INFORMATICA, 2011, Vol. 22, No. 2, 177–188

The Burg Algorithm with Extrapolation for Improving the Frequency Estimation

Kazys KAZLAUSKAS
Vilnius University Institute of Mathematics and Informatics
Akademijos 4, LT-08663 Vilnius, Lithuania
e-mail: kazys.kazlauskas@mii.vu.lt

Received: April 2010; accepted: December 2010

Abstract. The paper presents a novel method for improving the estimates of closely-spaced frequencies of a short length signal in additive Gaussian noise based on the Burg algorithm with extrapolation. The proposed method is implemented in two consecutive steps. In the first step, the Burg algorithm is used to estimate the parameters of the predictive filter, while in the second step the extrapolation technique of the signal is used to improve the frequency estimates. The experimental results demonstrate that the frequency estimates of the short length signal, using the Burg algorithm with extrapolation, are more accurate than the frequency estimates using the Burg algorithm without extrapolation.

Keywords: frequency estimation, short length signals, Burg algorithm, extrapolation.

1. Introduction

The autoregressive (AR) spectral analysis has become popular in many applications areas where harmonic components have to be detected and analyzed, e.g., radar (Haykin, 1979), geophysics (Yuou *et al.*, 1996), economics (Box and Jenkins, 1970), and signal processing (Waele and Broersen, 2000). In most problems of practical application, the covariance sequence of a signal is not known a priori, and thus the criterion, used to generate autoregressive model parameters for spectral estimation either directly from the given signal samples, or through estimates of the covariance sequence, determines the quality of the spectrum (Helme and Nikias, 1985). Linear prediction is useful in many signal processing applications, including spectral estimation (Lee, 1989), system identification (Marple, 1982), time series extrapolation (Kay, 1983), and speech recognition (Lipeika, 2010). Linear prediction estimates the current data sample as a linear combination of the past or future data samples. The optimal prediction coefficients are determined by minimizing the mean-square error. Since only the past or future data samples are used to estimate the current data sample value, that is called a one-sided linear prediction.

A better estimate of the present data sample would be expected, if we predict the present data sample based on both the past and future data samples simultaneously. The two-sided linear prediction (Hsue and Yagle, 1995) has been used in various signal processing applications, including spectral estimation (Lee, 1989), speech coding (David and Ramamurthi, 1991), linear phase filter design (Farden and Scharf, 1974), time series interpolation (Kay, 1983), and system identification (Marple, 1982).

A comparison of various estimators of AR parameters has showed that the Burg algorithm (Burg, 1967; Kay and Marple, 1981) is the preferred estimator for AR parameters (Broersen, 1997). The techniques choose the best set of AR parameters directly from the given signal samples by minimizing the sum of the average energies of the forward and backward linear prediction errors subject to the constraint that the optimum set of parameters satisfies the Levinson recursion. The Burg algorithm provides an increased spectral resolution over the conventional methods and the Yule–Walker technique based on biased autocorrelation estimates. The Yule–Walker algorithm can be severely biased. The least squares estimator and the forward-backward least-squares estimator have a greater variance than the Burg algorithm. In addition, they may yield unstable models (Helme and Nikias, 1985).

An important problem in several signal processing applications is the estimation of the frequencies and powers of sinusoids observed in additive noise. Many high resolution methods have been proposed, including the total least squares based autoregressive (AR) modeling (Swami and Mendel, 1991), the ESPRIT and its various variants (Roy and Kailath, 1989; Swindlehurst *et al.*, 1992).

The resolution for the AR spectrum has not been well defined because of its nonlinear dependence on signal power and model order. Since the denominator of AR spectrum is a polynomial of degree p with real coefficients, a maximum of $p/2$ independent poles or real frequency components could be resolved. The frequency separation necessary to resolve two neighboring components depends on the signal-to-noise ratio (SNR) as well as on the model order, and it is sensitive to the accuracy of the autocorrelation estimates.

