An Iterative Algorithm for Single-Frequency Estimation

Tyler Brown, *Member, IEEE*, and Michael Mao Wang, *Member, IEEE*

***Abstract*—An algorithm for the estimation of the frequency of a complex sinusoid in noise is proposed. The estimator consists of multiple applications of lowpass filtering and decimation, frequency estimation by linear prediction, and digital heterodyning. The estimator has a significantly reduced threshold relative to existing phase-based algorithms and performance close to that of maximum likelihood estimation. In addition, the mean-squared error performance is within 0.7 dB of the Cramér-Rao bound (CRB) at signal-to-noise ratios (SNRs) above threshold. Unlike many autocorrelation and phase-based methods, the proposed algorithm's performance is uniform across a frequency range of $-\pi$ to π. The computational complexity of the algorithm is shown to be favorable compared with maximum likelihood estimation via the fast Fourier transform (FFT) algorithm when significant zero-padding is required.**

I. Introduction

THE problem of estimating the frequency of a complex exponential in additive white noise from a set of samples is a fundamental and well-studied problem in estimation theory. Applications include radar, array signal processing, and frequency synchronization of communication systems. Estimation is typically performed on a signal of the form

$$x(n) = Ae^{j(\omega n+\varphi)} + z(n) \quad n = 0, 1, \ldots, N-1 \tag{1}$$

where $\omega, -\pi \leq \omega < \pi$ is the frequency, and A and φ, $-\pi \leq \varphi < \pi$ are unknown constants. The noise z is a zero-mean complex white Gaussian process with $z(n) = z_r(n) + jz_i(n)$. Its components $z_r(n)$ and $z_i(n)$ are real, uncorrelated, zero-mean Gaussian random variables with variance $\sigma^2/2$ (σ^2 is the variance of $z(n)$). Maximum likelihood (ML) estimation was studied by Rife and Boorstyn in [1], where the ML estimate was shown to be of the form

$$\hat{\omega} = \underset{w'}{\operatorname{argmax}} \left| \sum_{n=0}^{N-1} x(n) e^{-j\omega' n} \right|^2. \tag{2}$$

The ML estimate was shown in [1] to be unbiased at sufficiently high SNR $= A^2/\sigma^2$ and to exhibit the threshold effect typical of nonlinear estimators: Below an SNR threshold, the mean-squared estimation error increases rapidly. Above the threshold SNR, the mean-squared error (mse) approaches the Cramér–Rao bound (CRB) given by

$$\sigma^2_{CR} = \frac{6}{\left(\frac{A^2}{\sigma^2}\right) N(N^2-1)}. \tag{3}$$

The mean-squared error of the ML estimator is independent of the true frequency, except at the extreme band edges. The FFT algorithm may be used to approximate (2)

$$\hat{\omega} = \frac{2\pi}{N'} \underset{-N'/2 \leq m < N'/2}{\operatorname{argmax}} \left| \sum_{n=0}^{N-1} x(n) e^{-j(2\pi mn/N')} \right|^2. \tag{4}$$

However, padding is often required, i.e., $N' > N$, to obtain sufficient resolution. In such cases, the algorithm's complexity ($(N'/2)\log_2 N'$ complex multiplications) can be large.

A number of suboptimal algorithms that avoid the complexity of the ML estimator have been proposed. One class of such algorithms use the signal's sample autocorrelation function

$$\hat{r}(l) = \sum_{n=l}^{N-1} x^*(n-l)x(n) \tag{5}$$

as a basis for the frequency estimate. The simplest of these was proposed by Lank *et al.* [2] and is equal to the argument of accumulated lag l products

$$\hat{\omega} = \frac{1}{l} \arg \hat{r}(l). \tag{6}$$

For $l = 1$, this approach can be interpreted as a linear predictor estimator [3]. While the complexity of this approach is low, it does not attain the CRB, even at high SNR. Other autocorrelation-based estimators were proposed in [4] and [5]. In [4], the estimator is of the form

$$\hat{\omega} = \frac{\sum_{l=1}^{J} l \arg\left(\hat{r}(l)\right)}{\sum_{l=1}^{J} l^2} \tag{7}$$

where $J < K-1$ is a parameter of the algorithm. The algorithm in [5] interchanges the order of the argument and summation to give the estimator

$$\hat{\omega} = \frac{2}{\pi(J+1)} \arg\left(\sum_{l=1}^{J} \hat{r}(l) \right) \tag{8}$$

where, again, $J < K-1$ is a parameter of the algorithm. These estimators, rather than degrading rapidly below a threshold, show a smooth loss in performance as SNR is reduced, whereas

Manuscript received June 22, 2001; revised July 8, 2002. The associate editor coordinating the review of this paper and approving it for publication was Dr. Chong-Yung Chi.

The authors are with the Motorola Inc., Arlington Heights, IL 60004 USA (e-mail: Tyler.Brown@motorola.com; wangmm@cig.mot.com).

Digital Object Identifier 10.1109/TSP.2002.804096.

at high SNR, they exhibit small losses relative to the CRB. Their main disadvantage, however, is their limited frequency range. At frequencies outside of π/J and $2\pi/(J+1)$, respectively, the algorithms in [4] and [5] fail.

A second set of suboptimal algorithms operate on the signal's phase. Tretter proposed unwrapping the signal phase and performing linear regression to obtain a frequency estimate [6]. This approach was shown to approach the CRB at high SNR and is relatively simple computationally. The difficulty, however, is in the moderate-to-low SNR region where the phase unwrapping process becomes prone to errors. Kay addressed the phase unwrapping problem by only considering the phase differences [7]

$$\Delta(n) = \arg(x^*(n-1)x(n)) \quad n = 1, \ldots, N-1. \quad (9)$$

Kay shows that for large SNR, these differences may be expressed as

$$\Delta(n) = \omega + u(n) \quad (10)$$

where u is a zero-mean colored Gaussian process. This is a linear model whose minimum variance unbiased estimator is of the form

$$\hat{\omega} = \sum_{n=1}^{N-1} w(n)\Delta(n) \quad (11)$$

where $w(n)$ are weights derived from the correlation of the noise terms u. Like the ML estimator, this computationally simple estimator reaches the CRB for SNRs above a threshold. The threshold is larger than that observed with ML estimation (MLE), however, and therefore, performance at low SNR is substantially worse than the ML algorithm. In addition, at frequencies approaching π, the phase differences themselves wrap, causing large increases in mean-squared error. The estimator therefore performs poorly at frequencies near half the sampling frequency. Lang and Musicus [8] showed that the Kay estimator can be obtained from the Tretter estimator by summation by parts.

The threshold performance of Kay's method can be improved by filtering the received signal with a rectangular filter of length 2, thereby increasing the SNR by 3 dB [9]. In a manner similar to that used by Kay, the minimum variance unbiased estimator is applied to consecutive phase differences to obtain the frequency estimate. This method can be generalized to filtering with a rectangular filter of length M to give a gain of $10\log_{10}(M)$. As with the autocorrelation-based algorithms of (7) and (8), this performance improvement comes at the price of limited frequency range. Namely, the frequency range is limited to π/M. This limitation was addressed in [10] and [11]. In these approaches, a bank of filters are used to obtain a coarse estimate of the frequency. Effectively, this coarse estimate is used to reduce the received signal's frequency to within a small range. Lowpass filtering can then be used as in [9] to increase the SNR. Kay's phase-based estimation is then used to obtain the fine estimate. The final frequency estimate is just the sum of the coarse and fine frequencies. In [10], the filterbank is implemented with a DFT matrix, whereas in [11], cascaded stages of quadrature mirror filters are used. These methods yield lower thresholds than those obtained with Kay's method and may be attractive alternatives to direct computation of the likelihood function via the FFT. However, their threshold performance can vary widely, depending on the location of the frequency relative to the filterbank's response. Unlike the degradations at the band edges exhibited by Kay's method, degradation in these methods is periodic across the band, being related to the frequency responses of the bank's filters. In addition, these methods offer only modest improvements over Kay's method when the number of sample points is small. A 1-dB reduction in threshold was reported in [11] at $\omega = 0$ for a record length of 24. Unlike Kay's method, which attains the CRB at high SNR for all frequencies, the four channel filterbank (FCFB) method of [11] suffers a degradation of up to 0.9 dB, depending on frequency, with respect to the CRB.

This paper presents a computationally efficient iterative algorithm with threshold performance close to that of the ML estimator. Its performance is insensitive to frequency and has asymptotic performance within 0.5 to 0.74 dB of the CRB, depending on record length. The algorithm is based on the repeated use of an autocorrelation-based frequency estimator. In particular, the algorithm uses first-order linear prediction (i.e., Lank's algorithm) to form an initial estimate. The signal is then translated in frequency by this estimate to yield a new signal whose frequency is the negative of the error in the first estimate and therefore confined to a relatively narrow band around zero. Filtering is performed by accumulating samples, and the result is decimated to give a new signal with fewer samples. The process is iterated with the final estimate being the sum of the intermediate estimates modulo the initial sampling rate.

