16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland, August 25-29, 2008, copyright by EURASIP

CONSTRAINED POLE-ZERO LINEAR PREDICTION: AN EFFICIENT AND NEAR-OPTIMAL METHOD FOR MULTI-TONE FREQUENCY ESTIMATION

Toon van Waterschoot and Marc Moonen

Dept. E.E./ESAT, SCD-SISTA, Katholieke Universiteit Leuven
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
phone: +32 16 321709, fax: +32 16 321970
email: {tvanwate,moonen}@esat.kuleuven.be
web: http://homes.esat.kuleuven.be/∼tvanwate

ABSTRACT

Constrained pole-zero linear prediction (CPZLP) is proposed as a new method for parametric frequency estimation of multiple real sinusoids buried in noise. The method is based on a signal model that consists of a cascade of second-order constrained pole-zero models, thereby exploiting the linear prediction property of sinusoidal signals. The signal model is parametrized directly with the unknown frequencies, which are then estimated using a numerical optimization approach. By independently optimizing each second-order stage in the cascade model, a computationally efficient algorithm is obtained with a complexity that is linear in both the data record length and the number of sinusoids. The linear complexity allows for using relatively long data records, leading to high accuracy even at low signal-to-noise ratios (SNR). Simulation results confirm that the CPZLP algorithm nearly achieves the Cramér-Rao lower bound for SNR as low as 5 dB.

1. INTRODUCTION

The problem of estimating the frequencies of a sum of sinusoidal signals (multi-tone signals) buried in additive noise has received a lot of attention during the past decades. Solutions to this problem have been applied in many different areas, such as audio and speech processing, radar signal processing, telecommunications, etc. The existing methods are usually categorized as being either nonparametric or parametric. Nonparametric frequency estimation is directly based on Fourier transform theory, hence the signal is processed in a frame-based manner. The main drawback of nonparametric methods is their limited frequency resolution for finite frame length. Parametric methods, on the other hand, can achieve a higher resolution but require the postulation of a generating signal model. We refer to [1] for a recent overview of parametric frequency estimation methods.

A particular class of parametric methods exploits the linear prediction (LP) property of sinusoidal signals. It is well known that a sum of P sinusoids can be described exactly using an all-pole model of order $2P$, with mirror symmetric LP coefficients [1]. However, it has been shown that the all-pole model is not exact when noise is added, and in this case a pole-zero model of order $2P$ should be used [2]. Still, by constraining the poles and zeros to lie on common radial lines in the z-plane, the number of unknown parameters in the pole-zero model can be limited to P and the LP parameters can be uniquely related to the unknown frequencies [3]. The constrained pole-zero model has been widely applied in adaptive notch filtering (ANF), see, e.g., [3]-[5]. The ANF algorithms are however very sensitive to the choice of the initial conditions and the exponential forgetting factor, and in nonstationary scenarios memory resetting of the ANF is regularly required to enable sufficiently fast tracking.

In this paper, we describe a new parametric frequency estimation method that is based on the constrained pole-zero model proposed in [3], realized using a cascade of second-order sections with a direct frequency parametrization [4],[5]. The proposed method is referred to as constrained pole-zero linear prediction (CPZLP) and, in contrast to the ANF approach, the signal is processed in a frame-based manner. In the CPZLP method, the minimization of a least-squares (LS) objective for multi-tone frequency estimation is decoupled into a set of single-tone subproblems that can be solved consecutively by exploiting the cascade structure of the signal model. Each subproblem can be viewed as a single-variable unconstrained nonlinear optimization problem, and is solved iteratively using a numerical line search method [6, Ch. 3]. Because of the decoupling, the CPZLP method achieves a computational complexity that depends linearly on the frame length and on the number of second-order sections, even when Hessian information is used in the optimization method. As a consequence, relatively long frame lengths can be used to increase the noise robustness.

The paper is organized as follows. In Section 2, we introduce the constrained pole-zero signal model and derive the CPZLP algorithm by considering the decoupled optimization of the LS objective. We describe a line search method with three possible ways of calculating the search direction (steepest descent, Gauss-Newton, and quasi-Newton), and provide details on the gradient and Hessian calculation. Section 3 deals with the computational complexity of the CPZLP algorithm, and Section 4 contains Monte Carlo simulation results that illustrate the CPZLP performance in terms of frequency variance as compared to the Cramér-Rao lower bound (CRLB). Finally, Section 5 concludes the paper.