It is well known that the spectral width of a peak decreases and therefore the resolution increases with an increasing signal power and model order. However, an excessive model order may lead to line splitting or spurious spectral peaks (Quirk and Liu, 1983). One of the most attractive features of the AR spectrum estimation algorithms, which employ data samples directly, bypassing covariance estimates, is their ability to estimate the frequencies of closely-spaced spectral peaks. Therefore, it is very interesting to investigate the ability to estimate the frequencies of a short length process consisting of two or more sinusoids in additive Gaussian white noise.

The aim of this paper is to present a new approach to improve the frequency estimates of the short length signals in Gaussian additive noise, using the Burg algorithm and the extrapolation technique. The organization of the paper is as follows. Section 2 provides a description of the Burg algorithm. The power spectrum and frequency estimation technique are presented in Section 3. Section 4 is the core part of this paper. Finally, the paper discusses the experimental results and gives conclusions.

2. The Burg Algorithm

Suppose that we are given a signal $x(n),\ n = 1, 2, \ldots, N$, and let us consider the forward and backward linear prediction (LP) estimates of order $m = 1, 2, \ldots, p$

$$\hat{x}(n) = -\sum_{k=1}^{m} \alpha_m(k)x(n-k), \tag{1.1}$$

$$\hat{x}(n-m) = -\sum_{k=1}^{m} \beta_m(k)x(n+k-m), \tag{1.2}$$

where $\alpha_m(k)$ and $\beta_m(k)$ are the forward and backward prediction coefficients respectively, $x^t(n) = [x(n), x(n-1), \ldots, x(n-m)]$.

The Burg method is based on the concept of forward and backward finite impulse response (FIR) filters of the signal $x(n)$

$$f_m(n) = x(n) - \hat{x}(n) = \sum_{k=0}^{m} \alpha_m(k)x(n-k), \tag{2.1}$$

$$b_m(n) = x(n-m) - \hat{x}(n-m) = \sum_{k=0}^{m} \beta_m(k)x(n+k-m), \tag{2.2}$$

where $f_m(n)$ and $b_m(n)$ are the forward and backward prediction errors (residuals).

Note that $\alpha_m(0) = \beta_m(0) = 1$ by definition.

The forward filter output $f_m(n)$ and the backward filter output $b_m(n)$ depend on the column $(m+1)$-dimensional vector $x(n)$. In practice, we must choose $m < N$.

We assume that $x(n)$ is only available over the interval $1 \leqslant n \leqslant N$, so that FIR outputs can only be formed over the interval $m+1 \leqslant n \leqslant N$. The method combines filtering of the signal $x(n)$ in the forward and backward directions through the FIR filter is presented in one variance expression. It is shown that the forward and backward LP parameters for a stationary random process are complex conjugates, so the output $b_m(n)$ of the backward FIR filter may be expressed as (Proakis and Manolakis, 1996)

$$b_m(n) = \sum_{k=0}^{m} \alpha_m^*(k)x(n+k-m), \tag{3}$$

where the sign "*" means complex conjugate.

The coefficients in the backward FIR filter are the complex conjugates of the coefficients for the forward FIR filter, but they occur in the reverse order.

The FIR prediction error filter can be implemented through the lattice filter. The lattice filter is described by the set of order-recursive equations

$$\begin{aligned} f_m(n) &= f_{m-1}(n) + K_m b_{m-1}(n-1), \\ b_m(n) &= K_m f_{m-1}(n) + b_{m-1}(n-1), \quad m = 1, 2, \ldots, p, \end{aligned} \tag{4}$$

where K_m are the reflection coefficients of the mth recursion step.

The initial values for the residuals are $f_0(n) = b_0(n) = x(n)$. An essential characteristic of the Burg algorithm is that the number of residuals decreases with each recursion step. The Burg algorithm calculates the reflection coefficients K_m so that they minimize the sum of the forward and backward residual errors. This implies an assumption that the same autoregressive (AR) model can predict the signal forward and backward.

