ELSEVIER

Signal Processing 52 (1996) 283–297

SIGNAL PROCESSING

# A sliding Goertzel algorithm

Joe F. Chicharo, Mehdi T. Kilani*

*Department of Electrical and Computer Engineering, University of Wollongong, Northfields Avenue, Wollongong, NSW 2522, Australia*

Received 19 November 1993; revised 30 September 1994, 27 April 1995 and 15 March 1996

**Abstract**

This paper presents a sliding Goertzel algorithm to accurately estimate the Fourier coefficients of multifrequency (MF) sinusoidal signals buried in noise. The algorithm is based on second-order digital resonators that are tuned at the desired frequencies. The proposed method provides the following advantages when compared with the conventional Goertzel algorithm. Firstly, it computes Fourier coefficients in less than one signal period. Therefore, faster detection time is achieved, particularly when the greatest common divisor (GCD) of the input frequencies is small. Secondly, it is less prone to numerical overflow problems in fixed-point arithmetic implementation. Thirdly, the algorithm is quite suitable for time varying sinusoidal signal estimation. An analysis is undertaken to provide additional insight into the issue of required acquisition time versus the desired accuracy for the proposed algorithm. Extensive simulation tests are also included to demonstrate its performance.

**Zusammenfassung**

Dieser Beitrag stellt einen gleitenden Goertzel-Algorithmus vor, um die Fourier-Koeffizienten von im Rauschen verdeckten Multifrequenz(MF)-Sinussignalen genau zu schätzen. Der Algorithmus stützt sich auf digitale Resonatoren zweiter Ordnung, die auf die Nutzfrequenzen abgestimmt werden. Die vorgeschlagene Methode liefert, verglichen mit dem konventionellen Goertzel-Algorithmus, die folgenden Vorteile. Erstens, sie berechnet Fourier-Koeffizienten in weniger als einer Signalperiode. Deshalb wird eine kürzere Detektionszeit erreicht, besonders wenn der größte gemeinsame Teiler (ggT) der Eingangsfrequenzen klein ist. Zweitens, sie neigt bei einer Implementation in Festkomma-Arithmetik weniger zu numerischen Überlaufproblemen. Drittens, der Algorithmus ist durchaus zur Schätzung zeitveränderlicher Sinussignale geeignet. Eine Analyse wird vorgenommen, um zusätzlichen Einblick in die Beziehung zwischen benötigter Akquisitionszeit und Genauigkeit des vorgeschlagenen Algorithmus zu gewinnen. Umfangreiche Simulationstests werden ebenfalls dargestellt, um seine Leistungsfähigkeit zu demonstrieren.

**Résumé**

Cet article présente un algorithme de Goertzel glissant permettant d'estimer de manière précise les coefficients de Fourier de signaux sinusoïdaux multi-fréquences (MF) immergés dans du bruit. Cet algorithme est basé sur des résonnances digitales d'ordre deux réglées sur les fréquences désirées. La méthode proposée présente les avantages suivants vis-à-vis de l'algorithme de Goertzel conventionnel: tout d'abord, elle calcule les coefficients de Fourier sur moins qu'une période du signal. De ce fait, le temps de détection est réduit, particulièrement lorsque le plus grand diviseur commun (GCD) des fréquences d'entrée est petit. Deuxièmement, elle est moins sujette aux problèmes de dépassement de capacité numérique pour une implantation en précision finie. Troisièmement, l'algorithme est tout à fait adapté à l'estimation de signaux sinusoïdaux variant dans le temps.

* Corresponding author.


*PII* S0165-1684(96)00066-7

Une analyse permettant de mieux cerner le problème du temps d'acquisition par rapport à la précision requise est également opérée. Des tests de simulations extensifs sont de plus inclus afin de mettre en évidence les performances de l'algorithme.

*Keywords:* Goertzel algorithm; Multifrequency estimation

## 1. Introduction

In many applications such as dual-tone multifrequency (DTMF) signals [2], digital multifrequency (MF) receiver [6] and in very small aperture terminal (VSAT) satellite communication systems [8], it is desirable to detect frequency components simply and efficiently. In the above cases, the number of required Fourier coefficients are usually small, hence the direct discrete Fourier transform (DFT) is more efficient than fast Fourier transform (FFT) algorithms. A second-order realisation of the Goertzel filter is also favoured over the direct DFT because it results in reduced computational burden [7]. The transfer function of the filter that is used in the Goertzel algorithm is defined as follows [7]:

$$H_k(z^{-1}) = \frac{1 - e^{j\omega_k} z^{-1}}{1 - 2\cos\omega_k z^{-1} + z^{-2}}, \tag{1}$$

where $\omega_k = 2\pi k/N$. The signal flow graph of the transfer function given by Eq. (1) is shown in Fig. 1. Note that the Goertzel filter is composed of a recursive part (left-hand side of delay elements) and a nonrecursive part (right-hand side of the delay elements). The DFT coefficients are obtained as the output of the system after $N$ iterations. The recursive part is a second-order digital resonator. The resonant frequency of the resonator is set at equally spaced frequency points; that is, $\omega_k = 2\pi k/N$. In practice we only compute the recursive part of the filter at every sample update and the nonrecursive part is computed only after the $N$th time instant when the Fourier coefficients are to be determined.

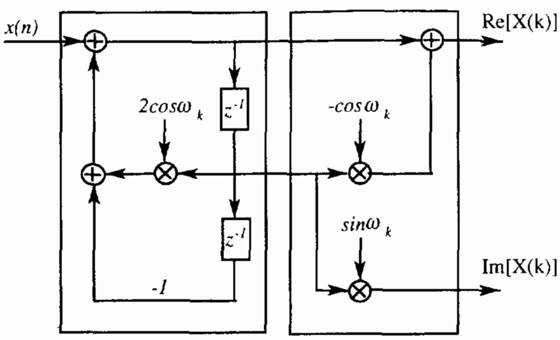



Fig. 1. Filter realisation of the Goertzel algorithm [7].

The conventional Goertzel filter is restricted in the sense that the Fourier coefficients of the input signal are given at equally spaced points in the frequency domain. This is a major disadvantage of the Goertzel technique particularly when the input sinusoids are arbitrarily located. In other words, the conventional Goertzel algorithm only finds the coefficients of an $N$ point DFT. Accurate sinusoidal coefficients are obtained if there exists a value for $k$ such that the input normalised frequencies become exactly equal to $2\pi k/N$. Otherwise, the estimate will be degraded due to the well-known leakage problem [4]. Consequently, $N$ must be large when the input frequencies are closely spaced or their frequencies are relatively prime factor. The resolution can be improved by using the modified Goertzel algorithm proposed by [3]. In the modified Goertzel algorithm, Eq. (1) is evaluated at the exact frequency of interest which means that the resonator is set arbitrarily at the input angular frequency. This modification results in faster acquisition time for estimating the amplitude of the input sinusoid. The modified Goertzel algorithm, however, requires the same acquisition time as the conventional Goertzel algorithm for estimating both the phase and amplitude. In fact, the conventional Goertzel algorithm yields accurate estimates of the sinusoidal parameters provided that the input frequency falls exactly at the resonant frequency of the resonator. Under these circumstances, both the modified and conventional Goertzel algorithms are equivalent.