2. CONSTRAINED POLE-ZERO LINEAR PREDICTION

2.1 Signal model

The observed signal $y(t)$ is assumed to consist of a sum of real sinusoids and additive noise,

$$y(t) = \sum_{n=1}^{P} A_n \cos(\omega_n t + \phi_n) + r(t),\ t = 1,\ldots,N \qquad (1)$$

with A_n the amplitude, $\omega_n \in [0,\pi]$ the radial frequency, and $\phi_n \in [0,2\pi)$ the phase of the nth sinusoid. While most parametric frequency estimators rely on the hypothesis that the noise $r(t)$ is white [1], we do not make explicit assumptions about the noise. The CPZLP algorithm has been tested both with white noise, see Section 4, and with colored noise, see [7]. Troughout this paper, it is assumed that the number of sinusoids P in the observed signal is known a priori, which is a common assumption in parametric frequency estimation [1], [3]-[5]. We should note that the CPZLP approach can

This research work was carried out at the ESAT laboratory of the Katholieke Universiteit Leuven, in the frame of K.U.Leuven Research Council CoE EF/05/006 Optimization in Engineering (OPTEC) and the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office IUAP P6/04 (DYSCO, 'Dynamical systems, control and optimization', 2007-2011), and was supported by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen). The scientific responsibility is assumed by its authors.

be elegantly extended to achieve joint order and frequency estimation [7]. The frequencies ω_n are assumed to be stationary over the observed data frame $t \in [1,N]$. Frequency tracking in a nonstationary environment is possible if CPZLP is preceded by an adaptive segmentation.

It is well known that a sum of P sinusoids can be described exactly as an autoregressive process of order $2P$. A sum of P sinusoids in additive white noise, similarly, can be modeled as an autoregressive moving average process of order $2P$, having zeros that coincide with the poles [2]. This observation has led to the constrained pole-zero model for signals that consist of sinusoidal or narrowband components in noise [3], which is given as

$$y(t) = \frac{\sum_{i=0}^{2P} \rho^i a_i z^{-i}}{\sum_{i=0}^{2P} a_i z^{-i}} e(t). \tag{2}$$

This model has been widely used for deriving ANF algorithms. The LP coefficients a_i in (2) are mirror symmetric because the poles are constrained to lie on the unit circle, i.e., $a_i = a_{2P-i}$ and also $a_0 = a_{2P} = 1$. Moreover, the zeros are constrained to lie on the same radial lines as the poles, at a constant distance ρ from the origin ($0 \leq \rho < 1$). Note that ρ is defined as the pole radius or pole contraction factor since in the prediction error filter, corresponding to the inverse signal model, ρ appears in the denominator. Throughout this paper, ρ is assumed to be a fixed parameter, the choice of which is however of great importance to the frequency estimation performance. The residual signal $e(t)$ is usually assumed to be an uncorrelated sequence, such as a white noise sequence or a Dirac impulse. With the aim of achieving direct frequency estimation, the model in (2) is sometimes rewritten using a second-order sections cascade structure [4], [5],

$$y(t) = \left(\prod_{n=1}^{P} \frac{1 - 2\rho\cos\theta_n z^{-1} + \rho^2 z^{-2}}{1 - 2\cos\theta_n z^{-1} + z^{-2}} \right) e(t) \tag{3}$$

with $\theta_n \in [0,\pi]$ denoting the angles of the pole-zero pairs in the upper half of the z-plane.

2.2 Decoupled optimization

The goal of the proposed frequency estimation method is to have the angles θ_n in the constrained pole-zero signal model (3) converge to the frequencies ω_n of the observed signal in (1). To this end, a LS objective is defined as follows:

$$V(\vartheta) = \frac{1}{N} \sum_{t=1}^{N} e^2(t,\vartheta) \tag{4}$$

with, from (3),

$$e(t,\vartheta) = \left(\prod_{n=1}^{P} \frac{1 - 2\cos\theta_n z^{-1} + z^{-2}}{1 - 2\rho\cos\theta_n z^{-1} + \rho^2 z^{-2}} \right) y(t) \tag{5}$$

and

$$\vartheta = [\theta_1 \quad \dots \quad \theta_P]^T. \tag{6}$$

Instead of directly minimizing $V(\vartheta)$ w.r.t. the parameter vector ϑ, we divide the minimization problem into P subproblems. Let the intermediate residual signal $e_n(t,\vartheta_n)$ be defined as the output of the nth section of the prediction error filter cascade,

$$e_n(t,\vartheta_n) = \left(\prod_{l=1}^{n} \frac{1 - 2\cos\theta_l z^{-1} + z^{-2}}{1 - 2\rho\cos\theta_l z^{-1} + \rho^2 z^{-2}} \right) y(t) \tag{7}$$

with

$$\vartheta_n = [\theta_1 \quad \dots \quad \theta_n]^T \tag{8}$$

and $e_P(t,\vartheta_P) = e(t,\vartheta)$. Then the nth subproblem is defined as follows:

$$\min_{\theta_n} V_n(\vartheta_n) = \min_{\theta_n} \frac{1}{N} \sum_{t=1}^{N} e_n^2(t,\vartheta_n). \tag{9}$$