The criterion E_m to be minimized with respect to K_m is the sum of squares of the forward and backward residuals in the mth recursion step

$$E_m = \sum_{n=p+1}^{N} \big\{ \big[f_m(n)\big]^2 + \big[b_m(n)\big]^2 \big\}, \quad m = 1, 2, \ldots, p. \tag{5}$$

Minimization of E_m with respect to the reflection coefficients K_m yields

$$\begin{aligned} \frac{\partial E_m}{\partial K_m} = 2 \sum_{n=p+1}^{N} \big\{ &\big[f_{m-1}(n) + K_m b_{m-1}(n-1)\big] b_{m-1}(n-1) \\ &+ \big[K_m f_{m-1}(n) + b_{m-1}(n-1)\big] f_{m-1}(n) \big\} = 0, \end{aligned} \tag{6}$$

from which the reflection coefficients follow

$$K_m = \frac{-2 \sum_{n=p+1}^{N} f_{m-1}(n) b_{m-1}(n-1)}{\sum_{n=p+1}^{N} \{[f_{m-1}(n)]^2 + [b_{m-1}(n)]^2\}}. \tag{7}$$

The FIR filter coefficients $\alpha_m(k)$ can be obtained from the reflection coefficients K_m via the Levinson–Durbin algorithm

$$\begin{array}{ll} \alpha_m(0) = 1, & \\ \alpha_m(m) = K_m, & m = 1, 2, \ldots, p, \\ \alpha_m(k) = \alpha_{m-1}(k) + K_m \alpha_{m-1}(m-k), & k = 1, 2, \ldots, m-1. \end{array} \tag{8}$$

At the end of the recursions, $\alpha_p(k)$ gives the prediction error filter estimated coefficients $\hat{\alpha}(k)$. The absolute value of K_m is always smaller than unity. Therefore, the stability of the estimated AR model is guaranteed. The Burg method not only minimizes the combined global error, but also it gives better estimates and a lower error, since it uses more data. The Burg method results in a high frequency resolution and is computationally efficient.

3. Estimation of the Power Density Spectrum

The algorithm computes the AR coefficients by (8). The frequency can be extracted from the autoregressive parameters $\hat{\alpha}(k)$. The signal is modeled as the output of the AR process with a zero mean white noise input $w(n)$. From the estimates of AR parameters, we form the power density spectrum estimate. The power density spectrum estimate of the signal $x(n)$ is given by Proakis and Manolakis (1996)

$$\hat{P}(f) = \frac{1}{f_s} \frac{1}{\left|1 + \sum_{k=1}^{p} \hat{\alpha}(k) e^{-2\pi jkf/f_s}\right|^2}, \tag{9}$$

where f is the frequency, and f_s is the sampling frequency.

Frequencies can be found by estimating the peaks of the power density spectrum $\hat{P}(f)$. If there are several spectral peaks in the power density spectrum function, a threshold must be set, and the peaks below this threshold belong to the noise. Another method to estimate the frequency is to form the estimation error filter polynomial $z^p + \hat{\alpha}(1)z^{p-1} + \cdots + \hat{\alpha}(p)$, and to calculate the roots z_i. The frequency estimates are the angles of the roots

$$\hat{f}_i = \frac{f_s}{2\pi} \mathrm{angle}\big(z_i\big). \tag{10}$$

If the signal is real, then the roots are complex conjugate. In that case, the roots in the upper or lower half of a complex plane are selected.

4. The Proposed Method

In the proposed method, the given signal $x(N_b + 1), \dots, x(N)$, is forward extrapolated to get the next sample $x_f(N+1)$, using the Burg algorithm for signal model coefficients. The coefficients, which were calculated by the Burg method, are updated each time to find the next samples $x_f(N+1), \dots, x_f(N+N_f)$, and all the previous samples are used to find the new AR coefficients. We repeated the extrapolation until we have got $x_f(N + N_f)$. Similarly, we also backward extrapolated the given signal $x(N_b + 1), \dots, x(N)$, to find the sample $x_b(N_b)$. This step was also repeated until we have got the sample $x_b(1)$. The extrapolated samples $x_f(N+1), \dots, x_f(N+N_f)$ and $x_b(1), \dots, x_b(N_b)$ are added to the given signal $x(N_b + 1), \dots, x(N)$ and the power density spectrum is calculated from the signal

$$\begin{aligned} x_e(n) = \big\{&x_b(1), \dots, x_b(N_b),\ x(N_b+1), \dots, x(N), \\ &x_f(N+1), \dots, x_f(N+N_f)\big\}, \end{aligned}$$

where $x_e(n),\ n = 1, 2, \dots, N_b + N + N_f$ is the sequence of the signal after two-sided extrapolation; N_b is the number of backward extrapolated samples; N_f is the number of forward extrapolated samples.