This method therefore performs multiple iterations of the frequency translation/filtering technique of [10] and [11]. More importantly, however, intermediate frequency estimates are obtained by a simple frequency estimator rather than a detection process based on a filterbank's output. This alleviates the frequency dependence due to the boundaries between channels. The technique of iteratively translating a signal in frequency to a more optimal range was suggested in [12]. However, there, the goal was the estimation of the frequencies of closely spaced sinusoids in noise, and no filtering was applied between iterations.

Section II describes the proposed estimator in two parts. First, the lag product estimator that is the building block of the iterative algorithm is described and its performance analyzed. Examination of this performance is used to motivate the iterative algorithm, which is described next. Section III describes the asymptotic performance of the algorithm and compares this performance with the CRB. Computational complexity is addressed in Section IV, where a complexity analysis of the proposed algorithm is presented along with a comparison of other estimators. Section V presents simulated results comparing the proposed algorithm with MLE, the FCFB algorithm, and Kay's algorithm.

II. Proposed Method

A. Basic Algorithm

The proposed algorithm consists of iteratively applying a basic frequency estimation algorithm. Before describing the

iteration process, this basic algorithm, which is based on the linear predictor estimator of (6) with $l = 1$, will be described. In this algorithm, the received data sequence x is first lowpass filtered with a rectangular filter with coefficients $h_i = 1$, $i = 0, 1, 2, \ldots, M-1$, and the result is decimated by M to give the new sequence v_M

$$v_M(n) = \sum_{m=0}^{M-1} x(nM+m), \quad n = 0, 1, \ldots, L-1 \tag{12}$$

where $L = N/M$ is assumed to be integer. The frequency estimate is then

$$\hat{\omega}(M) = \frac{1}{M} \arg\left(\sum_{n=1}^{L-1} v_M^*(n-1) v_M(n) \right). \tag{13}$$

Accumulating M samples decreases the error of the frequency estimate at the expense of a reduction in the estimator's frequency range. Namely, the range is reduced from π to π/M. As will be seen when the iteration process is described, values of $M > 1$ will only be used when the frequency to be estimated is very likely to be confined to the interval $[-\pi/M, \pi/M)$.

Two properties of the estimator's error will now be derived. The first property involves the decomposition of the error into the sum of continuous and discrete valued terms. Using (1) in (12) gives

$$v_M(n) = C_M(\omega) e^{j\omega Mn} + n_v(n) \tag{14}$$

where

$$C_M(\omega) = MA e^{jM\omega/2} \mathrm{sinc}_M\left(\frac{\omega}{2}\right) \tag{15}$$

$$\mathrm{sinc}_M(x) \triangleq \frac{\sin(Mx)}{M \sin(x)} \tag{16}$$

and n_v is a zero-mean complex white Gaussian process with independent real and imaginary parts and variance

$$E\left[n_v^*(n) n_v(n)\right] = \sigma_v^2 = M\sigma^2. \tag{17}$$

Substituting (14) into (13) gives

$$\hat{\omega}(M) = \frac{1}{M} \arg\left(\sum_{n=1}^{L-1} \left[C_M^*(\omega) e^{-j\omega M(n-1)} + n_v(n-1) \right] \times \left[C_M(\omega) e^{j\omega Mn} + n_v(n) \right] \right). \tag{18}$$

In Appendix A, it is shown that $\hat{\omega}(M)$ is of the form

$$\hat{\omega}(M) = \frac{1}{M} \arg\left(e^{j(M\omega+\alpha)} \right) = \frac{1}{M} (M\omega + \alpha)_{-\pi}^{\pi} \tag{19}$$

where α is a random variable distributed on $[-\pi, \pi)$, and the notation $(x)_{-a/2}^{a/2}$ is the modulo of x taken on $[-a/2, a/2)$, i.e., $(x)_{-a/2}^{a/2} = \mathrm{mod}_a(x + a/2) - a/2$. Using the identity

$$x = (x)_{-a/2}^{a/2} + a \left\lfloor \frac{x + \frac{a}{2}}{a} \right\rfloor \tag{20}$$

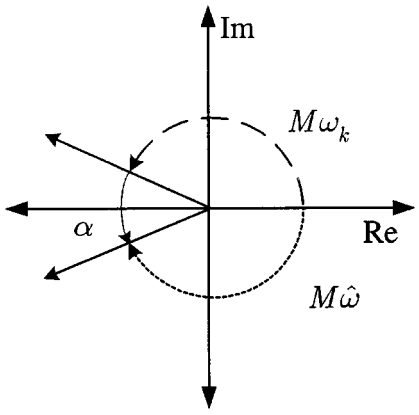

Fig. 1. Relationship between $M\omega$, $M\hat{\omega}(M)$, and α when phase wrapping occurs. In this example, $m = -1$, i.e., $M\hat{\omega}(M) = M\omega + \alpha - 2\pi$.

the error $e = \hat{\omega}(M) - \omega$ is then

$$e = \frac{1}{M}(M\omega + \alpha)_{-\pi}^{\pi} - \omega = \frac{\alpha}{M} + \frac{2m\pi}{M} \tag{21}$$

where $|m|$ is the number of times the phase $M\omega + \alpha$ wraps across the $-\pi/\pi$ boundary

$$m = -\left\lfloor \frac{M\omega + \alpha + \pi}{2\pi} \right\rfloor. \tag{22}$$

The notation $\lfloor x \rfloor$ represents the greatest integer less than or equal to x. The relationship between ω, α, and $\hat{\omega}(M)$ is illustrated in Fig. 1. The error in $\hat{\omega}(M)$ can be seen to be decomposed into two parts: a continuous-valued unwrapped error term that takes values in the interval $[-\pi/M, \pi/M)$, and a discrete-valued error term that is a multiple of $2\pi/M$ and is due to phase wrapping. As long as no phase wrapping occurs, the estimation error will therefore be less than π/M in magnitude.

A second property of the estimator is that its error is bounded by

$$|e| < \left(1 + \frac{1}{M}\right)\pi. \tag{23}$$

This may be obtained by observing that the estimates are restricted to the interval $[-\pi/M, \pi/M)$ and $\omega \in [-\pi, \pi)$.

Appendix A shows that the estimator is zero-mean with variance

$$\mathrm{var}[\hat{\omega}(M)] = \frac{\sigma_v^2}{M^2(L-1)^2 |C_M(\omega)|^2} \left(1 + \frac{(L-1)\sigma_v^2}{2|C_M(\omega)|^2}\right) \tag{24}$$

under the assumptions that 1) no phase-wrapping errors occur, and 2) the SNR A^2/σ^2 satisfies the condition

$$\frac{A^2}{\sigma^2} ? \frac{1}{(N-M)^{\frac{1}{2}} \left|\mathrm{sinc}_M\left(\frac{\omega}{2}\right)\right|^2}. \tag{25}$$

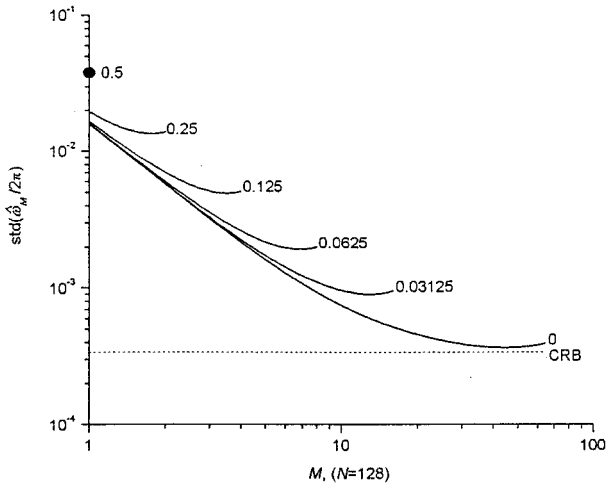

Fig. 2. Standard deviation (std) of frequency estimation error versus M for various frequencies $(\omega/2\pi = 0, 0.03125, 0.0625, 0.1250.25, 0.5)$ with $N = 128$ and SNR $= 0$ dB).