Note that the minimization in (9) is performed w.r.t. θ_n only, while the objective $V_n(\vartheta_n)$ depends on the entire vector ϑ_n. However, if the subproblems are solved consecutively, starting at $n = 1$, then in the nth subproblem, estimates for $\theta_1,\dots,\theta_{n-1}$ are available and only θ_n needs to be estimated. As a consequence, the subproblems are entirely decoupled and can be treated individually.

The solution to the nth subproblem is obtained iteratively using a line search optimization method [6, Ch. 3], i.e.,

$$\theta_n^{(k+1)} = \theta_n^{(k)} + \mu_k p^{(k)} \tag{10}$$

with $k \in \mathbb{N}$ the iteration index. The step length μ_k is determined using backtracking with Armijo's sufficient decrease condition [6, Ch. 3]. The search direction $p^{(k)}$ can be obtained with one of the following methods:

1. Steepest descent (SD):

$$p^{(k)} = -\frac{\partial}{\partial\theta_n} V_n\big(\hat{\vartheta}_n^{(k)}\big) \tag{11}$$

in which

$$\hat{\vartheta}_n^{(k)} = \Big[\hat{\theta}_1^{(\kappa_1)} \quad \dots \quad \hat{\theta}_{n-1}^{(\kappa_{n-1})} \quad \hat{\theta}_n^{(k)}\Big]^T \tag{12}$$

with κ_i, $i = 1,\dots,n-1$ the index of the final iteration in the ith subproblem, and k the current iteration index in the nth subproblem.

2. Gauss-Newton (GN):

$$p^{(k)} = -\frac{\Big(\frac{\partial}{\partial\theta_n}\mathbf{e}_n\big(\hat{\vartheta}_n^{(k)}\big)\Big)^T \mathbf{e}_n\big(\hat{\vartheta}_n^{(k)}\big)}{\Big(\frac{\partial}{\partial\theta_n}\mathbf{e}_n\big(\hat{\vartheta}_n^{(k)}\big)\Big)^T \Big(\frac{\partial}{\partial\theta_n}\mathbf{e}_n\big(\hat{\vartheta}_n^{(k)}\big)\Big)} \tag{13}$$

with

$$\mathbf{e}_n(\vartheta_n) = [e_n(1,\vartheta_n) \quad \dots \quad e_n(N,\vartheta_n)]^T. \tag{14}$$

3. Quasi-Newton with damped BFGS updating [6, Ch. 18] (BFGS) :

$$p^{(k)} = -B_k^{-1} \frac{\partial}{\partial\theta_n} V_n\big(\hat{\vartheta}_n^{(k)}\big) \tag{15}$$

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{u_k u_k^T}{u_k^T s_k} \tag{16}$$

with

$$s_k = \hat{\theta}_n^{(k+1)} - \hat{\theta}_n^{(k)} \tag{17}$$

$$v_k = \frac{\partial}{\partial\theta_n} V_n\big(\hat{\vartheta}_n^{(k+1)}\big) - \frac{\partial}{\partial\theta_n} V_n\big(\hat{\vartheta}_n^{(k)}\big) \tag{18}$$

$$u_k = \lambda v_k + (1-\lambda) B_k s_k \tag{19}$$

and

$$\lambda = \begin{cases} 1 & \text{if } s_k^T v_k \geq \gamma s_k^T B_k s_k \quad (20) \\ (1-\gamma)\dfrac{s_k^T B_k s_k}{s_k^T B_k s_k - s_k^T y_k} & \text{if } s_k^T v_k < \gamma s_k^T B_k s_k \quad (21) \end{cases}$$

The parameter $\gamma \in (0,1)$ is usually chosen as $\gamma = 0.2$ [8]. Since each CPZLP subproblem is a scalar optimization problem, the Hessian approximation B_k, the displacement vector s_k, the change of gradients vector v_k and its damped counterpart u_k are all scalars, and the general BFGS calculations in (16), (19)-(21) can be greatly simplified:

$$B_{k+1} = \max\Big(\frac{v_k}{s_k}, \gamma B_k\Big). \tag{22}$$

The iterative algorithm for solving the nth subproblem is terminated at iteration κ_n, either when

$$\left| p^{(\kappa_n)} \frac{\partial}{\partial \theta_n} V_n\big(\hat{\vartheta}_n^{(\kappa_n)}\big) \right| \leq \tau \tag{23}$$

with τ a specified tolerance, or when the maximum number of iterations is reached, i.e., $\kappa_n = k_{\max}$.