We tried three different cases to find the power density spectrum. First, the extrapolated signals $x_e(n)$ are averaged in the time domain, and the power density spectrum is calculated. Second, the coefficients of the predictive filters are averaged and then the power density spectrum is calculated from the averaged coefficients. Third, the power density spectrums of extrapolated signals are averaged and the final power density spectrum is obtained.

We estimated the mean frequency error (MFE) as follows:

$$\mathrm{MFE} = \frac{1}{M}\sum_{i=1}^{M} \left|f_i - \hat{f}_i\right|, \tag{11}$$

where M is the number of frequencies, f_i are true frequency values, and $\hat{f}_i$ are estimates of the true frequency values. The normalized power density spectrum (NPDS) estimate in dB is calculated according to the expression:

$$\mathrm{NPDS} = 10\log\frac{\hat{P}(f)}{\hat{P}_{\max}(f)}. \tag{12}$$

5. Simulation Results

In this section, we examine the performance of the proposed method and compare the results with that of the Burg algorithm. To investigate the abilities of the proposed method, we generated a signal from the signal generator comprised of two ($M = 2$), three ($M = 3$) or four ($M = 4$) sinusoids embedded in the noise

$$x(n) = s(n) + w(n) = \sum_{i=1}^{M} \cos(2\pi f_i n) + w(n), \tag{13}$$

for $n = 1, \ldots, N;$ f_i are frequencies, and $w(n)$ is a zero-mean white Gaussian noise with the unit variance $\sigma_w^2 = 1$. To get the desired Signal-to-Noise Ratio (SNR) from the signal generator, the output signal is defined by

$$x(n) = s(n) + kw(n), \tag{14}$$

in which the coefficient k is computed such that

$$\mathrm{SNR} = 10\log\frac{P_s}{k^2 P_w}, \tag{15}$$

where $P_s = \frac{1}{N}\sum_{n=1}^{N} s^2(n),\ P_w = \frac{1}{N}\sum_{n=1}^{N} w^2(n)$, and N is the length of the $s(n)$ and $w(n)$.

From (15) we obtain that for desired SNR, the coefficient k is calculated as follows

$$k = \frac{\sqrt{P_s}}{\sqrt{P_w}}10^{-\frac{\mathrm{SNR}}{20}}. \tag{16}$$

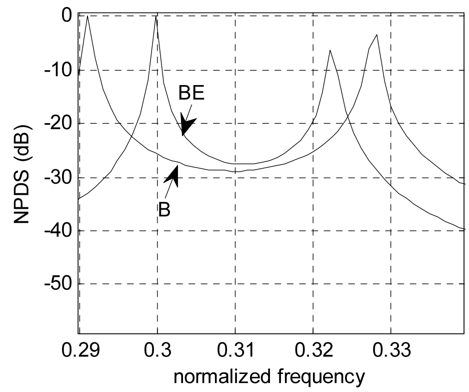

Fig. 1. The normalized power density spectrum versus the normalized frequency. SNR $=$ 15 dB, $N = 30,\ N_f = N_b = 15,\ p = 16;\ f_1 = 0.3,\ f_2 = 0.32$. B is the Burg algorithm; BE is the Burg algorithm with extrapolation. MFE1 $= 0.0043 \pm 0.0006$, MFE2 $= 0.0015 \pm 0.0005$. Monte Carlo runs are 200. The coefficients of the predictive filters are averaged (Case 2).

