AD-A261 839

②

NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
ELECTE
MAR 1 0 1993
S
E
D

PSEUDO WIGNER-VILLE DISTRIBUTION,
COMPUTER PROGRAM AND ITS APPLICATIONS
TO TIME-FREQUENCY DOMAIN PROBLEMS

by

Jae-Jin Jeon and Young S. Shin

Department of Mechanical Engineering

March 1, 1993

93-05095

UNCLASSIFIED

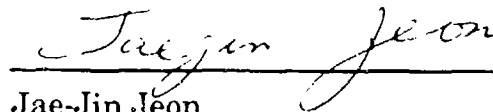Naval Postgraduate School
Monterey, California

Real Admiral T. A. Mercer   H. Shull
Superintendent   Provost
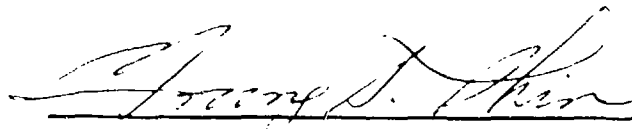

This report was prepared in conjunction with research conducted for and funded by the Naval Sea Systems Command, PMS390, Washington D.C.

This report was prepared by:

_____
Jae-Jin Jeon
Visiting Professor of Mechanical Engineering

_____
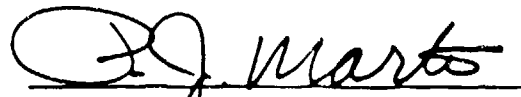Y. S. Shin
Professor of Mechanical Engineering

Reviewed by: Released by:

_____   _____
Matthew D. Kelleher   Paul J. Marto
Chairman   Dean of Research
Dept. of Mechanical Engineering

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information including suggestions for reducing this burden to Washington Headquaters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302 and to the Office of Management and Budget, Paperwork Reduction Project 0704-0188, Washington, DC 20503

1. AGENCY USE ONLY(Leave Blank)	2. REPORT DATE 1 Mar. 93	3. REPORT TYPE AND DATES COVERED 1 Oct 92 - 28 Feb 93

4. TITLE AND SUBTITLE

PSEUDO WIGNER-VILLE DISTRIBUTION, COMPUTER PROGRAM
AND ITS APPLICATION TO TIME-FREQUENCY DOMAIN PROBLEMS

5. FUNDING NUMBERS

6. AUTHOR(S)

Jae-Jin Jeon   and   Young S. Shin

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School

Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

NPS-ME-93-002

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Sea Systems Command, PMS 390

Washington, DC 20362-5101

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed are those of the authors and do not reflect the official policy or
position of DOD or US Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release:

Distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Machinery operating in non-stationary mode generates a signature which at each
instant of time has a distinct frequency.   A time-frequency domain representation
is needed to characterize such signature. Pseudo Wigner-Ville distribution is
ideally suited for portraying non-stationary signal in the time-frequency domain and
carried out by adapting the fast Fourier transform algorithm.   The important
parameters affecting the pseudo Wigner-Ville distribution are discussed and
sensitivity analyses are also performed.   Practical examples of an actual transient
signal are used to illustrate its dynamic features jointly in time and frequency.

14. SUBJECT TERMS

Pseudo Wigner-Ville Distribution, Analytic Signal,
Hilbert Transform, Gaussian Window,
Time-Frequency Domain

15. NUMBER OF PAGES

83

16. PRICE CODE

17 SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18 SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19 SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20 LIMITATION OF ABSTRACT SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)

TABLES OF CONTENTS

TABLES OF CONTENTS

PSEUDO WIGNER-VILLE DISTRIBUTION, COMPUTER PROGRAM AND ITS APPLICATIONS TO TIME-FREQUENCY DOMAIN PROBLEMS

by

Jae-Jin Jeon and Young S. Shin

ABSTRCT

Machinery operating in non-stationary mode generates a signature which at each instant of time has a distinct frequency. A time-frequency domain representation is needed to characterize such signature. Pseudo Wigner-Ville distribution is ideally suited for portraying non-stationary signal in the time-frequency domain and carried out by adapting the fast Fourier transform algorithm. The important parameters affecting the pseudo Wigner-Ville distribution are discussed and sensitivity analyses are also performed. Practical examples of an actual transient signal are used to illustrate its dynamic features jointly in time and frequency.

I. INTRODUCTION

The physical condition or state of health of machineries which operate in transient or other non-stationary modes are difficult to predict with any degree of accuracy. It is common to practice periodic preventive maintenance on these machineries in order to avoid failures and prolong the useful operating life of the equipment.

In order to assess the physical condition of machinery without complete disassembly, a physical measurement of its vibrations is conducted using an accelerometer. Other sensors, such as temperature or pressure transducers, could also be used. There are other methods, including motor current signature analysis on electrically driven machinery and wear debris analysis which could be used. However, vibrations are used predominantly for machinery condition monitoring. The vibrations are recorded in the time domain.

There is a need for a method to represent the time dependent events which occur with machinery operating in non-stationary modes. At each instant in time as the speed of the machinery changes, the frequency content will also change. The pseudo Wigner-Ville distribution(PWVD) is the method which was chosen to portray these time dependent changes. This is a continuation of work initially performed and published by Rossano, Hamilton and Shin [1].

The pseudo Wigner-Ville distribution is a three dimensional (time, frequency, amplitude) representation of an input signal and is ideally suited for describing transient or other non-stationary phenomena. The Wigner Distribution (WDF) has been used in the areas of optics [2,3,4] and speech analysis [5,6]. Wahl and Bolton [7] used it to identify structure-borne noise components. Flandrin et. al. [8] recently proposed its use in the area of machinery condition monitoring and diagnostics, while Forrester [9] is investigating its use in gear fault detection.

For such a non-stationary signal analysis, spectrogram is commonly used, which is based on the assumption that it is a collection of a short duration stationary signals. A major drawback of this approach is that the frequency resolution is directly affected by the duration of short stationary time, which subsequently determines the time resolution. A method for time-frequency domain signal characterization that overcomes this drawback is the Wigner distribution which was first introduced by Wigner [10] in 1932 to study the problem of statistical equilibrium in quantum mechanics. The

frequency and time resolutions of the Wigner distribution are not determined by the short duration but rather determined by the selection of desired resolution of the signal itself.

This paper discusses the important parameters affecting the PWVD in order to machinery condition monitoring and presents numerical examples of PWVD using synthetically generated signals. It is found that the PWVD is very effective in machinery condition monitoring which operates in non-stationary modes.

II. PSEUDO WIGNER-VILLE DISTRIBUTION

A.   Wigner Distribution Function

Signal associated with most vibrational phenomena are in general time varying, which means that their characteristics change with time and they have various features in different time frames. The general spectrogram usually requires a large time-bandwidth product to reduce the estimated bias and variability. In the case of a signal containing some transients or nonstationary conditions, the traditional approach in signal analysis fails to describe the dynamics of the signal's frequency component changes.

The general expression of the time-frequency distribution of a signal, $w(t,\omega)$ is given by, [11]

$$w(t,\omega) = \frac{1}{2\pi} \iiint e^{-j\theta t - j\tau\omega - j\theta u} \phi(\theta,\tau) \, s^*(u - \frac{\tau}{2}) \, s(u + \frac{\tau}{2}) \, du \, d\tau \, d\theta \qquad (1)$$

where $s(u)$ is the time signal, $s^*(u)$ is its complex conjugate, and $\phi(\theta,\tau)$ is an arbitrary function called the kernel. By choosing different kernels, different distributions are obtained. Wigner distribution is obtained by taking $\phi(\theta,\tau)$ = 1. The range of all integrations is from $-\infty$ to ∞ unless otherwise noted.

Substituting the kernel $\phi(\theta,\tau)$ = 1 to Eq. (1),

$$w(t,\omega) = \iint s^*(u - \frac{\tau}{2}) \, e^{-j\tau\omega} \delta(u - t) \, s(u + \frac{\tau}{2}) \, d\tau \, du \qquad (2)$$

From Eq. (2) the Wigner distribution is obtained,

$$w(t,\omega) = \int s^*(t - \frac{\tau}{2}) \, s(t + \frac{\tau}{2}) \, e^{-j\tau\omega} \, d\tau \qquad (3)$$

One of the basic frequency representations of a signal is the power density spectrum, which characterizes the signal's frequency component distribution. The power spectral density function $S(\omega)$ of a signal s(t) can be related to the Fourier transform of the signal's autocorrelation function $R(\tau)$:

$$S(\omega) = \int e^{-j\omega\tau} R(\tau) \, d\tau \tag{4}$$

with

$$R(\tau) = \int s(t) \, s(t+\tau) dt . \tag{5}$$

From this relation a time-dependent power spectral density function can be written as

$$w(t,\omega) = \int R_t(\tau) \, e^{-j\omega\tau} d\tau \tag{6}$$

where now $R_t(\tau)$ is a time-dependent or local autocorrelation function. Mark [12] argued for symmetry,

$$R_t(\tau) = s^*(t - \frac{\tau}{2}) \, s(t + \frac{\tau}{2}) \tag{7}$$

which gives the Wigner distribution function.

B. Properties of the Wigner distribution function

The properties of the WDF [13,14] are summarized and reinterpreted with this new formulation as follows:

(i) The WDF is a real-valued function.

$$w^*(t,\omega) = w(t,\omega) \tag{8}$$

5

(ii) The integral of the WDF with respect to frequency and time yields the instantaneous signal power and the signal's power density spectrum, respectively.

$$\int_{-\infty}^{\infty} w(t,\omega)\, d\omega = 2\pi |s(t)|^2 .$$

(9)

$$\int_{-\infty}^{\infty} w(t,\omega)\, dt = 2\pi |S(\omega)|^2 .$$

(iii) A time or frequency shift in the signal have the shift in the WDF.

If $s(t) \rightarrow s(t+t_0)$, then $w(t,\omega) \rightarrow w(t+t_0,\omega)$, (10)

if $s(t) \rightarrow e^{j\omega_0 t} s(t)$, then $w(t,\omega) \rightarrow w(t,\omega+\omega_0)$. (11)

(iv) The WDF is symmetrical in time for a given signal.

If $s(t) \rightarrow s(-t)$, then $w(t,\omega) \rightarrow w(-t,\omega)$, (12)

if $s(t) \rightarrow s^*(t)$, then $w(t,\omega) \rightarrow w(t,-\omega)$. (13)

(v) The WDF is not always positive.

(vi) The integration of the square of the WDF equals the square of the time integration of the signal's power. This is the counterpart of Parseval's relation of the WDF, called Mayol's fomula.

$$\int_{-\infty}^{\infty} |w(t,\omega)|^2\, dt\, d\omega = \left| \int_{-\infty}^{\infty} s^2(t)\, dt \right|^2$$

(14)

(vii) The WDF possesses basic symmetry under the interchange of time and frequency parameters with the Fourier transform of a given signal.

Let

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{j\omega t} S(\omega)\, d\omega;$$

(15)

then

$$w(t,\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{j\zeta t} S(\omega + \frac{\zeta}{2}) S^*(\omega - \frac{\zeta}{2}) \, d\zeta. \qquad (16)$$

III. IMPLEMENTATION WITH DIGITAL SIGNAL PROCESSING

A. Discrete Wigner Distribution Function

There are two distinct advantages for the calculation of the WDF. First, it has the form of the Fourier transform and the existing FFT algorithm can be adapted for its computation. Second, for a finite time signal, its integration is finite within the record length of the existing signal.

The discrete time Wigner distribution as developed by Claasen and Mecklenbrauker [13] is expressed by,

$$w(t, \omega) = 2 \sum_{\tau=-\infty}^{\tau=\infty} e^{-j2\omega\tau} s(t + \tau) s^*(t - \tau) \tag{17}$$

The discrete version of Eq. (17) for a sampled signal s(n), where n=0 to N-1, has the form,

$$w(\ell, k) = \frac{1}{N} \sum_{n=0}^{N-1} s(\ell + n) s^*(\ell - n) e^{-j\frac{4\pi}{N}nk}, \quad k=0,1,2,...N-1 \tag{18}$$

where s(m)=0 for m < 0 and m > N-1. However, in order to utilize the FFT algorithm, it must be assumed that the local autocorrelation function has a periodicity of N. This is just for operational convenience and should not apply to the interpretation of s(m) Eq. (18) can be rewritten as,

$$w[\ell, k + m(N/2)] = \frac{1}{N} \sum_{n=0}^{N-1} s(\ell + n) s^*(\ell - n) e^{-j\frac{4\pi}{N}n(k + m\frac{N}{2})}$$

$$= \frac{1}{N} \sum_{n=0}^{N-1} s(\ell + n) s^*(\ell - n) e^{-j\frac{4\pi}{N}nk} e^{-jmn2\pi} \tag{19}$$

$$= w(\ell, k)$$

since $e^{-jmn2\pi} = 1$ for m=integers.

Eq. (19) indicates that the WDF has a periodicity of N/2. Hence, even when the sampling of s(t) satisfies the Nyquist criteria, there are still aliasing components in the WDF. A simple approach to avoid aliasing is to use an analytic signal before computing the WDF. In 1948, J. Ville [15] proposed the use of the analytic signal in time-frequency representations of a real signal.

B.   Analytic Signal

An analytic signal is a complex signal which contains both real and imaginary components. The advantage of using the analytic signal is that in the frequency domain the amplitude of negative frequency components are zero. This satisfies mathematical completeness of the problem by accounting for all frequencies, yet does not limit the practical application since only positive frequency components have a practical interpretation. The imaginary part is obtained by Hilbert transform. The analytic signal may be expressed by,

$$s(t) = s_r(t) + j H\{s_r(t)\} \qquad (20)$$

where $H\{s_r(t)\}$ is a Hilbert transform and generated by the convolution of the impulse response h(t) of a 90-degree phase shift as follows:

$$H\{s_r(t)\} = s_r(t) * h(t)$$

$$h(t) = \frac{2\sin^2(\pi t / 2)}{\pi t}, \qquad t \neq 0, \qquad (21)$$

$$= 0, \qquad t = 0,$$

where * denotes the convolution. Rewriting Eq. (21) to discrete form,

$$H\{s_r(n)\} = \sum_{m=-\infty}^{\infty} h(n - m) \, s_r(m) \qquad (22)$$

9

The distribution resulting from an analytic signal being processed through the Wigner distribution is commonly termed as Wigner-Ville distribution.

C.   WDF with Digtal Signal Processing

To calculate the Wigner distribution of the sampled data, it is necessary that Eq. (18) be modified to Eq. (23), because the WDF has N/2 periodicity.

$$w(m\Delta t, k\Delta\omega) = 2\Delta t \sum_{n=0}^{2N-1} s[(m+n)\Delta t] \, s^*[(m-n)\Delta t] \, e^{-j2\pi mk/(2N)} \quad (23)$$

where $\Delta\omega = \pi / (2N\Delta t)$ and Δt is the sampling interval. The algorithm used in this paper is based on one written by Wahl and Bolton[7] and can be expressed as:

$$w(m\Delta t, k\Delta\omega) = \text{Re} \, [2\Delta t \, \text{FFT}(\text{corr}(i))]$$

$$\text{corr}(i) = s(m+i-1) \, s^*(m-i+1), \quad m \geq i$$
$$= 0, \qquad\qquad\qquad\qquad m < i \qquad\qquad (24)$$

where $\quad 1 \leq i \leq N+1$,

$$\text{corr}(2N-i+2) = \text{corr}^*(i), \qquad 2 \leq i \leq N$$

The frequency resolution, $\Delta\omega$, in Eq. (23) is different from that obtained by FFT of the original N point time record in two respects. The first difference is that the argument of the time signal and its conjugate contains a factor of 1/2, and secondly, the autocorrelation of the time signal is twice the length of the original record and therfore the FFT is evaluated over 2N points. The result is, that the WDF frequency resolution is one forth the resolution of an ordinary power spectrum density function.