2.3 Gradient and Hessian calculation

The above methods for calculating the search direction $p^{(k)}$ in the line search algorithm in (10) require gradient and Hessian information, which can be calculated using either $\frac{\partial}{\partial \theta_n} V_n(\vartheta_n)$ (in the SD and BFGS methods) or $\mathbf{e}_n(\vartheta_n)$ and $\frac{\partial}{\partial \theta_n}\mathbf{e}_n(\vartheta_n)$ (in the GN method). These quantities can be calculated in an efficient manner as follows.

From the definition of $e_n(t,\vartheta_n)$ in (7) it follows that

$$e_n(t,\vartheta_n) = \frac{1-2\cos\theta_n z^{-1}+z^{-2}}{1-2\rho\cos\theta_n z^{-1}+\rho^2 z^{-2}} e_{n-1}(t,\vartheta_{n-1}) \tag{24}$$

with $e_0(t) = y(t)$. This filtering operation can be executed for $t = 1,\ldots,N$ provided that the initial filter states are known. For simplicity, we set $e_n(t,\vartheta_n) = e_{n-1}(t,\vartheta_{n-1}) = 0$ for $t \leq 0$. The intermediate residual signal vector $\mathbf{e}_n(\vartheta_n)$ can be constructed from $e_n(t,\vartheta_n)$, $t = 1,\ldots,N$ as in (14).

By differentiating both sides of (24) w.r.t. θ_n, we obtain

$$\frac{\partial}{\partial \theta_n} e_n(t,\vartheta_n) = \frac{2(1-\rho)\sin\theta_n z^{-1}(1-\rho z^{-2})}{(1-2\rho\cos\theta_n z^{-1}+\rho^2 z^{-2})^2} e_{n-1}(t,\vartheta_{n-1}) \tag{25}$$

which can again be calculated for $t = 1,\ldots,N$ by setting $\frac{\partial}{\partial \theta_n} e_n(t,\vartheta_n) = e_{n-1}(t,\vartheta_{n-1}) = 0$ for $t \leq 0$. The derivative vector $\frac{\partial}{\partial \theta_n}\mathbf{e}_n(\vartheta_n)$ is constructed using $\frac{\partial}{\partial \theta_n} e_n(t,\vartheta_n)$, $t = 1,\ldots,N$.

Finally, the gradient $\frac{\partial}{\partial \theta_n} V_n(\vartheta_n)$ can be calculated by differentiating (9), i.e.,

$$\frac{\partial}{\partial \theta_n} V_n(\vartheta_n) = \frac{2}{N}\sum_{t=1}^{N}\left(\frac{\partial}{\partial \theta_n} e_n(t,\vartheta_n)\right) e_n(t,\vartheta_n) \tag{26}$$

$$= \frac{2}{N}\left(\frac{\partial}{\partial \theta_n}\mathbf{e}_n(\vartheta_n)\right)^T \mathbf{e}_n(\vartheta_n). \tag{27}$$

The above quantities should be evaluated at $\vartheta_n = \hat{\vartheta}_n^{(k)}$ for calculating the search direction in (11), (13), and (15). It follows from (12) that this evaluation can be achieved by replacing θ_n with its current estimate $\hat{\theta}_n^{(k)}$ in the filter transfer functions in (24) and (25), and by evaluating $e_{n-1}(t,\vartheta_{n-1})$ at $\vartheta_{n-1} = \hat{\vartheta}_{n-1}^{(\kappa_{n-1})}$ (which has already been done in the final iteration of the $(n-1)$th subproblem).

3. COMPUTATIONAL COMPLEXITY

The operations that are performed in each iteration k of the nth subproblem in the CPZLP algorithm are summarized in Table 1, with reference to the relevant equations and with the number of multiplications as a measure for computational complexity. The number of backtracking steps needed until Armijo's sufficient decrease condition is satisfied in iteration k of subproblem n, is denoted by $\beta_{n,k}$.

