Two computationally efficient algorithms for digital cyclic spectral analysis, the FFT Accumulation Method (FAM) and the Strip Spectral Correlation Algorithm (SSCA) are developed from a series of modifications on a simple time smoothing algorithm. The signal processing, computational, and structural attributes of time smoothing algorithms are presented with emphasis on the FAM and SSCA.

Computationally Efficient Algorithms for Cyclic Spectral Analysis

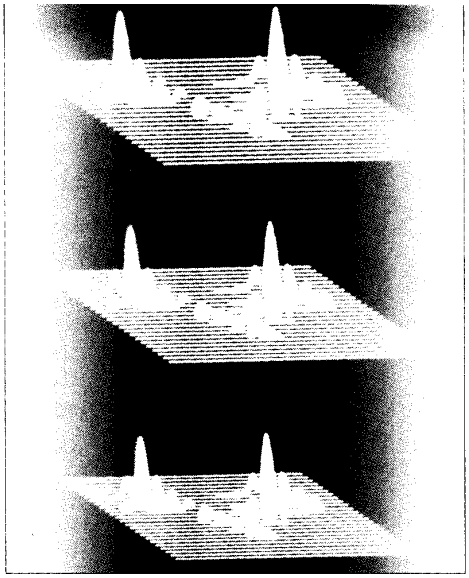
RANDY S. ROBERTS, WILLIAM A. BROWN,
and HERSCHEL H. LOOMIS, JR.

The need for computationally efficient cyclic spectral analysis algorithms becomes increasingly evident as cyclic spectral analysis grows in importance as a signal analysis tool [1]. For many signal analysis problems the computational complexity of cyclic spectral analysis far exceeds that of conventional spectral analysis. The reason for the computational complexity of cyclic spectral analysis lies in the nature of the estimation problem. Essentially, cyclic spectral analysis algorithms estimate the correlation between spectral components of signals. In the simplest case the spectral components of a real-valued signal are correlated, whereas in the most general case spectral components of two complex-valued signals are correlated. It is the potentially large number of correlation computations, rather than computing the spectral components, that makes cyclic spectral analysis computationally complex. Over the last six years several computationally efficient cyclic spectral analysis algorithms have evolved from the original methods introduced in [2] and [3]. The objective of this paper is to present these algorithms and describe their signal processing, computational, and structural properties.

Cyclic spectral analysis algorithms generally fall into two classes: those that average in frequency (frequency smoothing) and those that average in time (time smoothing). Although both classes of algorithms produce similar approximations to the cyclic spectrum, time smoothing algorithms are considered to be more computationally efficient for general cyclic spectral analysis. Frequency smoothing algorithms can be computationally superior to time smoothing algorithms in certain restricted cases, e.g., for estimating the cyclic spectrum for a few values of cycle frequency or estimating the cyclic spectrum for small time-frequency resolution product [8]. With computational efficiency for general cyclic spectral analysis as our primary motivation, we focus our attention in this paper on time smoothing algorithms.

Our discussion of time smoothing algorithms begins by describing an algorithm based on the time smoothed cyclic cross periodogram. This algorithm is considered to be

1053-5888/91/0400-0038 $1.00 ©1991 IEEE



the most fundamental time smoothing algorithm and serves to illustrate the general characteristics of time smoothing algorithms. A mathematical description of the basic algorithm that lends itself to the study of algorithm attributes such as frequency and cycle frequency resolution, and also computational complexity, is developed. The mathematical description is quite general and we apply it to other algorithms as well. From there we develop successively more sophisticated (and substantially less computationally complex) algorithms by refining the basic algorithm. After several intermediate algorithms we arrive at two computationally efficient algorithms: the FFT Accumulation Method (FAM) and the Strip Spectral Correlation Algorithm (SSCA).

In order to fully examine the algorithms we consider the problem of estimating the cyclic cross spectrum of two complex-valued sequences. This problem is the most general and computationally demanding problem of digital cyclic spectral analysis. [1]Simplification of the resulting expressions to special cases of the cross cyclic spectrum of two complex-valued sequences, such as the cyclic spectrum of a single real-valued sequence, are easily found by replacing references to $y(n)$ with $x(n)$ where needed. Computational and structural simplifications arising from the specialization are described.

BASIC TIME SMOOTHING ALGORITHMS

An implementation of the time smoothed cyclic cross periodogram

All time smoothing algorithms are based on the time smoothed cyclic cross periodogram [2]:

$$S^{\alpha}_{xy_T}(n,f)_{\Delta t}=\frac{1}{T}\left\langle X_T(n,f+\alpha/2)Y_T^*(n,f-\alpha/2)\right\rangle_{\Delta t} \qquad (1)$$

The time smoothed cyclic cross periodogram has the physical interpretation of correlating spectral components of $x(n)$ with spectral components of $y(n)$ over a time span of Δt *seconds. The spectral components* $X_T(n,f+\alpha/2)$ and $Y_T(n,f-\alpha/2)$, also called complex demodulates, are the complex envelopes of narrowband, bandpass components of a signal. Figure 1 illustrates the processing for a special case of Equation (1), namely, the time smoothed cyclic periodogram of a real signal. A data tapering window of length T seconds slides over the data for a time span of Δt seconds. At each instant the complex demodulates of the data within the window are computed. (Details of computing the complex demodulates are considered later. For now, note that the demodulates are lowpass sequences and have bandwidths on the order of the reciprocal of the data tapering window, i.e., $\Delta f = 1/T$ Hz.) After the complex demodulates have been computed, they are

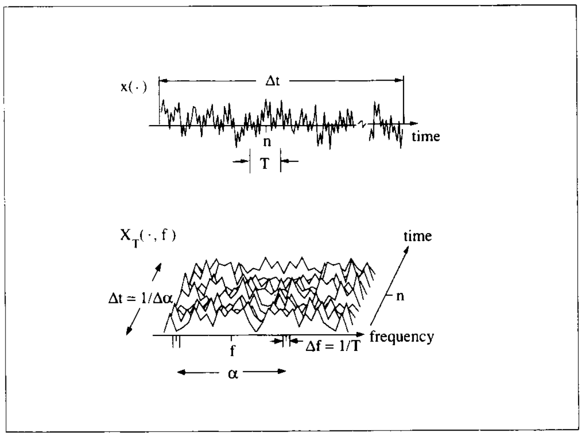


Fig. 1. Estimating the time smoothed cyclic periodogram of a single real-valued signal.

This work was supported in part by a grant from ESL Inc. with partial matching support from the California State MICRO Program (PI: W. A. Gardner).

[1] Complete cyclic spectral analysis of two complex sequences x(n) and y(n) requires the estimation of both $S^{\alpha}_{xy}(f)$ and $S^{\alpha}_{xy^*}(f)$.

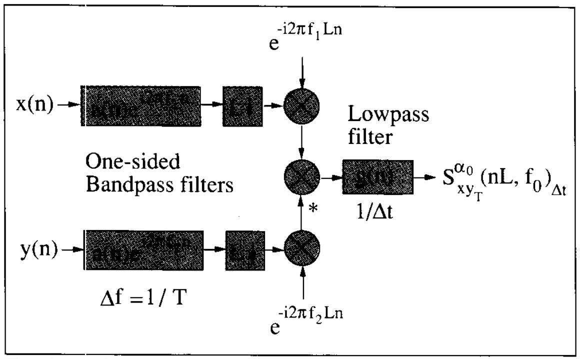


Fig. 2. Implementation of the time smoothed cyclic cross periodogram.

correlated by time averaging their conjugate product over an interval of Δt seconds to produce estimates of the cyclic spectrum. To estimate the cyclic spectrum at the point (f_0, α_0), demodulates separated in frequency by an amount α_0 and centered about a midpoint of f_0 are correlated. The quantities Δt and Δf are referred to as the time and frequency resolutions of the point estimate.