Before processing the WDF, a modified Hamming window is applied to the time domain signal to reduce the leakage caused by the discontinuity of the finite record of data, which will be called as data tapering. This type of

10

window is preferable since it alters the amplitude of fewer data points at the beginning and the end of the data block. A modified Hamming window, D(t) is given by:

$$D(t) = \begin{cases} 0.54 - 0.46 * \cos(10\pi t/T), & 0 \leq t \leq T/10, \\ 1.0, & T/10 \leq t \leq 9T/10, \\ 0.54 - 0.46 * \cos(10\pi(T-t)/T), & 9T/10 \leq t \leq T. \end{cases} \quad (25)$$
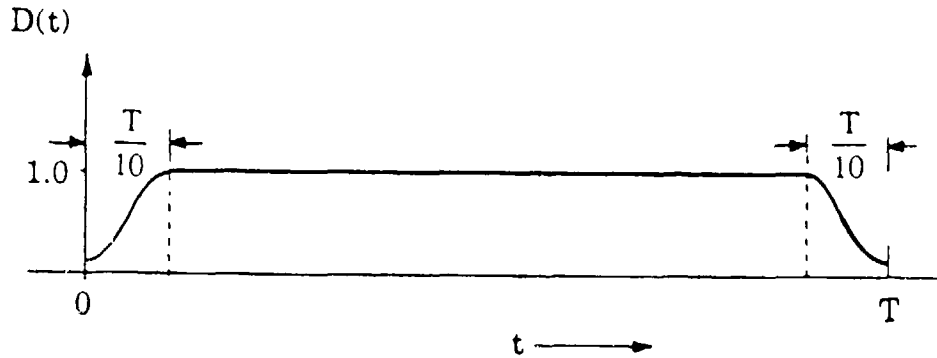
Fig. 1 Modified Hamming Window

Two other characteristics of the WDF should be also noted. First, the WDF of the sum of two signals is equal to the sum of the WDF of each signal plus cross term that appear when the cross-correlation of the two signal is non-zero. Second, the WDF may have negative values, which may be largely caused by interference due to the presence of these cross terms. In the case of input signals that contain multi-frequency components, the Wigner-Ville distribution of most signals are very complicated and difficult to interpret.

There are two methods to suppress the interference components of the WDF. Claasen and Mecklenbrauker[12] describe the application of a sliding window in the time domain before calculating WDF. The WDF obtained with a window function is called the Pseudo- Wigner distribution function. A second option is to smooth the WDF with a sliding averaging window in time-frequency plane. In both case the result is to deemphasize components

arising from calculations and to emphasize deterministic components. Obviously, averaging a Wigner-Ville distribution will result in a Pseudo Wigner-Ville distribution.

In this research, a sliding exponential window in the time-frequency domain was chosen. That is, a Gaussian window function, $G(t, \omega)$ is selected to reduce the interference and to avoid the negative values as follows: let

$$G(t,\omega) = \frac{1}{2\pi\sigma_t\sigma_\omega} e^{-(\frac{t^2}{2\sigma_t^2} + \frac{\omega^2}{2\sigma_\omega^2})} . \tag{26}$$

then

$$w(t,\omega) = \frac{1}{2\pi} \iint w(t',\omega') G(t-t',\omega-\omega') \, dt' \, d\omega' \quad >0 \tag{27}$$

where σ_t, $\sigma_\omega > 0$ and $\sigma_t\sigma_\omega \geq 1/2$ [16]. The time and the frequency resolution Δt and $\Delta\omega$ of this Gaussian window are related by,

$$\sigma_t = j \, \Delta t, \quad \sigma_\omega = k \, \Delta\omega \tag{28}$$

in the discrete form. Then the condition for the WDF to be positive in this case is

$$j \, \Delta t \, k \, \Delta\omega > 1/2. \tag{29}$$

This is the time-frequency version of Heisenberg's uncertainty relation[14]. If the segmentation of time and frequency for a given signal from Eq. (3) violates this uncertainty principle, the corresponding WDF may not be positive.

To perform the convolution on the sampled WDF, the Gaussian window function was applied to the range $\pm 2\sigma_t$ and $\pm 2\sigma_\omega$. Selecting ω and t to be the multiple of time and frequency steps, the sampled Gaussian window function is expressed by,

12

$$G(p,q) = \frac{1}{2\pi j k \Delta t \Delta \omega} e^{-\left(\frac{p^2}{2j^2} + \frac{q^2}{2k^2}\right)} \tag{30}$$

where p and q are an integer numbers in the range ±2j and ±2k, respectively. The convolution of the sampled WDF and the Gaussian window function can be evaluated as follows:

$$w'(\ell,m) = \frac{\Delta t \, \Delta \omega}{2\pi} \sum_{p=\ell-j}^{\ell+j} \sum_{q=m-k}^{m+k} w(p,q) \, G(p-\ell, q-m) \tag{31}$$

where $w'(\ell,m)$ is the smoothed WDF or Pseudo Wigner-Ville distribution.

Fig. 2 shows a block diagram for computational sequence of the Pseudo Wigner-Ville distribution. A time-varying signal sampled with the Nyquist rate is first high passed through a digital filter if the signal involves the zero frequency component, i.e., DC component, and converted into the analytic signal through a Hilbert transform. Then, the time-dependent correlation function is computed and the result is the WDF in terms of both time and frequency domain by FFT. The final step is to compute the convolution with a Gaussian window.
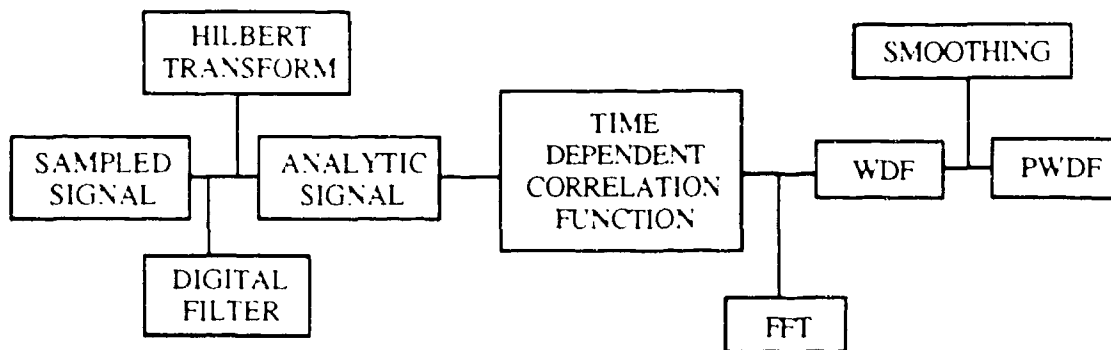

Fig. 2   Computational block diagram of Pseudo Wigner-Ville Distribution

D. Highpass Digital Filter

Filters are a paticularly important class of linear time-invariant system. Strictly speaking, the term frequency-selective filter suggests a system that passes certain frequency components and totally rejects all others, but in a broader context any system that modifies cetain frequencies relative to others is also call filter. The design of filters involves the following stages : (1) the specification of the desired properties of the system; (2) the approximation of the specifications using a causal discrete-time system; and (3) the realization of the system.

In this paper, Nonrecursive(finite impulse response - FIR) highpass filter was used for the elimination of undesired low frequency components. The basic design was to use a symmetric filter of the form,

$$y(i) = \sum_{k=-M}^{M} b_k \, s(i-k) \tag{32}$$

with

$$b_{-k} = b_k \tag{33}$$

and

$$b_k = \frac{\sin 2\pi BkT}{\pi k} \tag{34}$$

where b_k is the filter weights, $y(i)$ is the filtered signal, B is the cutoff frequency, T is a sampling interval and M is the span of the filter; 2M+1 weights are employed because of symmetry, only M+1 need be generated. The b_k weights are computed over the range -M to M. The weights are multiplied by a window function. Potter discusses a number of windows in the referenced work. His P310 window was found to be appropriate for filter implementation. It takes the form,

$$w_k = \frac{c_k}{w} \left[d_0 + 2 \sum_{p=3}^{3} d_p \cos \frac{\pi p k}{M} \right] \tag{35}$$

where

14

$$c_k = \frac{1}{2} \qquad k = \pm M \qquad\qquad (36)$$

$$= 1 \qquad\qquad \text{otherwise}$$

$$d_0 = 1$$
$$d_{-1} = d_1 = 0.684988$$
$$d_{-2} = d_2 = 0.202701$$
$$d_{-3} = d_3 = 0.0177127$$

and

$$w = d_0 + 2 \sum_{p=-3}^{3} d = 2.8108034 \qquad\qquad (37)$$

For a highpass filter with pass band from the cutoff frequency(B) to the maximum frequency, generate a low pass filter on the range 0 - B, and then subtract the central weight from unity and change the signs of the remainder of the weights.

15

IV. EXAMPLES AND DISCUSSIONS

Machinery operating in transient mode generates a signature in which the frequency content varies at each instant of time. To characterize such signatures and to understand the vibrational behavior of such machineries, time-frequency domain representation of the signal is needed. As discussed in the previous sections, Wigner distribution is a signal transformation that is particularly suited for the time-frequency analysis of nonstationary signals. There are many advantages of using PWVD for both steady and transient signals. However, there are also several disadvantages, for example, the drastic increase of peak value when the frequency content of signal changes abruptly. A computer program has been developed for PWVD. Two different versions are available at the present time: workstation and IBM PC compatible.

A. Harmonic Wave

Fig. 3 shows the PWVD of the pure sine wave with two frequency components (100Hz, 400Hz), respectively. The modified Hamming window was applied to the time domain signal and the Gaussian smoothing window function was applied on time-frequency domain Winger-Ville distribution. The slope of the end edges are due to data tapering by using the modified Hamming window. Fig. 4 shows the PWVD of the sine wave that have the 10 % and 50 % signal to noise ratio, respectively. The shape of PWVD is changed at the crest by the contamination of noise. The crest has the complicated shape with decreasing of signal to noise ratio. However, the PWVD well represents the signal components from the given signal with noise. The practical example is shown in Fig.14. The notation f_s and N used in the figures are sampling frequency and the total number of time data points.
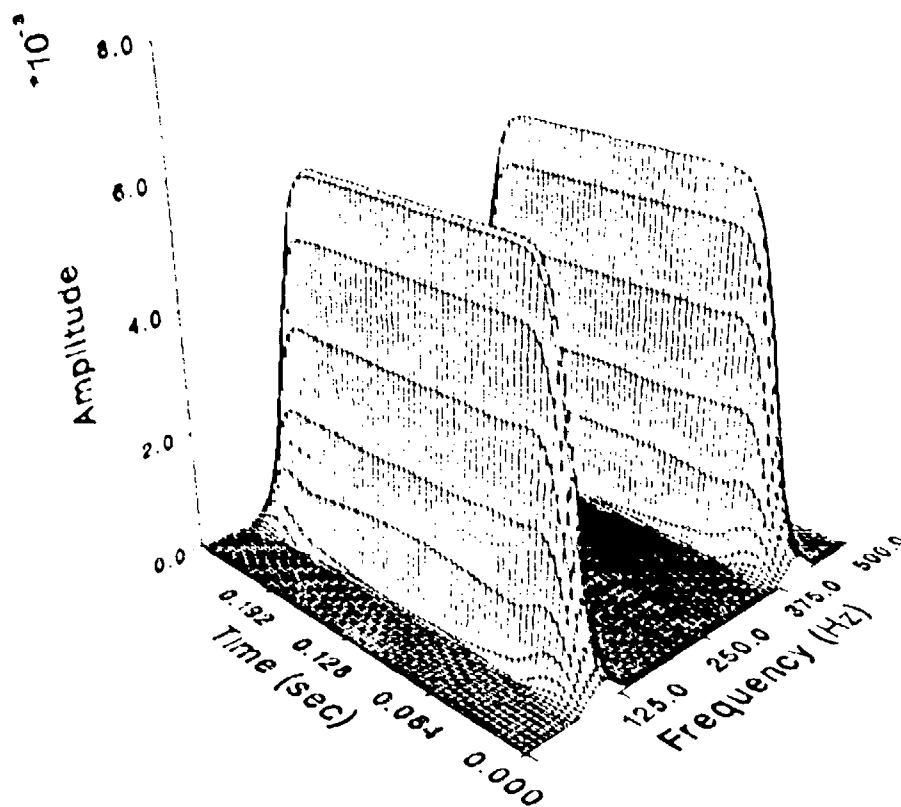
Fig. 3   Pseudo Wigner-Ville distribution of 100 and 400 Hz Pure Sine Waves
(f_s=1000 Hz, N=256 and Smoothing Window Size=10x10)

17
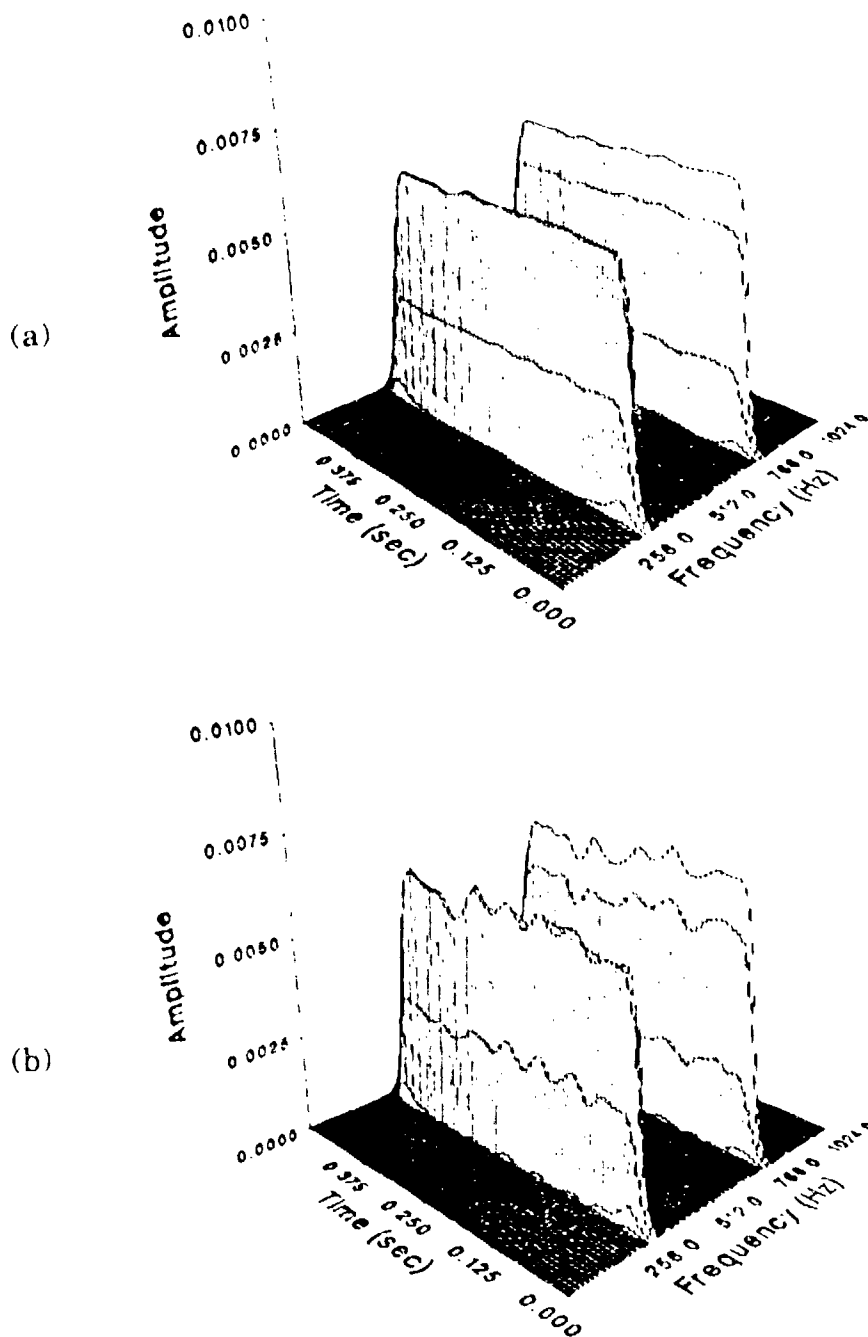
Fig. 4 Pseudo Wigner-Ville distribution of 300 and 750 Hz sine waves; signal to noise ratio (a) 10 % and (b) 50 % .
(f$_S$ = 2048 Hz, N=1024 and smoothing window size=18x18)