Using $L = N/M$, (15), and (17) in (24) yields an alternative form for the estimator variance:

$$\text{var}[\hat{\omega}(M)] = \frac{1}{M\left(\frac{A^2}{\sigma^2}\right)(N-M)^2 \text{sinc}_M^2\left(\frac{\omega}{2}\right)} \times \left(1 + \frac{(N-M)}{2M^2\left(\frac{A^2}{\sigma^2}\right)\text{sinc}_M^2\left(\frac{\omega}{2}\right)}\right). \quad (26)$$

This form is useful for understanding the motivation for the proposed algorithm. A plot of (26) as a function of M for various ω is shown in Fig. 2 for $N = 128$ and an SNR of 0 dB. Each curve in the plot represents a different frequency. For each frequency, the standard deviation (std) of the error is plotted for $M < \pi/\omega$. The value $\lfloor \pi/\omega \rfloor$ is the largest filter length that, after decimation, yields phase changes between samples of less than π. Using larger values of M will lead to phase changes between adjacent samples of greater than π. For $\omega/2\pi = 0.5$, no decimation is possible giving the single point at $M = 1$, where the standard deviation is 0.04. For $\omega/2\pi = 0.25$, decimation by a factor of two is possible with a resulting standard deviation of about 0.01. This continues until the $\omega = 0$ line, which is plotted through $M = 64$, which is the largest decimation factor possible for $N = 128$ since at least two samples are needed to calculate the frequency estimate. The standard deviation here is seen to be only slightly above the CRB. Note that for any frequency, the choice of M that minimizes estimation variance is slightly below the maximum value that could be used without incurring phase wrapping.

B. Iterated Algorithm

The previously described method can be used in an iterative fashion to provide frequency estimation with near- uniform performance across the range of frequencies from $-\pi$ through π. In addition, at high SNR, the estimation mean-squared error approaches the CRB. Before describing the algorithm in detail, an example that illustrates the fundamental idea will be presented.

Consider frequency estimation with the previously described algorithm for the parameters of Fig. 2, i.e., $N = 128$ received samples and SNR$= 0$ dB. Since the frequency may not be in the range $|\frac{\omega}{2\pi}| < 0.25$, it is not safe to perform filtering and decimation, i.e, estimation with the previously described algorithm can only be applied with $M = 1$. From Fig. 2, the error will, depending on frequency, have a standard deviation between about 0.01 and 0.04. This initial estimate can then be used to remove the unknown frequency of the received samples. This is done by digitally heterodyning the received samples with a complex tone with frequency equal to the negative of the initial estimate. The result of this frequency translation is a residual signal whose frequency is random with zero mean and standard deviation of no more than 0.04. Frequency estimation is then performed on this signal but with filtering and decimation by $M = 2$. This is safe to do as long as the phase does not increase or decrease by more than π rad in two samples, which is equivalent to saying that the residual signal's frequency is less than 0.25 in magnitude. Since this signal's frequency is centered at zero with a standard deviation of no more than 0.04, this is very likely to be the case. From Fig. 2, for $M = 2$ and a frequency less than 0.25, the standard deviation of the error will be less than 0.02. At this point, an estimate of ω can be obtained by adding the initial estimate obtained with $M = 1$ and the estimate of the translated data obtained with $M = 2$. This estimate can then be used to translate the original data resulting in a signal whose frequency is again random but with standard deviation of less than 0.02. This process continues with M doubling at each iteration until $M = 32$, where estimation is performed on a signal whose frequency is distributed tightly around zero. Again, referring to Fig. 2, this last stage of estimation is seen to have error very near the CRB. The algorithm's frequency estimate of ω is then just the sum of the frequency estimates obtained with $M = 1, 2, 4, \ldots, 32$.

The proposed algorithm, which will be termed iterative linear prediction (ILP), is illustrated in Figs. 3 and 4. For notational reasons, define the sequence $\tilde{x}_0$ as the set of original received samples

$$\tilde{x}_0(n) \triangleq x(n) \quad n = 0, 1, \ldots, N-1. \quad (27)$$

Form an initial estimate $\hat{\omega}_0$ with error e_0 from $\tilde{x}_0$ using $M = 1$ in the previously described algorithm:

$$\begin{aligned}\hat{\omega}_0 &= \arg\left[\sum_{n=1}^{N-1} \tilde{x}_0^*(n-1)\tilde{x}_0(n)\right] \\ &= \omega + e_0. \end{aligned} \quad (28)$$

This estimate is used to translate the received samples in frequency to give the residual signal

$$\begin{aligned}\tilde{x}_1(n) &= e^{-j\hat{\omega}_0 n}x(n) \\ &= e^{j(\omega-\hat{\omega}_0)n} + z(n)e^{-j\hat{\omega}_0 n} \\ &= e^{j\Delta\omega_1 n} + z(n)e^{-j\hat{\omega}_0 n} \end{aligned} \quad (29)$$

where

$$\Delta\omega_1 \triangleq (-e_0)_{-\pi}^{\pi} \quad (30)$$

can be interpreted as the frequency of $\tilde{x}_1$. From the results of Section II-A, the error e_0 may represented as $e_0 = \alpha_0 + 2\pi m_0$,

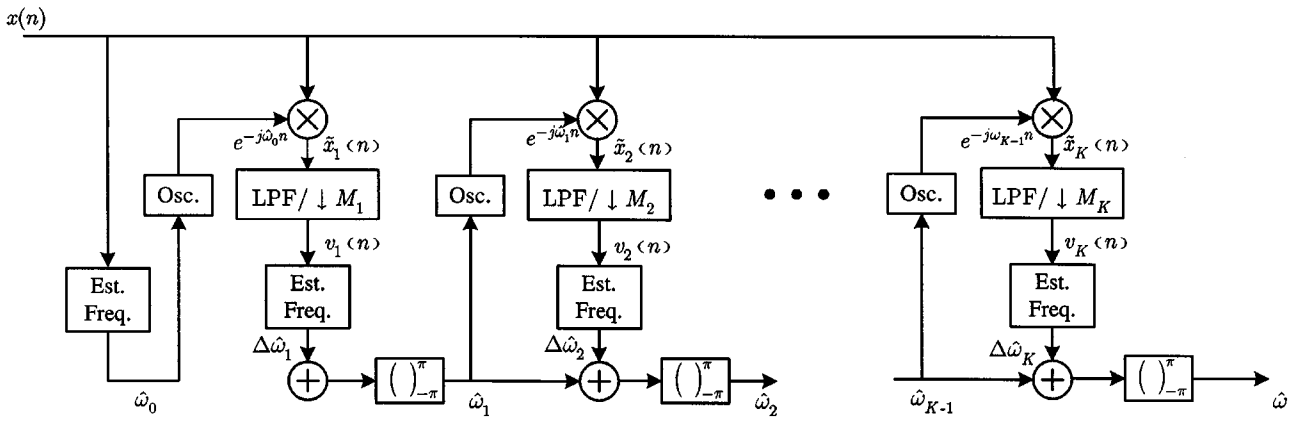

Fig. 3. Block diagram of proposed estimator.

$-\pi \leq \alpha < \pi$, and therefore, the frequency of the first residual signal is equal to the negative of the initial estimate's unwrapped error $\Delta\omega_1 = -\alpha_0$. This is true regardless of whether phase wrapping occurred, i.e., regardless of the value of m_0, and therefore, the error does not increase as ω approaches $\pm\pi$. At sufficiently high SNR , this unwrapped error will be small, and therefore, the residual signal's frequency will be confined to a relatively narrow interval about zero. The residual signal $\tilde{x}_1$ is then filtered by summing M_1 consecutive samples and decimating by M_1 to give the sequence

$$v_1(n) = \sum_{m=0}^{M_1-1} \tilde{x}_1(M_1 n + m) \tag{31}$$

where N/M_1 is assumed to be integer. This sequence is then used to form an estimate of $\Delta\omega_1$

$$\begin{aligned} \Delta\hat{\omega}_1 =& \frac{1}{M_1} \arg\left[\sum_{n=1}^{N/M_1-1} v_1^*(n-1)v_1(n)\right] \\ =& \Delta\omega_1 + e_1. \end{aligned} \tag{32}$$

A refined estimate of ω, $\hat{\omega}_1$ is then formed by summing $\hat{\omega}_0$ and $\Delta\hat{\omega}_1$ and taking the result modulo 2π:[1]

$$\hat{\omega}_1 = (\hat{\omega}_0 + \Delta\hat{\omega}_1)_{-\pi}^{\pi}. \tag{33}$$

Substituting (32) into (33) yields

$$\hat{\omega}_1 = (\hat{\omega}_0 + \Delta\omega_1 + e_1)_{-\pi}^{\pi}. \tag{34}$$

Using (28) and (30) in (34) gives

$$\begin{aligned} \hat{\omega}_1 =& (\omega + e_1 + 2m_0\pi)_{-\pi}^{\pi} \\ =& (\omega + e_1)_{-\pi}^{\pi}. \end{aligned} \tag{35}$$

After one iteration, we see that as long as $|\omega + e_1| < \pi$, the estimation error is e_1. This error, however, is the error in estimating the frequency of the first residual signal $\Delta\omega_1$. As long as this frequency is sufficiently small to prevent phase-wrapping errors during decimation, the mean-square of e_1 will be less than the mean-square of e_0. This process then continues and, barring phase-wrapping errors, the mean-squared error reduces at each iteration.