The computational complexity of the entire CPZLP algorithm can then be calculated as follows. As an example, we derive the total number of multiplications M_{BFGS} when the BFGS method is applied. From Table 1, we have

$$M_{\text{BFGS}} = \sum_{n=1}^{P}\left(\sum_{k=1}^{\kappa_n}\big((13+3\beta_{n,k})N+(17+5\beta_{n,k})\big)\right) \tag{28}$$

$$= (13N+17)\sum_{n=1}^{P}\kappa_n + (3N+5)\sum_{n=1}^{P}\left(\sum_{k=1}^{\kappa_n}\beta_{n,k}\right) \tag{29}$$

$$= (13N+17)\bar{\kappa}P + (3N+5)\sum_{n=1}^{P}\kappa_n\bar{\beta}_n \tag{30}$$

with $\bar{\kappa}$ the average number of iterations per subproblem and $\bar{\beta}_n$ the average number of backtracking steps per iteration in subproblem n, i.e.,

$$\bar{\kappa} = \frac{1}{P}\sum_{n=1}^{P}\kappa_n, \quad \bar{\beta}_n = \frac{1}{\kappa_n}\sum_{k=1}^{\kappa_n}\beta_{n,k}. \tag{31}$$

Assuming that the average number of backtracking steps per iteration is the same for all subproblems, i.e., $\bar{\beta}_1 = \ldots = \bar{\beta}_P = \bar{\beta}$, the computational complexity for the three different methods can be written as

$$\boxed{\begin{aligned} M_{\text{SD}} &= \bar{\kappa}P\big[(13+3\bar{\beta})N+(14+5\bar{\beta})\big] \\ M_{\text{GN}} &= \bar{\kappa}P\big[(14+3\bar{\beta})N+(15+5\bar{\beta})\big] \\ M_{\text{BFGS}} &= \bar{\kappa}P\big[(13+3\bar{\beta})N+(17+5\bar{\beta})\big] \end{aligned}} \tag{32}$$

From the above expressions, it is clear that the computational complexity is linear w.r.t. both the frame length N and the number of sinusoids P. As a consequence of decoupling the problem into scalar subproblems, the GN and BFGS methods are not significantly more expensive than the SD algorithm, although they do take into account Hessian information in the optimization algorithm. The actual complexity depends on the average number of iterations and backtracking steps per iteration in the P subproblems. This leads to the peculiar observation that the fastest converging method will also have the lowest complexity, which is in contrast with the traditional trade-off between convergence speed and complexity.

4. SIMULATION RESULTS

Monte Carlo simulations were carried out to validate the performance of the CPZLP algorithm. The observed signal is a sum of $P = 3$ sinusoids, with amplitudes $[A_1,A_2,A_3] = [1,0.5,1.5]$, radial frequencies $[\omega_1,\omega_2,\omega_3] = [0.25,0.4,0.7]\pi$, and phases $[\phi_1,\phi_2,\phi_3] = [0,0.8,1.5]\pi$. The pole radius is fixed to $\rho = 0.95$, which appears to be an optimal value for most sinusoidal frequency estimation problems [7]. The optimization algorithm parameters are set as recommended in [6]: the initial step length is $\mu_k^{(0)} = 1$, the contraction factor determining the step length $\mu_k^{(m)} = \eta^m \mu_k^{(0)}$ in the mth backtracking step is $\eta = 0.9$, the scaling factor determining Armijo's sufficient decrease condition is $c = 10^{-4}$, Powells parameter in the damped BFGS update is $\gamma = 0.2$ [8], the termination criterion tolerance is $\tau = 10^{-6}$, and the maximum number of iterations per subproblem is $k_{\max} = 30$. The initial estimate $\hat{\theta}_n^{(0)} = \pi/3$ is chosen equal for all three subproblems, to illustrate the sensitivity of the algorithm w.r.t. the choice of initial conditions. An additional rescue procedure is implemented, which restarts the iterative procedure for subproblem n with a different initial estimate if $\kappa_n = k_{\max}$. When after five rescue restarts, subproblem n still remains unsolved, we set $\hat{\theta}_n = \pi/2$ and continue with subproblem $n+1$.

The CPZLP algorithm is evaluated w.r.t. frequency bias and frequency variance, defined as ($n = 1,2,3$)

$$\text{bias}\big(\hat{\theta}_n^{(\kappa_n)}\big) = E\big\{\hat{\theta}_n^{(\kappa_n)}\big\} - \omega_n \tag{33}$$

$$\text{var}\big(\hat{\theta}_n^{(\kappa_n)}\big) = E\big\{(\hat{\theta}_n^{(\kappa_n)} - \omega_n)^2\big\}. \tag{34}$$

The expectation operator $E\{\cdot\}$ in (33)-(34) is approximated by averaging over 100 simulation runs, with different realizations of the Gaussian white noise signal $r(t)$. The CPZLP algorithm was found to produce approximately unbiased frequency estimates for $N \geq 256$ and SNR ≥ 0 dB (SD), $N \geq 512$ and SNR ≥ 25 dB (GN), and $N \geq 512$ and SNR ≥ 10 dB (BFGS).