18

B.   Harmonic Wave with Stepwise Frequency Changes

Fig. 5 shows the PWVD of the sine wave with 500 Hz in the time from 0.085 sec to 0.17 sec. The PWVD well represents the time delay of the signal. The Fig. 6 shows (a) the sine wave with stepwise frequency changes, 100 Hz, 250 Hz and 500 Hz and (b) its PWVD. The PWVD shows the time delay and frequency component of the signal. The wide spread of PWVD at the edge of each frequency region is noticed. This phenomenon is caused by the discontinuity of the signal in time domain and the leakage in digital signal processing. This effect may be reduced by applying the data tapering to the actual signal block. Nevertheless the PWVD represented the characteristics of the signal well. PWVD can portray the characteristics of the steady state signals involving time delay and multi-frequency components . If different size of the smoothing window are applied, the PWVD amplitude changes, but the total energy remains unchanged.
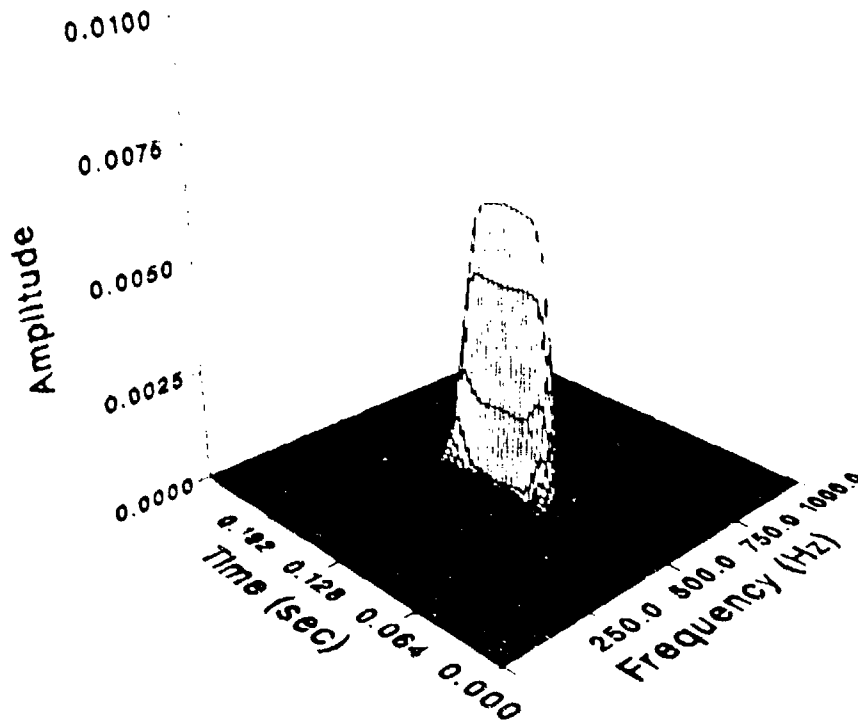
Fig. 5   500 Hz sine wave with finite duration
(f_s=2000 Hz, N=512 and Smoothing Window Size=10x10)
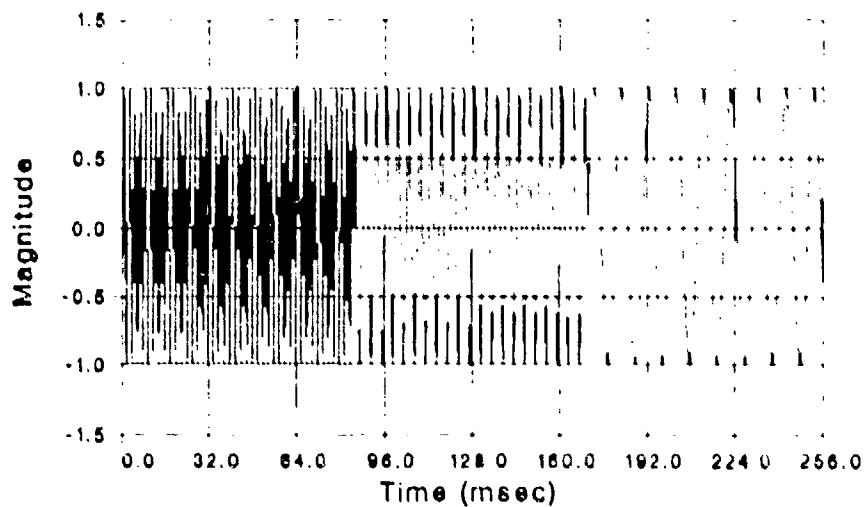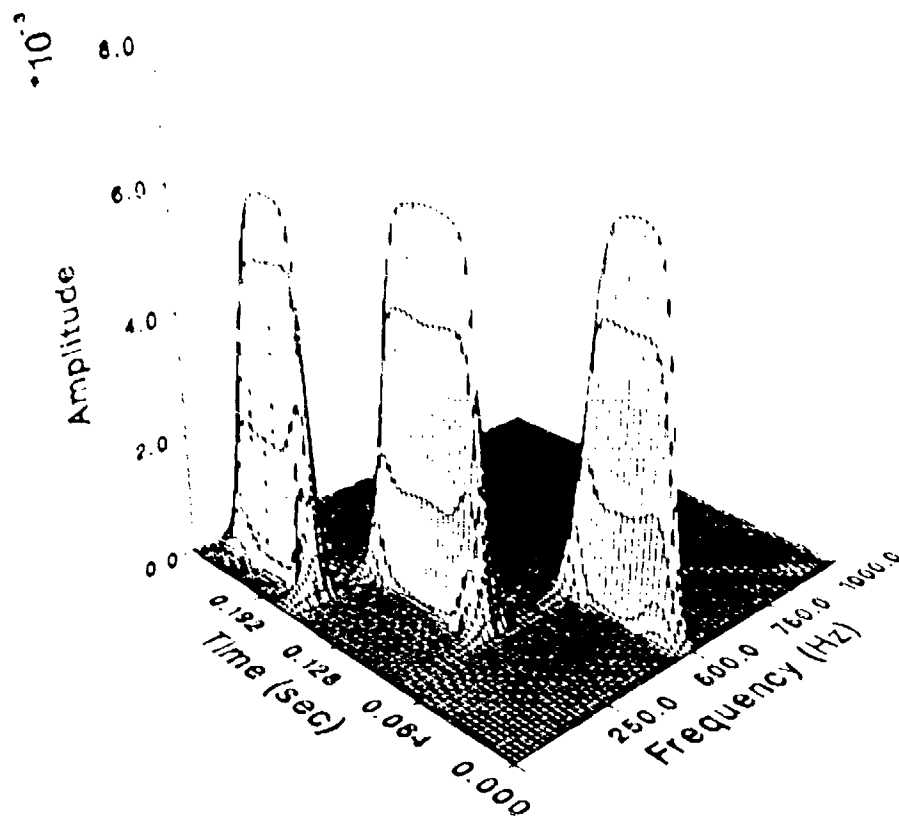
(a) Time signal

(b) PWVD

Fig. 6   Sine Wave with Stepwise Frequency Changes: 100, 250 and 500 Hz
(f_s=2000 Hz, N=512 and Smoothing Window Size=10x10)

20

C. Composite Signal with Two Frequency Components at Each Time

The PWVDs of the nonstationary signals were studied and the results were shown in Fig. 7 through 10. Fig. 7 shows (a) the time signal composed of two sweeping frequency components at each time, one increasing and the other decreasing with the same rate, and (b) its Wigner-Ville distribution (before applying the smoothing window) (c) its contour plot of WDF and (d) its pseudo Wigner-Ville distribution (after applying the smoothing window), respectively.

The effect of cross (or interference) term is significant and appeared in the average frequency region. This is one of the disadvantages of using Wigner-Ville distribution but it is a characteristic of the distribution. When Gaussian window was applied to Wigner-Ville distribution, the effect of cross term disappeared. The main lobe of PWVD is wider and its amplitude is significantly reduced. The large peak at the intersection point of two sweeping frequency signals is mainly caused by the doubling effect of amplitudes of two signals.
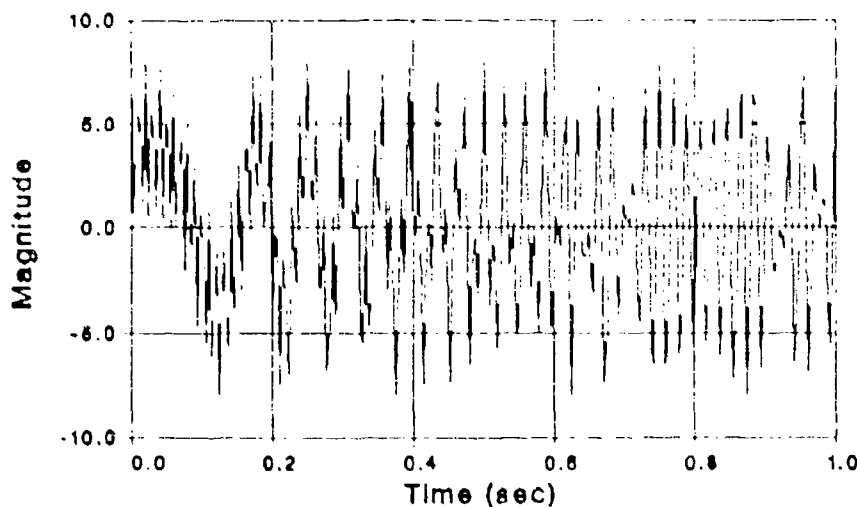
(a) Time signal

Fig.7 (continued)
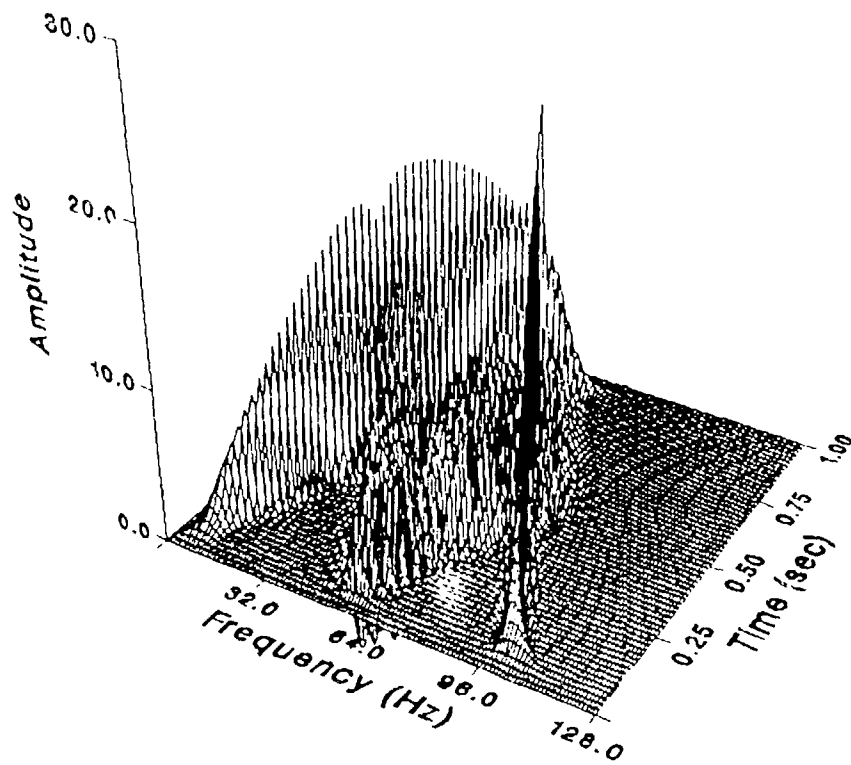
(b) WDF

(c) Contour plot of WDF

Fig.7 (continued)
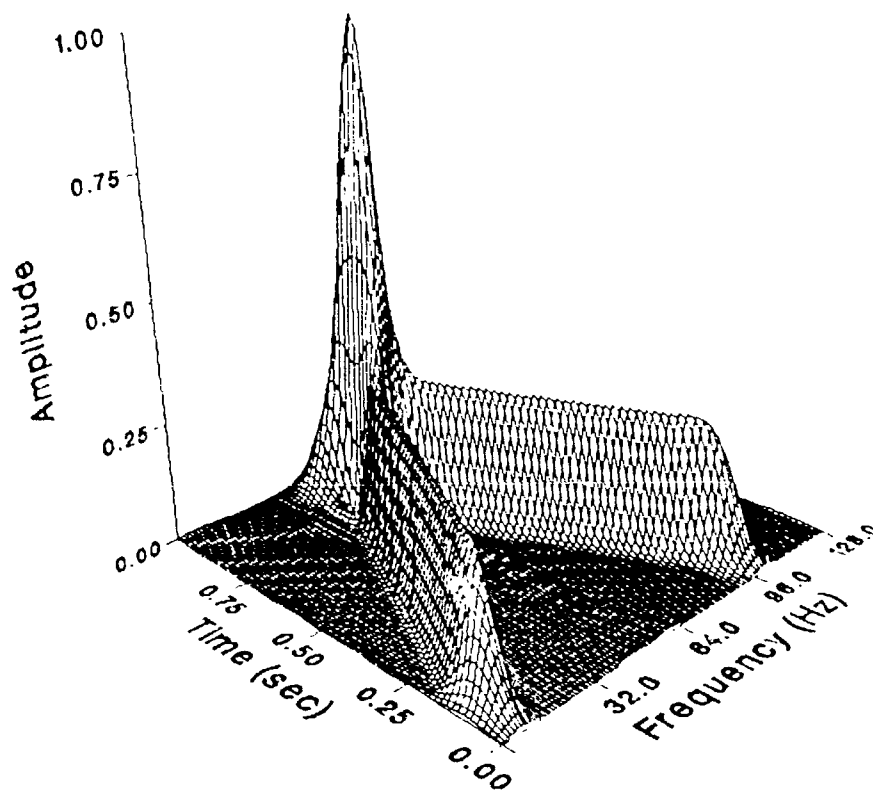
(d) PWVD

Fig. 7 Composite Signal with Two Frequency Components at Each Time

$$s(t)=4\cos(2\pi\ 32t^2) + 4\ \cos\{2\pi(40+32(2-t)]t\}$$

$(f_s=256\ Hz,\ N=256$ and Smoothing Window Size=10x10)

23

D.   Linear Chirp Signal

Another type of a non-stationary signal sweeps up and down in frequency is called a linear chirp signal and is shown in Fig. 8(a). This signal has only one frequency component at each time. The effect of cross terms appears in the Wigner-Ville distribution, as shown in Fig. 7(b). The smoothing window was applied to Wigner-Ville distribution and the result is shown in Fig. 7(d). Fig. 8 (c) is the contour plot of WDF. As expected, the effect of cross term is significantly reduced. However, the unusual peak (called 'ghost' peak) appeared at the point where the direction of sweep changes. To understand the cause of this phenomenon, the PWVD was integrated along the frequency axis and it was found that the square root of the resultant amplitude was the amplitude of original time signal, implying that the energy content remained constant. The following function was used to generate the linear chirp signal:

$$s(t) = \sin\left[2\pi\left(30 + \frac{220(i-1)}{256}\right)t\right], \qquad 1 \le i \le 256$$

$$s(t) = -\sin\left[2\pi\left(30 + \frac{220(512-i)}{256}\right)(0.256 - t)\right], \qquad 256 \le i \le 512$$

(38)

where t = (i-1) dt and dt=0.0005.
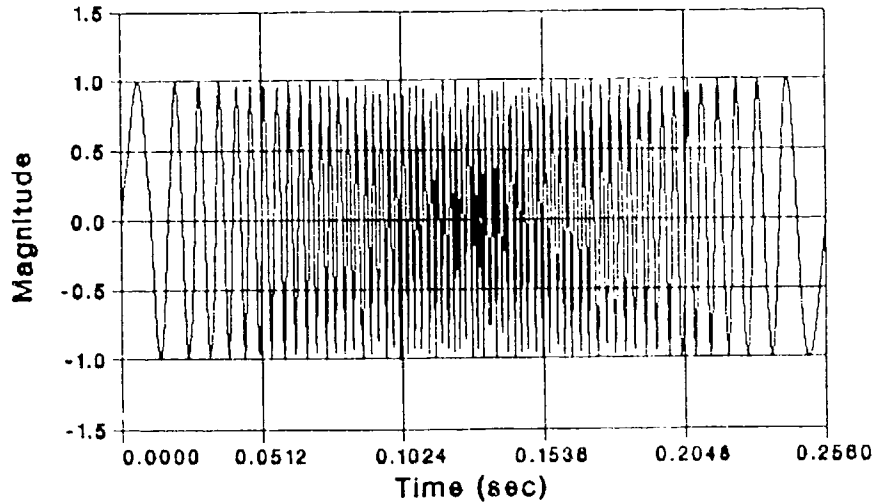
(a) Time signal

Fig.8 (continued)