When using the conventional and modified Goertzel algorithms, accurate phase and amplitude estimates of the sinusoidal components are obtained at the end of one period of the signal. For situations where the greatest common divisor (GCD) of the input frequencies is 1 Hz, the period of the signal is equal to 1 s. Consequently, a large number of samples are required which increases the acquisition time as well as the possibility of overflow at the output of the Goertzel filter for fixed-point arithmetic implementation [1].

The notch Fourier transform (NFT) approach reported in [9] also proposes an efficient method for computing the Fourier coefficients at arbitrary frequencies. Assuming that the signal is composed of $m$ sinusoidal components with known frequencies, the NFT method utilises $(m-1)$ serially connected second-order finite impulse response (FIR) notch filters which only let one component pass through. This approach [9] provides a significant computational advantage and faster detection time when compared to the DFT method. However, its performance is seriously degraded when the components are buried in noise.

In this paper, an input sinusoid is applied to a second-order digital resonator whose resonant frequency is tuned exactly to the input frequency. The characteristics of the signal at the output of the resonator are derived in the time domain by means of a $z$-transformation. An algorithm similar to the real valued Goertzel algorithm is then established which estimates the phase and amplitude of the input sinusoid for successive values of $n$. Hence, the proposed algorithm is referred to as a sliding Goertzel (SG) algorithm. This new approach provides the following features when compared with the conventional and modified Goertzel algorithms. Firstly, it computes Fourier coefficients of sinusoids in less than one signal period, thus leading to faster acquisition time. This is due to the fact that individual sinusoids are separated and enhanced through the resonator very fast. In other words, the effect of noise and other input components at the output of the resonator will be negligible after a short period of time. Secondly, since the Fourier coefficient estimates are updated at every sample time then the proposed algorithm is capable of tracking rapid changes in the signal parameters (phase and amplitude). Finally, the required number of samples is significantly reduced which means that the possibility of numerical overflows reduces when the proposed method is implemented in hardware with fixed-point arithmetic [1]. The processing gain (PG) factor defined as the ratio of the output SNR to the input SNR [4] has been analytically obtained for the proposed algorithm. This factor is used to establish the relationship between the desired accuracy and the acquisition time for the proposed algorithm.

This paper is organised as follows. In Section 2, the problem statement is given and the sliding Goertzel algorithm as a solution is proposed in Section 3. Simulation tests are included in Section 4. Finally, Section 5 concludes the paper.

## 2. Problem statement

One problem associated with the DFT and Goertzel techniques is that they cannot accurately compute Fourier coefficients until the end of a complete data block. This means that in order to accurately estimate the phase and amplitude of sinusoidal components of an MF signal, the required number of samples should be taken over the whole period of the input sequence. Clearly, the period of an MF signal depends on the constituent input frequencies. In situations where the input frequencies are relatively prime or very closely spaced, a large number of samples will be required which results in an undesirable increase in acquisition time. Under these circumstances, higher resolution is needed to accurately estimate the sinusoids. The resolution can be increased by increasing the input block length, thus resulting in an increased acquisition time. In the case where the GCD of the input frequencies is equal to $\Delta f$ Hz, the minimum number of samples ($N_{\min}$) which is required to accurately compute the sinusoidal coefficients is given by (see Appendix A)

$$N_{\min} = \frac{f_s}{\Delta f}, \tag{2}$$

where $f_s$ is the sampling frequency ($f_s$ is assumed to be an integer factor $\Delta f$). It is evident that $N_{\min}$ is inversely proportional to $\Delta f$, that is, for a small value of $\Delta f$ a large number of samples are required. Note that when using the DFT and the Goertzel algorithms the phase and amplitude must be fixed during each block of data. If the input signal parameters vary within each frame of data, the estimates are severely degraded. This means that the required large number of samples given by Eq. (2) also reduces the tracking capability of the Goertzel technique when the sinusoidal parameters are time varying.

## 3. The sliding Goertzel algorithm

This section presents a sliding Goertzel algorithm which updates the Fourier coefficients at every sample time. The proposed technique provides faster acquisition time when compared with the conventional Goertzel algorithm and is therefore suitable for the situations when the parameters of sinusoids change relatively fast. Consider a real valued signal composed of arbitrary frequencies expressed as follows:

$$x(n) = \sum_{k=1}^{s} a_k \cos \omega_k n + b_k \sin \omega_k n + v(n), \tag{3}$$

where $s$ is the number of components and $v(n)$ is the noise component. The task is to compute the coefficients ($a_k$ and $b_k$) for the sinusoidal components defined by $x(n)$. Let us begin by examining the case where the input signal consists of a single component as follows:

$$x_k(n) = a_k \cos \omega_k n + b_k \sin \omega_k n. \tag{4}$$

Applying $x_k(n)$ to the resonator, in terms of $z$-transform, the output, $Y_k(z^{-1})$, is given by

$$Y_k(z^{-1}) = X_k(z^{-1}) H_r(z^{-1}), \tag{5}$$

where $X_k(z^{-1})$ is

$$X_k(z^{-1}) = \frac{a_k(1 - \cos \omega_k z^{-1}) + b_k \sin \omega_k z^{-1}}{1 - 2\cos \omega_k z^{-1} + z^{-2}}. \tag{6}$$

$H_r(z^{-1})$ is the transfer function of a second-order resonator. Taking the inverse $z$-transform of Eq. (5), it can be shown that the output, $y_k(n)$, is given by (see Appendix B)

$$\begin{aligned} y_k(n) = {} & \frac{b_k}{2\sin^2 \omega_k}\{(n+2)\cos \omega_k \sin[\omega_k(n+1)] \\ & -(n+1)\sin[\omega_k(n+2)]\} \\ & + \frac{a_k}{2\sin^2 \omega_k}\{(n+2)\sin \omega_k \sin[\omega_k(n+1)]\}. \end{aligned} \tag{7}$$

Evaluating Eq. (7) at the $n$th and $(n-1)$th sample time and after some manipulations we obtain

$$\begin{aligned} y_k(n) = {} & \frac{n}{2\sin \omega_k}\{a_k \sin[\omega_k(n+1)] \\ & - b_k \cos[\omega_k(n+1)]\} \\ & + \frac{1}{2\sin^2 \omega_k}\{2a_k \sin \omega_k \sin[\omega_k(n+1)] \\ & + b_k \sin \omega_k n\} \end{aligned} \tag{8}$$

and

$$\begin{aligned} y_k(n-1) = {} & \frac{n}{2\sin \omega_k}(a_k \sin \omega_k n - b_k \cos \omega_k n) \\ & + \frac{1}{2\sin^2 \omega_k}[(b_k \cos \omega_k + a_k \sin \omega_k)\sin \omega_k n]. \end{aligned} \tag{9}$$

It is evident that as the number of samples increases, the output values will also increase in magnitude. This is expected since the input frequency is equal to the resonance frequency of the resonator. For large values of $n$, the second terms in Eqs. (8) and (9) can be ignored. Hence after some straightforward manipulations, the following expressions (for the estimates of $a_k$ and $b_k$) are obtained:

$$\begin{aligned} \hat{A}_k(n) &= \frac{2(y_k(n) - y_k(n-1)\cos \omega_k)}{n} \\ &\approx a_k \cos \omega_k n + b_k \sin \omega_k n, \\ \hat{B}_k(n) &= -\frac{2\sin \omega_k y_k(n-1)}{n} \\ &\approx -a_k \sin \omega_k n + b_k \cos \omega_k n. \end{aligned} \tag{10}$$