A system based on Equation (1) is illustrated in Fig. 2 (for now let the time decimation parameter $L = 1$). In this approach the complex demodulates are obtained by filtering the input sequences with one-sided bandpass filters and frequency shifting the filter outputs to baseband. To compute the point estimate at (f_0, α_0), the center frequencies of the filters are set to $f_1 = f_0 + \alpha_0 / 2$ and $f_2 = f_0 - \alpha_0/2$. Following the notation in [4], the input filters in Fig. 2 have memory lengths of $T = N' T_s$ seconds and therefore bandwidths on the order of $1/T = f_s/N'$Hz. (T_s is the sampling interval and f_s is the sampling frequency.) Mathematically, computation of the complex demodulates is expressed as

$$X_T(n,f) = \sum_{r=-N/2}^{N/2} a(r)x(n-r)e^{-i2\pi f(n-r)T_s} \qquad (2)$$

where $a(r)$ is a data tapering window of length $T = N'T_s$ seconds. For convenience, the quantity Δa is defined to be the bandwidth of the input filters; hence, for this system the bandwidth of the input filters is also the frequency resolution of the estimate: $\Delta f = \Delta a = f_s/N'$.

After the complex demodulates have been computed they are correlated over a time span of Δt seconds. In Fig. 2 the correlation operation is performed by a complex multiplier followed by a lowpass filter. The lowpass filter has a memory length of $\Delta t = NT_s$ seconds and a bandwidth on the order of $1/\Delta t = f_s/N$ Hz. The correlation operation is expressed as

$$S^{\alpha_0}_{xy_T}(n, f_0)_{\Delta t} = \sum_r X_T(r, f_1)Y^*_T(r, f_2)\, g(n-r) \qquad (3)$$

where $g(n)$ is a data tapering window of width $\Delta t = NT_s$ seconds. The resulting output sequence $S^{\alpha_0}_{xy_T}(n, f_0)_{\Delta t}$ is the spectral cross correlation estimate at (f_0, α_0) and is composed of a bias (if spectral correlation is present) whose magnitude indicates the degree of spectral correlation, and a random component that can be interpreted as measurement noise. For a reliable estimate it is necessary to have $\Delta t \gg T$, or equivalently, to have the time-frequency resolution product be much greater than unity: $\Delta t \Delta f \gg 1$. It is shown in [2] that the time smoothed cyclic cross periodogram converges to the cyclic cross spectrum in the limit, as $\Delta t \to \infty$ followed by $\Delta f \to 0$, if the time windows $a(n)$ and $g(n)$ are properly normalized. Therefore, if $\sum_n a^2(n) = \sum_n g(n) = 1$ we have

$$\lim_{\Delta f\to 0}\lim_{\Delta t\to\infty} S^{\alpha_0}_{xy_T}(n,f_0)_{\Delta t} = S^{\alpha_0}_{xy}(f_0) \qquad (4)$$

where the cyclic cross spectrum $S^{\alpha_0}_{xy}(f_0)$ is defined by

$$S^{\alpha_0}_{xy}(f_0) = \sum_{k=-\infty}^{\infty} R^{\alpha_0}_{xy}(k)e^{-i2\pi f_0 kT_s} \qquad (5)$$

and the cyclic cross correlation $R^{\alpha_0}_{xy}(k)$is defined by

$$R^{\alpha_0}_{xy}(k) = \lim_{\Delta t\to\infty} \left\langle x(nT_s + kT_s)y^*(nT_s)e^{-i2\pi(n+k/2)T_s} \right\rangle_{\Delta t} \qquad (6)$$

See [2] and [5] for further details on the above relationships.

An interesting representation of the basic system is found by combining expressions for the complex demodulates with Equation (3) to get [4]

$$S^{\alpha_0}_{xy_T}(n,f_0)_{\Delta t} = \sum_q \sum_r m(q,r)x(n-q)y^*(n-r)e^{-i2\pi\alpha_0 nT_s} \qquad (7)$$

Equation (7) is a quadratic transformation of the input sequences with the kernel $m(q,r)$. By manipulating Equations (2), (3), and (7), the kernel for the basic system is found to be

$$m(q,r) = \sum_p g(p)a(q-p)a(r-p)e^{i2\pi f_0(q-r)T_s}e^{i\pi\alpha_0(q+r)T_s} \qquad (8)$$

In general, most cyclic spectral analysis algorithms can be mathematically described in the form of Equation (8), that is, as a quadratic transformation of the input sequences with a kernel that depends on the system parameters f_0, α_0, Δf, and Δt [2], [5]. The interesting feature of this representation is that the system parameters are contained solely within $m(q,r)$. Hence, Equation (8) describes how system parameters influence estimation of the cyclic cross spectrum. Although the system kernel provides a compact mathematical representation of cyclic spectral analysis algorithms, a transformed version of this representation provides greater insight into the algorithms.

In terms of the rotated Fourier Series Transform (FST) of the kernel $m(q,r\ )$

$$M(\alpha,f) = \sum_q \sum_r m(q,r)e^{-i2\pi(f+\alpha/2)qT_s}e^{i2\pi(f-\alpha/2)rT_s} \qquad (9)$$

the output is expressed as [4]

$$S^{\alpha_0}_{xy_T}(n,f_0)_{\Delta t}=T_s\sum_{\Re}\int_{-f_s/2}^{f_s/2} M(\beta,\nu)\,S^{\beta}_{x}(\nu)\,d\nu\, e^{i2\pi(\beta-\alpha_0)nT_s}+R(n)\quad(10)$$

where the $\Sigma_{\Re}$-indicated summation is over the region $-f_s/2 < \beta - \alpha_0 \le f_s/2$, and $R(n)$ is a random component that can be interpreted as measurement noise. Note that Equation (10) also describes how system parameters (represented by $M(\alpha,f\ )$) influence the estimation of $S^{\alpha}_{xy}(f)$, the underlying cyclic cross spectrum. Cycle features of $S^{\alpha}_{xy}(f)$ that are within the domain of $M(\alpha,f)$ are summed in cycle frequency and integrated in frequency to produce the output value of the estimate. Cycle features outside the domain of $M(\alpha,f)$ are suppressed. Thus, by understanding the properties of $M(\alpha,f)$ we can understand how system parameters affect the estimate. As a general rule, if $\Delta t\Delta f \gg 1$ then the kernel transform $M(\alpha,f)$ can often be approximated by the separable form [2]

$$M(\alpha,f)\approx G_{1/\Delta t}(\alpha-\alpha_0)H_{1/T}(f-f_0)\qquad(11)$$

The components of the kernel, $G_{1/\Delta t}(\alpha)$ and $H_{1/T}(f)$, are frequency windows that are simply related to the FSTs of data tapering windows used in the algorithm under consideration, and have approximate bandwidths of $1/\Delta t$ and $1/T$ Hz respectively. Typically, $M(\alpha,f)$ is a two dimensional pulse-like function surrounded by sidelobes. The sidelobes of $M(\alpha,f)$ are generally small compared to the mainlobe but can be troublesome in some measurements. The region of support of the main lobe of $M(\alpha,\ f)$ is called a Cyclic Spectrum Analyzer (CSA) cell. Equation (11) indicates that a CSA cell has a width on the order of $1/\ \Delta t$ in cycle frequency and a length on the order of $1/T$ in frequency. As described by Equation (10), features of $S^{\alpha}_{xy}(f)$ within a CSA cell are transmitted to the output while features outside of the region are suppressed. Thus, the width of a CSA cell determines the cycle frequency resolution $\Delta\alpha$ and the length of the cell determines the frequency resolution Δf. Figure 3 depicts an idealized (i.e., no sidelobes or skirts) CSA cell located at $(f_0,\ \alpha_0)$. For a proper measurement the cycle frequency resolution must be small enough to resolve the cycle features of $S^{\alpha}_{xy}(f)$ and the frequency resolution must be small enough to resolve $S^{\alpha}_{xy}(f)$ in frequency. Note that cycle features exterior to a CSA cell but within the sidelobes of $M(\alpha,\ f)$ can contribute to the estimate of $S^{\alpha}_{xy}(f)$. This undesirable effect is called cycle leakage and is minimized by designing $M(\alpha,f\ )$ to have sufficiently low sidelobes.