Induction may be used to show that the frequency estimate at iteration k is

$$\hat{\omega}_k = (\omega + e_k)_{-\pi}^{\pi}. \tag{36}$$

Assume that the frequency estimate at iteration $k-1$ is

$$\hat{\omega}_{k-1} = (\omega + e_{k-1})_{-\pi}^{\pi}. \tag{37}$$

Translating with this estimate then gives

$$\begin{aligned} \tilde{x}_k(n) =& e^{-j\hat{\omega}_{k-1}n} x(n) \\ =& e^{j(\omega-\hat{\omega}_{k-1})n} + z(n)e^{-j\hat{\omega}_{k-1}n} \\ =& e^{j\Delta\omega_k n} + z(n)e^{-j\hat{\omega}_{k-1}n} \end{aligned} \tag{38}$$

where

$$\Delta\omega_k \triangleq (\omega - \hat{\omega}_{k-1})_{-\pi}^{\pi}. \tag{39}$$

Substituting (37) into (39) gives

$$\Delta\omega_k = (-e_{k-1})_{-\pi}^{\pi}. \tag{40}$$

Filter and decimate the result to give the residual sequence v_k

$$v_k(n) = \sum_{m=0}^{M_k-1} \tilde{x}_k(M_k n + m) \quad n = 0, 1, \ldots, \frac{N}{M_k - 1}. \tag{41}$$

The frequency of v_k is then estimated with error

$$\begin{aligned} \Delta\hat{\omega}_k =& \frac{1}{M_k} \arg\left[\sum_{n=1}^{N/M_k-1} v_k^*(n-1)v_k(n)\right] \\ =& \Delta\omega_k + e_k. \end{aligned} \tag{42}$$

Using (40) in (42) gives

$$\Delta\hat{\omega}_k = (-e_{k-1})_{-\pi}^{\pi} + e_k. \tag{43}$$

The frequency estimate at iteration k, by definition, is

$$\hat{\omega}_k = (\Delta\hat{\omega}_k + \hat{\omega}_{k-1})_{-\pi}^{\pi}. \tag{44}$$

[1]Practically, the modulo operation is only needed in the last iteration.

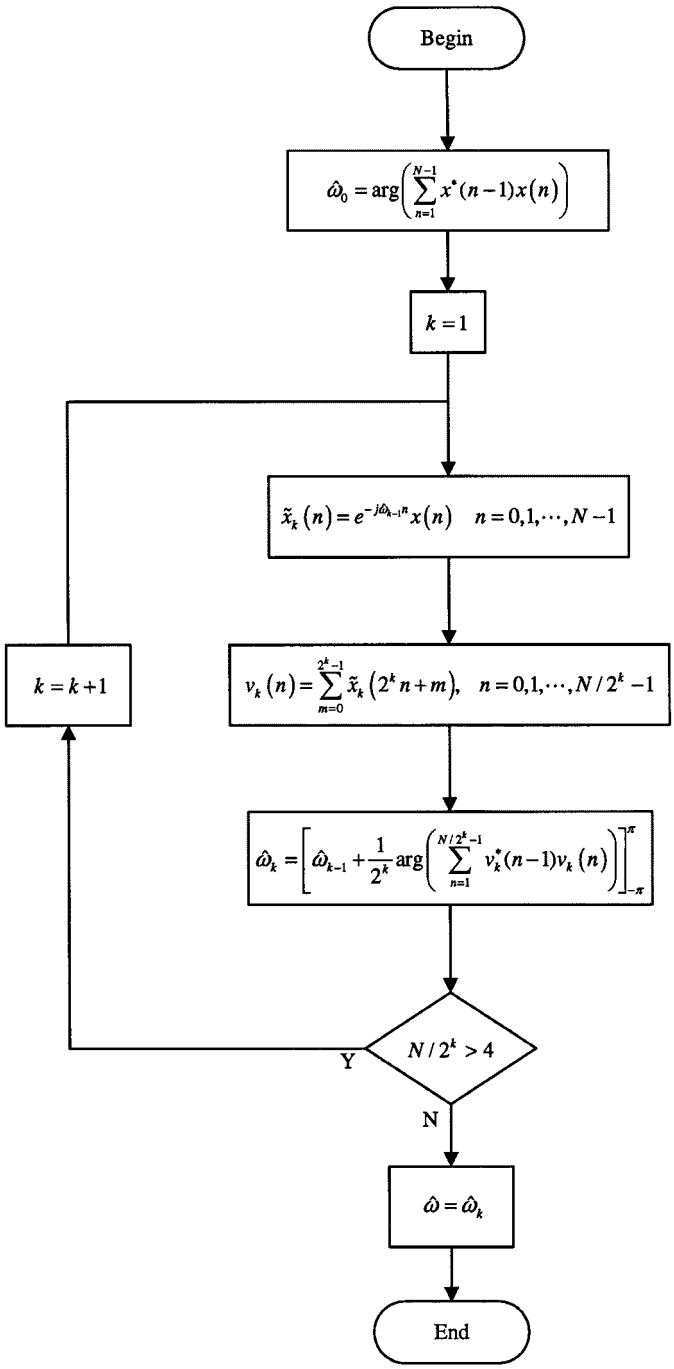

Fig. 4. Summary of proposed algorithm.

Substituting (37) and (43) into (44) yields the desired result in (36). The algorithm's frequency estimate $\hat{\omega}$ is then the frequency estimate at the final iteration

$$\begin{aligned}\hat{\omega} =& \hat{\omega}_K \\ =& (\omega + e_K)_{-\pi}^{\pi}. \end{aligned} \tag{45}$$

Using the decomposition of (22), the error at iteration k may be expressed

$$e_k = \frac{\alpha_k}{M_k} + \frac{2\pi m_k}{M_k}. \tag{46}$$

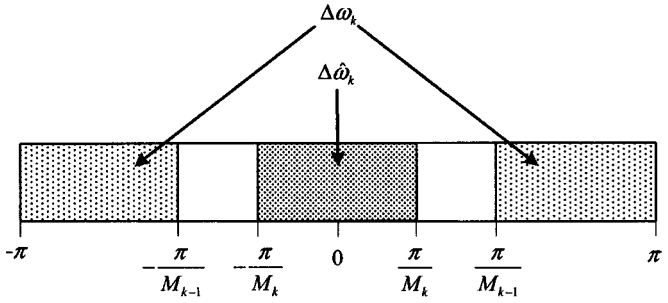

Fig. 5. Relationship between the residual signal's frequency at iteration k, $\Delta\omega_k$, and its estimate $\Delta\hat{\omega}_k$ when phase wrapping occurred in the previous iteration. The error is seen to satisfy $|\Delta\omega_k - \Delta\hat{\omega}_k| > (\pi/M_{k-1}) - (\pi/M_k)$.

In the absence of phase wrapping, $m_k = 0$ and the error is bounded by $|e_k| \leq \pi/M_k$. By extension, when no phase-wrapping occurs during any of the iterations, the sequence of errors is bounded by the decreasing sequence π/M_k, $k = 1, 2, \ldots, K$, and in this sense, the algorithm converges. Once phase-wrapping occurs, however, $m_k \neq 0$, and these bounds no longer apply. In fact, for the case when M_k doubles at each iteration, it will be shown that once the error is outside the band $[-\pi/M_k, \pi/M_k]$ at any iteration $k-1 > 0$, it will fall outside the corresponding band in each successive iteration. Specifically, if $M_k/M_{k-1} = 2$ and at iteration $k-1 > 0$, $|e_{k-1}| > \pi/M_{k-1}$, then $|e_k| > \pi/M_k$. The derivation of this result follows.