This algorithm has a similar parameterisation to the real valued conventional Goertzel algorithm which is now presented in sliding form. In the conventional Goertzel algorithm the left-hand side of Eq. (10) is only evaluated at the $N$th sample time and the resonator is tuned at equally spaced frequency points, that is $\omega_k = 2\pi k/N$. The proposed and modified Goertzel techniques are similar in the sense that both employ a resonator tuned exactly at the input frequency location. However, the modified Goertzel algorithm takes $\hat{A}_k$ and $\hat{B}_k$ as the estimates of $a_k$ and $b_k$, respectively [3]. In the proposed algorithm, the equality on the right-hand side of Eqs. (10) provides a suitable means for updating both the phase and amplitude at every sample time. In other words, by solving the resulting linear equations (defined by Eq. (10)), the sliding algorithm for Fourier coefficients is obtained as follows:

$$\begin{bmatrix} a_k(n) \\ b_k(n) \end{bmatrix} \approx \begin{bmatrix} \cos \omega_k n & -\sin \omega_k n \\ \sin \omega_k n & \cos \omega_k n \end{bmatrix} \begin{bmatrix} \hat{A}_k(n) \\ \hat{B}_k(n) \end{bmatrix}. \tag{11}$$

From Eqs. (10) it can be shown that

$$\hat{A}_k^2(n) + \hat{B}_k^2(n) \approx a_k^2 + b_k^2. \tag{12}$$

This means that the amplitude of the sinusoidal component can be updated at every sample time without evaluating Eq. (11). The phase associated with each sinusoid is also updated as follows:

$$\phi_k(n) = -\tan^{-1}\left(\frac{a_k(n)}{b_k(n)}\right). \tag{13}$$

For the case where the input signal is composed of $s$ sinusoids buried in noise, $s$ resonators are required where each resonator is tuned at a particular input sinusoidal frequency.

At this point it is important to consider the issue of the acquisition time ($t_{acq}$). Note that although the algorithm provides the sinusoidal component coefficients at every sample update, one does not need to evaluate Eqs. (10) and (11) at every sample time but only after the transient time has elapsed and the output of the resonator has settled. In other words, accurate estimates of the Fourier coefficients are available after a specific sampling number (say $N_{acq}$) when the contribution of noise and other components become insignificant at the output of the resonator. Hence, we can state that

$$t_{acq} = N_{acq}/f_s. \tag{14}$$

### 3.1. Criteria for selection of $N_{acq}$

For the proposed technique, the acquisition time depends on the power of the noise and the required resolution. In other words, when the input spectral lines are closely spaced or the input signal is heavily corrupted by noise, a greater value for $N_{acq}$ will be required. In situations where the frequency of the input sinusoid is different from the resonance frequency of the resonator, the output is given by (see Appendix B)

$$y_i(n) = \frac{1}{-2(\cos\omega_k - \cos\omega_i)} \times \left\{ \frac{-C_i \sin\omega_i}{\sin\omega_k} \sin[\omega_k(n+1)] + C_i \sin[\omega_i(n+1)] \right\}, \tag{15}$$

where $\omega_k$ is the resonance frequency of the resonator while $\omega_i$ and $C_i$ are the frequency and amplitude of the input sinusoid, respectively. From Eq. (15), it can be noted that the output magnitude is inversely proportional to the factor of $(\cos\omega_k - \cos\omega_i)$ which means that the contribution of the unwanted component will be greatest when it is close to the desired component. However, because the resonator is exactly tuned at the input frequency locations, the output component associated with the corresponding input sinusoid becomes dominant very quickly. Note that the magnitude of the output component associated with the desired sinusoid increases linearly with time (see Eq. (8)) while the output component due to the unwanted frequency is constant. It is also expected that the algorithm will also provide good performance under low SNR conditions. This is due to the fact that the resonator can be considered as an ideal matched filter for sinusoids buried in noise.

The contribution (or leakage) of the $i$th ($i \neq k$) sinusoid on the estimate of the $k$th amplitude at the $N_{acq}$th sampling time is obtained by substituting Eq. (15) into Eq. (10). After some manipulations it can be shown that

$$L(i) = \frac{C_i}{N_{acq}(\cos\omega_k - \cos\omega_i)} \times \left\{ \sin\omega_i - \sqrt{\sin^2[\omega_i(N_{acq}+1)] + \sin^2\omega_i N_{acq} - 2\sin\omega_i N_{acq} \sin[\omega_i(N_{acq}+1)]\cos\omega_k} \right\}. \tag{16}$$

Eq. (16) will be used to determine the amount of leakage resulting from other sinusoids on the desired frequency. In order to obtain the output noise power at the $N_{acq}$th sampling time, the proposed approach can be regarded as being the same as the conventional Goertzel algorithm where the input sinusoid is assumed to be equal to the resonant frequency of the resonator. In other words, the output noise power at the $N_{acq}$th sample time is equivalent to the noise power accumulated at each bin of an $N_{acq}$ point DFT. Hence, the output noise is written as

$$V_k = \frac{1}{N_{acq}} \sum_{n=0}^{N_{acq}-1} v(n) e^{-j(2\pi k/N_{acq})n}, \tag{17}$$

where $v(n)$ is assumed to be a zero mean Gaussian white noise sequence with power $\sigma_v^2$. Since Eq. (17) is a linear transformation then the output noise is also expected to be a zero mean Gaussian sequence whose power is given by [4]

$$\begin{aligned} E(|V|^2) &= \frac{1}{N_{acq}^2} \sum_{n=0}^{N_{acq}-1} \sum_{m=0}^{N_{acq}-1} E[v(m)v(n)] \\ &\qquad \times e^{-j(2\pi/N_{acq})n} e^{j(2\pi/N_{acq})m} \\ &= \frac{\sigma_v^2}{N_{acq}^2} \sum_{n=0}^{N_{acq}-1} \sum_{m=0}^{N_{acq}-1} e^{-j(2\pi/N_{acq})m} e^{+j(2\pi/N_{acq})n} \\ &= \frac{\sigma_v^2}{N_{acq}}. \end{aligned} \tag{18}$$

From Eqs. (16) and (18), the SNR at the output of the resonator can be expressed as

$$\frac{S_0}{N_0} \approx \frac{(C_k^2/2)}{\sigma_v^2/N_{acq} + \sum_{i=1, i\neq k}^{m} (L^2(i)/2)}. \tag{19}$$

The first and second terms in the denominator of Eq. (19) are the noise and unwanted sinusoidal powers, respectively. From Eq. (3) the input SNR per sinusoid is given by

$$\frac{S_i}{N_i} = \frac{C_k^2}{2\sigma_v^2}. \tag{20}$$

The processing gain (PG) [4] which is defined in terms of the ratio of the output SNR to the input SNR is therefore given by

$$\mathrm{PG} = \frac{S_0/N_0}{S_i/N_i} = \frac{S_0}{N_0} \approx \frac{\sigma_v^2}{\sigma_v^2/N_{acq} + \sum_{i=1, i\neq k}^{m} (L^2(i)/2)}. \tag{21}$$

Eq. (21) provides a suitable means for selecting the value of $N_{acq}$ in terms of the desired accuracy. Further, Eq. (21) clearly shows that the criteria for selecting $N_{acq}$ depend on the required accuracy and signal conditions. For the case of a single sinusoid buried in white noise Eq. (21) reduces to

$$\mathrm{PG} = N_{acq}. \tag{22}$$

Since an estimate for phase and amplitude of sinusoid is available at every sample time, the value of $N_{acq}$ can be selected arbitrarily large to get more accurate results. This is not the case for the conventional Goertzel algorithm. The algorithm recipe is summarised in Table 1. For comparison purposes, the conventional Goertzel algorithm is also given in Table 2. The signal flow graph of the proposed algorithm is shown in