24
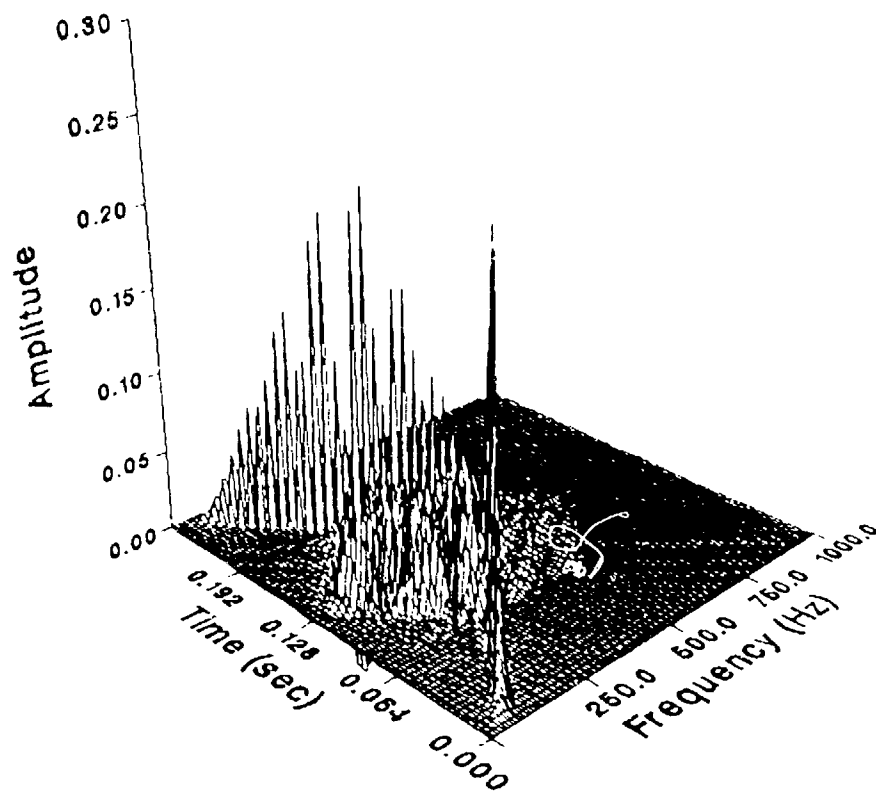
(b) WDF

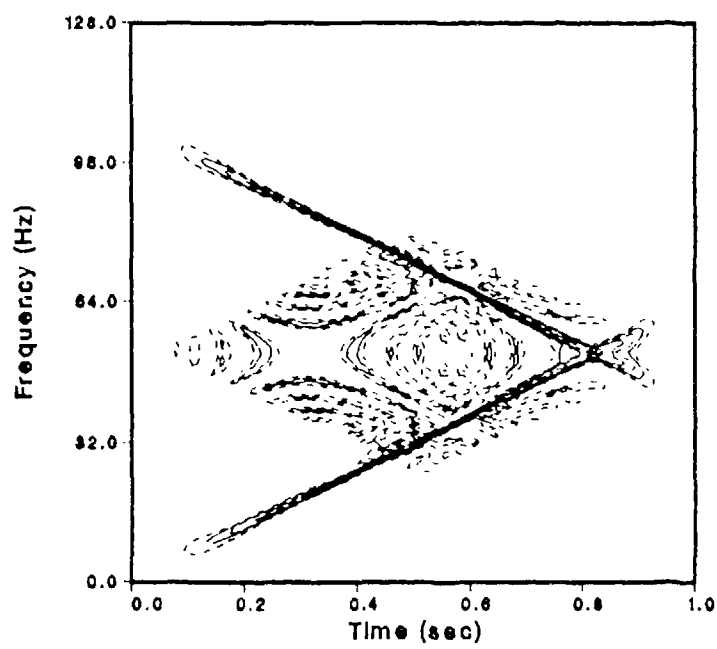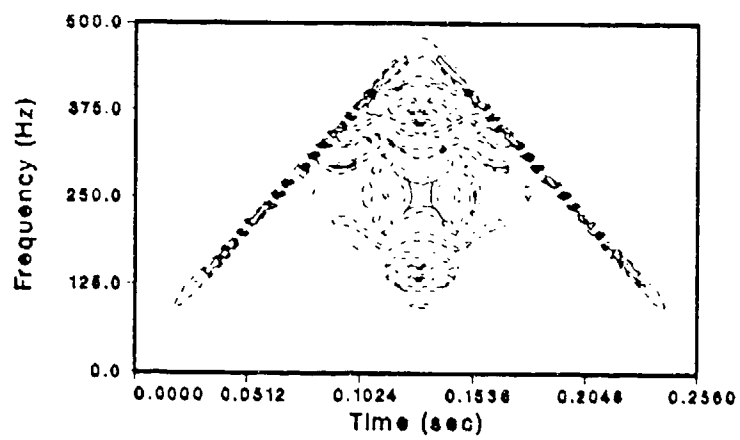(c) Contour plot of WDF

Fig.8 (continued)
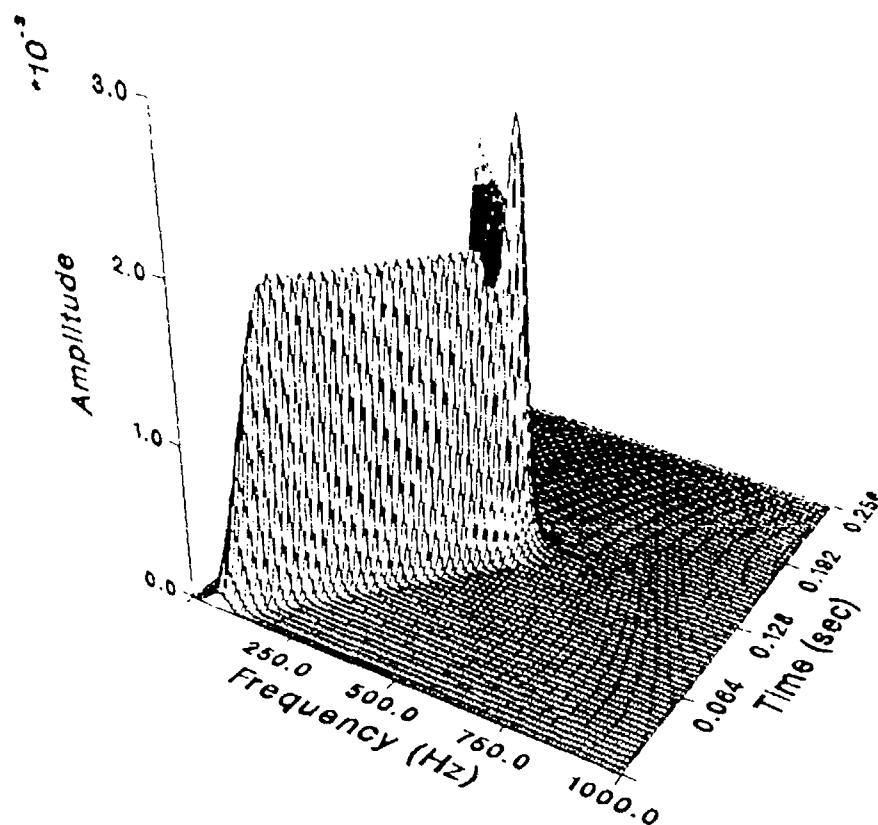
(d) PWVD

Fig. 8 Linear Chirp Signal with One Frequency Component at Each Time
(f$_s$=2000 Hz, N=512 and Smoothing Window Size=16x16)

E. Composite Signal of Sweeping-up and Steady Frequency

The signal which sweeps up along the frequency for first 0.5 second and holds to a constant frequency for next 0.5 second was considered. This signal is typical speed profile of start-up stage of pump. Fig. 9 shows (a) PWVD and (b) its contour plot. The interesting phenomenon was observed in PWVD that the sweep-up portion of signal (first half seconds) has a lower amplitude and wider main lobe compared with the steady frequency region of signal (second half seconds). When the PWVD was integrated along the frequency axis and it was found that the resultant amplitudes in these two regions are same. The following functions were used to generate the desired signal:

$$s(t) = 4\cos(2\pi 32t^2), \qquad 0 \leq t \leq 0.5 \text{ sec.}$$
$$s(t) = 4\cos(2\pi 64t), \qquad 0.5 \leq t \leq 1.0 \text{ sec.} \tag{39}$$

Fig. 10 is the PWVD of the signal which sweeps up along the frequency with a logarithmic rate, that is, the sweep rate is propotional to the square root of time. It was found that the maximum magnitude of the PWVD increases with increasing the frequency. This fact is shown that the PWVD of the stable signal has a larger magnitude than the unstable signals although having the same magnitude in time domain and the PWVD is the good tool for the analysis of the stability of the signal. The following functions were used to generate the desired signal:

$$s(t) = 4 \cos \{2\pi (30 + 60 t^{1/2}) t\}. \tag{40}$$

The instantaneous frequency of the Eq.(40) is the derivative of the argument of the cosine function which is

$$f(t) = 90 t^{1/2} + 30 \tag{41}$$
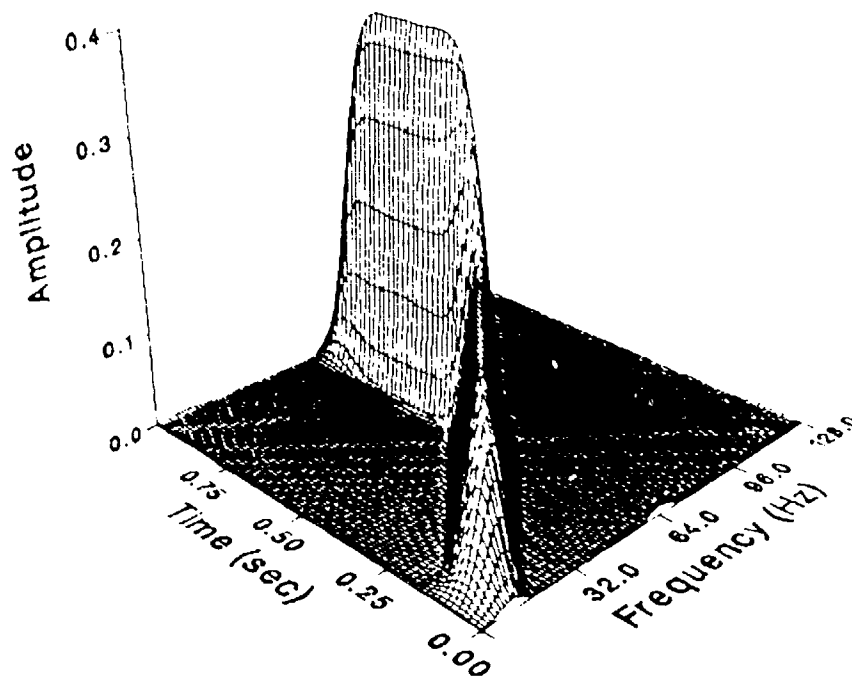
(a) PWVD
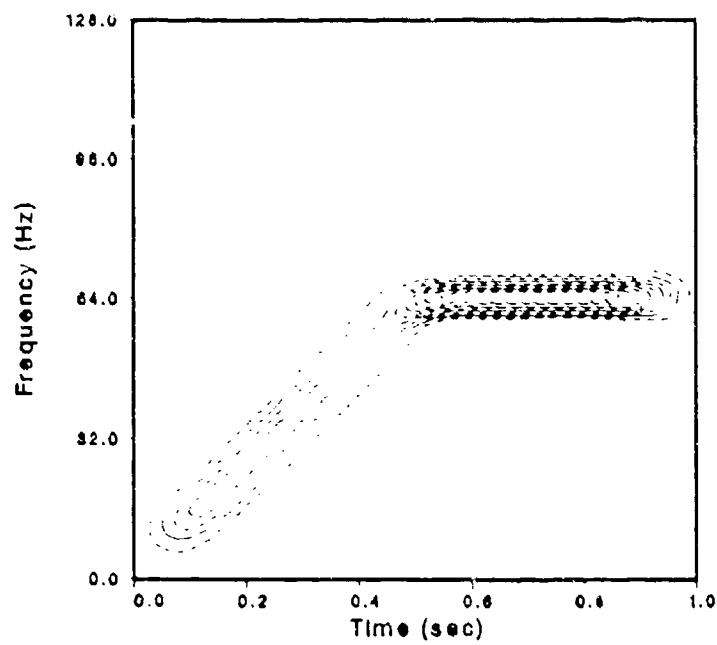
(b) Contour plot of PWVD

Fig. 9 PWVD of a Composite Signal of Sweeping-up and Steady Frequency
(f_s=256 Hz, N=256 and Smoothing Window Size=10x10)

28
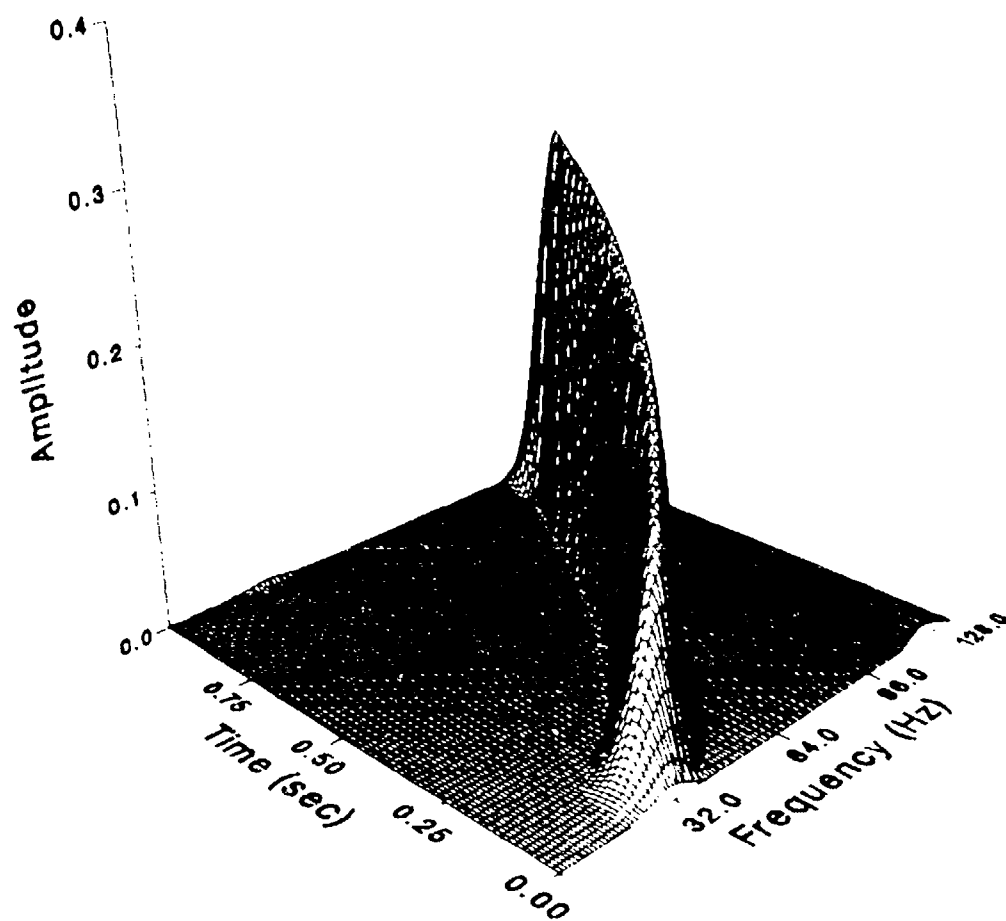
Fig. 10 PWVD of a signal of sweeping-up with a logarithmic rate with time.
(f_S=256 Hz, N=256 and Smoothig Window Size = 10x10)