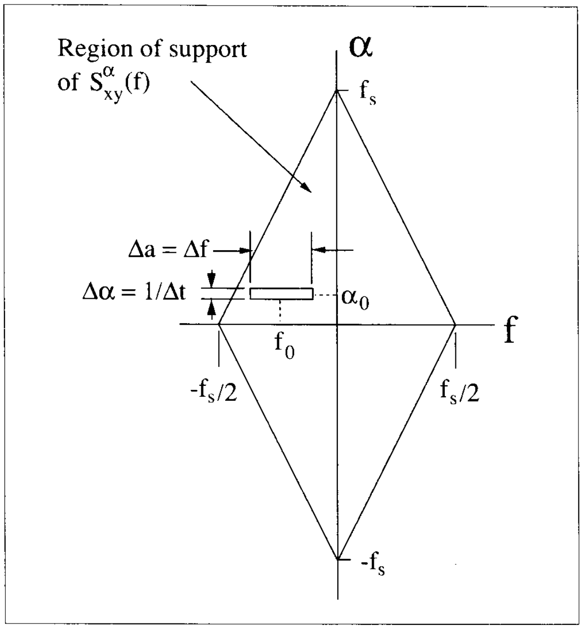

Region of support of $S^{\alpha}_{xy}(f)$
α
f_s
$\Delta a = \Delta f$
$\Delta\alpha = 1/\Delta t$
α_0
f
f_0
$-f_s/2$
$f_s/2$
$-f_s$

Fig. 3. A Cyclic Spectrum Analyzer (CSA) cell in the bifrequency plane.

The kernel transform is a useful means for studying the frequency resolution and cycle frequency resolution of an algorithm. As an example, the kernel transform for the basic system is

$$M(\alpha,f)=G(\alpha-\alpha_0)A\left(f-f_0+\frac{\alpha-\alpha_0}{2}\right)A^{*}\left(f-f_0-\frac{\alpha-\alpha_0}{2}\right)\qquad(12)$$

where $A(f)$ and $G(\alpha)$ are the FSTs of $a(n)$ and $g(n)$. (Note that the bandwidth of $A(f)$ is on the order of $\Delta a = 1/T$ and that of $G(a)$ on the order of $1/\Delta t$.) Since $f_1 = f_0 + \alpha/2$ and $f_2 = f_0 - \alpha_0/2$, the transform kernel can also be expressed as

$$M(\alpha,f)=G(\alpha-\alpha_0)A\left(f-f_1+\frac{\alpha}{2}\right)A^{*}\left(f-f_2-\frac{\alpha}{2}\right)\qquad(13)$$

Examination of Equations (12) or (13) reveals that most of the energy in $M(\alpha,f)$ is indeed confined to the

TABLE I

BASIC TIME SMOOTHING

Computation Section	Number of Complex Multiplications	
	Cyclic Cross Spectrum of Two Complex Signals	Cyclic Spectrum of a Single Real Signal
Filtering	$2\ N^2N'$	N^2N'
Frequency Shift	$2\ N^2$	N^2
Correlate	N^2N'	$N^2N'/4$

Computational complexity of the basic time smoothing algorithm in terms of the number of complex multiplications. Note that complete coverage of the bifrequency plane requires N N′ point estimates.

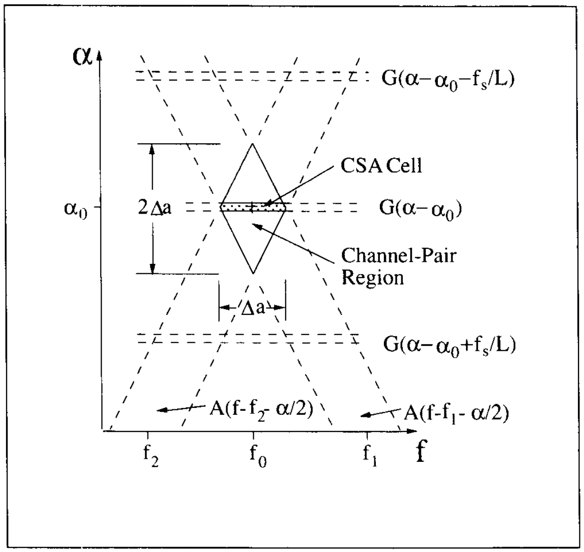


Fig. 4. The region of support of $M(\alpha, f)$ for the basic time smoothing algorithm.

region $|f - f_0| \leq \Delta a/2$ and $|\alpha - \alpha_0| \leq 1/2\Delta t$ (cf. Fig. 3). Hence, the frequency resolution of this algorithm is again seen to be $\Delta f = \Delta a = f_s/N'$, and cycle frequency resolution is $\Delta\alpha = 1/\Delta t = f_s/N$. Finally, the time-frequency resolution product is $\Delta t \Delta f = N/N'$.

In addition to determining the resolutions of a cyclic spectral analysis algorithm the CSA cell concept is useful for determining the computational complexity of an algorithm. Consider the problem of estimating the cyclic cross spectrum of two complex signals everywhere in the bifrequency plane at a given time instant. To perform the estimation approximately NN' point estimates must be calculated (this is the area of the region of support of $S_{xy}^{\alpha}(f)$ divided by the area of a CSA cell). All of the input data can be filtered with two banks of N one-sided bandpass filters for a total of N^2N' complex multiplications and the complex demodulates calculated with an additional $2N^2$ complex multiplications. Additionally, N^2N' complex multiplications are needed to compute the spectral correlations for all point estimates (assuming that $g(n)$ is a rectangular window with unity height). Table I summarizes the computational complexity of the algorithm in terms of the number of complex multiplications required to compute estimates of the cyclic cross spectrum of two complex signals and the cyclic spectrum of a single real signal.

To put these numbers into perspective a numerical example is considered. Say that we want to estimate the cyclic cross spectrum of two complex sequences for $\Delta f = 1/8$ $(T_s = 1)$ and $\Delta t\, \Delta f = 16384$. For these values, N'=8 and N=131072. If the computations are performed on a uniprocessor computer (i.e., one ALU) capable of producing one complex product every 150 ns, then the computation would take approximately 18.6 hrs. For the case of estimating the cyclic spectrum of a real signal the symmetry relations $S_x^{\alpha}(-f) = S_x^{\alpha}(f)$ and $S_x^{-\alpha}(f) = S_x^{\alpha}(f)^*$ imply that $S_x^{\alpha}(f)$ need only be estimated in one quadrant of the bifrequency plane. Thus, the previous estimation problem with a single real signal would require approximately 7.9 hrs of computation. This example serves to illustrate the "brute force" approach to digital cyclic spectral analysis. Algorithms that are considerably more efficient are presented in the remainder of this paper.