Table 1
The new proposed sliding Goertzel algorithm

for $k = 1, \ldots, m$

**Step 1:** The second-order digital resonator is set exactly at the $k$th input frequency. $N_{acq}$ is obtained according to the desired processing gain using Eq. (21).
**Step 2:** All the state conditions are set to zero.
**Step 3:** The input signal is applied to the resonator, and at the $N_{acq}$th sampling time, the following equations are computed:

(1) $\hat{A}_k(N_{acq}) = \dfrac{2(y_k(N_{acq}) - y_k(N_{acq}-1)\cos\omega_k)}{N_{acq}}$,

(2) $\hat{B}_k(N_{acq}) = -\dfrac{2\sin\omega_k\, y_k(N_{acq}-1)}{N_{acq}}$.

**Step 4:** Estimates for sinusoidal coefficients are computed as

$$\begin{bmatrix} a_k \\ b_k \end{bmatrix} \approx \begin{bmatrix} \cos\omega_k N_{acq} & -\sin\omega_k N_{acq} \\ \sin\omega_k N_{acq} & \cos\omega_k N_{acq} \end{bmatrix} \begin{bmatrix} \hat{A}_k(N_{acq}) \\ \hat{B}_k(N_{acq}) \end{bmatrix}.$$

Table 2
The conventional Goertzel algorithm

for $k = 1, \ldots, m$

**Step 1:** The second-order digital resonator is set at frequency $2\pi k/N$ where $k$ is selected such that $2\pi k/N$ becomes as close as possible to the input normalised frequency. Note that $N$ must be equal to $f_s/\Delta f$, if accurate estimates of sinusoids are required.
**Step 2:** All the state conditions are set to zero.
**Step 3:** The input signal is applied to the resonator, and at the $N$th time instant, coefficients are computed as follows:

(1) $a_k = \dfrac{2(y_k(N) - y_k(N-1)\cos\omega_k)}{N}$,

(2) $b_k = -\dfrac{2\sin\omega_k\, y_k(N-1)}{N}$.

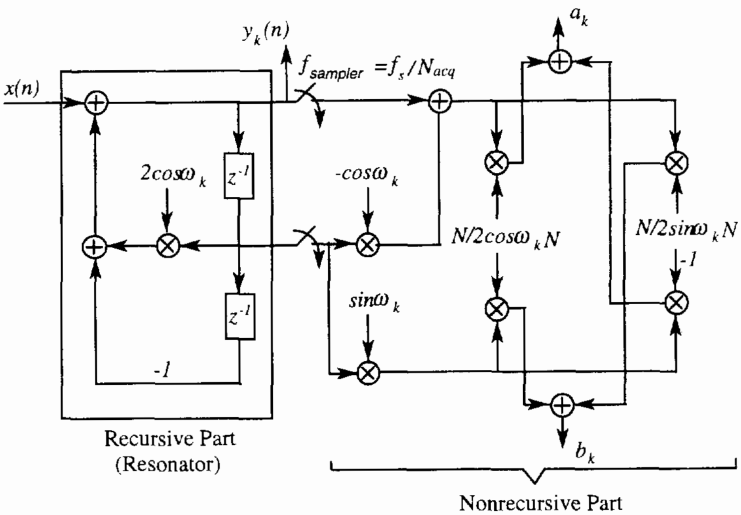



Fig. 2. Digital resonator and sliding Goertzel algorithm realisation.

Fig. 2. The frequency of the sampler in Fig. 2 is given by

$$f_{\text{sampler}} = \frac{1}{t_{\text{acq}}} = \frac{f_s}{N_{\text{acq}}}. \tag{23}$$

Note that the conventional, modified and proposed sliding Goertzel algorithms need only one addition and one multiplication per output sample for implementing the recursive part. In order to implement the nonrecursive part, the modified and conventional Goertzel algorithm requires two multiplications and one addition while the proposed approach requires six multiplications and three additions. Since the nonrecursive part is performed only at the $N_{\text{acq}}$th time instant, the overall computational complexity has been slightly increased by four more multiplications and two more additions per input sinusoid.

## 4. Simulation results

Simulation tests have been carried out to evaluate the performance of the proposed algorithm. Two cases are considered. First, the algorithm is used for the detection of the DTMF signal. The DTMF signal is composed of two frequencies which are selected from a low and a high frequency group. In this case, the GCD of the two components is equal to 1 Hz. Secondly, the algorithm will also estimate the phase and amplitude of a single sinusoid whose parameters are time varying within the successive sequences.

Table 3
Parameters of DTMF signal

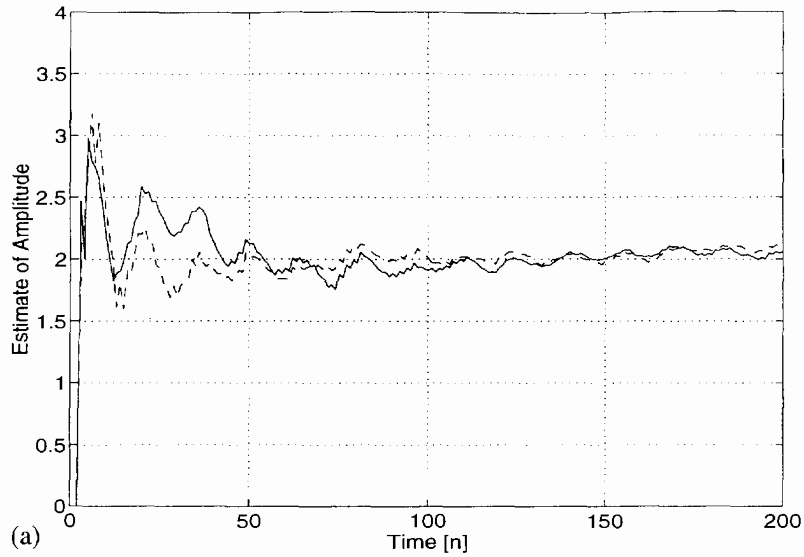
| | Low freq. component | | High freq. component |
|---|---|---|---|
| Freq. | 941 Hz | | 1209 Hz |
| Amplitude | 2 | | 2 |
| Phase | 1 (rad) | | 1 (rad) |
| Sampling freq. | | 8000 Hz | |
| SNR | | 3 dB | |

**Example 1.** The proposed sliding Goertzel algorithm is used for the task of Fourier coefficient estimation of a DTMF signal. Consider the following DTMF signals:

$$x(n) = \sum_{i=l,h} C_i \cos\left(\frac{2\pi n f_i}{f_s} - \phi_i\right) + v(n), \tag{24}$$

where $v(n)$ is zero mean white noise. The parameters of the signal are given in Table 3.