F. Sweep Rate Effect

The effect of sweep rate on PWVD was investigated. The sweep rate is the frequency change per unit time. The power spectrum density of a typical swept sine is shown in Fig. 11. The incorrect assumption is often made that a swept sine of constant amplitude has a flat spectrum. As can be seen from the plot this is not so. Fig. 12 shows the PWVDs of the linear chirp signal with a various sweep rates:(a) has zero sweep rate and (b) has lower sweep rate than (c). It can be seen that the amplitude of PWVD decreases with increasing sweep rate but energy remains unchanged. This result appeared to be caused by Heisenberg's uncertainty relation between time and frequency. However, based on this study, it is clear that the 'ghost' peak (see Fig. 8) appears due to the instantaneous zero sweep rate at the point where the direction of sweep changes. Also the peak value is affected by the size of smoothing window.
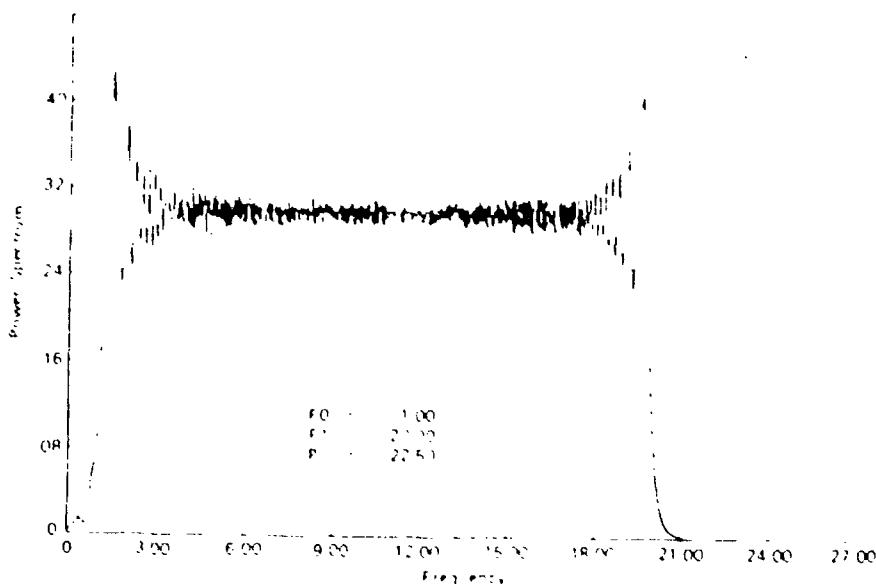
Fig.11 Swept sine wave spectrum.
(Frequency range 1 to 20 Hz, sweep time 22.5 sec)

30
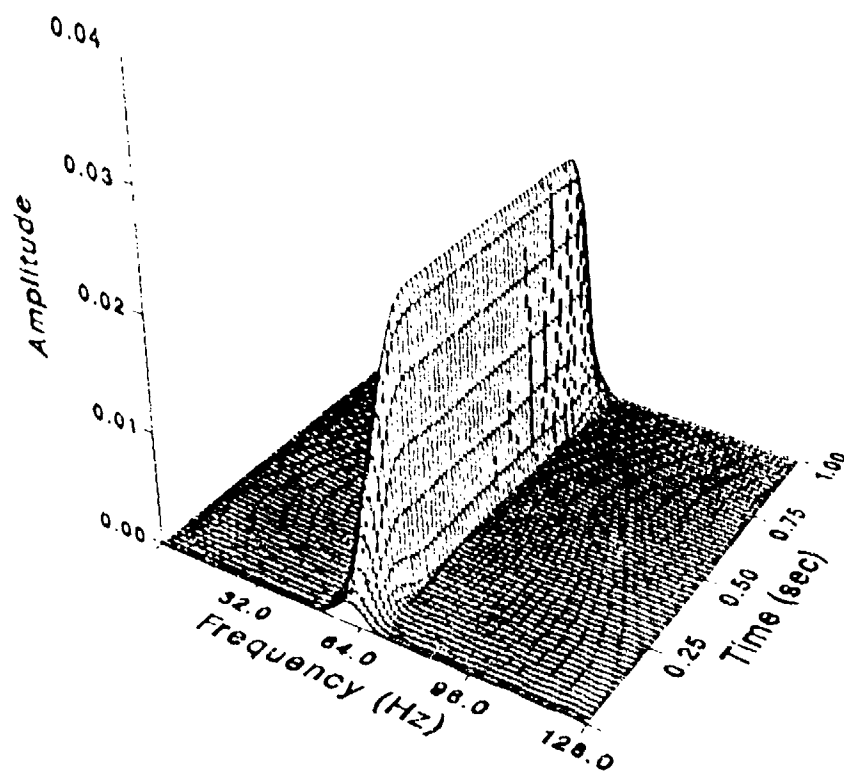
(a) $s(t) = \cos(2\pi\,60\,t)$
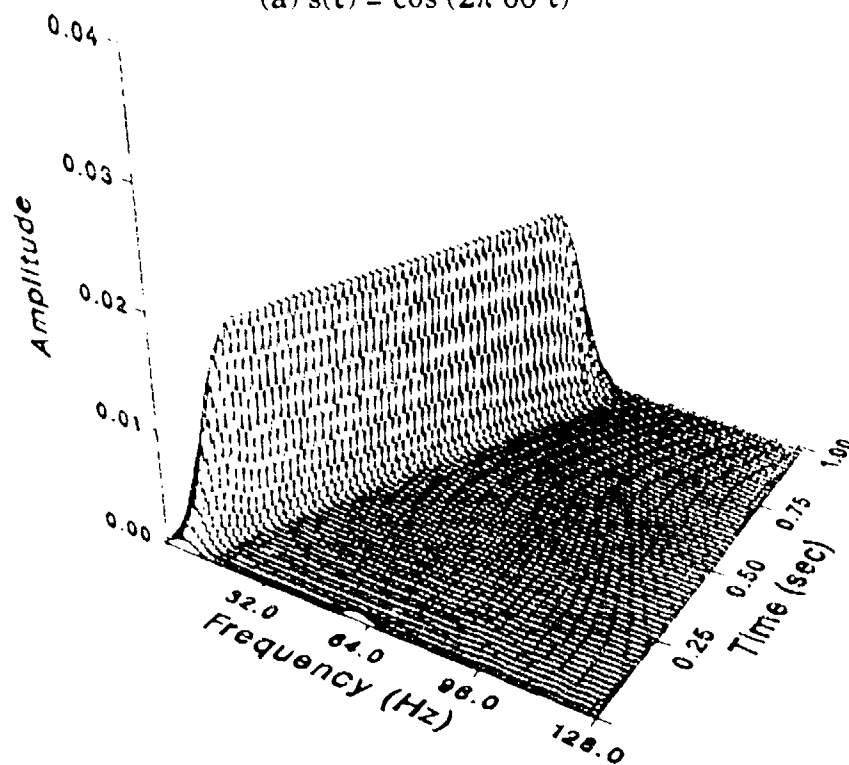
(b) $s(t) = \cos(2\pi\,32\,t^2)$
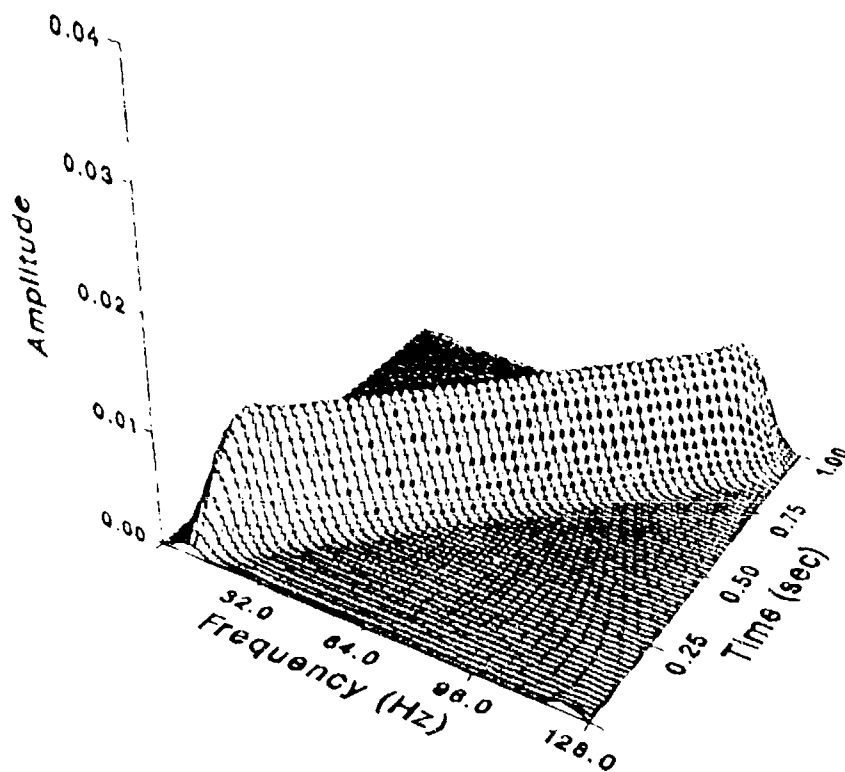
(c) $s(t) = \cos(2\pi\ 64\ t^2)$

Fig. 12. The Effect of Sweep Rates To Pseudo Wigner-Ville Distribution
(f_s=256 Hz, N=256 and Smoothing Window Size=10x10)

G. Harmonic Wave with Some Glitches

The interesting phenomena on the signal with an abnormal components as a fault were investigated. Fig 13 shows the PWVD of the harmonic wave with glitch at a small region of the time record: (a) is the time signal, (b) is the PWVD and (c) is the contour plot of the PWVD. It can be seen that the PWVD of the signal Fig. 13(a) well represents the loacation of each glitch and its frequency components. This characteristic of the PWVD is useful to detect the faults or glitch and to monitor the condition on the vibrational machinery having the periodicity such as a gear train. The general rotating machinery has a periodic signal pattern in time domain.

(a)
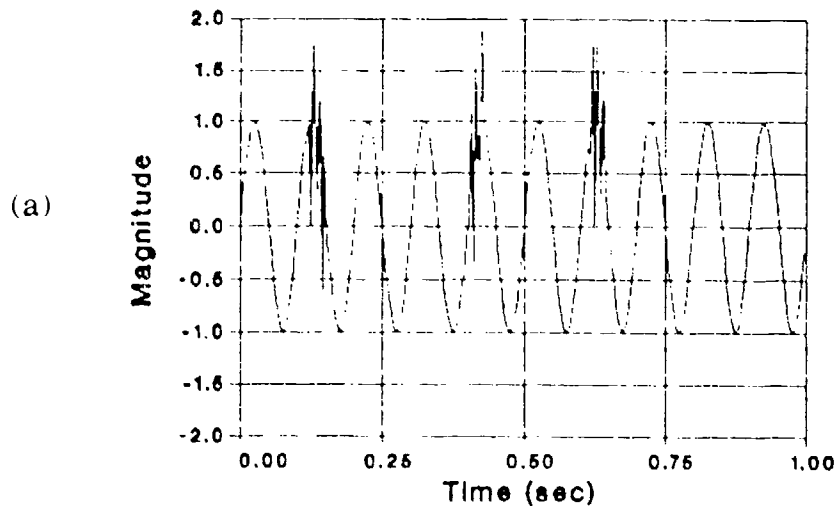
Fig. 13 (continued)
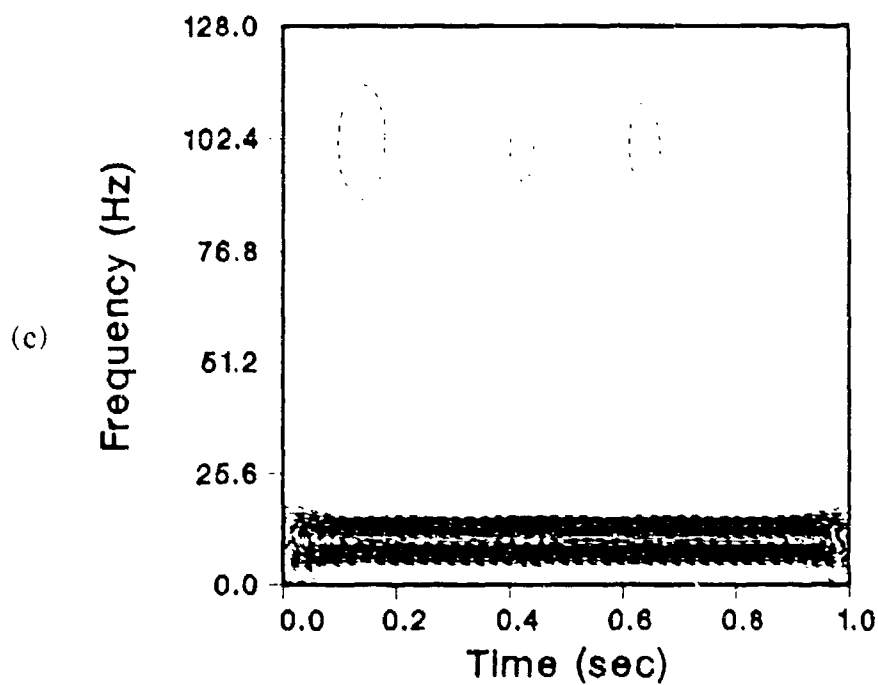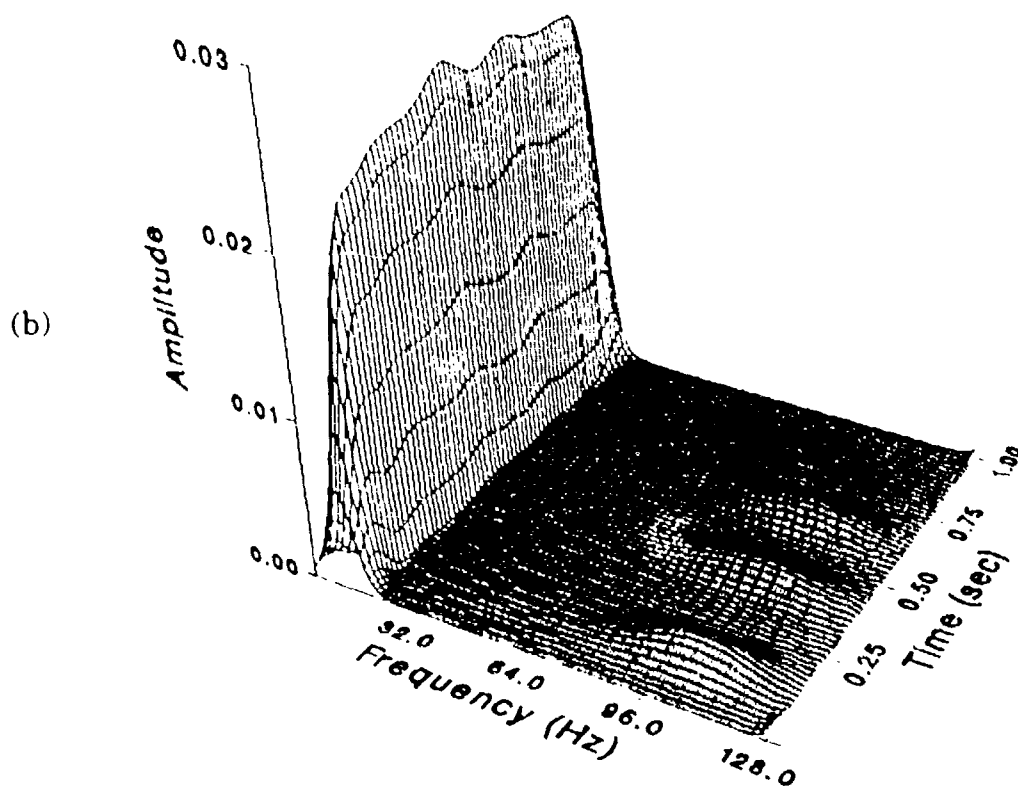
Fig. 13 PWVD od the signal with gltches: (a) time signal, (b) PWVD and (c) contour plot of PWVD.