Assume that at iteration $k-1 > 0$, $|e_{k-1}| > \pi/M_{k-1}$. From (23), the error e_{k-1} lies in the union $(-\pi - \pi/M_{k-1}, -\pi/M_{k-1}) \bigcup (\pi/M_{k-1}, \pi + \pi/M_{k-1})$, and therefore, $\Delta\omega_k = (-e_{k-1})_{-\pi}^{\pi}$ lies in the union $[-\pi, -\pi/M_{k-1}), \bigcup (\pi/M_{k-1}, \pi)$. However, by definition, the estimate $\Delta\hat{\omega}_k$ is restricted to lie in the interval $[-\pi/M_k, \pi/M_k)$, and therefore, from Fig. 5, it can be seen that

$$\begin{aligned}|e_k| =& |\Delta\hat{\omega}_k - \Delta\omega_k| \\ >& \frac{\pi}{M_{k-1}} - \frac{\pi}{M_k} \\ =& \frac{\pi}{M_k}. \end{aligned} \tag{47}$$

It can be seen therefore that once phase-wrapping causes the error at iteration $k > 0$ to exceed π/M_K in magnitude, the subsequent errors $e_{k+1}, e_{k+2}, \ldots, e_K$ will exceed $\pi/M_{k+1}, \pi/M_{k+2}, \ldots, \pi/M_K$, respectively. In such cases, the iterative process effectively fails, and the final error will have a nonzero bias term of the form $2\pi m_K/M_K$ with $m_K \neq 0$.

In summary, under high SNR conditions, frequency estimation is performed on residual signals that, on average, decrease in frequency with each iteration, thereby allowing the use of increased filtering at each iteration. This increased filtering results in residual signals with frequencies that are progressively more concentrated about zero. The process repeats until the final stage, where the residual signal's frequency is estimated with an accuracy close to the CRB. Just how close to this bound the estimator can come will be described in Section III.

III. Asymptotic Performance Analysis

The analysis in the previous sections indicates that as long as the error in the last iteration satisfies $|\omega + e_K| < \pi$, the algorithm's error is just equal to the last iteration's estimation error e_k. At asymptotically high SNR, the frequency of the residual signal entering the last iteration $\Delta\omega_K$ will be small, and therefore

$$\text{sinc}^2_{M_K}\left(\frac{\Delta\omega_K}{2}\right) \approx 1. \tag{48}$$

Using this approximation in the expression for the basic algorithm's variance (26) yields

$$\begin{aligned}\text{var}(e_K) =& \frac{1}{M_K\left(\frac{A^2}{\sigma^2}\right)(N-M_K)^2\text{sinc}^2_{M_K}\left(\frac{\Delta\omega_K}{2}\right)} \cdot \\ & \times\left[1+\frac{N-M_K}{2M_K^2\left(\frac{A^2}{\sigma^2}\right)\text{sinc}^2_{M_K}\left(\frac{\Delta\omega_K}{2}\right)}\right] \\ \approx& \frac{1}{M_K\left(\frac{A^2}{\sigma^2}\right)(N-M_K)^2} \\ & \times\left[1+\frac{N-M_K}{2M_K^2\left(\frac{A^2}{\sigma^2}\right)}\right].\end{aligned} \tag{49}$$

Appendix A shows that e_K is zero mean at high SNR, and therefore, at least asymptotically, the estimator's variance is equal to its mean-squared error:

$$\begin{aligned}E\left\{(\hat{\omega}-\omega)^2\right\} &\triangleq \text{mse}(\hat{\omega}) \\ &= \text{var}(e_K).\end{aligned} \tag{50}$$

Finally, when the SNR A^2/σ^2 satisfies

$$\frac{A^2}{\sigma^2} \gg \frac{N-M_K}{2M_K^2} \tag{51}$$

we have, from (49)

$$\begin{aligned}\text{mse}\,(\hat{\omega}) \approx& \frac{1}{M_K\left(\frac{A^2}{\sigma^2}\right)(N-M_K)^2} \\ =& \frac{1}{M_K^3\left(\frac{A^2}{\sigma^2}\right)\left(\frac{N}{M_K}-1\right)^2} \\ =& \frac{\left(\frac{N}{M_K}\right)^3}{N^3\left(\frac{A^2}{\sigma^2}\right)\left(\frac{N}{M_K}-1\right)^2}.\end{aligned} \tag{52}$$

For the special case of $M_1 = 2$, $M_2 = 4, \ldots, M_K = N/4$ considered earlier, the requirement in (51) becomes

$$\frac{A^2}{\sigma^2} \gg \frac{6}{N} \tag{53}$$

and therefore, the high SNR requirement is inversely proportional to N. For comparison purposes, this result can be referenced to the CRB given in (3):

$$\frac{\text{mse}(\hat{\omega})}{\sigma^2_{CR}} = \frac{\left(\frac{N}{M_K}\right)^3(N^2-1)N}{6\left(\frac{N}{M_K}-1\right)N^3}. \tag{54}$$

It is easy to show that (54) takes on its minimum value when $N/M_k = 3$, which is to say that at the last iteration, samples of the residual signal are accumulated to form three segments, which are then used for frequency estimation. For $N(N^2-1)/N^3 \approx 1$, the degradation in this case is 0.51 dB. For $N/M_k = 4$ and $N/M_k = 2$, the degradation increases to 0.74 and 1.25 dB, respectively.

IV. Complexity Analysis

The complexity of the ILP algorithm can be assessed by breaking the algorithm into the four steps of

1) correlation calculation;
2) angle calculation;
3) frequency translation;
4) estimation update.

The following analysis assumes that N is a power of 2 and that in addition to the initial estimation $K = \log_2 N - 2$ iterations are performed with the filter lengths doubling at each iteration, i.e., $M_0 = 1$, $M_1 = 2$, $M_2 = 4, \ldots, M_k = N/4$. Correlation calculation requires $N/2^{k-1} - 1$ complex multiplications and $N/2^{k-1} - 2$ complex additions at iteration k, yielding totals of $2N - \log_2 N - 3$ and $2N - 2\log_2 N - 2$, respectively. Angle calculation requires one arctangent evaluation per iteration, for a total of $\log_2 N - 1$.

For all but the last iteration, the N received samples are translated in frequency by multiplication with a complex exponential. One method of generating a complex exponential is with a numerically controlled oscillator (NCO) [13]. The simplest NCO consists of an overflowing phase accumulator that addresses a sin/cos look-up table. A sampled complex exponential function is generated by initializing the phase accumulator to an initial value and incrementing the accumulator once per clock cycle by a quantity proportional to the desired frequency. The accumulator contents are then used to index into a sin/cos look-up table. Using this arrangement, one sample of the complex exponential can be generated per accumulator update. For our purposes, the frequency translation stage then requires N complex multiplies and $N-1$ real additions per iteration for totals of $N(\log_2 N - 2)$ and $(N-1)(\log_2 N - 2)$, respectively. Finally, a total of $\log_2 N - 2$ real additions are needed to update the frequency estimate, as given in (44) for iterations one through K.

The total operation counts are summarized in Table I. For comparison purposes, the operation counts for the FCFB and Kay algorithms, as derived in Appendix B, are also shown in Table II. The complexity of the FFT approach depends on the amount of zero-padding used and is therefore a function of the frequency resolution required. The resolution required in turn depends on the operating SNR of the estimator since the limited resolution effectively sets a lower bound on the estimation error that can be obtained. A comparison of ILP and FFT complexity will be given in Section V.

V. Simulation Results

The performance of the ILP, FCFB, and Kay algorithms was evaluated through simulation for $N = 24$ and $N = 128$. Simulations with $N = 24$ used filter lengths that doubled at each

TABLE I
ILP COMPLEXITY BREAKDOWN

	Complex		Real		Arctangents
	Multiplications	Additions	Multiplications	Additions	
Correlation Calculation	$2N-\log_2 N-3$	$2N-2\log_2 N-2$	—	—	—
Angle Calculation	—	—	—	—	$\log_2 N-1$
Frequency Translation	$N(\log_2 N-2)$	—	—	$(N-1)(\log_2 N-2)$	—
Frequency Update	—	—	—	$\log_2 N-2$	
Total	$(N-1)\log_2 N-3$	$2N-2\log_2 N-2$		$N(\log_2 N-2)$	

TABLE II
RELATIVE COMPLEXITY OF SINGLE FREQUENCY ESTIMATORS IN THE FCFB METHOD. NUMBER OF STAGES IS $p \le \log_2 N - 1$

	Complex		Real		Arctangents
	Multiplications	Additions	Multiplications	Additions	
Kay	$N-1$	$N-2$	$N-1$	$N-2$	$N-1$
FCFB	$2^p N$	$3N2^p-2N-2\times4^p$	$N/2^p-1$	$N/2^p-2$	$N/2^p-1$
ILP	$(N-1)\log_2 N-3$	$2N-2\log_2 N-2$	—	$N(\log_2 N-2)$	$\log_2 N-1$

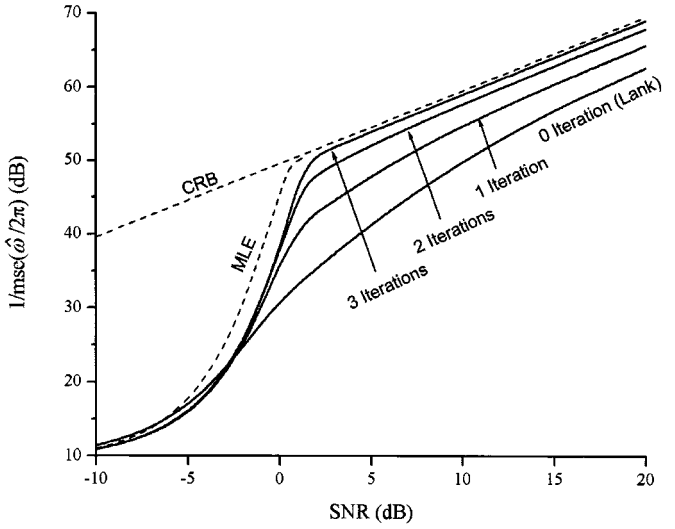

Fig. 6. ILP algorithm for zero through three iterations $K = 0, 1, 2, 3$ ($\omega/2\pi = 0.2$, $N = 24$).

iteration, i.e., $M_k = 2^k$, $k = 0, \ldots, 3$. Similarly, for $N = 128$, the filter lengths were $M_k = 2^k$, $k = 0, \ldots, 5$. The number of Monte Carlo runs was at least 200 000 for $N = 24$ and at least 500 000 when $N = 128$. The FCFB method was simulated with decimation.