Fig. 3 shows the resulting trajectories for amplitude and phase estimates of the input signal components. The amplitude estimates when the frequencies have not been transmitted are also depicted in Fig. 4. From



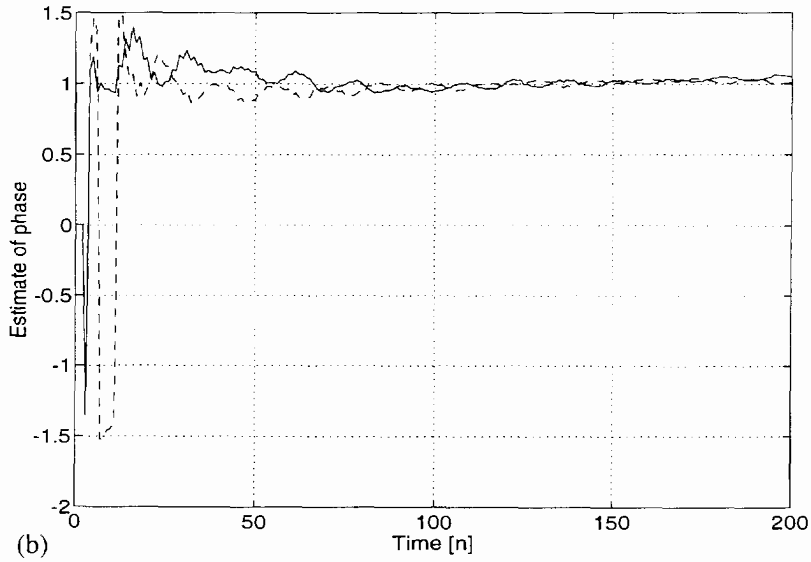




Fig. 3. Sliding computation of (a) amplitude ($C_l, C_h$) and (b) phase ($\phi_l, \phi_h$) of the DTMF (low frequency component (dashed line), high frequency component (solid line)).

Fig. 4 it is clear that the contribution of noise is insignificant after 100 samples when compared with the magnitude of the sinusoidal component. Fig. 5 corresponds to the case when the two closest frequencies ($f_l$ = 852 Hz and $f_h$ = 1336 Hz) are transmitted while the resonators are tuned at 941 and 1209 Hz. It is evident that when the two most adjacent frequencies are transmitted, after 100 samples the receiver does not falsely detect another DTMF signal. Once the transient time of the filter has elapsed, detection can be accomplished by using an appropriate threshold. In other words, a valid detection of the signal is assumed if the magnitude of the detected signal is larger than the threshold. The results clearly show that the sinusoidal coefficients are computed very fast indeed. Note that if the conventional or modified Goertzel

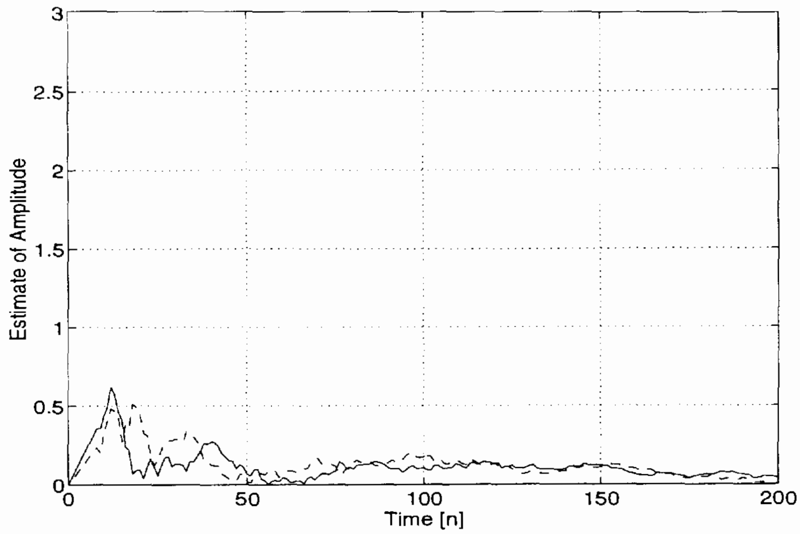



Fig. 4. Sliding amplitude computation ($C_l, C_h$) using sliding Goertzel algorithm in the presence of unit variance, zero mean white noise only (low frequency component (dashed line), high frequency component (solid line)).

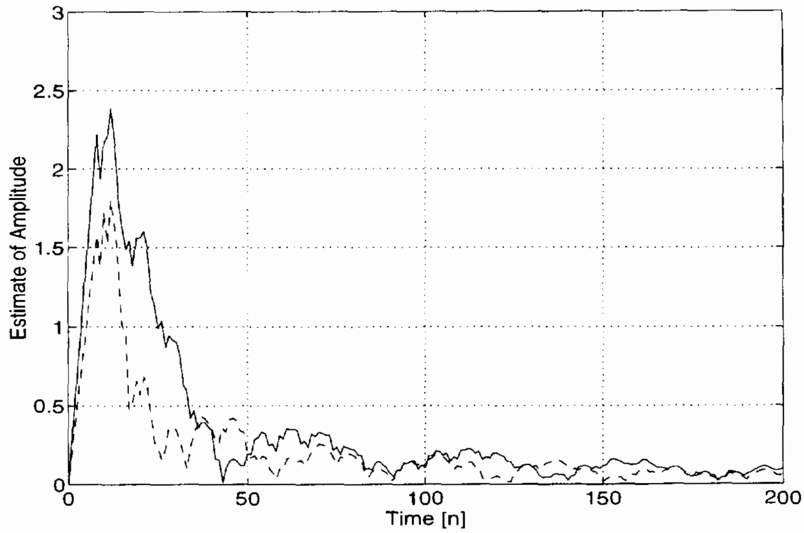



Fig. 5. Sliding amplitude computation ($C_l, C_h$) using sliding Goertzel algorithm when the two most adjacent frequencies ($f_l = 852$ Hz, $f_h = 1336$ Hz) are transmitted (low frequency component (dashed line), high frequency component (solid line)).

algorithms are used, at least 8000 samples are needed to accurately estimate both the phase and amplitude of the components.

Based on 100 independent experiments, the average and standard deviation of amplitude estimates of the low frequency component ($f_l$) using the proposed, conventional and modified Goertzel algorithms were computed at the $N_{\text{acq}} = 256$th time instant. This means that we have applied the signal to the resonator, and at the 256th sampling time, the coefficients are computed and the results are given in Table 4. For the conventional Goertzel algorithm, the index $k$ is found to be equal to 30. Note that since the conventional and modified Goertzel algorithms cannot estimate phase from the 256 samples available, these have not been included in Table 4. It is also evident that the sliding

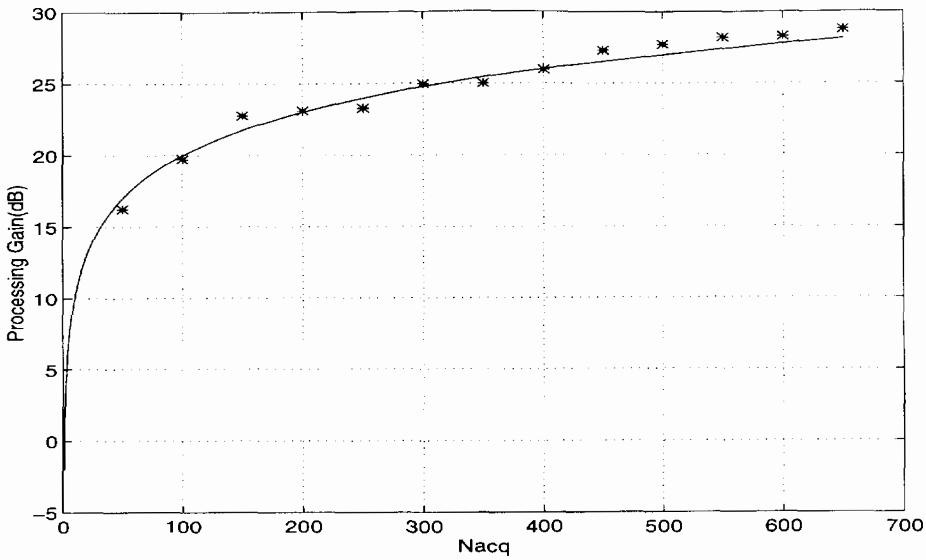



Fig. 6. Theoretical (solid line) and experimental (marked points) processing gain (PG) versus $N_{acq}$.