Time smoothing with decimation

The computational efficiency of the previous algorithm can be improved by decimating the outputs of the bandpass filters by a suitable factor. Equivalently, data is shifted into the filters in blocks of L samples where $L<N'$; thus, only N/L samples are processed for each point estimate and the overall computational complexity of the algorithm is reduced by the factor L. Since the filter outputs are over sampled by a factor of N' the sampling rate can be reduced to f_s/L, $L \leq N'$ before aliasing occurs. However, if the sampling rate is reduced by the maximum factor $L = N'$, then the effects of cycle leakage can be substantial. With decimation in effect the estimate by Equation (3) is modified to

$$S_{xy_T}^{\alpha_0}(nL, f_0)_{\Delta t} = \sum_r X_T(rL, f_1) Y_T^*(rL, f_2) g_c(n-r) \qquad (14)$$

Equation (10) still applies except that $M(\alpha, f)$ now also accounts for decimation. Shifting the input sequences into the system in blocks has the effect of mutating the lowpass filter G(.) in Equations (12) and (13) into the comb filter [2]

$$G_c(\alpha) = \sum_r g_c(r) e^{-i2\pi\alpha r L T_s} \qquad (15)$$

$$= \sum_n G\left(\alpha + \frac{nf_s}{L}\right) \qquad (16)$$

where $G(\alpha)$ consists of one period of $G_c(\alpha)$,

$$G(\alpha) = \begin{cases} G_c(\alpha) & |\alpha| < \dfrac{f_S}{2L} \\ 0 & \text{otherwise} \end{cases}$$

Consequentially, the kernel transform is now expressed as (cf. Equation (13))

$$M(\alpha, f) = \sum_n G\left(\alpha - \alpha_0 + \frac{nf_s}{L}\right) A\left(f - f_1 + \frac{\alpha}{2}\right) A^*\left(f - f_2 - \frac{\alpha}{2}\right) \qquad (17)$$

It is convenient to consider Equation (17) as consisting of two components: the product of frequency shifted channelizer transfer functions

$$M_1(\alpha, f) = A\left(f - f_1 + \frac{\alpha}{2}\right) A^*\left(f - f_2 - \frac{\alpha}{2}\right) \qquad (18)$$

and the comb filter

$$G_c(\alpha - \alpha_0) = \sum_n G\left(\alpha - \alpha_0 + \frac{nf_s}{L}\right) \qquad (19)$$

The bandwidth of $A(f)$ is on the order of Δa and $G_c(\alpha)$ has teeth with bandwidths on the order of $1/\Delta t$ spaced f_s/L Hz apart in cycle frequency. The approximate

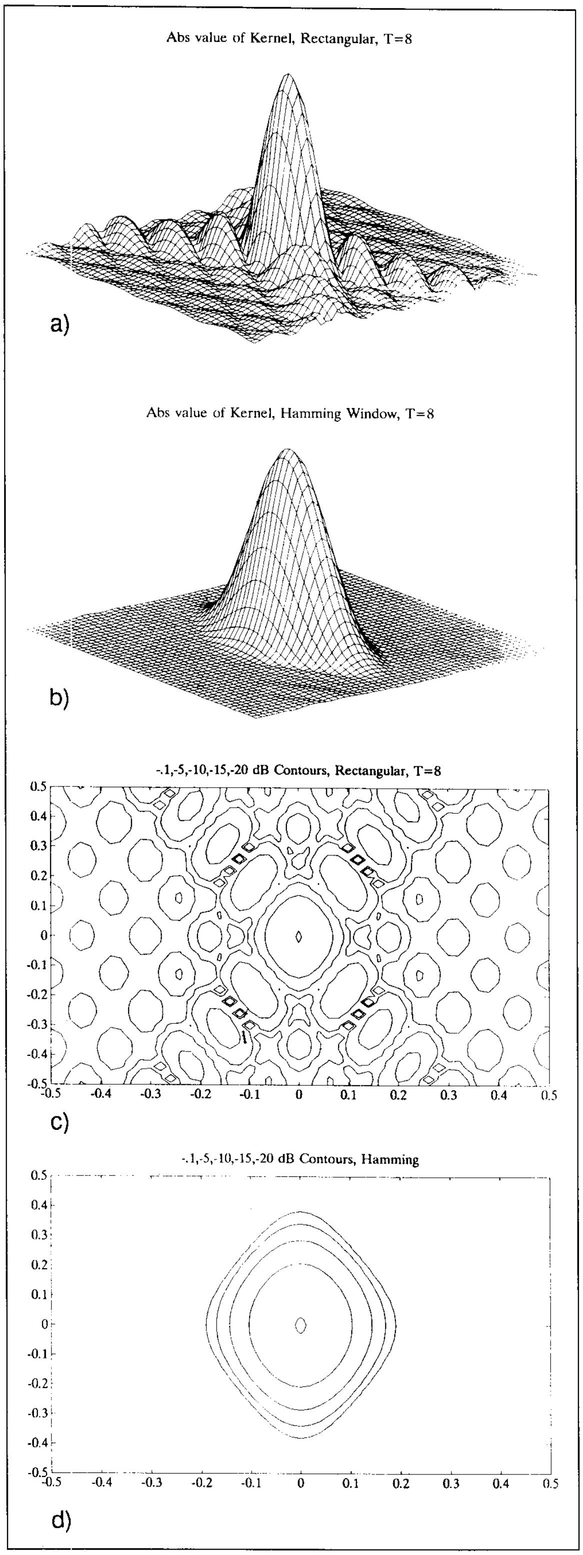


Fig. 5 Channelizer transfer function for a) a rectangular window and b) a Hamming window. Corresponding contour plots are c) rectangular and d) Hamming.

regions of support of $M_1(\alpha, f)$ and $G_c(\alpha - \alpha_0]$ are illustrated in Fig. 4.

The diamond shaped region in Fig. 4 is called a channel-pair region, and it is the idealized (no sidelobes or skirts) region of support of $M_1(\alpha, f)$. A channel-pair region is approximately diamond shaped with length and width on the order of $2\Delta a$ and Δa respectively. The intersection of a channel-pair region with the region of support of the $n=0$ tooth of $G_c(\alpha - \alpha_0)$ at (f_0, a_0) defines a CSA cell. Using the fixed ratio $P = N/L$ we can express the cycle frequency resolution of the decimated algorithm as $\Delta\alpha = f_s/PL$. Note that the frequency resolution of the algorithm remains the same as before: $\Delta f = f_s/N'$. Although the boundaries of the CSA cell and channel--pair region in Fig. 4 are depicted as well defined, this is generally not the case. Both $M_1(\alpha, f)$ and $G_c(\alpha - \alpha_0)$ can have significant skirts and sidelobes beyond the regions indicated in Fig. 4. Figure 5a illustrates the magnitude of $M_1(\alpha, f)$ for $a(n)$ a rectangular data tapering window and Fig. 5b shows $M_1(\alpha, f)$ for $a(n)$ a Hamming window. Contour plots of Figs. 5a and 5b are shown in Figs. 5c and 5d respectively.

Recall that cycle leakage occurs if the sidelobes of $M(\alpha, f)$ are large enough to admit nearby cycle features. In the decimated algorithm, cycle leakage can also occur if teeth of $G_c(\alpha - \alpha_0)$ other than the one at α_0 intersect the channel-pair region. Overlap of a channel-pair region with $n \neq 0$ teeth of $G_c(\alpha - \alpha_0)$ occurs if L is too small or if the channel-pair region is larger than indicated in Fig. 4. An additional effect, known as cycle aliasing, occurs if the frequencies of leaked cycle features exceed the decimated sampling rate. We postpone determination of appropriate values for L until the discussion of the FAM.

FFT BASED TIME SMOOTHING ALGORITHMS

Time smoothing with a Fourier transform

Consider frequency shifting the product sequence in Fig. 2 by an amount ε from α_0 to $\alpha_0 + \varepsilon$ (see Fig. 6 for a block diagram of the processing). In this case, the output of the system is given by

$$S_{xy_T}^{\alpha_0+\varepsilon}(n,f_0)_{\Delta t} = \sum_r X_T(r,f_1)\, Y_T^*(r,f_2)\, g(n-r)\, e^{-i2\pi\varepsilon r T_s} \quad (20)$$

Following some manipulations the kernel transform is found to be

$$M(\alpha,f) = G(\alpha-\alpha_0-\varepsilon)A(f-f_1+\alpha/2)A^*(f-f_2-\alpha/2) \quad (21)$$

The region of support for the kernel transform is depicted in Fig. 7. It is seen that $M(\alpha, f)$ is a pulse centered at $(f_0, \alpha_0 + \varepsilon)$ with a width of $\Delta\alpha = 1/\Delta t$ and length of $\Delta f = \Delta a - |\varepsilon|$, where $|\varepsilon| < \Delta a$. If several values of ε are desired, evaluation of the sum in Equation (20) can be simplified by discretizing the values of ε to be ε

= $q\Delta\alpha$. In this case the output of the algorithm is expressed as

$$S_{xy_T}^{\alpha_i+q\Delta\alpha}(n,f_j)_{\Delta t}=\sum_r X_T(r,f_1)Y_T^*(r,f_2)g(n-r)\,e^{-i2\pi rq/N} \quad (22)$$

in which the sum in Equation (22) can be evaluated with an N-point FFT. Thus, point estimates with constant cycle frequency can be computed in blocks by Fourier transforming the product sequence instead of averaging the product sequences individually as in Equation (1).

