# Digital Implementations of Spectral Correlation Analyzers

William A. Brown, III, *Member, IEEE*, and Herschel H. Loomis, Jr., *Senior Member, IEEE*

*Abstract*—**The spectral correlation function plays a central role in the analysis and design of signal processing systems operating on cyclostationary signals. In applications such as detecting the presence of a cyclostationary signal buried in noise and interference, estimation of the spectral correlation function is an integral part of the required processing. This paper addresses the issues involved in the design of computationally efficient algorithms for spectral correlation estimation and the resulting impact on high-speed digital realization. First, to develop the theoretical framework for understanding the realizations studied, the spectral correlation analyzer is characterized as a periodically time-varying quadratic system with a kernel possessing certain gross properties. The mean and variance of the output are expressed in terms of the kernel and the spectral correlation function of the input. Then, three realizations, each with different advantages, are analyzed in detail. One approach is based on the frequency-smoothing method of cross spectral analysis. The remaining approaches are variants of the time-smoothing method in which the time-smoothing integral is transformed into a Fourier transform operation. For each of these realizations, an exact expression for the quadratic system kernel is given, the digital implementation is developed, and a detailed complexity analysis is presented. High speed pipeline realizations of the algorithms are analyzed, and related special issues are discussed. Finally, examples involving the calculation of the spectral correlation function in near-real time for broad-band communications signals are presented and discussed.**

## I. Introduction

CYCLOSTATIONARY waveforms are persistent random waveforms with statistical parameters that vary periodically with time. Many naturally occurring and man-made systems with underlying periodically time-variant structures give rise to observed functions of time that are usefully modeled as cyclostationary waveforms. For example, most digital modulation schemes used in radio communication systems produce signals which, in connection with detection, estimation, synchronization, and emitter localization problems, are appropriately modeled as cyclostationary. The *spectral correlation function* (SCF), which is the cross spectrum of a signal and a frequency shifted version of itself, provides a second-order statistical description in the frequency domain of such signals. In proposed solutions to certain problems, such as detecting the presence of a cyclostationary signal buried in noise and interference, estimation of the SCF of the received waveform is an integral part of the required processing. This paper discusses some of the issues involved in the design of digital implementations of spectral correlation analyzers and presents some promising candidate algorithms.

The theory and utility of cyclostationary signal models and the SCF, also called the cyclic spectrum, are discussed in references [1]–[4], [7], and [8]. The basic time- and frequency-smoothing methods of spectral correlation analysis were introduced in [1] and proof of their equivalence was given in [2] and [3]. Methods which more fully exploit the computational efficiency of the FFT, namely, the *FFT accumulation method* (FAM) and the *strip spectral correlation analyzer* (SSCA), were introduced in [4] and discussed in [5] and [6]. This paper provides a comprehensive treatment of these computationally efficient algorithms including a detailed computational complexity analysis and an illustrative example. The kernel method introduced in [3] and developed in [3], [4], and in this paper, allows unified analysis of all these nonparametric methods. Recent applications in which computationally efficient estimation of the SCF is important include signal interception [9], [10], time-difference-of-arrival measurement [11], [12], adaptive spatial filtering [13], and high-resolution direction finding [17].

A general quadratic system representation for spectral correlation analyzers based on conventional cross spectral analysis techniques is studied in Section II. The analyzer tuning and resolution parameters and cycle frequency leakage performance are shown to be simply related to the system kernel transform. An expression for the output signal-to-noise ratio for Gaussian input waveforms is given in terms of the kernel transform and the input waveform SCF. The kernel transforms associated with alternative configurations conducive to efficient digital implementation are studied in Section III. A detailed hardware complexity analysis for the frequency-smoothing method and two fast time-smoothing methods is given in Section IV. A specific example involving detecting the presence of a cyclostationary signal buried in noise is discussed in Section V and some concluding remarks are given in Section VI.

The SCF for a discrete-time real-valued signal $x(n)$ is defined as the Fourier-series transform (discrete-time

Manuscript received February 14, 1991; revised January 12, 1992.

W. A. Brown, III, is with Mission Research Corporation, Monterey, CA 93940.

H. H. Loomis, Jr., is with the Naval Postgraduate School, Monterey, CA 93943.

IEEE Log Number 9205108.