Table 4
The average and standard deviation of phase and amplitude estimates of the low frequency component ($f_l$) of the DTMF signal (Example 1) using conventional, modified and the proposed sliding Goertzel algorithms ($N_{acq} = 256$, SNR = 3 dB)

| | | Sliding Goertzel algorithm | Modified Goertzel algorithm | Conventional Goertzel algorithm |
|---|---|---|---|---|
| Amplitude ($f_l$) | Ave. | 2.0158 | 1.9874 | 1.9610 |
| | STD | 0.0910 | 0.1115 | 0.1312 |
| Phase ($f_l$) | Ave. | 0.9900 | — | — |
| | STD | 0.0441 | — | — |

Goertzel algorithm provides similar accuracy for amplitude computation as does the modified Goertzel algorithm. Both the sliding and modified Goertzel algorithms perform better than the conventional Goertzel algorithm.

As stated earlier the accuracy of the proposed algorithm can be increased by selecting a greater value for $N_{acq}$ when the sinusoidal coefficients are to be determined. Fig. 6 shows the processing gain (PG) versus $N_{acq}$. To estimate the power of the output noise, the algorithm has been performed over 100 independent trials for each value of $N_{acq}$. The theoretical values of PG versus $N_{acq}$ are also included in Fig. 6 and clearly confirm the theoretical analysis presented in Section 3. As expected, as $N_{acq}$ increases, the accuracy also improves. Hence the value of $N_{acq}$ can be selected to yield the required level of estimate accuracy. Naturally as $N_{acq}$ increases, the acquisition time also increases which means that one needs to trade acquisition time for accuracy, or vice versa.

The performance of the sliding Goertzel algorithm under various SNR conditions has also been examined. At each SNR condition, the estimates are computed based on 100 independent experiments. The standard deviation and average error of the estimates for the amplitude and phase are shown in Figs. 7 and 8, respectively. The average errors of the estimates for amplitude and phase were computed as follows:

$$\overline{\Delta C} = |(\overline{\hat{C}} - C_{\text{actual}})/C_{\text{actual}}|,$$
$$\overline{\Delta \phi} = |(\overline{\hat{\phi}} - \phi_{\text{actual}})/\phi_{\text{actual}}|. \tag{25}$$

In Figs. 7 and 8, the parameter $N$ indicates the sample number at which the algorithm was performed. It is seen that the algorithm exhibits good performance at high SNR. Next we consider the condition when a mismatch exists between the resonance frequency of the resonator and the actual input sinusoidal frequencies. This reflects the practical situation where there may exist some deviation in input frequencies. For

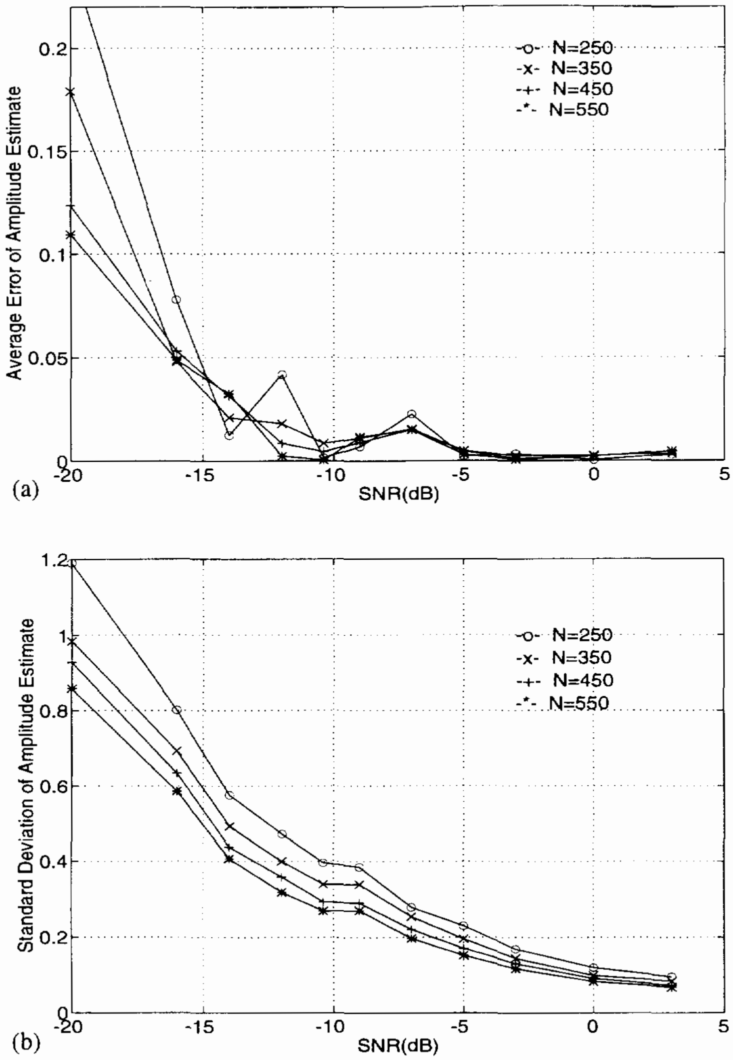
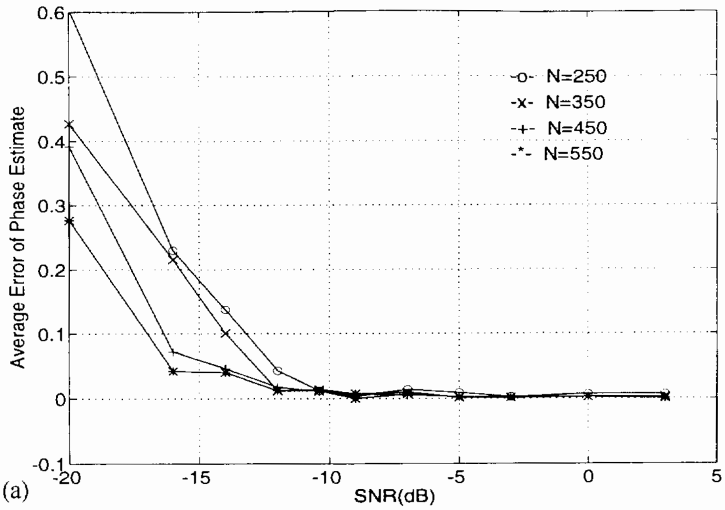



Fig. 7. (a) Average error and (b) standard deviation of amplitude estimate versus SNR.