Fig. 6 shows simulated results for the ILP algorithm for zero through three iterations with $N = 24$ and $\omega/2\pi = 0.2$. Note the case of no iterations is equivalent to that proposed in [2]. As the number of iterations increases, the performance approaches that of MLE. With three iterations, the threshold is 1 dB greater than obtained with MLE. Above this threshold, the ILP algorithm's mean-squared error is within 0.5 dB of the CRB, as predicated from the results of Section III.

Fig. 7 compares the performance of the proposed ILP algorithm with Fowler's FCFB method ($p = 1$ stage), Kay's method, and Lank's method. Normalized frequencies of 0.2, 0.25, and 0.4 were simulated. At a frequency of 0.2, the threshold for the

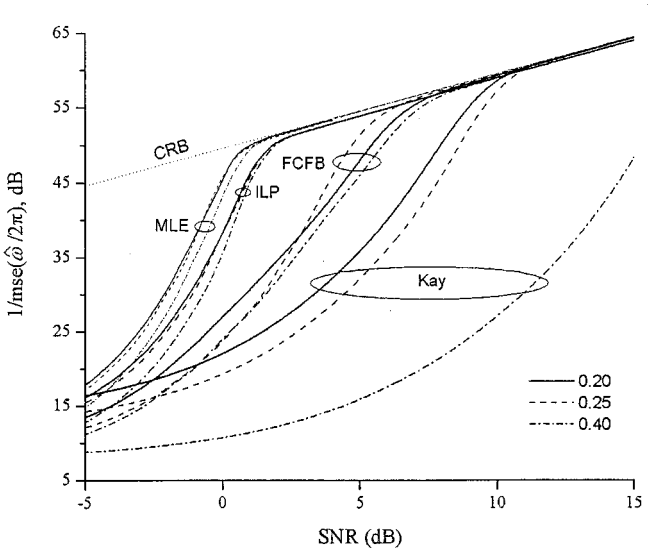

Fig. 7. Mean-squared error comparison among phase-based algorithms ($\omega/2\pi = 0.2, 0.25, 0.4$, $N = 24$). The FCFB method was simulated with the $p = 1$ stage.

ILP algorithm is seen to be 5 dB below that of the FCFB method and more than 8 dB below that of Kay's method. The thresholds of the MLE and ILP algorithms are seen to be independent of frequency, whereas Kay's method exhibits large variations with frequency, and the FCFB's threshold varies by about 2 dB over the three frequencies plotted. Notice that a change in normalized frequency of only 0.05 increases the FCFB's threshold by 1 dB.

Performance variation with frequency is further illustrated in Fig. 8, where contours of inverse mean-squared error in decibels are plotted versus SNR and ω. The right-most contour represents an inverse mean-squared error of 58 dB, where each subsequent contour to the left is 3 dB less. The threshold occurs where contours are closely spaced. The MLE, ILP, and FCFB ($p = 1$) algorithms are shown. Threshold performance of the MLE algorithm is relatively independent of frequency, whereas the FCFB's threshold varies by about 2 dB with the highest values occurring between filterbank centers, i.e., at $\omega/2\pi = 0.125$. The performance of the ILP algorithm is nearly uniform across frequency.

Results for $N = 128$ are shown in Fig. 9 for $\omega = 0$ and $\omega/2\pi = 0.2$. The ILP threshold is seen to be about 2 dB higher than that of MLE, independent of frequency, whereas the FCFB method with $p = 5$ stages has thresholds that vary between 5 and 12 dB above that of MLE.

Fig. 10 compares the ILP algorithm with the FFT method for $N = 24$ and an N'-point FFT. Frequencies are randomly distributed with a density that is uniform over the interval $[-0.45, 0.45]$. Below threshold, FFT performance agrees well with MLE. However for high SNR, the FFT method's performance plateaus at a value that increases with N'. This phenomenon results from calculating the ML statistic at only a finite set of frequencies. The plateaus in inverse-mse are seen to occur at about $12N'^2$, which is the inverse of the quantization error variance associated with quantization of the frequency estimate to N' frequency bins. For an operating SNR of 2

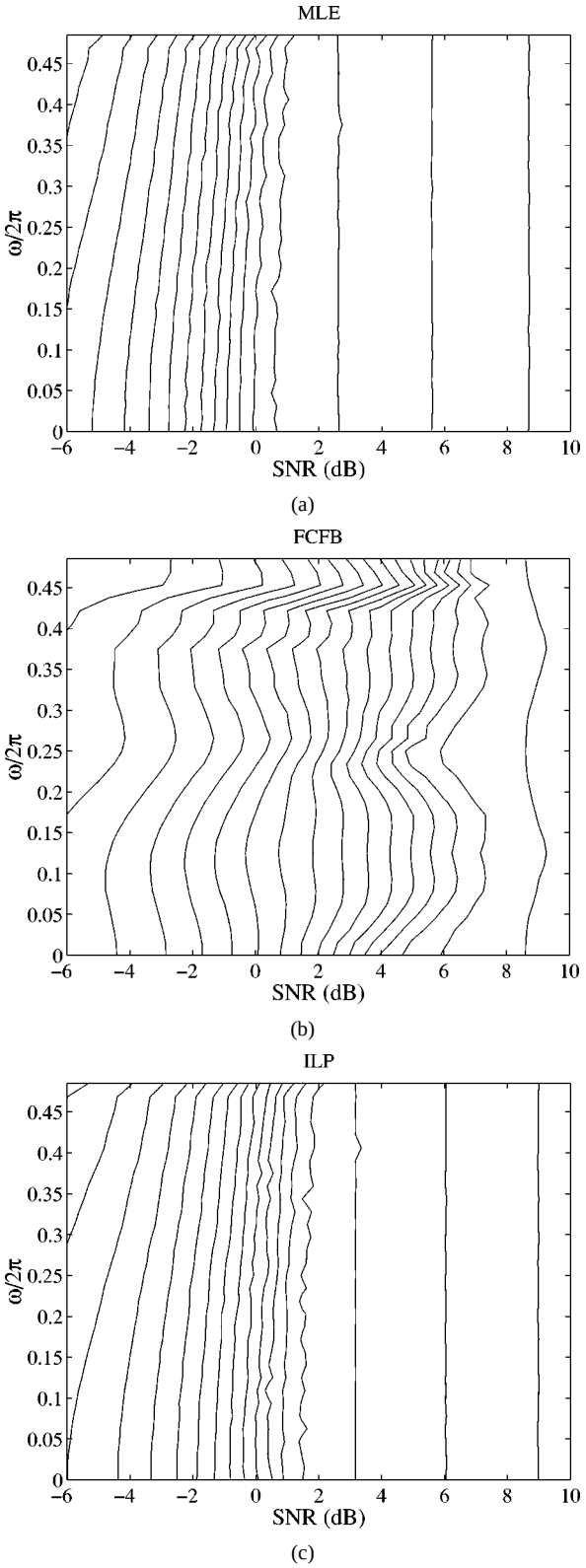

Fig. 8. Frequency versus SNR contours comparing mean-squared error of estimators. Right-most contour corresponds to 1mse$(\hat{\omega}) = 58$ dB. (Top) MLE. (Middle) FCFB with $p = 1$ stage. (Bottom) ILP with $K = 3$ iterations.