We have used $L = 200$ Monte Carlo simulations in the case of additive noise. The same signals were used to demonstrate the superiority of the proposed method as compared with the Burg algorithm. In Figs. 1–4, we show examples of the averaged normalized power density spectrums versus the normalized frequency: the plots B are the spectrums obtained by the Burg algorithm, and the plots BE are the spectrums obtained by the proposed method; MFE1 is the normalized mean frequency error of the Burg algorithm, and MFE2 is the normalized mean frequency error of the proposed method. In Figs. 1–2, we show a signal comprised of two sinusoids with frequencies $f_1 = 0.3$ and $f_2 = 0.32$ embedded in a noise. In Fig. 3, we show a signal comprised of three sinusoids with frequencies $f_1 = 0.2,\ f_2 = 0.24$, and $f_3 = 0.28$ embedded in a noise. In Fig. 4, we show a signal comprised of four sinusoids with frequencies $f_1 = 0.2,\ f_2 = 0.23,\ f_3 = 0.26$, and $f_4 = 0.29$ embedded in a noise.

We analyze a signal comprised of two sinusoids with frequencies $f_1 = 0.3$ and $f_2 = 0.32$ embedded in a noise. Table 1 illustrates the normalized frequency error estimates averaged by $L = 200$ experiments and their confidence intervals $\Delta = \pm t_{\alpha/2;L-1}\frac{\hat{\sigma}}{\sqrt{L}}$, in which $\hat{\sigma}$ is the estimate of the standard deviation and α is the significance level. The value $t_{\alpha/2;L-1}$ is the point of Student's distribution with $L-1$ degrees of freedom which cuts the $\alpha/2$ part of the distribution. In case $\alpha = 0.05$ and $L = 200$, we find from Student's distribution table that $t_{0.025;199} = 1.9720$.

In the column "Case 1" of Table 1, the signal $x(n),\ n = 1,2,\ldots,N$ (rows B) and the extrapolated signal $x_e(n),\ n = 1,2,\ldots,N_b + N + N_f$ (rows BE) are averaged in the time domain for $L = 200$ experiments, and then the power density spectrums are calculated using (9). Then, from (10), we calculate the frequency estimates $\hat{f}_1$ and $\hat{f}_2$, and from (11), we obtain the mean frequency error estimates. In the column "Case 2", the coefficients of the predictive filters of the Burg algorithm (rows B) and the coefficients of the Burg algorithm with extrapolation (rows BE) are averaged and then the power density spectrums are calculated. In the column "Case 3", the power density spectrums

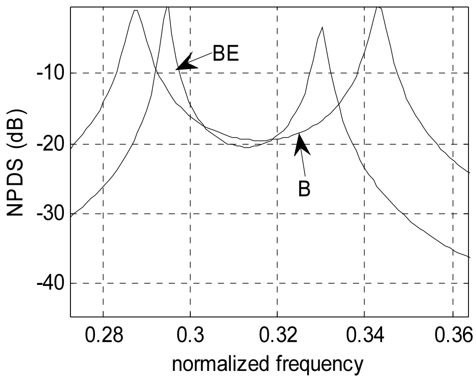

Fig. 2. The normalized power density spectrum versus the normalized frequency. SNR $=$ 15 dB, $N = 20$, $N_f = N_b = 10$, $p = 12$; $f_1 = 0.3$, $f_2 = 0.32$. B is the Burg algorithm; BE is the Burg algorithm with extrapolation. MFE1 $= 0.0081 \pm 0.00019$, MFE2 $= 0.0037 \pm 0.00021$. Monte Carlo runs are 200. The coefficients of the predictive filters are averaged (Case 2).

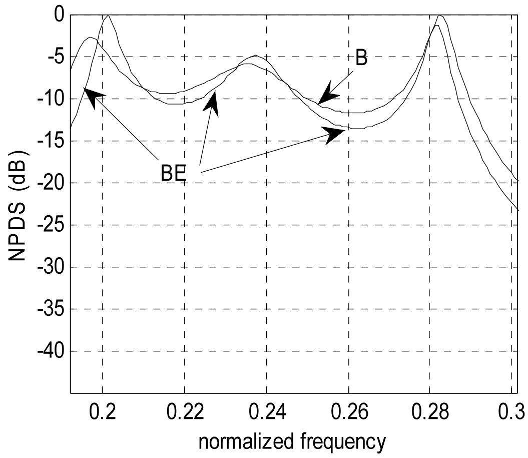

Fig. 3. Normalized power density spectrum versus normalized frequency. SNR $=$ 15 dB, $N = 30$, $N_f = N_b = 10$, $p = 12$, $f_1 = 0.2$, $f_2 = 0.24$, $f_3 = 0.28$. B – Burg algorithm; BE – Burg algorithm with extrapolation. MFE1 $= 0.0054 \pm 0.00036$, MFE2 $= 0.00318 \pm 0.00032$. Monte Carlo runs are 200. The coefficients of the predictive filters are averaged (Case 2).

of the signals $x(n)$ are averaged (rows B) and the power density spectrums of the signals $x_e(n)$ are averaged (rows BE), and then the power density spectrums are calculated.