(f$_s$=256 Hz, N=256, smoothing window size 10x10)

H.   Actual Fan Signal

The acceleration signal of a fan was measured at the steady state speed and the result was shown in Fig. 14. The crest has the complicated shape on time axis as shown in Fig. 4. The first peak is the fundamental frequency of the blade rate and the second peak is 3rd harmonics. The third peak is the fundamental frequency of motor by the pole. The measured vibration signal was contaminated with the noise. If the measured signal involves the faults, the PWVD will represent the different pattern having the abnormal frequency components in comparison with the normal condition with time.
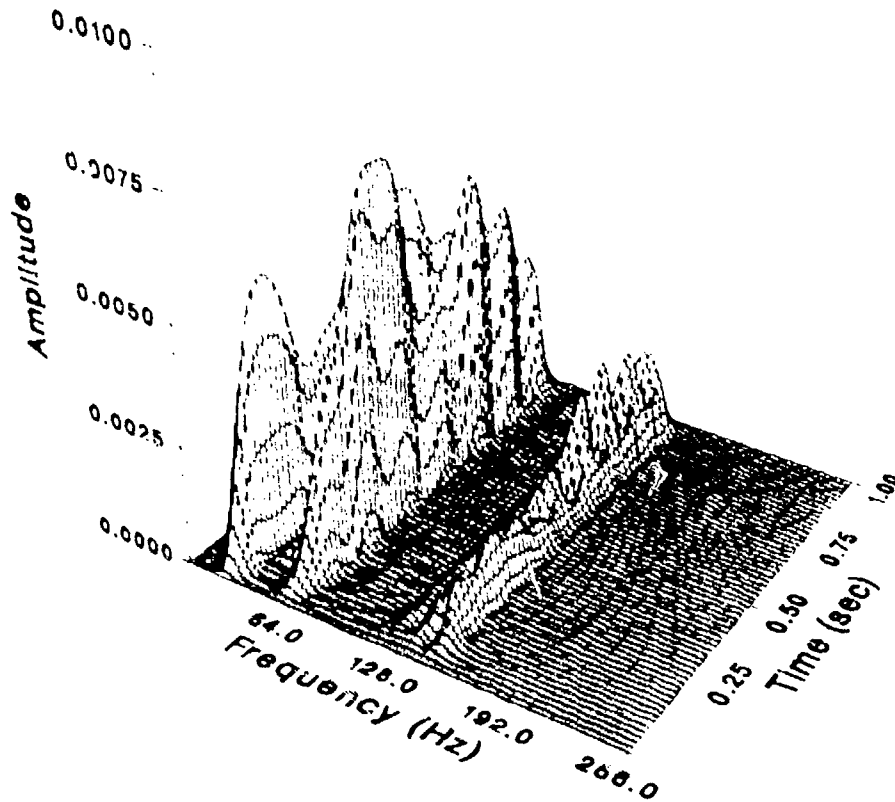
Fig. 14 PWVD of the actual fan signal in the constant speed.
(f_s = 512 Hz, N=512, smoothing window size 13x13)

I.   Actual Pump Start-up RPM Signal

The start-up transient speed of the pump was measured and the results were shown in Fig. 15. The time signal is measured by magnetic sensor and has the same magnitude independently on time. The PWVD is shown in Fig. 15(a) and the contour view is shown in Fig. 15(b). The contour plot shows that the speed of the pump runs up when initially started, reaches the maximum RPM and coasts down gradually. Near the maximum speed during the run up, the sweep rate was rapidly decreased and, as a result, the peak value was rapidly increased. When the sweep rate is close to zero at the normalized time of 0.4, the amplitude attains the maximum value.

(a)

Fig. 15 (continued)

36
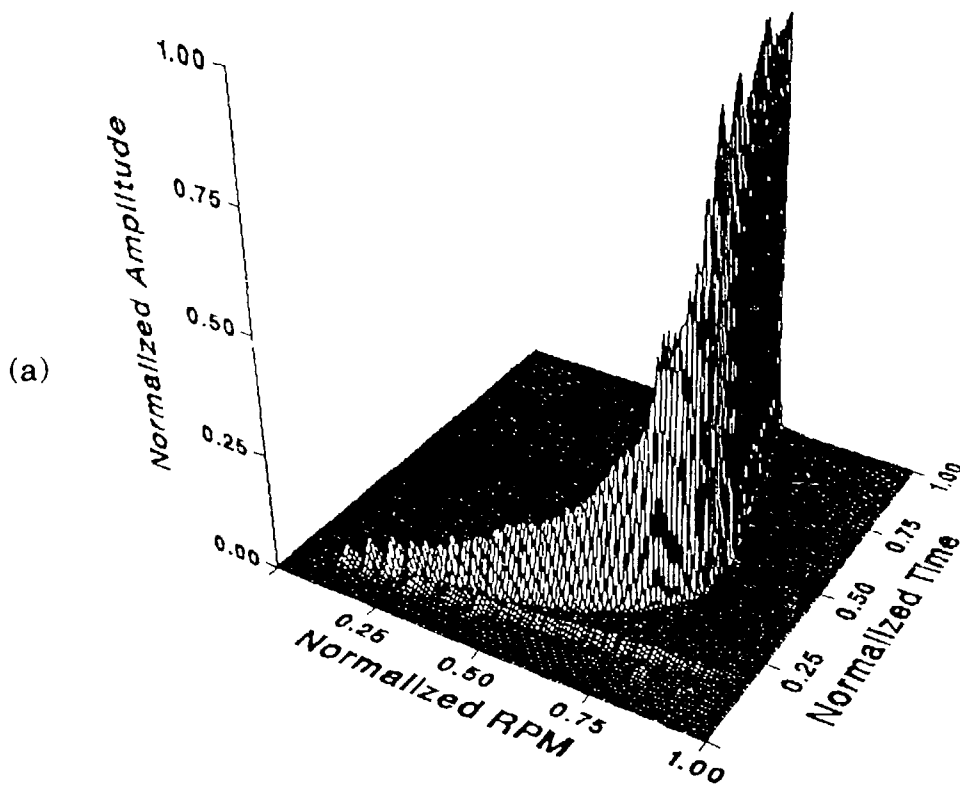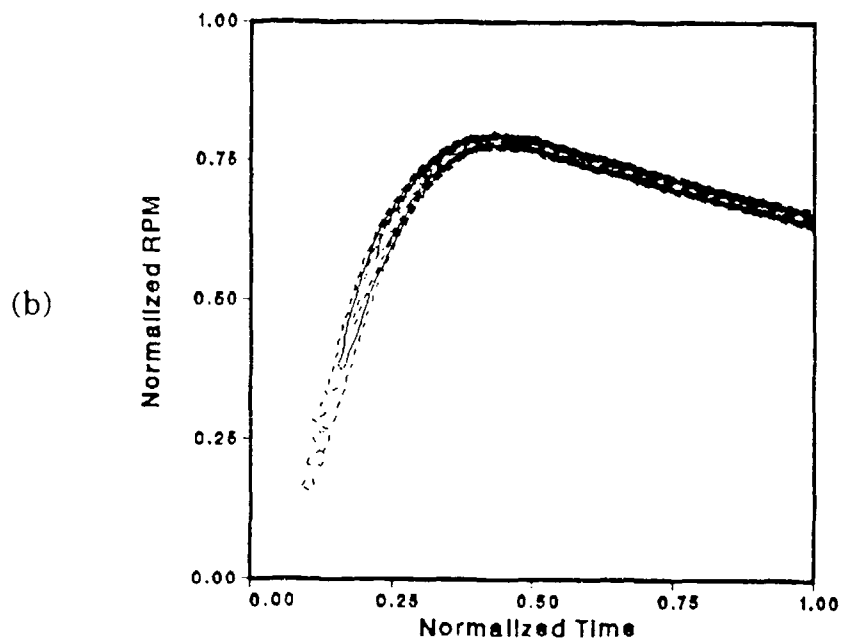
Fig. 15.   Pseudo Wigner-Ville Distribution of Transient Speed of the Pump;
(a) PWVD and (b) contour plot.

37

V. CONCLUSIONS

The pseudo Wigner-Ville distribution has been investigated and applied to analyzing non-stationary signals typical of transient machinery signatures. The results of this research will be a valuable asset for condition monitoring of transient machinery. The following conclusions can be drawn:

(1) The pseudo Wigner-Ville distribution is ideally suited for portraying non-stationary time signals.

(2) The use of modified Hamming window to time signals is effective to reduce the edge effect of discontinuity.

(3) The use of the analytic signal in calculating the Wigner distribution eliminates aliasing problem.

(4) The Gaussian window function for smoothing the Wigner-Ville distribution is very effective and the presence of cross terms is significantly reduced.

(5) Both the amplitude and the main lobe of the pseudo Wigner-Ville distribution is significantly affected by the sweep rate. As the absolute sweep rate increases, the amplitude of the PWVD decreases and the main lobe becomes wider.

(6) The PWVD characterizes the time-frequency domain distribution of the signal well and may be useful tool for the machinery condition monitoring.

ACKNOWLEDGMENTS

REFERENCES

1. Rossano, G.W., Hamilton, J.F., and Shin, Y.S., "The Pseudo Wigner-Ville Distribution as a Method for Machinery Condition Monitoring of Transient Phenomia", Proceedings of the 2nd International Machinery Monitoring & Diagnostics Conference, Los Angeles, CA, pp. 167-173, October 22-25, 1990.

2. Bastiaans, M.J., "The Wigner Distribution Function Applied to Optical Signals and Systems," Optics Communications, vol. 25, no. 1, pp. 26-30, April 1978.

3. Bastiaans, M.J., "Wigner Distribution Function and its Application to First-Order Optics," Journal of the Optical Society of America, vol. 69, no. 12, pp. 1710-1716, Dec 1979.

4. Bartelt, H.O., Brenner, K.H. and Lohmann, A.W., "The Wigner Distribution Function and its Optical Production," Optics Communications, vol. 32, no. 1, pp. 32-38, Jan 1980.

5. Riley, M., Speech Time-Frequency Representations, Kluwer Academic Publishers, 1989.

6. Velez, E. and Absher, R., "Transient Analysis of Speech Signals using the Wigner Time-Frequency Representation," IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 2242-2245, May 1989.

7. Wahl, T., and Bolton, J., "The Use of the Wigner Distribution to Analyze Structural Impulse Responses," International Congress on Recent Developments in Air and Structure-Borne Sound, Mar 1990.

8.   Flandrin, P., Garreau, D., and Puyal, C., "Improving Monitoring of PWR Electrical Power Plants 'In Core' Instrumentation with Time-Frequency Signal Analysis," IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 2246-2249, May 1989.

9.   Forrester, B., "Analysis of Gear Vibration in the Time-Frequency Domain," Proceedings of the 44th Meeting of the Mechanical Failures Prevention Group, pp. 225-234, Apr 1990.

10. Wigner, E., "On the Quantum Correction for Thermodynamic Equilibrium", Physics Review, vol. 40, pp. 749-759, June 1932.

11. Cohen, Leon, "Time-Frequency Distribution - A Review," Proceeding of the IEEE, Vol.77, No.7, pp.941-981, 1989.

12. Mark, W. D., "Spectral Analysis of the convolution and Filtering of Non-Stationary Stochastic Processing," Journal of Sound and Vib., Vol.11, pp.19-63, 1970.

13. Claasen, T. and Mecklenbrauker, W., "The Wigner Distribution - A Tool for Time-Frequency Signal Analysis," Philips Journal of Research, Vol.35, Part I: pp.217-250, Part II: pp.276-300, Part III: pp.372-389, 1980.

14. Yen, N., "Time and Frequency Representation of acoustic signals by means of the Wigner Distribution Function: Implementation and Interpretation," Journal of Acoustical Society of America, Vol.81, No.6, pp.1841-1850, 1987.

15. Ville, J., "Theorie et Applications de la Notion de Signal Analytique," Cables et Transmission, Vol. 2a, No.1, pp.61-74, 1948.

16. Cartwright, N. D., "A Non-Negative Wigner-Type Distribution," Physica, 83A, pp.210-212, 1976.

17.   Otens, R. K., and Enochson, L., Applied Time Series Analysis, John Wiley and Sons, Chap.4, 1978.

APPENDIX A. USER'S GUIDE OF THE PROGRAM

This program calculates either the Wigner-Ville distribution or the smoothed Wigner-Ville(Pseudo Wigner-Ville) distribution(PWVD) of a time series. This program uses the FORTRAN 77 language and it is possible to run at workstation and IBM-PC compatible computer. The user supplies the real data of the time history, the sampling parameters, the digital filter parameters, the smoothing parameters and the output parameters. The program outputs 2-D array containing the Wiger- Ville distribution (rwdf.out) and the smoothed Wigner-Ville distribution, i.e, the pseudo Wigner-Ville distribution (rswdf.out) dependently output parameters.

A-1. PROGRAM INPUT/OUTPUT (Main Program)

This program is interactive and displays an explanations about a desired values. Fig. A-1 is a flowchart of this program and the computational block diagram is shown in Fig. 2.

1) INPUT

This program displays the input variables as follow.

Enter name of signal input file

The user must supply the input filename under 25 characters.

Number of the sampled data point

The user must input the number of sample data(dp). If dp is larger than 2048, the user must change np in a available memory size of the user's computer and recompile the source program. The dp must be a multiple of 2 because of FFT.

4 3
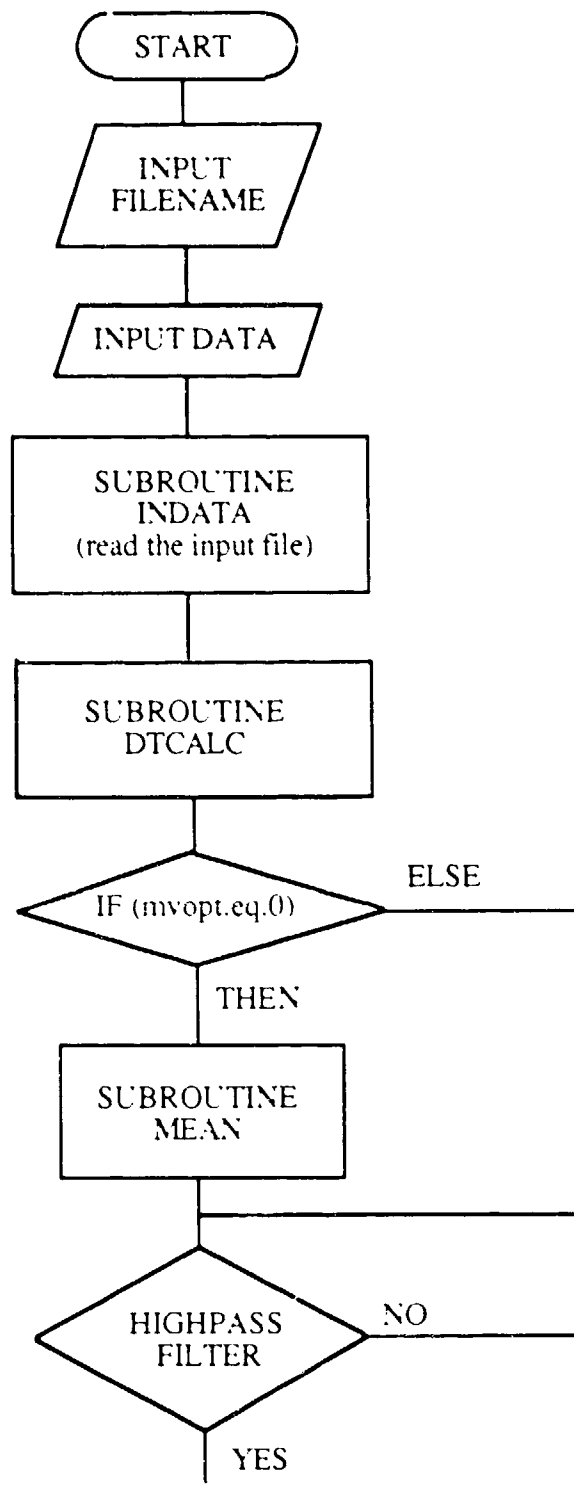
Fig. A-1 Flow chart (continued)
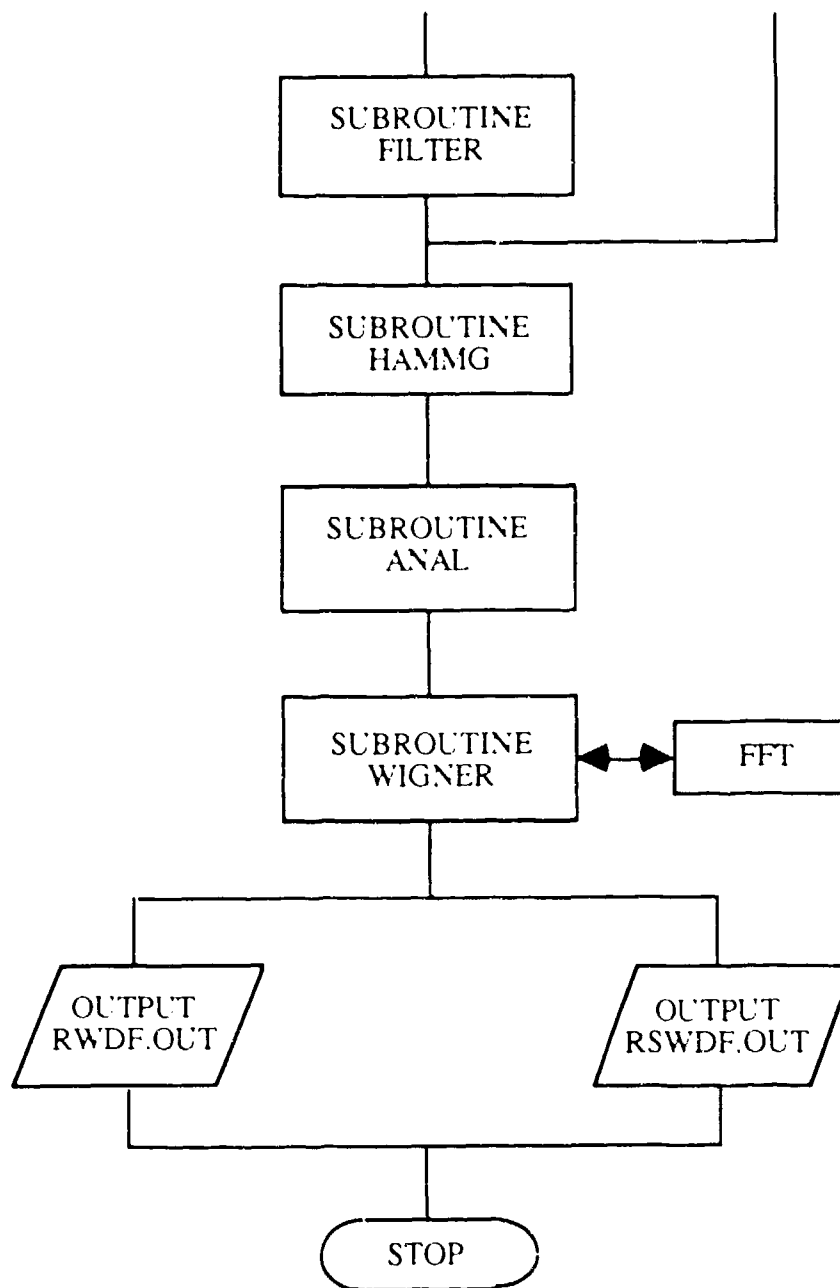
Fig. A-1 Flow chart

The next step is,

Do you wish to remove the mean value ?
Enter 1 for yes or 0 for no.