For complete coverage of the bifrequency plane a bank of bandpass filters is required to produce the necessary complex demodulates. An efficient method for producing the required complex demodulates is based on a sliding FFT [6]. In this approach the frequencies of the filter bank are discretized to $f_k = k(f_s/N'), k = -N'/2...(N'/2)-1$. The channel-pair regions associated with all pairs of complex demodulates are located at (f_j, α_i) where the frequency coordinates are

$$f_j=\frac{f_k+f_l}{2} \quad (23)$$

$$=\frac{k+l}{2}\left(\frac{f_s}{N'}\right) \quad (24)$$

and the cycle frequency coordinates are

$$\alpha_i=f_k-f_l \quad (25)$$

$$=(k-l)\left(\frac{f_s}{N'}\right) \quad (26)$$

Figure 8 shows the pattern of channel-pair regions for $N' = 8$. The ordered pair associated with each channel-pair region in Fig. 8 is the coordinate of the region written in terms of indices: $(f_j, \alpha_i) \rightarrow (j, i)$. For an N'-point channelizer there are $(N')^2$ possible combinations of channelizer streams; hence, there are at most $(N')^2$ channel-pair regions. Due to symmetry, estimation of the cyclic spectrum of a single real signal requires only $(N')^2/4$ channel-pair regions (one quadrant of the bifrequency plane).

A troubling aspect of this approach is the nonuniform frequency resolution of the point estimates (recall that $\Delta f = \Delta a - |\varepsilon|$). Near the top and bottom of a channel-pair region the frequency resolution approaches zero, and as $\Delta f \rightarrow 0$ so does the time-frequency resolution product. As a result, point estimates at the ends of channel-pair regions become increasingly unreliable (i.e., the estimates have high variability). Several methods have been developed to deal with this problem. One approach, developed in [7], combines estimates in adjacent channel-pair regions to obtain new estimates

$$S_{xy_T}^{\alpha_0}(f_0+\varepsilon_1/2)_{\Delta t}=(S_{xy_T}^{\alpha_0}(f_0)_{\Delta t}+S_{xy_T}^{\alpha_0}(f_0+\mathrm{sign}(\varepsilon_1)/2N')_{\Delta t})/2 \quad (27)$$

The estimates in this method have a frequency resolution of $\Delta f = \Delta a$ provided that $A(f)$ is rectangular. Another method, emphasized here, is to retain only those estimates within $\pm\Delta a/2$ of the center of the channel-pair region [4], [5]. Although this approach is simple to implement, the frequency resolution and time-frequency resolution product still vary within a channel-pair region. Furthermore, discarding estimates in this manner leaves coverage gaps in the bifrequency plane. However, if the input signals are resolvable with $\Delta f \approx \Delta a$ then loss of coverage is not detrimental since cycle features extend to at least one channel-pair region. See [5] for other ways to eliminate coverage gaps.

The FFT accumulation method

The FFT Accumulation Method (FAM) incorporates all of the ideas so far discussed [4], [5]. Channelization is performed by an N'-point FFT that is hopped over the data in blocks of L samples. The outputs of the FFT are frequency shifted to baseband to obtain decimated complex demodulate sequences. After the complex demodulates are computed, product sequences $X_T(nL, f_k)\, Y_T^*(nL, f_l)$ are formed and Fourier transformed with a P-point FFT. (Recall that $P=N/L$ where N is the total number of samples in the input sequences.) The output of the FAM is therefore

$$S_{xy_T}^{\alpha_i+q\Delta\alpha}(nL,f_j)_{\Delta t}=\sum_r X_T(rL,f_k)Y_T^*(rL,f_l)g_c(n-r)e^{-i2\pi rq/P} \quad (28)$$

and the kernel transform is given by Equation (17) with $f_0 = f_j$ and $\alpha_0 = \alpha_i + q\Delta\alpha$. The cycle frequency resolution of the FAM is identical to that of the decimated algorithm described above (see "Time smoothing with decimation"); thus, $\Delta\alpha = f_s/PL$. The frequency resolution of the FAM is identical to that of the Fourier transform smoothing algorithm of the preceding Section; thus, $\Delta f(q) = \Delta a - |q|\Delta\alpha$ and since $\Delta a = f_s/N'$

$$\Delta f(q)=\left(1-|q|\frac{N'}{PL}\right)\frac{f_s}{N'} \quad (29)$$

Likewise, the time-frequency resolution product is variable and is given as

$$\Delta t\Delta f=\Delta t(\Delta a-|q|\Delta\alpha) \quad (30)$$

$$=\frac{N}{N'}-|q| \quad (31)$$

The time-frequency resolution product of the FAM is

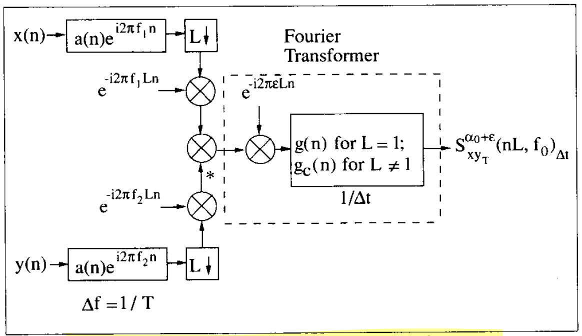


Fig. 6. The basic time smoothing algorithm with Fourier transformer output.

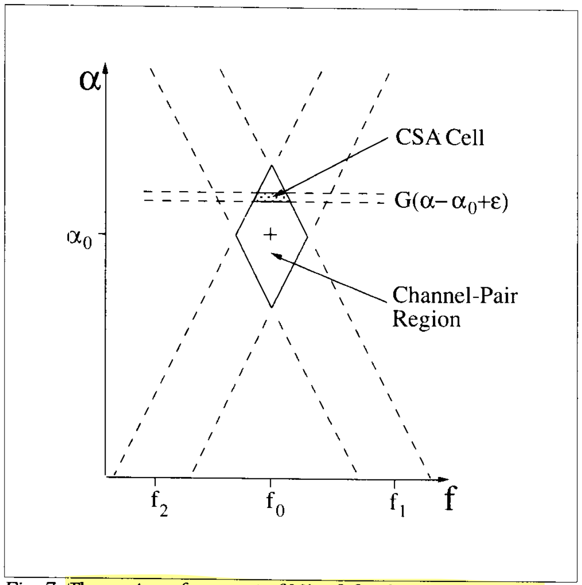


Fig. 7. The region of support of M(α, f) for the FFT Accumulation Method.

typically referenced to the center of a channel-pair region ($q=0$) so that $\Delta t\Delta f = N/N' = PL/N'$). To minimize the variability of point estimates near the top and bottom of the channel-pair regions we can retain only those estimates within $\pm\Delta a/2$ of the center of the region and discard all others. In terms of the index parameter q, only estimates within the range

$$-\frac{\Delta a}{2} \le q\Delta\alpha < \frac{\Delta a}{2} \tag{32}$$

or

$$-\frac{PL}{2N'} \le q \le \frac{PL}{2N'} - 1 \tag{33}$$

in Equation (28) are retained.

Both cycle leakage and cycle aliasing can be minimized by restricting the amount of overlap of the $n \neq 0$ teeth of $G_c(\alpha - \alpha_0)$ with the channel-pair region. Diminishing the overlap is accomplished in two ways. First, L must be selected to ensure that the passbands of $G_c(\alpha - \alpha_0)$ are sufficiently removed from the channel-pair region. In practice, a decimation factor of $L = N'/4$ has been found to be a good compromise between maintaining computational efficiency and minimizing cycle leakage and cycle aliasing [4]. Second, cycle leakage can be reduced by minimizing the skirts and sidelobes of $M_1(\alpha, f)$ and $G_c(\alpha)$. From Equation (18) it is evident that a channelizer with steep transition bands and low sidelobes produces low skirts and low sidelobes in the channel-pair region. Thus, a data tapering window whose Fourier transform has low skirts and low sidelobes (e.g., the Hamming window) is desirable. A data tapering window for the output lowpass filter is not as crucial as a data tapering window for the input bandpass filters. For simplicity $g_c(p)$ is often taken to be a rectangular window.