the given example, assume ±1.0% frequency deviation in the low frequency component of the input signal. Using 100 different trials, the standard deviation and the average of the Fourier coefficients were determined and the results are depicted in Table 5. The input frequency was randomly selected from a maximum possible range of values. As can be observed, the performances (in terms of accuracy) of the proposed and modified Goertzel algorithms are similar and both provide more accurate estimates when compared with the conventional Goertzel algorithm. Obviously, the accuracy depends on the degree of uncertainty in the

Table 5
The average and standard deviation of phase and amplitude estimates of the low frequency component (Example 1) using sliding, modified and conventional Goertzel algorithms for 1% deviations in frequency ($N_{acq}$ = 256, SNR = 3 dB, freq. deviation = 1%)

| | | Sliding Goertzel algorithm | Conventional Goertzel algorithm | Modified Goertzel algorithm |
|---|---|---|---|---|
| Amplitude | Ave. | 2.0434 | 2.0454 | 1.8806 |
| ($f_l$) | STD | 0.1305 | 0.1285 | 0.2253 |
| Phase | Ave. | 1.0120 | — | — |
| ($f_l$) | STD | 0.1036 | — | — |



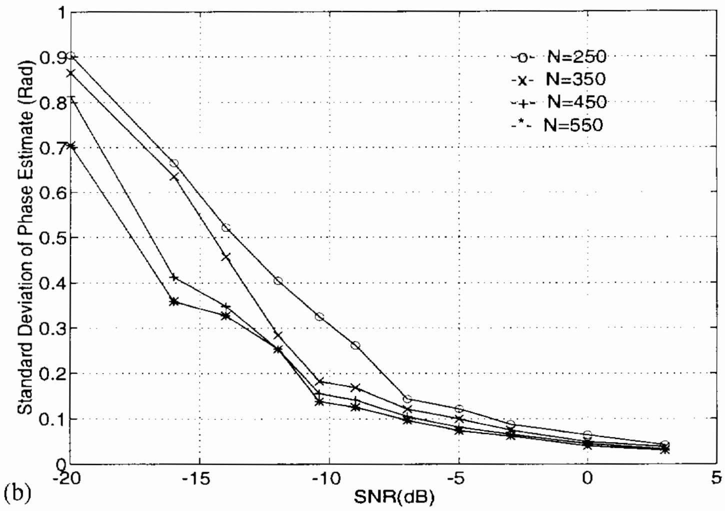




Fig. 8. (a) Average error and (b) standard deviation of phase estimate versus SNR.

input frequency, that is, a more accurate estimate can be obtained as the amount of mismatch reduces. In order to avoid overflows due to the recursive part of the algorithm (resonator), the state variables of the resonator are checked at every $N_{acq}$ sample time, and if the magnitude at any node is halfway to saturation, all state variables are scaled down to a small value near zero.

**Example 2.** In this example we consider the situation where the signals are time varying. Even though the proposed technique provides an estimate update at every sample time, it is necessary to wait for a minimum of $N_{acq}$ samples before the estimate can be considered reliable. Note, however, that $N_{acq}$ is less than one period of the signal. Fig. 9 depicts the amplitude and phase estimates at every 30 sample interval of a single sinusoid whose phase and amplitude changes within successive sequences. The input and sampling frequency were chosen to be equal to 941 and 8000Hz, respectively. The SNR was set at 20 dB and with step changes in amplitude (from 1 to 2) and phase (from 0 to 1 rad) and these were applied simultaneously to the input sinusoid. From Fig. 9, it is evident that the given

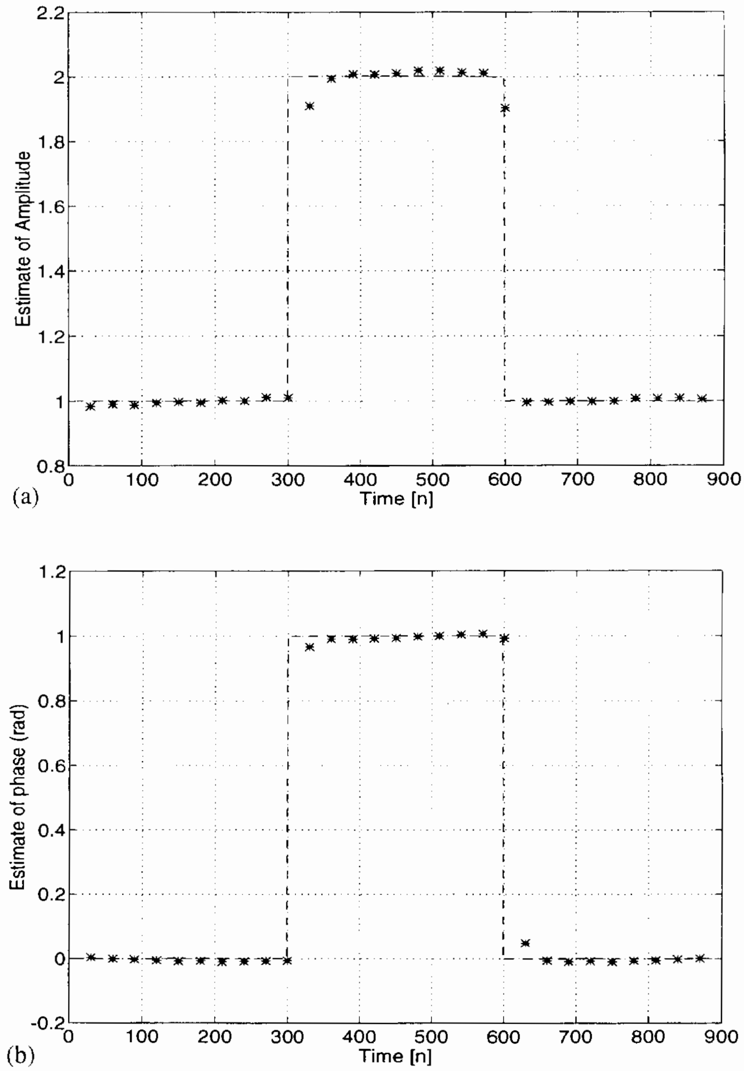



Fig. 9. (a) Amplitude and (b) phase computation of single sinusoid at every 30 samples with step changes in amplitude and phase.

algorithm is capable of tracking the rapid changes in the parameters of the input sinusoid.

## 5. Conclusion

A sliding algorithm similar to the real valued Goertzel algorithm is proposed for estimating the Fourier coefficients of input signals composed of arbitrary frequencies. The algorithm implementation is based on second-order digital resonators which are tuned to the input frequencies. The proposed method computes accurate Fourier coefficients within one period of the signal. This leads to a significant reduction in acquisition time (as compared to the conventional Goertzel algorithm), particularly when the common factor divisor of the input frequencies is small. This means that the proposed algorithm is better suited to the estimation of Fourier coefficients when the parameters are time varying within successive sequences. Simulation tests for the Fourier coefficient estimation of a typical DTMF signal were conducted to evaluate the relative performance of the sliding, modified and conventional Goertzel algorithms.