Analysis of the results presented in Table 1 shows that the frequency estimates obtained by the proposed method (BE) are more accurate as compared with the Burg algorithm (B) in cases where SNR changes from -5 to 50 dB, signal length N changes from 15 to 40 points, and the predictive filter order p changes from 10 to 20.

The performance of the considered methods has been compared by varying the number of available data from $N = 20$ to $N = 60$. For each value of N, a Monte Carlo

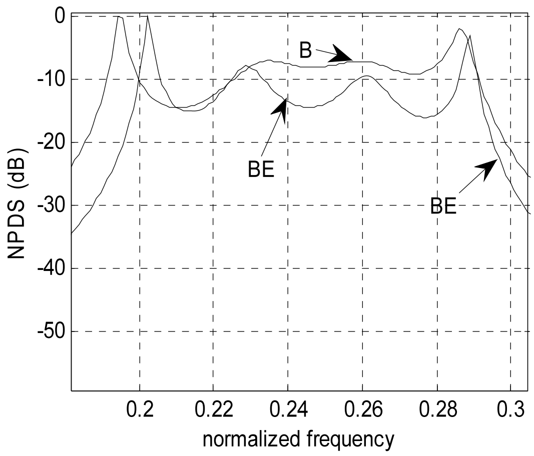

Fig. 4. The normalized power density spectrum versus the normalized frequency. SNR $=$ 30 dB, $N = 30,\ N_f = N_b = 10,\ p = 14,\ f_1 = 0.2,\ f_2 = 0.23,\ f_3 = 0.26,\ f_4 = 0.29$. B is the Burg algorithm; BE is the Burg algorithm with extrapolation. MFE1 $= 0.00745 \pm 0.00025$, MFE2 $= 0.00566 \pm 0.00053$. Monte Carlo runs are 200. The coefficients of the predictive filters are averaged (Case 2).

Table 1

The normalized mean frequency error (MFE) estimates and their confidence intervals Δ

		Case 1	Case 2	Case 3
SNR $= 50$, $N = 15$, $p = 10$,	B ----	0.0079 ± 0.00011	0.0085 ± 0.00011	0.0083 ± 0.00015
$N_f = N_b = 10$	BE ----	0.0072 ± 0.00009	0.0072 ± 0.00006	0.0073 ± 0.00008
SNR $= 20,\ N = 30,\ p = 20,$	B ----	0.0059 ± 0.00014	0.0064 ± 0.00017	0.0062 ± 0.00017
$N_f = N_b = 15$	BE ----	0.0027 ± 0.00012	0.0029 ± 0.00017	0.0029 ± 0.00017
SNR $= 20,\ N = 20,\ p = 15,$	B ----	0.0076 ± 0.00033	0.0067 ± 0.00053	0.0072 ± 0.00042
$N_f = N_b = 10$	BE ----	0.0051 ± 0.00045	0.0042 ± 0.00046	0.0045 ± 0.00035
SNR $= 15,\ N = 30,\ p = 20,$	B ----	0.0061 ± 0.00015	0.0075 ± 0.00026	0.0062 ± 0.00015
$N_f = N_b = 15$	BE ----	0.0030 ± 0.00099	0.0038 ± 0.00019	0.0033 ± 0.00019
SNR $= 15,\ N = 20,\ p = 12,$	B ----	0.0089 ± 0.00027	0.0108 ± 0.00019	0.0098 ± 0.00050
$N_f = N_b = 10$	BE ----	0.0045 ± 0.00075	0.0069 ± 0.00021	0.0062 ± 0.00055
SNR $= 10,\ N = 40,\ p = 20,$	B ----	0.0079 ± 0.00033	0.0074 ± 0.00013	0.0077 ± 0.00021
$N_f = N_b = 15$	BE ----	0.0037 ± 0.00014	0.0036 ± 0.00011	0.0042 ± 0.00015
SNR $= 10,\ N = 30,\ p = 20,$	B ----	0.0067 ± 0.00024	0.0081 ± 0.00028	0.0071 ± 0.00014
$N_f = N_b = 15$	BE ----	0.0045 ± 0.00017	0.0050 ± 0.00026	0.0045 ± 0.00018
SNR $= 5,\ N = 30,\ p = 20,$	B ----	0.0071 ± 0.00033	0.0086 ± 0.00046	0.0080 ± 0.00033
$N_f = N_b = 15$	BE ----	0.0063 ± 0.00031	0.0069 ± 0.00038	0.0062 ± 0.00021