The computational complexity of the FAM, in terms of the number of complex multiplications required to compute estimates of the cyclic cross spectrum of two complex signals and the cyclic spectrum of a single real signal, is given in Table II. In order to give some feel for the magnitude of calculations typically encountered when estimating the cyclic cross spectrum with the FAM, a numerical example is given. Consider the previous problem of estimating the cyclic cross spectrum of two complex signals where a time-frequency resolution product of $\Delta t\Delta f = 16384$ and a frequency resolution of $\Delta f = 1/8$ is desired. Channelization for this problem requires approximately 2×10^6 complex multiplications; computing and Fourier transforming all product sequences requires approximately 4×10^6 and 67×10^6 complex multiplications respectively. If a complex multiplication is performed every 150 ns, this computation would take approximately 11 seconds to perform. Estimating the cyclic spectrum of a real signal with the same processing parameters would take on the order of 2.75 seconds.

A close look at the FAM reveals that the algorithm has a high degree of parallelism [8]. The parallelism in the FAM is a direct result of the independence of the product sequences $X_T(p,f_k)Y^*_T(p,f_l)$ for each k and l during and after their computation. A convenient way to describe the parallelism in the FAM is through a series of time-sequenced signal flow graphs (SFGs). The time sequencing of the signal flow graphs indicates the ordering of computations in the algorithm, that is, the output of SFG(i) is the input to SFG(i+1). The ordered

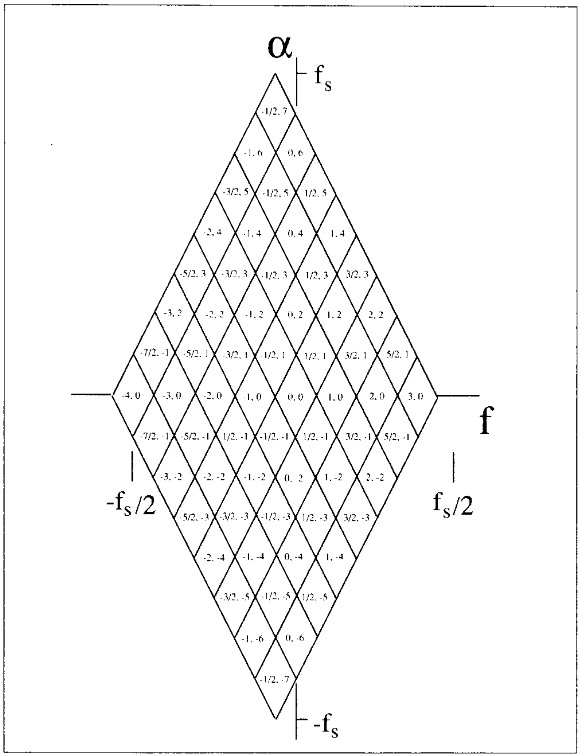


Fig. 8. Tiling the bifrequency plane with the FFT Accumulation Method for N′=8

TABLE II
FFT ACCUMULATION METHOD

Computation Section	Number of Complex Multiplications	
	Cyclic Cross Spectrum of Two Complex Signals	Cyclic Spectrum of a Single Real Signal
Channelizer:		
Data Tapering	2 N'P	†
N'-point FFT	$PN'\log_2 N'$	$P(N'/2)\log_2 N'$
Frequency Shift	2 N'P	N'P
Cross Multiply	$P(N')^2$	$P[(N')^2]/4$
FFT Product Sequences	$(N')^2(P/2)\log_2 P$	$[(N')^2/4](P/2)\log_2 P$

Computational complexity of the FAM in terms of the number of complex multiplications. † Note that N'P real multiplications are required for data tapering.

signal flow graph description also has the advantage of easy translation into digital architectures. For the FAM the signal flow graphs are:

SFG(1) Computation of the Complex Demodulates
SFG(2) Cross Multiplication, Complex Demodulates
SFG(3) P -point FFT array (including I/O buffers)

These three signal flow graphs are shown in Fig. 9.[2] For clarity the FFT signal flow graphs in SFG(1) and SFG(3) are shown as blocks. Additionally, only one node of SFG(3] is shown.

Operationally, data is input to SFG(1) in blocks of L samples and output in N'-point blocks. Each block of data contains $X_T(\cdot, f_k)$ and $Y^*_T(\cdot, f_k)$, $-N'/2 \leq k \leq (N'/2)-1$. As illustrated in Fig. 9, data from SFG(1) is input to SFG(2]. SFG(2), which cross multiplies all complex demodulates, is detailed in Fig. 10 for $N' = 8$. The nodes of SFG(2) in Fig. 10 are labeled according to channel--pair regions associated with the node (cf. Fig. 8). In general, SFG(2) consists of 2N' complex data paths and $(N')^2$ complex multiplier nodes configured in an $N' \times N'$ array. Complex demodulate streams flow through a node of SFG(2), are cross multiplied and fed into nodes of SFG(3). SFG(3) Fourier transforms the output product sequence streams of SFG(2). For each node in SFG(2) there is a corresponding node in SFG(3). The array structure of SFG(3) is similar to that of SFG(2) except that nodes of SFG(3) are not connected to one another – due to the independence of the product sequences. Each node of SFG(3) consists of an input buffer and a P-point FFT processor. The buffers convert the product sequence streams output from SFG(2) into P-point blocks suitable for Fourier transforming by the FFT processors.

The parallel description of the FAM can be used as a guide to mapping the FAM calculation onto a multiprocessor computer, or for designing application-specific architectures. In either case the algorithm is easily partitioned. The most natural partition for the FAM is by channel-pair region. If processing elements compute and Fourier transform all product sequences concurrently, then the amount of time required to estimate the cyclic cross spectrum decreases by a factor of $(N')^2$. Such an implementation significantly decreases the time required to perform the FAM computation since computing and Fourier transforming the product sequences represent the bulk of the computations. Whatever the implementation, it is important to maintain a smooth flow of data between the signal flow

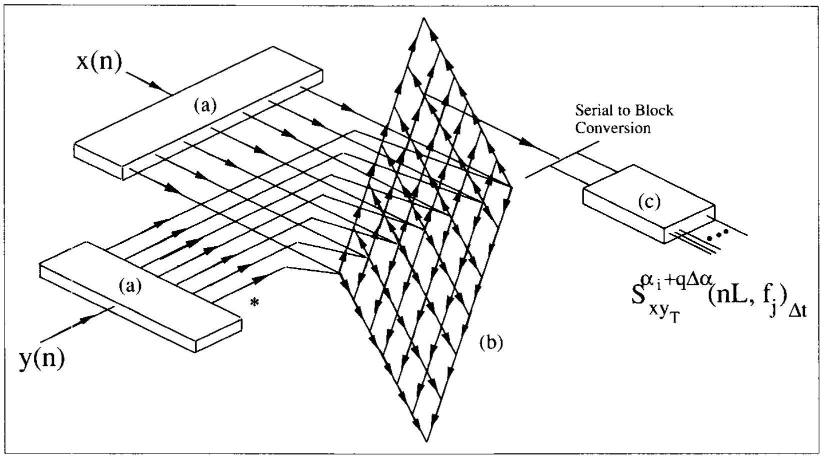

x(n)
(a)
Serial to Block Conversion
(c)
(a)
*
$S^{\alpha_i+q\Delta\alpha}_{xy_T}(nL, f_j)_{\Delta t}$
y(n)
(b)

Fig. 9. Interconnection of signal flow graphs in the FFT Accumulation Method.
a) N'- point FFT channelizers. b) SFG(2) for cross multiplication of complex demodulates.
c) P-point FFT processors.