The frequency variance is displayed in Figs. 1(a)-(c) as a function of different frame lengths $N \in [64,8192]$, with SNR = 15 dB. The CRLB for estimating ω_n, $n = 1,2,3$, from the true signal model in (1) is also shown in Fig. 1, and was calculated under the assumption that the sinusoidal frequencies are not near 0 and π [9, Ch.

Table 1: CPZLP complexity comparison: number of multiplications in iteration k of subproblem n

calculation of	Eqs.	SD	GN	BFGS
gradient	(24)-(26)	$10N+6$	$10N+6$	$10N+6$
Hessian	(13),(17),(18),(22)	0	N	2
search direction	(11),(13),(15)	1	2	2
termination criterion	(23)	1	1	1
step length	[6, p. 37]	$(1+\beta_{n,k})(3N+5)$	$(1+\beta_{n,k})(3N+5)$	$(1+\beta_{n,k})(3N+5)$
parameter estimate	(10)	1	1	1
TOTAL		$(13+3\beta_{n,k})N+(14+5\beta_{n,k})$	$(14+3\beta_{n,k})N+(15+5\beta_{n,k})$	$(13+3\beta_{n,k})N+(17+5\beta_{n,k})$

3] and sufficiently separated from each other [7]. In this case, the Fisher information matrix is diagonal and the CRLBs for the different frequencies are independent and equal to [7], [9, Ch. 3]

$$\mathrm{CRLB}(\omega_n) = \frac{6}{N(N+1)(2N+1)\mathrm{SNR}_n} \tag{35}$$

with $\mathrm{SNR}_n = A_n^2/(2\sigma_r^2)$ and σ_r^2 the noise variance. It can be seen that with the GN method, only $\mathrm{var}(\hat{\theta}_1^{(\kappa_1)})$ comes close to the CRLB, which is probably due the proximity of $\hat{\theta}_1^{(0)}$ to ω_1 and the relatively good SNR_1. The BFGS method performs much better, with all three variance curves staying near the CRLB for $N \geq 512$. Figs. 1(d)-(f) show the frequency variance versus $\mathrm{SNR} \in [-10, 40]$ dB, with $N = 2048$. With the GN and BFGS methods there is a clear treshold effect, i.e., the variance suddenly drops for $\mathrm{SNR} \geq 25$ dB (GN) and $\mathrm{SNR} \geq 15$ dB (BFGS). In favorable estimation conditions the treshold effect can occur at SNR as low as 5 dB (which is illustrated by the $\mathrm{var}(\hat{\theta}_1^{(\kappa_1)})$ curve in Figs. 1(e)-(f)).

To have an idea of the actual computational complexity, the required number of iterations κ_n and the average number of backtracking steps per iteration $\bar{\beta}_n$ are plotted for $n = 1, 2, 3$ as a function of N and SNR in Fig. 2. It is clear that the SD method is not suited for the frequency estimation problem under consideration. The GN method requires more iterations than the BFGS method, but due to the fact that GN consistently produces estimates that meet Armijo's sufficient decrease condition without backtracking ($\bar{\beta}_n \equiv 0$), it is computationally cheaper than BFGS.

5. CONCLUSION

We have presented a new parametric frequency estimation method for multiple real sinusoids corrupted by noise. The so-called CPZLP algorithm provides frame-based frequency estimation by optimizing the parameters of a cascade of second-order constrained pole-zero filter sections in a decoupled and consecutive fashion. Each of the unknown frequencies is estimated using a line search optimization algorithm, which has been implemented with three popular line search methods (SD, GN, and BFGS). The computational complexity of the CPZLP algorithm is linear w.r.t. both the number of sinusoids and the frame length, such that long data frames can be used and hence noise robustness is increased. Monte Carlo simulation results show that the BFGS method is particularly promising, since it provides unbiased and near-optimal frequency estimates for frame lengths larger than 512 samples and SNR as low as 5 dB in favorable estimation conditions and 15 dB in worse conditions. Since the required number of iterations and backtracking steps has a profound effect on the actual complexity, the faster converging GN and BFGS methods are computationally much more interesting than the SD method. Further work [7] includes an extension of the CPZLP algorithm to multi-pitch estimation and an approach to joint order and frequency estimation.