The user inputs the desired value in according to the characteristics of input time signal. If the time signal have the DC components, the user had better select mvopt=1.

Do you want to apply highpass digital
filter to the original data ? (y/n)

If 'y' is selected, the desired cutoff (half-power point) frequency is inputed after displaying the follow command.

Enter the cutoff frequency of
the digital highpass filter (in Hz)

This program uses the non-recursive(finite impulse response - FIR) highpass filter for the elimination of undesired low frequency components. If the user wants to pass the highpass filter, input the desired cutoff frequency in Hz.

The next procedure is,

Input the desired reduction size
input 1 for   64 by 32
input 2 for   128 by 64
input 3 for   128 by 128
input 4 for   256 by 128

This input statement describes the desired output file size for the output and the plotting because the original Wigner distribution involving the all domains needs the huge memory size.

As the next step, this program chooses the

The desired smoothing window size

input the smoothing parameter for
frequency in Gaussian function(nf)

input the smoothing parameter for
time in Gaussian function(mt)

The smoothing parameters for frequency (nf) and time (mt) depend on the number of the sampled data. The user inputs the integer values satisfying the following condition.

$$
nf * mt \geq \frac{dp}{\pi} \tag{A-1}
$$

We recommend to select the square smoothing window size.

2) OUTPUT

This program generates the two output files. One is the file for the unsmoothed Wigner-Ville distribution and the other is the file for the smoothed Wigner-Ville distribution. The output files consist of one column along with the time and frequency axis dependently the output parameter as following descriptions.

0.00000E+00   : initial time
0.10000E+01   : time record length

47

```
0.00000E+00                          : starting frequency
0.25600E+03                          : end frequency
             128            64       : reduction size
0.53830E-06
0.49456E-06
0.18271E-06
          .                          : the results of PWVD or WVD

          .

          .

          .
```

The graphic output of the results used of CA-DISSPLA version 11.0. The 3-D graphic program describes the unsmoothed or smoothed Wigner-Ville distribution(or pseudo Wigner-Ville distribution). The 2-D graphic program describes the contour plot of the results and the time series.

A-2. EXPLANATIONS OF INPUT VARIABLES

Variable names	Descriptions
inname	Input file containing the real time signal.
dp	The number of the sampled data point.
mt	Smoothing parameter for time in Gaussian Fn.
nf	Smoothing parameter for frequency in Gaussian Fn.
mvopt	Option with respect to removing mean value if mvopt = 1, then zero mean. if mvopt = 0, then no.
bw	Cutoff frequency of highpass filter

A-3. SUBROUTINES

1) indata(dp,tin,ain)

This subroutine reads the input file 'inname'. The input data file involves the time and amplitude in real value. The reading format is a free format.

2) dtcalc(dp,tin,dt)

This subroutine calculates the mean time interval dt from the input time signal. Because the obtained data from A/D converter or signal processor doesn't have the exactly same interval. Delta time(dt) is given as follow,

$$dt = (total\ record\ length) / (number\ of\ data\ point-1) \qquad (A-2)$$

3) mean(dp,ain)

This subroutine calculates the mean value and removes the mean value of the signal under the condition of the input variable mvopt. In the case the signal has the DC component, it is recommended to remove the mean value. A highpass digital filter must use for the elimination of DC component.

4) filter(m,b,t,bk)

This subroutine generates a nonrecursive (finite impulse response - FIR) filter weights and uses the method devised by Potter, Bickford and Glaze. Nonrecursive highpass filter was used for the elimination of

undesired low frequency components. The variables m, b, t, and bk are the number of the weights, the cutoff frequency of the filter, the time interval and the storage array of the filter weights, respectively.

The basic design was to use a symmetric filter of the form,

$$y(i) = \sum_{k=-m}^{m} b_k \, s(i-k) \qquad \text{(A-3)}$$

with

$$b_{-k} = b_k \qquad \text{(A-4)}$$

and

$$b_k = \frac{\sin 2\pi bkt}{\pi k} \qquad \text{(A-5)}$$

where b_k is the filter weights, $y(i)$ is the filtered signal, $s(i)$ is the original signal, b is the cutoff frequency, t is a sampling interval and mo is the span of the filter; 2m+1 weights are employed because of symmetry, only m+1 need be generated. The b_k weights are computed over the range -m to m. The weights are multiplied by a window function. Potter discusses a number of windows in the referenced work. His P310 window was found to be appropriate for filter implementation. It takes the form,

$$w_k = \frac{c_k}{w'} \left[d_0 + 2 \sum_{p=-3}^{3} d_p \cos \frac{\pi pk}{m} \right] \qquad \text{(A-6)}$$

where

$$c_k = \frac{1}{2} \qquad k = \pm m$$
$$= 1 \qquad \text{otherwise} \qquad \text{(A-7)}$$

$$d_0 = 1$$
$$d_{-1} = d_1 = 0.684988$$
$$d_{-2} = d_2 = 0.202701$$
$$d_{-3} = d_3 = 0.0177127$$

51

and

$$w = d_0 + 2 \sum_{p=-3}^{3} d = 2.8108034 \qquad \text{(A-8)}$$

For a highpass filter with pass band from the cutoff frequency(B) to the maximum frequency, generate a low pass filter on the range 0 - B, and then subtract the central weight from unity and change the signs of the remainder of the weights.

The filtered signal in the main program is generated by using the calcultaed filter weights as follows.

```
do     k=-m,m
  j=k
  if (k.lt.0) then
     j=k*(-1)
  endif
  .
  .
  .
  bb=-bk(j)
  if (k.eq.0) then
     bb=1.-bk(j)
  endif
  y(i)=y(i)+bb*s(i-k)
end do
```

(A-9)

5)   hammg(dp,dt,pi,ain)

This subroutine is a data tapering by using the modified Hamming window. It is often desirable to taper a random time series at each end to enhance certain characteristics of the spectral estimates. Tapering is multiplying the time series by a data window analogous to multiplying the correlation function by a lag window. Thus tapering the time series is

equivalent to applying a convolution operation to the raw Fourier transform. The purpose of tapering when viewed from its frequency domain effect is to suppress large side lobes in effective filter obtained with the raw transform. When looked at from the time domain, the object of tapering is to round off potential discontinuities at each end of the finite segment of the time history being analyzed.

The used data window in this program is a modified Hamming window as given in Eq.(A-10).

$$W(t) = \begin{cases} 0.54 - 0.46 * \cos(10\pi t/T) & 0 \leq t \leq T/10. \\ 1.0 & T/10 \leq t \leq 9T/10 \\ 0.54 - 0.46 * \cos(10\pi(T-t)/T) & 9T/10 \leq t \leq T \end{cases} \qquad (A\text{-}10)$$

6)   anal(dp,pi,ain,s)

This subroutine converts a real signal to an analytic one by using the Hilbert transform given as follow,

$$h(n) = \frac{2}{\pi} \frac{\sin^2(\pi n/2)}{n}, \qquad n \neq 0,$$
$$= 0, \qquad n = 0.$$

(A-11)

The Wigner distribution has a periodicity of $N/2$, where N is the number of data. Hence, even when the sampling of signal satisfies the Nyquist criteria, there are still aliasing components in the Wigner distribution function. If we sample the signal with Nyquist rate, its power density spectrum does not overlap with its own components. For Wigner distribution function, there are additional spectrum components causing the aliasing interference. However, a simple way to alleviate this problem for practical purpose is to increase the seperation of the spectrum groups by either doubling sampling rate or by interpolating with additional data

points according to sampling theorem before the transform process. Another approach to avoid the aliasing is to use only the positive part the signal's frequency components, the analytical signal, before computing the Wigner distribution function. The analytic signal approach is convenient since it can be easily obtained with the Hilbert transform, which can also use the efficient FFT algorithm.

This program uses the later case. The real time signal converts to complex variable by the Hilbert transform, that is, the imaginary part is generated by the convolution of the impulse response h(n) shown as follow,

$$H\{s_r(n)\} = \sum_{m=-\infty}^{\infty} h(n-m)\, s_r(m) \tag{A-13}$$

where s_r is the original input signal.

The analytic version of the real signal is made up of the real signal plus an imaginary part composed of the Hilbert transform of the real signal.

$$s(t) = s_r(t) + j\, H\{s_r(t)\} \tag{A-14}$$

7)   wigner(dp,dt,pi,s,c,mm,nn,wdf,mt,nf)

This subroutine calculates the Wigner distribution function. The Wigner distribution function is obtained by using fast Fourier transform (FFT) of the local correlation of signal. At first, the local correlation of the signal is obtained and 2N point FFT is carried out. This program generates the results wdf(i,j) of the Wigner distribution function and pseudo Wigner-Ville distribution function on the given time series. The Wigner distribution has 1/2 of the frequency resolution of the power spectrum because 2N points FFT and the argument of the time signal and its conjugate contains a factor of 1/2. That is, df = 1/(4*dp*dt).

To avoid the negative values and the elimination of the interference by the cross correlation term, this program uses the smoothing technique with Gaussian window function as following Eq.(A-15).

$$G(t,w) = \frac{1}{2\pi\sigma_t\sigma_\omega} e^{-\frac{t^2}{2\sigma_t^2} \cdot \frac{\omega^2}{2\sigma_\omega^2}}$$ (A-15)

The smoothing is obtained by the convolution integration the Wigner distribution and a Gaussian window function as given Eq(26). To perform the convolution on the sampled wdf(i,j), the Gaussian function was first truncated so that it spanned the range $\pm 2\sigma_t$ and $\pm 2\sigma_\omega$. To avoid the negative value, Gaussian window size must have a larger or equal integer value mt and nf than $(dp/\pi)^{1/2}$. For example, in the case of dp=1024, at least Gaussian window size must be 18x18 or larger. We recommend to select the square windows size.

8)   fft(dp,pi,c)

This subroutine is a program about the Fast Fourier Transform by Jim Cooley's method.

APPENDIX B.   PROGRAM LIST

B-1.   PROGRAM LIST ABOUT THE CALCULATION OF WDF

PROGRAM PWVD

```
**************************************************************************
*                                                                        *
*         PSEUDO WIGNER-VILLE DISTRIBUTION FUNCTION          *
*              PC VERSION 1.0            JAN.  1993          *
*                    (by using highpass digital filter)          *
*                                                                        *
**************************************************************************
*
*
*       This is the program about the pseudo Wigner-Ville
*       distribution(PWVD). The PWVD is a three dimensional(time,
*       frequency, amplitude) representation of an input
*       signal and is ideally suited for portraying transient
*       phenomena.
*
*             VARIABLES
*
*       dp      = the number of the sampled data point
*       mm      = output parameter for time
*       nn      = output parameter for frequency
*       mt      = smoothing parameter for time in Gaussian Fnc.
*       nf      = smoothing parameter for frequency
*                  in Gaussian Fnc.
*       inname  = input filename (sampled data)
*       mvopt   = option with respect to removing mean value
*                  if mvopt=1, then zero mean
*                  if mvopt=0, then no
```

```
*      df      = frequency resolution
*      dt      = sampling interval
*
*          ARRAYS
*
*      tin(i)       = sampled time data
*      ain(i)       = sampled magnitude data
*      fain(i)      = filtered data
*      wdf(i,j)     = reduced array of PWVD
*      bk(i)        = filter weights
*      s(i)         = analytic signal
*      c(i)         = local auto-correlation or the results of FFT
*

*****************************************************************
*                                                               *
*          VARIABLE DECLARATION                                 *
*                                                               *
*****************************************************************
***************--- setting number of data points ---*****************
*      Notes: For sample sizes greater than 2048, change
*             the parameter np.
*

       integer dp,dp2,mvopt,redopt,mm,nn,nf,mt
       parameter (np=2048)
       real pi,tin(np),ain(np),wdf(256,128),dt,fain(np),
1    bk(1100)
       complex s(np*2),c(np*2)
       character*25 inname
       character as
       pi=atan(1.)*4.
**************************************************************
*      --- Print description of program ---
       print*,'*******************************************'
       print*,'        Pseudo Wigner-Ville Distribution'
```

57

```fortran
      print*,'*********************************************'
      print*

*     --- Set input parameters ---

      print*,' Enter name of signal input file'
      read(5,901) inname
      print*
      print*,'Number of the sampled data point'
      read(5,902) dp

      dp2=dp*2

      do 100 i=1,dp2
100   s(i)=cmplx(0.,0.)
*

      print*,' Do you wish to remove the mean value?'
      print*,' Enter 1 for yes or 0 for no'
      read(5,902) mvopt
      print*
*

      print*,'Do you want to apply a highpass digital'
      print*,'filter to the original data ? (Y/N)'
      read(*,903) as

      if (as.eq.'Y'.or.as.eq.'y') then
         print*,'Enter the cutoff frequency of'
         print*,'the digital highpass filter (in Hz) '
         read(*,*) bw

      end
if
*
      fmin=0.
```

```
      print*,'Input the desired reduction size'
      print*,' input 1 for 64 by 32 '
      print*,' input 2 for 128 by 64 '
      print*,' input 3 for 128 by 128 '
      print*,' input 4 for 256 by 128 '
      read(5,902) redopt
      print*
      if (redopt.eq.1) then
         mm=dp/32
         nn=dp2/64
      elseif (redopt.eq.2) then
         mm=dp/64
         nn=dp2/128
      elseif (redopt.eq.3) then
         mm=dp/128
         nn=dp2/128
      else
         mm=dp/128
         nn=dp2/256
      endif
*

      print*,'Input the desired smoothing window size '
      print*
      print*,' input the smoothing parameter for frequency'
      print*,' in Gaussian function'
      read(5,902) nf
      print*,' input the smoothing parameter for time'
      print*,' in Gaussian function'
      read(5,902) mt


**************************************************************
*          The calculation part of the program             *
**************************************************************
```

```
*               Read the sampled data file

        open(4, file=inname,status='old')
        call indata(dp,tin,ain)
        print*,'finished subr. indata'
        close(4)

*    calculate the mean time interval

        call dtcalc(dp,tin,dt)
        print*,'finished subr. dtcalc'

        if (mvopt.eq.1) then
           call mean(dp,ain)
           print*, 'finished subr. mean'
        endif
*
*    Signal modifications
*
*    Application of highpass digital filter

        if (as.eq.'Y'.or.as.eq.'y') then
        mo=dp/2

*    calculate the filter weighting

        call filter(mo,bw,dt,bk)

*    pass the highpass filter

        do 160 i=1,dp
160     fain(i)=0.

        do 200 i=1,dp
```

```fortran
        do 170 k=-mo,mo
          j=k
          if (k.lt.0) then
            j=k*(-1)
          endif
          j=j+1
          ll=i-k
          if (ll.lt.1) then
            ll=ll+dp
          elseif (ll.gt.dp) then
            ll=ll-dp
          endif
          bb=-bk(j)
          if (k.eq.0) then
            bb=1-bk(j)
          endif
          fain(i)=fain(i)+bb*ain(ll)
170       continue
200     continue

        do 210 i=1,dp
210     ain(i)=fain(i)

        print*, 'finished digital filtering'
        endif

*   Window application(modified hamming windows)

        call hammg(dp,dt,pi,ain)
        print*, 'finished subr. hammg'


*   Conversion of real signal to an analytic one

        call anal(dp,pi,ain,s)
```

```fortran
      print*, 'finished subr. anal'

*

      open(9,file='rwdf.out',status='new')
      open(10,file='rswdf.out',status='new')
*
*     Writing the WDF to output file          *
          ttime=dp*dt
          df=1./(4.*dp*dt)
          fmax=2.*dp*df
          nx=dp2/nn
          ny=dp/mm
          write(9,904) tin(1),ttime,fmin,fmax
          write(9,*) nx,ny


*     Calculation of the Wigner distribution:

          print*,'Calculating the PWVD'
          call  wigner(dp,dt,pi,s,c,mm,nn,wdf,mt,nf)
          print*,'finished wigner ville distribution function'
          close(9)
*
*     Writing of reduced & smoothed WVD to output file
*

          write(10,904) tin(1),ttime,fmin,fmax
          write(10,*) nx,ny
          do 500 i=1,dp/mm
            do 500 j=1,dp2/nn
                write(10,905) wdf(j,i)
500           continue

*     Format statements
```

```
901      format(a25)
902      format(i6)
903      format(a1)
904      format(e12.5,/,e12.5,/,e12.5,/,e12.5)
905      format(2x,e12.5)


         close(10)

         print*,'terminate the excution of Wigner-Ville distribution'

         stop
         end


*******************************************************************
*                                                                 *
*                    SUBROUTINES                                  *
*                                                                 *
*******************************************************************


*******************************************************************
         subroutine indata(dp,tin,ain)
*********************************************************************** *
         integer dp
         real tin(*),ain(*)

********simple loop to read in time & amplitude*************

         do 100 j=1,dp
            read(4,*) tin(j) , ain(j)
100      continue
         return
         end


*******************************************************************
```