[2] Note that the structure illustrated in Fig. 9 can be used to complete other quadratic surfaces, e.g. the cross ambiguity function, by modifying the pre- and post- quadratic tranform blocks (blocks (a) and (c) in Fig. 9).

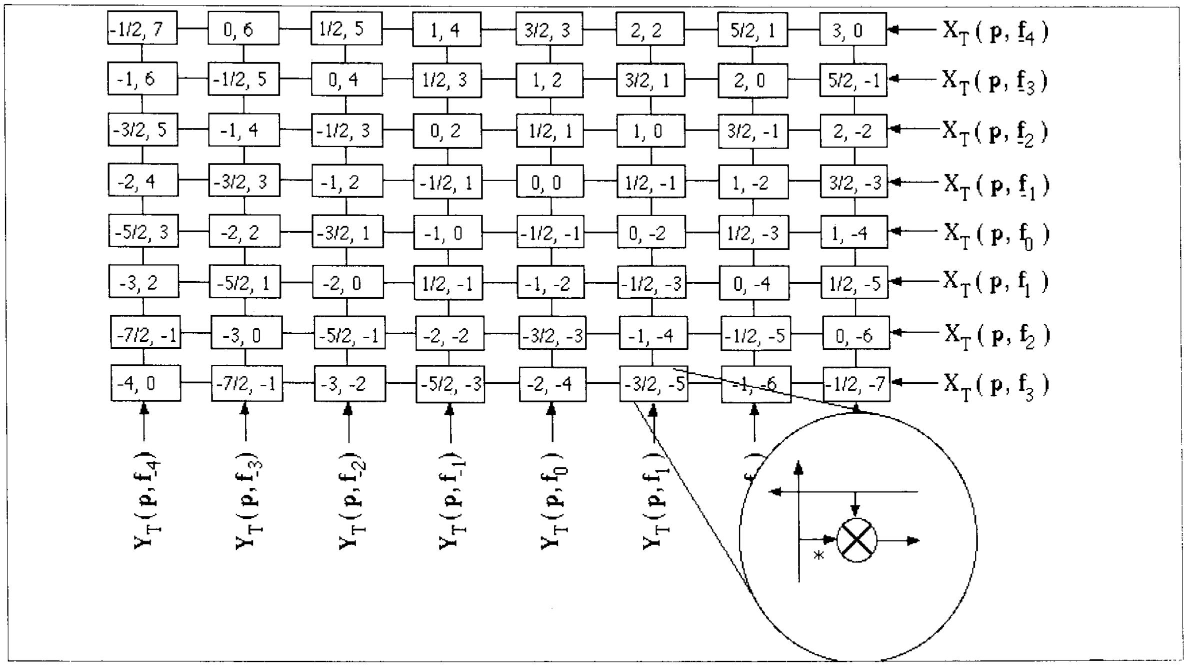


Fig. 10. SFG(2) in the FFT Accumulation Method for N′ = 8.

graphs, i.e., to computationally balance the realization. As an example, in [8] it is shown that architectures based explicitly on the parallel description of the FAM contain redundant arithmetic units. However, by computationally balancing designs based explicitly on the parallel description, efficient architectures can be realized.

The strip spectral correlation algorithm

The second FFT-based time smoothing algorithm is the Strip Spectral Correlation Algorithm (SSCA) [4], [5]. In the SSCA the complex demodulates $X_T(n,f_k)$ directly multiply $y^*(n)$. As a result, the point estimates produced by the SSCA lie along the frequency-skewed family of lines $\alpha = 2f_k - 2f$ (see Fig. 11). A strip of point estimates is computed using the formula

$$S_{xy_T}^{f_k+q\Delta\alpha}\left(n,\frac{f_k}{2}-q\frac{\Delta\alpha}{2}\right)_{\Delta t}=\sum_r X_T(r,f_k)y^*(r)g(n-r)e^{-i2\pi qr/N} \quad (34)$$

(Recall that $f_k = k(f_s/N')$, $-N'/2 \le k \le (N'/2) - 1$). The SSCA uses an N-point FFT to compute the sum in

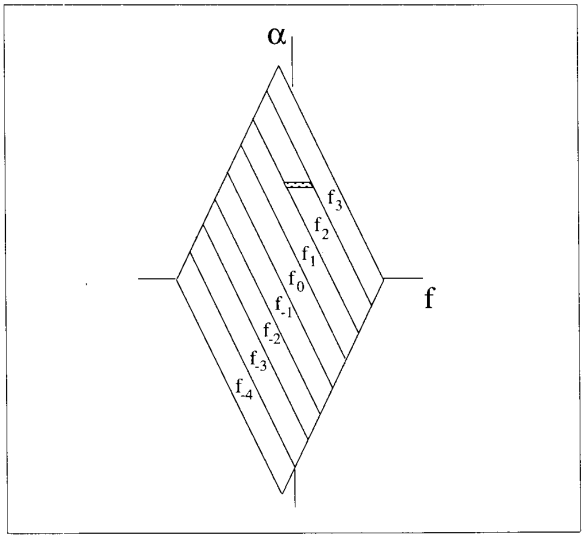


Fig. 11. Tiling the bifrequency plane with the Strip Spectral Correlation Algorithm for N′ = 8. A single CSA cell is shown..

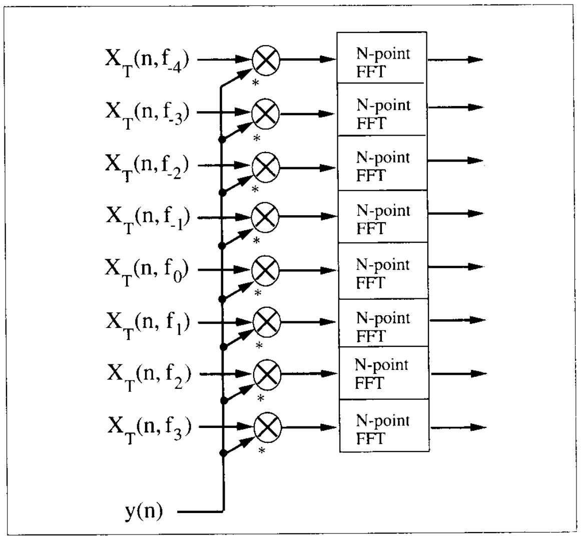


Fig. 12. Signal Flow Graphs for the Strip Spectral Correlation Algorithm excluding SFG(1) (the channelizer for x(n)).

TABLE III
STRIP SPECTRAL CORRELATION ALGORITHM

Computation Section	Number of Complex Multiplications	
	Cylic Cross Spectrum of Two Complex Signals	Cyclic Spectrum of a Single Real Signal
Channelizer:		
Data Tapering	NN′	†
N′-point FFT	N(N′/2)log$_2$N′	N(N′/2)log$_2$N′
Frequency Shift	NN′	NN′/2
Compute Product Sequences	NN′	NN′/2
FFTs Product Sequences	N′(N/2)log$_2$N	(N′/2)(N/2)log$_2$N

Computational complexity of the SSCA in terms of the number of complex multiplications. † Note that N N′ real multiplications are required for data tapering.

Equation (34). Note that the sampling rate of $X_T(n, f_k)$ cannot be decimated; in order for $X_T(n, f_k)$ to properly multiply $y^*(n)$ both sequences must be sampled at the same rate.