REFERENCES

[1] H. C. So, K. W. Chan, Y. T. Chan, and K. C. Ho, "Linear prediction approach for efficient frequency estimation of multiple real sinusoids: algorithms and analyses," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2290–2305, July 2005.

[2] S. M. Kay, "The effects of noise on the autoregressive spectral estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 5, pp. 478–485, Oct. 1979.

[3] A. Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 4, pp. 983–996, Aug. 1985.

[4] M. V. Dragošević and S. S. Stanković, "An adaptive notch filter with improved tracking properties," *IEEE Trans. Signal Process.*, vol. 43, no. 9, pp. 2068–2078, Sept. 1995.

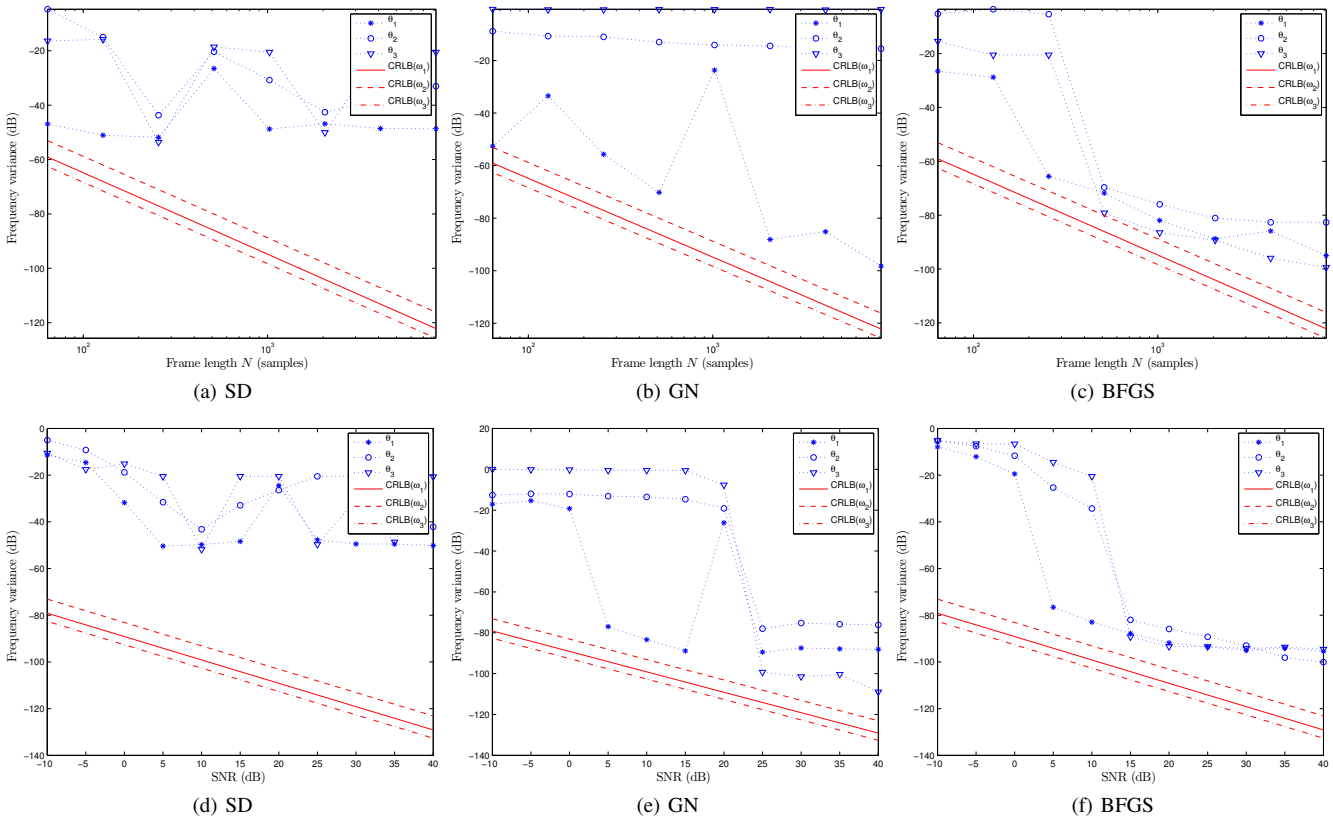
[5] G. Li, "A stable and efficient adaptive notch filter for direct frequency estimation," *IEEE Trans. Signal Process.*, vol. 45, no. 8, pp. 2001–2009, Aug. 1997.

[6] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York: Springer, 2006.