## Appendix A

To prove Eq. (2), consider the following composite signal:

$$x(n) = \sum_{i=1}^{s} C_i e^{j(2\pi f_i/f_s)n}, \tag{A.1}$$

where $C_i$ and $f_i$ are the amplitude and frequency of the $i$th complex sinusoidal input component, respectively. The DFT of a sequence $x(n)$ of length $N$ is then given by

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j(2\pi/N)nk}, \quad 0 \leqslant k \leqslant N-1. \tag{A.2}$$

Substituting Eq. (A.1) into Eq. (A.2), we obtain

$$X(k) = C_l \sum_{n=0}^{N-1} e^{-j(2\pi/N)n(k_l - Nf_l/f_s)} + \sum_{\substack{i=1,\\ i\neq l}}^{s} \sum_{n=0}^{N-1} C_i e^{-j(2\pi/N)n(k_l - Nf_i/f_s)}. \tag{A.3}$$

To compute the $l$th component ($C_l$), the appropriate index $k_l$ is required so that $X(k_l) = NC_l$. This implies that the first summation and the second term in Eq. (A.3) be equal to $N$ and zero, respectively. To meet these two requirements we use the fact that

$$\sum_{n=0}^{N-1} e^{-j(2\pi/N)n(k-r)} = \begin{cases} N & \text{for } k-r = mN \text{ and } m \text{ is an integer,} \\ 0 & \text{for other integer values of } r. \end{cases} \tag{A.4}$$

In order to satisfy the first requirement we obtain

$$k_i - \frac{Nf_i}{f_s} = 0 \quad \text{for } i = 1, \ldots, s. \tag{A.5}$$

Assuming there exist at least two frequencies (say $f_p$ and $f_q$) that have no GCD than $\Delta f$, we have

$$f_p = M_p \Delta f, \quad f_q = M_q \Delta f. \tag{A.6}$$

From Eqs. (A.5) and (A.6), we obtain

$$\frac{M_p}{M_q} = \frac{k_p}{k_q}. \tag{A.7}$$

Since $M_p$ and $M_q$ are relatively prime, the indexes $k_p$ and $k_q$ should be equal to the values of $M_p$ and $M_q$, respectively. Applying this fact to Eq. (A.5), the required block length is given as follows:

$$N = \frac{f_s M_p}{f_p} = \frac{f_s}{\Delta f}. \tag{A.8}$$

Using Eq. (A.8), it can be easily shown that the second summation in Eq. (A.3) is equal to zero. Now we show that the block length given by Eq. (A.8) is a minimum. Considering the case of $j = p$ and $i = q$ in Eq. (A.3), we obtain

$$k_p - \frac{Nf_q}{f_s} = M_p - \frac{N}{f_s/\Delta f} M_q = r. \tag{A.9}$$

Since $M_p$ and $M_q$ are relatively prime, the value of $r$ becomes noninteger for $N < (f_s/\Delta f)$. Therefore the second term in Eq. (A.3) will not be identical to zero and this ends the proof.

## Appendix B

When the input frequency is the same as the resonance frequency of the resonator, the $z$-transform of the output is given by

$$y_k(z^{-1}) = \frac{a_k(1 - \cos\omega_k z^{-1}) + b_k \sin\omega_k z^{-1}}{(1 - 2\cos\omega_k z^{-1} + z^{-2})^2}. \tag{B.1}$$

Considering the following $z$-transform pairs [5]:

$$p_1(n) = n \sin\omega_k n \Leftrightarrow$$
$$P_1(z^{-1}) = \sin\omega_k z^{-1} \frac{1 - z^{-2}}{(1 - 2\cos\omega_k z^{-1} + z^{-2})^2}, \tag{B.2}$$

$$q_1(n) = n \cos\omega_k n \Leftrightarrow$$
$$Q_1(z^{-1}) = \frac{\cos\omega_k z^{-1} - 2z^{-2} + \cos\omega_k z^{-3}}{(1 - 2\cos\omega_k z^{-1} + z^{-2})^2}, \tag{B.3}$$

together with the following equalities:

$$\cos\omega_k p_1(n) - \sin\omega_k q_1(n) = n\sin[\omega_k(n-1)], \tag{B.4}$$

$$\sin\omega_k q_1(n) + \cos\omega_k p_1(n) = n\sin[\omega_k(n+1)], \tag{B.5}$$

with some straightforward manipulations the following $z$-transform pair will be obtained:

$$p_2(n) = (n+2)\sin[\omega_k(n+1)] \Leftrightarrow$$
$$P_2(z^{-1}) = 2\sin\omega_k \frac{1-\cos\omega_k z^{-1}}{(1-2\cos\omega_k z^{-1}+z^{-2})^2}, \tag{B.6}$$

$$q_2(n) = n\sin[\omega_k(n+1)] \Leftrightarrow$$
$$Q_2(z^{-1}) = 2\sin\omega_k z^{-1} \frac{\cos\omega_k - z^{-1}}{(1-2\cos\omega_k z^{-1}+z^{-2})^2}. \tag{B.7}$$

From Eqs. (B.6) and (B.7), it can be found that

$$\frac{1}{2\sin^2\omega_k}(P_2(z^{-1})\cos\omega_k - Q_2(z^{-1})z) = \frac{\sin\omega_k z^{-1}}{(1-2\cos\omega_k z^{-1}+z^{-2})^2}. \tag{B.8}$$

Using Eqs. (B.8) and (B.6), the output of the resonator in time domain, $y_k(n)$, can be obtained. Now let us consider the case when the input frequency ($\omega_i$) is different from the resonant frequency of the resonator ($\omega_k$). Without loss of generality consider the following input sinusoid which is only composed of a sine term:

$$x(n) = C_i \sin\omega_i n. \tag{B.9}$$

For the input signal given by Eq. (B.9), the resonator output in the $z$-domain is given by

$$Y_i(z) = \frac{C_i\sin\omega_i z^{-1}}{(1-2\cos\omega_i z^{-1}+z^{-2})(1-2\cos\omega_k z^{-1}+z^{-2})}. \tag{B.10}$$

Using partial-fraction expansion, we have

$$Y_i(z) = \frac{1}{-2(\cos\omega_i - \cos\omega_k)} \times \left(\frac{-C_i\sin\omega_i}{1-2\cos\omega_k z^{-1}+z^{-2}} + \frac{C_i\sin\omega_i}{1-2\cos\omega_i z^{-1}+z^{-2}}\right). \tag{B.11}$$

From Eq. (B.11), the output in time domain is easily obtained and given by Eq. (15).

## References

[1] J.A. Beraldin and W. Steenaart, "Overflow analysis of a fixed-point implementation of the Goertzel algorithm", *IEEE Trans. Circuits and Systems*, Vol. 36, No. 2, February 1989, pp. 322–324.

[2] F.Q. Braun, "Nonrecursive digital filters for detecting multifrequency code signals", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-23, No. 3, June 1975, pp. 250–256.

[3] S.L. Gay, J. Hartung and G.L. Smith, "Algorithms for multichannel DTMF detection for the WE DSP32 family", *Proc. Internat. Conf. Acoust. Speech Signal Process.*, Glasgow, Scotland, 1989, pp. 1134–1137.

[4] F.J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform", *Proc. IEEE*, Vol. 66, No. 1, January 1978, pp. 51–83.

[5] E.I. Jury, *Theory and Application of the z-Transform Method*, Wiley, New York, 1964.

[6] I. Koval and G. Gara, "Digital MF receiver using discrete Fourier transform", *IEEE Trans. Comm.*, Vol. COM-21, No. 12, December 1973, pp. 1331–1335.

[7] A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1975.

[8] R.A.Z. Simington and T.M.P. Percival, "New frequency domain technique for DSP based VSAT modems", *Proc. IREE Conf.*, Sydney, Australia, 1991, pp. 428–431.

[9] Y. Tadokoro and K. Abe, "Notch Fourier transform", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. 35, No. 9, September 1987, pp. 1282–1288.