63

```fortran
        subroutine dtcalc(dp,tin,dt)
***********************************************************
*
*       This subroutine calculates the delta t of the signal
*
*

        integer dp
        real tin(*),dt
*
        dtsum = 0.0

        do 100 i = 1 , dp-1

            delt = tin(i+1) - tin(i)
            dtsum = dtsum + delt


100     continue

        dt = dtsum / float(dp-1)


        return
        end


***********************************************************
        subroutine mean(dp,ain)
***********************************************************
*
*   This subroutine calculates and removes the mean value
*   of the signal.
*

        integer dp
        real ain(*),meanv
*
```

```
          asum = 0.0
          do 100 i = 1 , dp
              asum = asum + ain(i)
100       continue
          meanv = asum / dp
          do 200 i = 1 , dp
              ain(i) = ain(i) - meanv
200       continue

          return
          end


*******************************************************************
          subroutine filter(mo,b,t,bk)
*******************************************************************
*
*         Routine generates FIR filter weights.
*         Method devised by Potter, Bickford and Glaze.
*         There are a total of 2M+1 weights...filter generates M+1.
*
*         --- variables ---
*         t = the sampling interval in second.
*         bw = cutoff(half-power point) of the filter in Hz;
*                  must be on the range from o to 1/2t.
*
*         Results are stored in bk
*
*    ---- Note; in the case of highpass filter, the value of
*                  weight b0 must use 1-b0 instead of b0.
*
          dimension bk(*),d(3)
          data d0/0.35577019/,d(1)/0.2436983/,d(2)/0.07211497/,
         * d(3)/0.00630165/
```

65

```fortran
          pi=atan(1.)*4.
          m=mo
*         first generate plain boxcar weights
          fact=2.*b*t
          bk(1)=fact
          fact=fact*pi
          do 5 i=1,m
          fi=i
5         bk(i+1)=sin(fact*fi)/(pi*fi)
*         trapezoidal weighting at end
          bk(m+1)=bk(m+1)/2.
*         Now apply the Potter p310 window
          sumg=bk(1)
          do 15 i=1,m
          sum=d0
          fact=pi*float(i)/float(m)
              do 10 k=1,3
10                sum=sum+2.*d(k)*cos(fact*float(k))
          bk(i+1)=bk(i+1)*sum
15        sumg=sumg+2.*bk(i+1)
          m1=m+1
          do 20 i=1,m1
20        bk(i)=bk(i)/sumg
          return
          end


*******************************************************
          subroutine  hammg(dp,dt,pi,ain)
*******************************************************
*
*         This subroutine applies a modified hamming window
*         to the signal ain(t)
*
          integer dp
```

```
      real  pi,ain(*),dt,mtime,del1,del2,const
*
      mtime=(dp-1)*dt
      del1=0.1*mtime
      del2=0.9*mtime
      const=pi/del1

       do 100 j = 1 , dp

       t = (j-1) * dt

       if (t.le.del1) then

            ain(j) = ain(j) * (.54-0.46*cos(const*t))

       elseif ((t.ge.del2).and.(t.le.mtime)) then

            ain(j)=ain(j)*(.54-0.46*cos(const*(mtime-t)))

       endif

100    continue

       return
       end


**********************************************************
       subroutine anal(dp,pi,ain,s)
**********************************************************
*
*      This subroutine converts a real signal to an
*      analytic one by using Hilbert transform.
*
*      s(i) = analytic signal.
```

```
*
            integer dp
            real  pi,ain(*),sum,sumb,val,sval
            complex  s(*)
*

            do 100 i=1,dp
                sum=0.0

             do 200 j=1,dp
                sumb=0.0

                if (i-j.eq.0) go to 200

                    n=i-j
                    val=pi*n/2.
                    sval=sin(val)
                    sumb=ain(j)*sval*sval/val


200         sum=sum+sumb

                s(i)=cmplx(ain(i),sum)


100     continue

        return
        end


*************************************************************
        subroutine  wigner(dp,dt,pi,s,c,mm,nn,wdf,mt,nf)
*************************************************************
*

*       This subroutine calculates the WDF of the signal

*

*
```

```fortran
        integer dp,dp2
        real pi,dt,coef,wdf(256,128),hg(-60:60,-60:60)
        complex s(*),dum,c(*)
        dp2 = dp*2
        coef = 2.0 * dt
        df=1./(4.*dp*dt)
        nf2=nf*2
        mt2=mt*2
        f1=float(mt)
        f2=float(nf)
*
*       Gaussian function
*
        val=1./((2.*pi)**2*f1*f2*df*dt)
        do 20 j=-mt2,mt2
          q1=float(j)
          do 10 i=-nf2,nf2
            q2=float(i)
            cf=-((q1*q1)/(2.*f1*f1))-((q2*q2)/(2.*f2*f2))
            hg(i,j)=val*exp(cf)
10        continue
20      continue
*
*       initialize wdf(i,j)
*
        do 100 i=1,256
        do 100 j=1,128
100       wdf(i,j)=0.

        do 5000  j = 1 , dp

*    local auto-correlation

          do 1000 i = 1 , dp+1
```

```fortran
            if (j.ge.i) then
              dum = s(j-i+1)
            else
              dum = cmplx(0.,0.)
            endif
            c(i) = coef * (s(j+i-1)*conjg(dum))
              if (i.ne.1.and.i.ne.dp+1) then
                c(dp2-i+2) = conjg(c(i))
              endif
1000        continue

        call fft(dp,pi,c)

        ik=mod((j-1),mm)
        if (ik.eq.0) then

        do 1500 i=1,dp2,nn
              write(9,1400) real(c(i))
1400             format(2x,e12.5)
1500        continue
        endif
          m1=0
          do 4000
          m=1,dp,mm
            m1=m1+1
            n1=0
            if (abs(j-m).le.mt2) then
            do 3000 n=1,dp2,nn
              n1=n1+1
              do 2500 kk=n-nf2,n+nf2
                k1=kk
                if(kk.lt.1) k1=kk+dp2
                if(kk.gt.dp2) k1=kk-dp2
                wdf(n1,m1)=wdf(n1,m1)+real(c(k1))*hg(kk-n,j-m)*df*dt
```

70

```fortran
2500          continue
3000           continue
          endif
4000       continue
5000      continue
      return
      end


*************r*******************************************
      subroutine fft(dp,pi,c)
********************************************************
*
*      This subroutine is the Fast Fourier Transform
*
      integer dp,dp2,val,coef,coef1
      real pi
      complex  dum,c(*),dum3,dum2
*
      dp2 = dp*2
      const=float(dp2)
      val=alog(const)/alog(2.)+.1
      j=1
      do 40 i=1,dp2-1
         if (i.ge.j) go to 10
         dum3=c(j)
         c(j)=c(i)
         c(i)=dum3
10    k=dp
20    if (k.ge.j) go to 30
         j=j-k
         k=k/2
         go to 20
30    j=j+k
40  continue
```

```
      do 70 n=1,val
         coef=2**n
         coef1=coef/2
         dum2=cmplx(1.,0.)
         theta=pi/float(coef1)
         dum=cmplx(cos(theta),-sin(theta))
         do 60 j=1,coef1
            do 50 i=j,dp2,coef
               ii=i+coef1
               dum3=c(ii)*du
               m2 c(ii)=c(i)-
               dum3
               c(i)=c(i)+dum3
50          continue
            dum2=dum2*dum
60       continue
70    continue
      return
      end
```

B-2. PROGRAM LIST OF 3-D PLOT

```
        PROGRAM 3DPLOT
*
*       This program uses the graphic package CA-DISSPLA to plot
*       the results of WDF
*
*       ttime      = time record length
*       tini       = initial time
*       fmin       = start frequency
*       fmax       = stop frequency
*       nx         = the number of the frequency data at time n x dt
*       ny         = the number of the time data at frequency m x df
*
* ----------- Declaring variables -----------
*
        real rwdf(32768)
        integer nx,ny
        character*25 fname

        write(*,*) 'input file name ?'
        read(*,20) fname
20      format(a25)
*
        open(15,file=fname,status='old')

        read(15,*) tini
        read(15,*) ttime
        read(15,*) fmin
        read(15,*) fmax
        read(15,*) nx,ny
        write(*,*) tini,ttime,fmin,fmax,nx,ny
```

```fortran
      n=nx*ny
      do 100 i=1,n
          read(15,*) rwdf(i)
100   continue

      close(15)

      smax=rwdf(1)
      do 200 i=1,n
      if (smax.lt.rwdf(i)) then
          smax=rwdf(i)
      endif
200   continue

      write(*,*) 'smax =', smax
      write(*,*) 'input maximum z-axis value'
      read(*,*) fac

      call pdev('ln03', ieer)

*     plotting

      call hwshd
      call swissm
      call shdchr(90.,1,0.002,1)
      call height(0.2)
      call physor(0.7,0.625)
      call area2d(7.5,9.75)
      call messag('WIGNER-VILLE DISTRIBUTION $',100,1.1,8.2)
      call blsur
      call volm3d(8.,8.,9.)
      call x3name('Frequency (Hz) $',100)
      call y3name('Time (sec) $',100)
```

74

```
call z3name('Amplitude $',100)
call vuangl(-60.,30.,30.)
call zaxang(90.)
tstep=ttime/4.
fstep=(fmax-fmin)/4.
tmax=tini+ttime
call graf3d(fimn,fstep,fmax,tini,tstep,tmax,0.,'SCALE',fac)

call surmat(rwdf,1,nx,1,ny,1)

call end3gr(0)
call endpl(o)
call donepl

stop
end
```

B-3. PROGRAM LIST OF CONTOUR PLOT

```
        PROGRAM CONTOUR
*
*       This program uses the graphic package CA-DISSPLA to plot
*       the results of WDF
*
*       ttime       = time record length
*       tini        = initial time
*       fmin        = start frequency
*       fmax        = stop frequency
*       nx          = the number of the frequency data at time n x dt
*       ny          = the number of the time data at frequency m x df
*
*       ----------- Declaring variables ----------
*
        real r(64,128)
        common work(9000)
        integer nx,ny
        character*25 fname

        write(*,*) 'input file name ?'
        read(*,20) fname
20      format(a25)
*
        open(15,file=fname,status='old')

        read(15,*) tini
        read(15,*) ttime
        read(15,*) fmin
        read(15,*) fmax
        read(15,*) nx,ny
        write(*,*) tini,ttime,fmin,fmax,nx,ny
```

```fortran
      smax=0.
      do 100 i=1,ny
          do 100 j- 1,nx
              read(15,*) r(i,j)
              if (r(i,j).lt.0.) r(i,j)=0.
              if (smax.lt.r(i,j)) then
                  smax=r(i,j)
              endif
100       continue

      close(15)

*       Normalizing

      do 200 i=1,ny
      do 200 j=1,nx
          r(i,j)=r(i,j)/smax
200   continue

      write(*,*) 'smax =', smax
      write(*,*) 'input maximum z-axis value'
      read(*,*) fac

      call pdev('ln03', ieer)

*       plotting

      call hwshd
      call swissm
      call shdchr(90.,1,0.002,1)
      call height(0.2)
      call page(8.5,11.)
      call physor(1.5,1.5)
```

```
        call area2d(6.,6.)
        call headin('WIGNER-VILLE DISTRIBUTION CONTOUR$',
     *              100,1.1,1)
        call blsur
        call yname('Freque icy (Hz) $',100)
        call xname('Time (sec) $',100)
        call yaxang(0.)
        tstep=ttime/5.
        fstep=(fmax-fmin)/5.
        tmax=tini+ttime
        call graf(tini,tstep,tmax,fimn,fstep,fmax)

        call frame
        call bcomon(9000)
        scale=1./30.
        call conmak(r,ny,nx,scale)
*
*       contour plot
*
        call conlin(0,'SOLID', 'NOLABELS',1,3)
        call conlin(1.,'DASH', 'NOLABELS',1,1)
        call conang(50.)
        call raspln(0.25)
        call contur(2,'LABELS','DRAW')

        call endpl(o)
        call donepl

        stop
        end
```

B-4. PROGRAM LIST OF PLOT OF TIME SERIES

```
        PROGRAM TIMEPLOT
*
*       This program uses the graphic package CA-DISSPLA to plot
*       the time series.
*
*       ----------- Declaring variables ----------
*
        dimension x(4000),y(4000)
        character*60 title
        character*25 inname

        write(*,*) 'input file name ?'
        read(*,1000) inname
1000    format(a25)
        write(*,*) 'title ?'
        read(*,1001) title
1001    format(a60)

        write(*,*) 'the number of the data point'
        read(*,*) np
        write(*,*) 'maximum scale of the time in a figure'
        read(*,*) xmax
        write(*,*) 'maximum scale of the magnitude in the figure'
        read(*,*) ymax
*
        open(8,file=inname,status='old')

        do 100 i=1,np
            read(8,*) x(i),y(i)
100     continue
```

```
      close(8)

      call pdev('ln03', ieer)

*     plotting

      call hwshd
      call swissm
      call shdchr(90.,1,0.002,1)
      call height(0.2)
      call page(8.5,11.)
      call physor(1.5,1.5)
      call area2d(6.,6.)
      call xname('Time (sec) $',100)
      call yname('Frequency (Hz) $',100)
      call headin(title,60,1.1,1)
      call thkfrm(0.01)
      call yaxang(90.)
      call graf(0.,xmax/4.,xmax,-ymax,0.5,ymax)
      call grid(1,1)
      call curve(x,y,np,0)

      call endpl(o)
      call donepl

      stop
      end
```

DISTRIBUTION LIST

No. of Copies

1. Defense Technical Information Center   2
   Cameron Station
   Alexandria, VA 22304-6145

2. Library, Code 52   2
   Naval Postgraduate School
   Monterey, CA 93943-5002

3. Department Chairman, Code ME   1
   Department of Mechanical Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5004

4. Professor Y. S. Shin, Code ME/Sg   3
   Department of Mechanical Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5004

5. Dr. Jae-Jin Jeon, Code ME/Je   2
   Department of Mechanical Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5004

6. Dr. Kam W. Ng   1
   Office of Naval Research
   Code 122
   800 N. Quincy Street
   Arlington, VA 22217-5000

7. Mr. Allen Woods   1
   Submarine Maintenance and Support Office
   Naval Sea Systems Command, PMS 390
   Washington, DC 20362-5101

8.   Mr. Robert Hagan   1
     Submarine Maintenance and Support Office
     Naval Sea Systems Command, PMS 390
     Washington, DC 20362-5101

9.   Mr. Myron Aleridge   1
     Submarine Maintenance and Support Office
     Naval Sea Systems Command, PMS 390
     Washington, DC 20362-5101

10. Mr. Robert A. Rohrbaugh   1
     Ship Silencing Evaluation Division
     DTRC Detachment
     Bremerton, WA 98314-5215