[7] T. van Waterschoot, M. Diehl, and M. Moonen, "Constrained pole-zero linear prediction: Optimization of cascaded biquadratic notch filters for multi-tone and multi-pitch estimation," ESAT-SISTA Technical Report TR 07-115, Katholieke Universiteit Leuven, Belgium, Feb. 2008.

[8] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculation," in *Proc. 7th Biennal Conf. Numerical Analysis*, Dundee, UK, G. A. Watson, Ed. Berlin: Springer-Verlag, 1977, pp. 144–157.

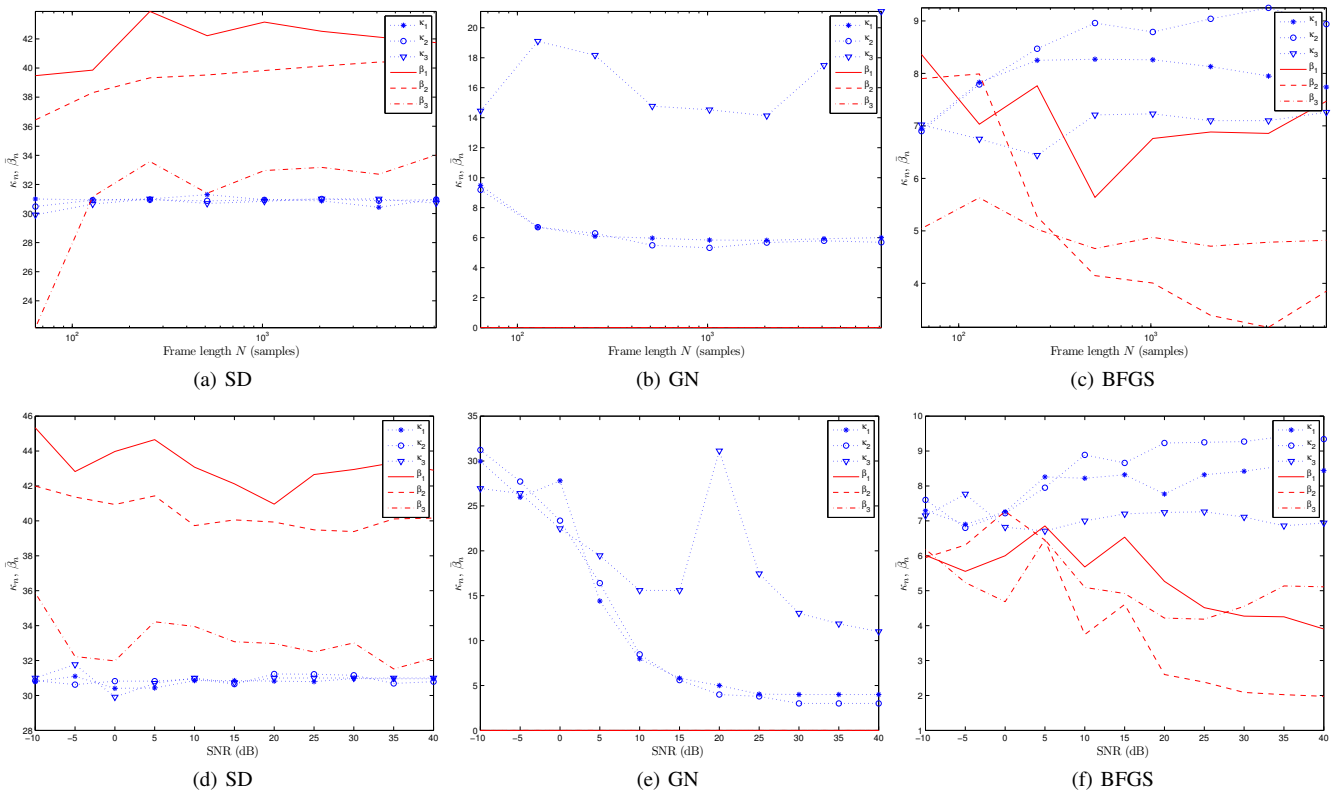
[9] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, New Jersey: Prentice-Hall, 1993.



(a) SD (b) GN (c) BFGS

(d) SD (e) GN (f) BFGS

Figure 1: CPZLP frequency variance and CRLB, (a)-(c) versus frame length N, (d)-(f) versus SNR.



(a) SD (b) GN (c) BFGS

(d) SD (e) GN (f) BFGS

Figure 2: CPZLP number of iterations κ_n and average number of backtracking steps per iteration $\bar{\beta}_n$, (a)-(c) versus N, (d)-(f) versus SNR.