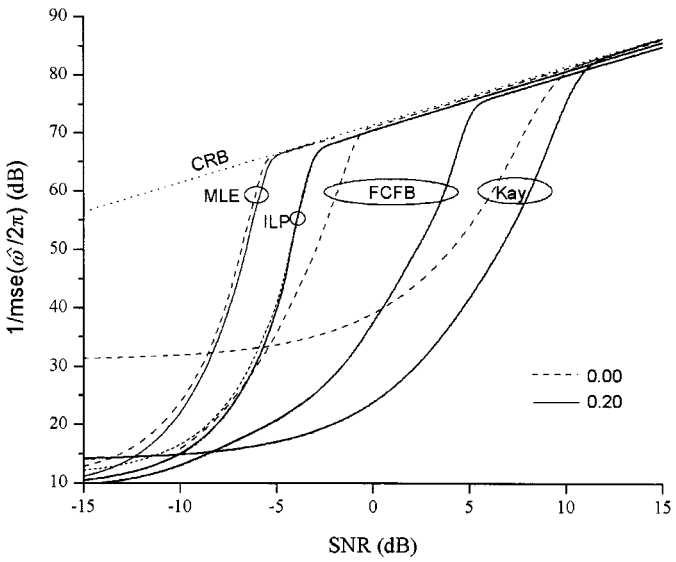

Fig. 9. Mean-squared error comparison for $\omega = 0, \omega/2\pi = 0.2$, and $N = 128$. The FCFB method was simulated with $p = 5$ iterations.

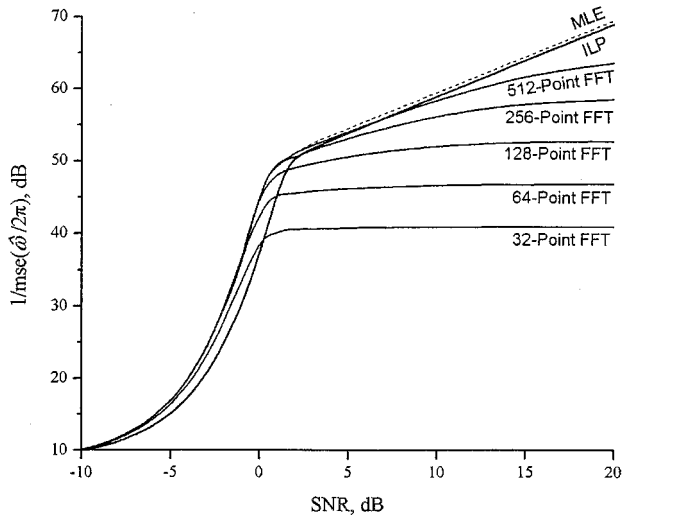

Fig. 10. Performance comparison between ILP and FFT ($\omega/2\pi \in [-0.45, 0.45]$, $N = 4$). At the SNR threshold, which is about 2 dB, a 256-point FFT is required to obtain similar mean-squared error to the ILP algorithm.

dB, approximately at the ML threshold, we see that at least a 256-point FFT is required for the FFT to have equivalent performance to ILP. An FFT of length N'-point FFT requires $(N'/2)\log_2 N'$ complex multiplications and $N' \log_2 N'$ additions [14]. Including an additional N' complex multiplies for the magnitude calculation results in a total of about 1150 complex multiplies. In contrast, the ILP algorithm using $M_1 = 2$, $M_2 = 4$, and $M_3 = 8$ requires $23 + 11 + 5 + 3 = 42$ complex multiplies for lag product calculation and $3 \times 24 = 72$ complex multiplies for frequency translation, for a total of 114. The number of complex multiplies required for the ILP algorithm is therefore about one tenth of those required for the FFT. However, the ILP algorithm requires three arctangent computations.

VI. Conclusion

This paper presented a computationally efficient frequency estimation algorithm called iterative linear prediction. In this algorithm, linear prediction frequency estimation is used to obtain

an initial estimate of the unknown frequency. Translation in frequency of the received signal by this initial estimate results in a signal whose frequency is equal to the negative of the initial estimate and is therefore confined to a relatively narrow band about zero. This allows lowpass filtering to be applied, which results in increased SNR and, therefore, improved frequency estimation in the next iteration. This approach yields a significantly reduced threshold relative to the algorithms of Fowler and Kay; this is a performance that is close to that of maximum likelihood estimation. In addition, the mean-squared error performance is within 0.7 dB of the Cramer-Rao bound at SNRs above threshold. Unlike many frequency estimation methods, the proposed algorithm's performance is uniform across the frequency range $[-\pi, \pi)$. This may be attributed to the continuous-valued estimates, made at each iteration, that produce estimation errors at each iteration that are largely insensitive to the frequency of the original signal. This is in contrast to the case where a detection process is used to form a discrete-valued coarse estimate of the received signal's frequency.

This work addressed the problem of estimating an unknown frequency in the range of $-\pi$ to π from a set of N samples. In many applications, however, the unknown frequency may be known *a priori* to lie within some range, say, $-c\pi$ to $c\pi$, $0 < c < 1$. As described in [9], this problem can be converted to the problem treated here by prefiltering and decimation. This will result in a reduction in the number of samples, and if the frequency range is sufficiently small, as few as three or four samples may remain after decimation. In this case, iteration is not possible, and the algorithm reduces to that of Lank. If the original data is filtered and decimated so that the resulting frequency range is from $-\pi$ to π, then the number of samples that emerge will be approximately equal to twice the maximum number of cycles of the complex exponential that could have been received in the original set of samples. The conclusion is that iteration is possible, and therefore, the algorithm is beneficial when the maximum number of cycles received is at least three. Future investigation in this area may include the study of performing frequency estimation at each iteration with algorithms other than linear prediction. The algorithms of [4] and [5], for example, may also be used at the expense of increased complexity. Alternatively, frequency estimation may be performed with Kay's algorithm. Our investigations, however, have shown that this method results in poor performance both in terms of threshold level and uniformity of threshold with frequency.

Appendix A
Mean and Variance of Basic Algorithm

This appendix derives the mean and variance of the estimator in (18) under a high SNR assumption and under the assumption that phase-wrapping does not occur. The first step is to expand the argument of the arg function

$$\begin{aligned}\sum_{n=1}^{L-1} &\left[C_M^*(\omega)e^{-j\omega M(n-1)} + n_v^*(n-1)\right] \\ &\times \left[C_M(\omega)e^{j\omega Mn} + n_v(n)\right] \\ &= (L-1)\left(|C_M(w)|^2 e^{j\omega M} + z\right) \end{aligned} \tag{A.1}$$

where

$$\begin{aligned} z =& \frac{1}{L-1}\left[C_M(\omega)e^{jM\omega}n_v^*(0)\right. \\ &\left. + C_M^*(\omega)e^{-j(L-2)M\omega}n_v(L-1)\right] \\ &+ \frac{1}{L-1}e^{jM\omega}\left[\sum_{l=1}^{L-2} C_M(\omega)e^{jlM\omega}n_v^*(l)\right. \\ &\left. + C_M^*(\omega)e^{-jlM\omega}n_v(l)\right] \\ &+ \frac{1}{L-1}\sum_{l=1}^{L-1} n_v^*(l-1)n_v(l). \end{aligned} \tag{A.2}$$

Next, define the scaled and rotated version of z as

$$\bar{z} = \frac{e^{-jM\omega z}}{|C_M(\omega)|^2}. \tag{A.3}$$

The real and imaginary parts of $\bar{z}$ are zero mean with variances

$$\begin{aligned} \text{var}\,(\bar{z}_r) =& \frac{\sigma_v^2}{(L-1)^2\,|C_M(\omega)|^2} \\ &\times \left(2L - 3 + \frac{(L-1)\sigma_v^2}{2\,|C_M(\omega)|^2}\right) \end{aligned} \tag{A.4}$$

$$\begin{aligned} \text{var}\,(\bar{z}_i) =& \frac{\sigma_v^2}{(L-1)^2\,|C_M(\omega)|^2} \\ &\times \left(1 + \frac{(L-1)\sigma_v^2}{2\,|C_M(\omega)|^2}\right). \end{aligned} \tag{A.5}$$

Using (A.1) and (A.3), the estimator in (18) can be expressed as

$$\begin{aligned} \hat{\omega}(M) =& \frac{1}{M}\arg\left[e^{jM\omega}(1+\bar{z})\right] \\ =& \frac{1}{M}\arg\left[e^{j(M\omega+\alpha)}\right] \end{aligned} \tag{A.6}$$

where $\alpha = \arg(1+\bar{z})$, but $1+\bar{z}$ can be written

$$\begin{aligned} e^{j\alpha} &= \frac{1+\bar{z}}{|1+\bar{z}|} \\ &= \exp\left\{j\tan^{-1}\frac{\bar{z}_i}{1+\bar{z}_r}\right\}. \end{aligned} \tag{A.7}$$