SNR is the signal-to-noise ratio; N is the signal length; p is the order of the predictive filter; $N_f,\ N_b$ are numbers of forward and backward extrapolation points. B is the Burg algorithm; $f_1 = 0.3$ and $f_2 = 0.32$. BE is the Burg algorithm with extrapolation. Monte Carlo runs are equal to 200.

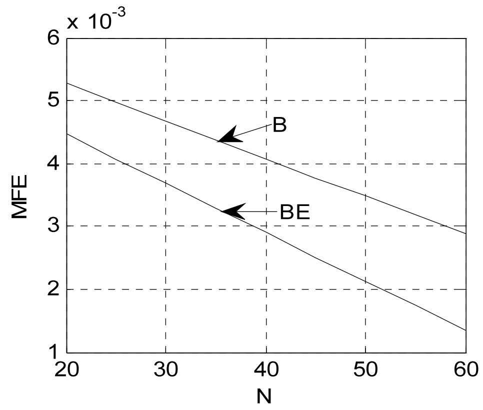

Fig. 5. The normalized MFE versus N: B is the Burg algorithm, BE is the Burg algorithm with extrapolation. SNR $= 15, N_f = N_b = 15, p = 15$.

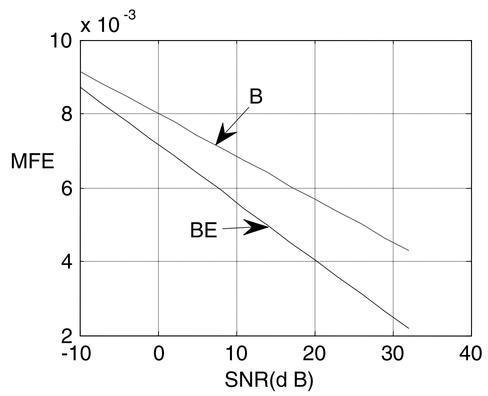

Fig. 6. The normalized MFE versus SNR: B is the Burg algorithm, BE is the Burg algorithm with extrapolation. $N = 30,\ N_f = N_b = 15,\ p = 15$.

simulation of 200 independent runs has been carried out. The mean frequency error (11) has been used as the performance index. The results are reported in Fig. 5. Finally, a set of Monte Carlo simulations with an SNR ranging from -10 to 40 dB has been performed, using $N = 30$ and MFE as the performance index. The results are shown in Fig. 6. It can be observed that the proposed algorithm outperforms the Burg algorithm and allows us to obtain a better frequency estimation accuracy. Other Monte Carlo simulations, performed by using different lengths of signals and SNRs, have led to the same conclusion.

6. Conclusion

The limitations of the Burg algorithm for the AR power spectrum estimation are the frequency bias and line-splitting in processing the sinusoidal signals in noise. On the other hand, its attractive features are high-resolution spectral estimates with short signal records, an efficient recursive implementation, and guaranteed stable models. In this paper, a new method, based on the Burg algorithm and extrapolation technique, has been proposed for estimating closely-spaced frequencies of the short length signals in the noisy environment. The simulation results have shown that in many cases the proposed method considerably reduces the frequency bias and the confidence intervals of frequency estimates as compared with the Burg algorithm.

Acknowledgements. The authors are grateful to anonymous referees for their constructive and helpful comments and suggestions.