The properties of the SSCA are determined by studying the CSA cells produced by the algorithm. The SSCA kernel transform for a CSA cell located at (f_0,α_0) is [4]

$$M(\alpha,f)=G(\alpha-\alpha_0)A\left(f-f_0+\frac{\alpha-\alpha_0}{2}\right) \tag{35}$$

where the allowable values of f_0 and α_0 *are*

$$f_0=\frac{f_k}{2}-q\frac{\Delta\alpha}{2} \tag{36}$$

$$\alpha_0=f_k+q\Delta\alpha \tag{37}$$

In Equation (35), $G(\alpha)$ and $A(f)$ are the FSTs of $g(n)$ and $a(n)$. From Equations (35) – (37), the region of support of $A(f-f_0+(\alpha-\alpha_0)/2)$ is found to be a strip approximately Δa wide along the line $\alpha = 2f_k - 2f$. Thus, CSA cells in the SSCA have a constant length of Δa which implies that the SSCA has a constant frequency resolution of $\Delta f = \Delta a$. The uniform frequency resolution of the SSCA is in marked contrast to the nonuniform frequency resolution of the FAM. The width of a CSA cell, determined by the bandwidth of $G(\alpha)$, is gleaned from Equation (35) to be $1/\Delta t = f_s/N$. Hence, the cycle frequency resolution of the SSCA is $\Delta\alpha = 1/\Delta t = f_s/N$. The time-frequency resolution product of the SSCA is therefore $\Delta t\Delta f=N/N'$. Figure 11 shows the bifrequency plane tiled with strips of $A(\cdot)$. In all, there are N' strips in the cyclic cross spectrum estimate and $N'/2$ in the cyclic spectrum estimate.

Table III summarizes the computational complexity of the SSCA in terms of the number of complex multiplications required to estimate the cyclic cross spectrum of two complex signals and the cyclic spectrum of a single real signal. As an example of the computational complexity, the previous problem where $\Delta f = 1/8$ and $\Delta t\Delta f = 16348$ which corresponds to $N' = 8$ and $N = 131072$, would require approximately two seconds to execute. Estimating the cyclic spectrum of a single real signal with the same parameters would take approximately one second.

Like the FAM, the SSCA is a highly parallel algorithm and the parallel description of the SSCA is similar to the parallel description of the FAM but less complex [8]. The parallel description of the SSCA consists of three time-sequenced signal flow graphs:

SFG(1) Computation of Complex Demodulates
SFG(2) Computation of Product Sequences
SFG(3) *N*-point FFT (including I/O buffers)

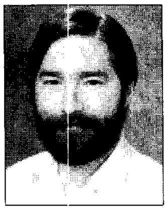
SFG(1) is identical to that of SFG(1) in the FAM parallel description except that no sample rate decimation occurs ($L=1$). SFG(2) is composed of N' complex multiplier nodes configured in a linear array. Figure 12 shows the SSCA signal flow graphs, excluding SFG(1), for $N' = 8$. Complex demodulate streams from SFG(1) enter SFG(2) on the left and product sequence streams exit SFG(2) on the right for SFG(3). The array structure of SFG(3) follows directly from the array structure of SFG(2). Additionally, the nodes of SFG(3) are identical to nodes of SFG(3) in the FAM parallel description (except N-point FFTs are used instead of P-point FFTs). The prior comments on the use of the parallel description to implement the FAM are also applicable to the SSCA.

SUMMARY

We began the discussion with a simple algorithm based on a direct implementation of the time smoothed cyclic cross periodogram. Although this algorithm is not computationally attractive for estimating the cyclic (cross) spectrum over the entire bifrequency plane, it illustrates the essence of time smoothing algorithms. Several representations of the basic time smoothing algorithm were developed next. The transformed kernel representation is perhaps the most important characterization of cyclic spectral analysis algorithms due to its generality and conceptual utility. From this representation the signal processing attributes of an algorithm, such as frequency and cycle frequency resolution, and phenomena such as cycle leakage can be studied. Additionally, the computational complexity of cyclic spectral analysis algorithms is readily determined using this representation. From the basic time smoothing algorithm several improved algorithms were developed. Modifications to reduce the computational complexity of the basic algorithm included decimating the complex demodulates prior to forming product sequences and time averaging the frequency shifted product sequences with FFTs. With the addition of an FFT based input channelizer, we arrived at the computationally efficient FFT Accumulation Method. A second computationally efficient algorithm, the Strip Spectral Correlation Algorithm was developed as an alternative to the FAM. Both of these algorithms have highly parallel structures and are readily implemented on general purpose computers or, if execution time is critical, specialized multiprocessor signal analyzers.

Randy S. Roberts (M'89) was born in Oildale, CA, on October 26, 1957. He received the B.S. degree in electrical engineering in 1980, the M.S. degree in electrical engineering in 1982, and the Ph.D. degree in electrical engineering in 1989, all from the University of California at Davis.

From 1982 to 1984 he was a member of the Technical Staff at Signal Science Inc., Santa Clara, CA, working in the area of digital signal processing. He is currently a staff member at the Los Alamos National Laboratory, Los Alamos, NM. His research interests include statistical signal processing, parallel algorithms and architectures, and image processing.



William A. Brown (S'73-M'78) was born in Richland, WA, on February 13, 1952. He received the B.S. degree from California State University, Chico, in 1974, the M.S. degree from Illinois Institute of Technology, Chicago, in 1975, and the Ph.D. degree from University of California, Davis, in 1987, all in electrical engineering.

From 1977 to 1981 he was a member of the technical staff at ARGOSystems, Sunnyvale, CA, working on passive sonar signal processing. From 1981 to 1987 he was engaged in doctoral research on the theory of cyclostationary signals. Since 1987, he has been a staff scientist for Mission Research Corporation, Monterey, CA, working on communication signal interception as well as mitigation of ionospheric propagation effects on radar tracking and imaging systems.

Herschel H. Loomis, Jr. (S'59-M'63-SM'75) was born in Wilmington, Delaware on May 31, 1934. He graduated from Wilmington Friends School in 1952. He received the B.E.E. Degree from Cornell University in 1957, the M.S. in Electrical Engineering from the University of Maryland in 1959 and the Ph.D. degree from Massachusetts Institute of Technology in 1963.

From 1957 until 1959 he served as an electronic engineer with the United States Navy. He was Assistant Professor, Associate Professor and Professor of Electrical and Computer Engineering at the University of California, Davis, California from 1963 until 1981. He served as Department Chairman at UC Davis from 1970 until 1975. From 1981 to 1983, he was Naval Electronics Systems Command Chair Professor at the Naval Postgraduate School in Monterey. He is currently Professor of Electrical and Computer Engineering at the Naval Postgraduate School.

He has served as a consultant to Lawrence Livermore Laboratory, Fairchild Semiconductor, and Signal Science, Inc.

Dr. Loomis is a Senior Member of the IEEE, and is a member of the Association for Computing Machinery, Eta Kappa Nu, Phi Kappa Phi, Sigma Xi and Tau Beta Pi. He is graduate scholarship chairman for the Special Interest Group on Design Automation of the ACM.

REFERENCES

[1] W. A. Gardner, "Exploitation of spectral redundancy in man-made signals," *IEEE SP Magazine*, this issue.

[2] W. A. Gardner, *Statistical Spectral Analysis: A Nonprobabilistic Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[3] W. A. Gardner, "Measurement of spectral correlation," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. ASSP-34, No. 5, pp. 1111-1123, October 1986.

[4] W. A. Brown and H. H. Loomis, Jr. "Digital implementations of spectral correlation analyzers," *Proceedings of the Fourth Annual ASSP Workshop on Spectrum Estimation and Modeling*, Minneapolis MN. August 1988. pp. 264-270.

[5] W. A. Brown,"On the Theory of Cyclostationary Signals," Ph.D Dissertation, Department of Electrical Engineering and Computer Science, University of California, Davis, 1987. (Thesis Advisors: W. A. Gardner and H. H. Loomis, Jr.)

[6] R. Crochiere and L. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[7] B. G. Agee and W. A. Gardner, "Cyclic spectrum analysis study: executive summary and final briefing," ARGOSystems Technical Report No. B83-0014, June 1984.

[8] R. S. Roberts, "Architectures for Digital Cyclic Spectral Analysis," Ph.D Dissertation, Dept. of Electrical Engineering and Computer Science, University of California, Davis, 1989. (Thesis Advisors: W. A. Gardner and H. H. Loomis, Jr.)