spectrum of a vowel frame, and the logarithm of the all-pole spectrum obtained by Levinson-Durbin's method and PSO method. In Fig. 2, the logarithm of Fourier spectrum of the same vowel frame, and the logarithm of all-pole spectrum obtained by Burg's method and PSO method is observed. As it can be seen, the accuracy of spectrum estimation by PSO method is considerably improved compared to the classic methods. One of the drawbacks of classic methods is their error in estimation of gain b_0, which is largely eliminated using PSO method. Perhaps, one reason for the good performance of the PSO method in spectrum estimation is that the number of problem parameters is not so high, and, for example, it is equal to P+1 in AR model of speech signal and in this paper 13.



(a) Male voice (b) Female voice

Fig. 2 – The log-spectrum of a vowel speech frame (solid line), estimated AR spectrum using Burg's method (dashed line), and estimated AR spectrum using PSO method (dash-dotted line).

5. Conclusion

Classic methods for parametric all-pole signal spectrum estimation cannot reach the global optimum (minimum) error. In this paper, the PSO metaheuristic method was used for estimating parameters of all-pole model of speech signal. Results obtained from male and female voice show that the all-pole spectrum estimation error of PSO method is considerably decreased compared to both Levinson-Durbin's and Burg's classic methods. However, the amount of calculations needed when using PSO method for estimating spectrum are much higher than the calculations required when using classic methods, and thus, utilizing this method in real-time applications becomes impossible and a solution must be found for this problem. Spectrum estimation using PSO method in real-time might be achieved using parallel and general-purpose hardwares like GPU.

References

[1] Aleksandar Lazinica, Particle Swarm Optimization, In-Tech Publishers, 2009.
[2] Bo Li, "Improved Algorithm for AR Model Spectral Estimation", Journal of Computational Information Systems, vol. 8, no. 17, pp. 7003-7010, 2012.
[3] E.C. Ifeachor, B.W. Jervis, Digital Signal Processing, A Practical Approach, 2nd Edition, Prentice-Hall, 2002.
[4] Jan Erkelens, Jesper Jensen, Richard Heusdens, "A data-driven Approach to Optimizing Spectral Speech Enhancement Methods for Various Error Criteria", Speech Communication, vol. 49, pp. 530-541, 2007.
[5] J.G. Proakis, D.G. Manolakis, Digital Signal Processing, 4th Edition, Prentice-Hall, 2007.
[6] Piet M.T. Broersen, Automatic Autocorrelation and Spectral Analysis, Springer, 2006.
[7] Luis Weruaga, "All-Pole Estimation in Spectral Domain", IEEE Transactions on Signal Processing, pp. 1-10, 2007.
[8] Vijay K. Madisetti, "The Digital Signal Processing Handbook", 2nd Edition, CRC Press, 2009.