References

Box, G.E.P., Jenkins, G.M. (1970). *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco.

Broersen, P.M.T. (1997). The ABC of autoregressive order selection criteria. In: *Proc. Sysid Conf.*, Kitakyushu, Japan, pp. 231–236.

Burg, J.P (1967). Maximum entropy spectral analysis. In: *Proc. 37th Meet. Soc. Explorational Geophys.*, Oklahoma City.

David, S., Ramamurthi, B. (1991). Two-sided filter for frame-based prediction. *IEEE Transactions on Signal Processing*, 39(4), 789–794.

Farden, D.C., Scharf, L.L. (1974). Statistical design of non recursive digital filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 22(3), 188–196.

Haykin, S. (1979). *Nonlinear Methods of Spectral Analysis.* Springer, Berlin.

Helme, B.I., Nikias, Ch.L. (1985). Improved spectrum performance via a data-adaptive weighted Burg technique. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(4), 903–910.

Hsue, J., Yagle, A.E. (1995). Similarities and differences between one-sided and two-sided linear prediction. *IEEE Transactions on Signal Processing*, 43(1), 345–349.

Yuou, P., Baert, E., Loutre, M.F. (1996). Spectral analysis of climate data. *Survey Geophys.*, 17, 619–663.

Kay, S.M. (1983). Some results in linear interpolation theory. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(3), 746–749.

Kay, S.M., Marple, S.L. (1981). Spectrum analysis – a modern perspective. *Proc. IEEE*, 69, 1380–1419.

Lee, A.C. (1989). A new autoregressive method for high-performance spectrum analysis. *J. Acoust. Soc. Am.*, 86, 150–157.

Lipeika, A. (2010). Optimization of formant feature based speech recognition. *Informatica*, 21(3), 361–374.

Marple, S.L (1982). Fast algorithms for linear prediction and system identification filters with linear phase. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(6), 942–953.

Proakis, J.G., Manolakis, D.G. (1996). *Digital Signal Processing: Principles, Algorithms, and Applications*. Printice Hall, New Jersey.

Quirk, M.P., Liu, B. (1983). Improving resolution for autoregressive spectral estimation by decimation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(3), 630–637.

Roy, R., Kailath, T. (1989). ESPRIT – Estimation of signal parameters via rotational invariance technique. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7), 984–995.

Swami A., Mendel, J.M. (1991). Cumulant-based approach to the harmonic retrievel and related problems. *IEEE Transactions on Signal Processing*, 39(5), 1099–1109.

Swindlehurst, A.L., Ottersten, B., Roy, R., Kailath, T. (1992). Multiple invariance ESPRIT. *IEEE Transactions on Signal Processing,* 40(4), 867–881.

Waele, S., Broersen, P.M.T. (2000). The Burg algorithm for segments. *IEEE Trans. on Signal Processing*, 48(10), 2876–2880.

K. Kazlauskas received a PhD degree from Kaunas Polytechnic Institute and a doctor habilius degree from Institute of Mathematics and Informatics and Vytautas Magnus University. He is a senior researcher of the Recognition Processes Department at the Vilnius University Institute of Mathematics and Informatics and a professor at the Informatics Department of Vilnius Pedagogical University. His research interests include signal processing, parameter estimation, and digital system design.

Dažnių įverčių pagerinimas naudojant Burgo algoritmą ir ekstrapoliaciją

Kazys KAZLAUSKAS

Straipsnyje pasiūlytas naujas metodas, pagrįstas Burgo algoritmu ir signalo reikšmių ekstrapoliacija, skirtas trumpų signalų stebimų Gauso triukšmuose dažnių skiriamosios gebos pagerinimui. Metodą sudaro dvi dalys: pirma, naudodami Burgo algoritmą, įvertiname prognozės filtro parametrus, ir antra, ekstrapoliuojame signalo reikšmes pirmyn ir atgal bei apskaičiuojame taip “pailginto“ signalo spektrą. Eksperimento rezultatai parodė, kad trumpų signalų stebimų triukšmuose dažnių skiriamoji geba yra didesnė, kai naudojamas Burgo algoritmas bei signalo reikšmių ekstrapoliavimas pirmyn ir atgal.