When the variance of the components of $\bar{z}$ satisfy $\text{var}(\bar{z}_r) \ll 1$ and $\text{var}(\bar{z}_i) \ll 1$, the standard small angle approximation applies, and thus

$$\alpha \cong \bar{z}_i. \tag{A.8}$$

From (A.4) and (A.5), these conditions will be satisfied when

$$\frac{\sigma_v^2}{(L-1)^{\frac{1}{2}}\,|C_M(\omega)|^2} \ll 1. \tag{A.9}$$

Condition (A.9) can be expressed in terms of the SNR A^2/σ^2, the number of samples accumulated M, and the record length N through the relations $\sigma_v^2 = M\sigma^2$ and $L = N/M$, and the definition of $C_M(\omega)$ in (15):

$$\frac{\sigma^2}{(N-M)^{\frac{1}{2}}A^2\left|\text{sinc}_M\left(\frac{\omega}{2}\right)\right|^2} = 1. \tag{A.10}$$

TABLE III
OPERATION COUNTS FOR THE p-STAGE FCFB ESTIMATOR

	Complex		Real		Arctangents
	Multiplications	Additions	Multiplications	Additions	
Filtering	—	$2N(2^p-1)$	—	—	—
Detection	2^pN	$2^pN-2\times4^p$	—	—	—
Phase Deltas	—	—	—	—	$N/2^p$
Freq. Est	—	—	$N/2^p-1$	$N/2^p-2$	—
Total	2^pN	$3\times2^pN-2N-2\times4^p$	$N/2^p-1$	$N/2^p-2$	$N/2^p-1$

Under the assumption of no-phase wrapping, $|M\omega+\alpha|<\pi$, and from (A.6) and (A.8), we have

$$\hat{\omega}=\frac{1}{M}\arg\left(e^{j(M\omega+\alpha)}\right)=\omega+\frac{\bar{z}_i}{M}. \tag{A.11}$$

The estimate is seen to be unbiased with variance

$$\text{var}(\hat{\omega})=\frac{\sigma_v^2}{(L-1)^2M^2\left|C_M(\omega)\right|^2}\left(1+\frac{(L-1)\sigma_v^2}{2\left|C_M(\omega)\right|^2}\right). \tag{A.12}$$

APPENDIX B
COMPLEXITY OF KAY AND FCFB ESTIMATORS

This Appendix presents a complexity analysis of the Kay and FCFB estimators. In the Kay estimator, the phase differences in (9) require $N-1$ complex multiplications, $N-2$ complex additions, and $N-1$ arctangents. Assuming the weighting coefficients are precalculated and stored in memory, the linear combination of phase differences given in (11) require an additional $N-1$ real multiplications and $N-2$ real additions.

The single-stage FCFB estimator takes the original sequence of N samples and forms four new sequences by digital heterodyning with sequences of the form $\{e^{-j(\pi/2)n}\}$, $\{e^{-j0}\}$, $\{e^{j(\pi/2)n}\}$, and $\{e^{j\pi n}\}$. Because this step only involves changing the signs of the real and imaginary parts, the complexity incurred in this step will be assumed to be negligible. Each of the resulting sequences is filtered with a length two filter with coefficients $[1\ 1]$ and decimated. This step therefore requires $2N$ complex additions. Each sequence also requires the calculation of a detection statistic that is equal to the magnitude of the sum of first-order lag products, and therefore, all four decision statistics can be calculated with $4\times(N/2-1)$ complex multiplications and $4\times(N/2-2)$ complex additions for the lag product calculation and an additional four complex multiplications to form the complex magnitude. Kay's phase estimation is performed on the sequence with the largest decision statistic. Note that each lag product was already computed in the calculation of the winning decision statistic, and thus, the phase deltas require only $N/2-1$ arctangents. Again, assuming the weights are precomputed, the frequency estimate requires $N/2-1$ real multiplications and $N/2-2$ real additions.

A breakdown of the operation counts for the p-stage FCFB estimator is given in Table III. In the first stage of filtering, $N/2$ complex adds are performed for each of the four sequences. In the second stage, the length of the sequences is cut in half, but their number increases by four, and therefore, for p stages, the number of complex adds is

$$2N+4\left(\frac{2N}{2}\right)+4^2\left(\frac{2N}{2^2}\right)+\cdots+4^{p-1}\frac{2N}{2^{p-1}}=2N(2^p-1).$$

Each of the 4^p sequences has $N/2^p$ points, and therefore, the decision statistic requires $4^p(N/2^p-1)$ complex multiplications to form the lags, an additional 4^p complex multiplications to form the magnitude, and $4^p(N/2^p-2)$ complex additions. Frequency estimation is performed on the winning sequence of length $N/2^p$ and therefore requires $N/2^p-1$ arctangents, $N/2^p-1$ real multiplications, and $N/2^p-2$ real additions. Overall, the number of operations increases linearly with N, although for maximum performance, the number of stages p also increases with N. For example, in [11], one, three, and four stages were reported to be required to obtain optimum threshold performance for 32, 64, and 128 points, respectively. The number of complex multiplication and additions increases exponentially with p, but the number of arctangents required decreases exponentially with p.

ACKNOWLEDGMENT

The authors would like to thank C. Thron and W. Xiao for their valuable comments. They would also like to thank the reviewers for their useful comments.

REFERENCES

[1] D. Rife and R. Boorstyn, "Single-tone parameter estimation from discrete-time observations," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 591–598, Sept. 1974.

[2] G. Lank, I. Reed, and G. Pollon, "A semicoherent detection and doppler estimation statistic," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-9, pp. 151–165, Mar. 1973.
[3] L. B. Jackson and D. W. Tufts, "Frequency estimation by linear prediction," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, vol. ASSP-26, 1978, pp. 352–356.
[4] M. Fitz, "Further results in the fast estimation of a single frequency," *IEEE Trans. Commun.*, vol. 42, pp. 862–864, Feb. 1994.
[5] M. Luise and R. Reggiannini, "Carrier frequency recovery in all-digital modems for burst-mode transmissions," *IEEE Trans. Commun.*, vol. 43, pp. 1169–1178, Feb. 1995.
[6] S. Tretter, "Estimating the frequency of a noisy sinusoid by linear regression," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 832–835, Nov. 1985.
[7] S. Kay, "A fast and accurate single frequency estimator," in *IEEE Trans. Acousti., Speech, Signal Processing*, vol. 37, Dec. 1989, pp. 1987–1990.
[8] S. Lang and B. Musicus, "Frequency estimation from phase differences," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 37, 1989, pp. 2140–2143.
[9] D. Kim, M. Narasimha, and D. Cox, "An improved single frequency estimator," *IEEE Signal Processing Lett.*, vol. 3, pp. 212–214, July 1996.
[10] S. Umesh and D. Nelson, "Computationally efficient estimation of sinusoidal frequency at low SNR," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1997, pp. 2797–2800.
[11] M. Fowler and J. Johnson, "Extending the threshold and frequency range for phase-based frequency estimation," *IEEE Trans. Signal Processing*, vol. 47, pp. 2857–2863, Oct. 1999.
[12] D. Nelson and K. Short, "A channelized cross spectral method for improved frequency resolution," in *Proc. IEEE-SP Int Symp. Time-Freq. Time-Scale Anal.*, 1998, pp. 101–104.
[13] J. Thierney, C. M. Radar, and B. Gould, "A digital frequency synthesizer," *IEEE Trans. Audio Electroacoust.*, vol. AU-19, pp. 48–57, 1971.
[14] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989, pp. 587–598.

Tyler Brown (M'95) received the B.S.E.E. degree from the University of Minnesota, Minneapolis, in 1987, the M.S.E.E. degree from the University of Colorado, Boulder, in 1990, and the Ph.D. degree from the University of Minnesota, in 1995.

He joined Motorola, Arlington Heights, IL, in 1995, where he is now a Distinguished Member of Technical Staff. At Motorola, he has worked on GSM subscriber and CDMA cellular infrastructure development. His current research interests are wireless communications and signal processing.

Dr. Brown is a member of the IEEE Communications and Signal Processing Societies.

Michael Mao Wang (M'99) received the B.S. and M.S. degrees in electrical engineering from Nanjing Institute of Technology, Nanjing, China, in 1984 and 1987, respectively, and the M.S. degree in biomedical engineering in 1992 and the Ph.D. degree in electrical engineering in 1995 from the University of Kentucky, Lexington.

He is currently a Distinguished Member of Technical Staff at Motorola Cellular Infrastructure Group, Arlington Heights, IL. His current research interests include signal processing and wireless communications.

Dr. Wang is a member of the IEEE Communications and Signal Processing Societies.