Fourier transform) of the cyclic correlation function $R_x^\alpha(k)$,

$$S_x^\alpha(f) \triangleq \sum_{k=-\infty}^{\infty} R_x^\alpha(k) e^{-i2\pi fk}$$

where

$$R_x^\alpha(k) \triangleq \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} [x(n+k)e^{-i\pi\alpha(n+k)}] \cdot [x(n)e^{i\pi\alpha n}]^*.$$

Thus $S_x^\alpha(f)$ is the cross spectrum of the pair of complex-valued frequency-shifted signals $x(n)e^{-i\pi\alpha n}$ and $x(n)e^{i\pi\alpha n}$, where $f$ is the cross spectrum frequency variable and the parameter $\alpha$, called the cycle frequency, is the relative frequency shift. If the squared signal has finite average power then there is at most a countable number of values of $\alpha$ for which $S_x^\alpha(f) \not\equiv 0$ [3, p. 391]. For $\alpha = 0$ the SCF reduces to the ordinary power spectrum

$$S_x(f) = S_x^0(f) = \Sigma_k R_x^0(k)\, e^{-i2\pi fk}.$$

The symmetry relationships $S_x^\alpha(-f) = S_x^\alpha(f)$ and $S_x^{-\alpha}(f) = S_x^\alpha(f)^*$ and the periodicity associated with discrete time, $S_x^{\alpha+n}(f+m+n/2) = S_x^\alpha(f)$, which are easily verified from the defining expressions, imply that the entire function is determined by $S_x^\alpha(f)$ for $\{0 \le f \le 1/2, 0 \le \alpha \le 1-2f\}$.

Since the SCF can be interpreted as a cross spectrum, the principles of spectral analysis apply, including the concepts of time and frequency resolution, the time- and frequency-smoothing equivalence, and spectral leakage. For a fixed cycle frequency $\alpha$, conventional cross spectral analysis techniques such as the time-smoothing and frequency-smoothing methods provide procedures for estimating $S_x^\alpha(f)$ from a given finite segment of the received signal $x(n)$ [2], [3]. However, in many applications the values of cycle frequency for which the spectral correlation is nonzero are not known *a priori*. This necessitates estimating the function for a range of cycle frequencies encompassing the entire region of possible support. As demonstrated in Section II-B, the inherent cycle frequency resolution provided by conventional cross spectral analysis methods is $\Delta\alpha = 1/\Delta t$, where $\Delta t$ is the length of the observation interval. Therefore, a complete characterization of spectral correlation for a finite observation interval requires estimation of the SCF for a discrete set of spectrum frequencies $f_k = k\Delta f$, where $\Delta f$ is the frequency resolution of the analyzer, and cycle frequencies $\alpha_k = k\Delta\alpha$ over the region $\{0 \le f_k \le 1/2, 0 \le \alpha_k \le 1 - 2f_k\}$ of the bifrequency plane.

Statistically reliable estimation requires that $\Delta t\Delta f \gg 1$ [3]. Combining this requirement with the cycle frequency resolution relationship $\Delta\alpha = 1/\Delta t$ yields $\Delta\alpha \ll \Delta f$. Thus the required cycle frequency spacing is typically much finer than the spectrum frequency spacing. The number of individual estimates $\hat{S}_x^{\alpha_k}(f_k)$ required for complete analysis is thus $\Delta t/4\Delta f$ which, for fixed $\Delta f$, increases linearly with observation time. (In contrast, the number of estimates required for conventional spectral analysis is simply $1/2\Delta f$.) Since the computational cost of each estimate also increases roughly linearly with $\Delta t$, the computational cost of complete analysis using conventional methods is proportional to $\Delta t^2$. Two of the methods described in this paper are based on the application of the FFT and consequently reduce this dependence to roughly a $\Delta t \log \Delta t$ dependence.

The hardware realization section of the paper introduces a formalism to quantitatively describe the notion of "near real-time;" that is, the *factor of real time*. The application of this factor permits a hardware realization to be characterized by the factor by which it differs from real-time performance. The hardware section also presents in detail the way in which pipeline processors can be used to obtain high throughput realizations of the FFT, and these techniques are used to obtain the two highly efficient time-smoothing based methods, the FAM and the SSCA. Another paper, [6], presents a survey of the three methods discussed here, but it lacks the rigorous theoretical foundation of this paper and formal treatment of performance.

For simplicity, the treatment in this paper is restricted to spectral correlation analysis of a single real-valued signal. The extension to multiple and complex-valued signals can be found in [3] and [4]. Although the extension to complex-valued signals is straightforward, it is important to note that complete analysis of a complex-valued signal requires estimation of both $S_{xx}^\alpha(f)$ and $S_{xx^*}^\alpha(f)$, where $S_{ab}^\alpha(f)$ is the Fourier-series transform of $R_{ab}^\alpha(k)$, and

$$R_{ab}^\alpha(k) \triangleq \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} a(n+k)b^*(n)\, e^{-i2\pi\alpha n}\, e^{-i\pi\alpha k}.$$

Also, the degree of symmetry in the spectral correlation functions is reduced from that which exists for real-valued signals; in general for complex-valued signals, $S_{xx}^\alpha(-f) \ne S_{xx}^\alpha(f)$ and $S_{xx^*}^{-\alpha}(f) \ne S_{xx^*}^\alpha(f)^*$. The relationship between the SCF of a real-valued signal $x_r(t)$ and the SCF's of its complex envelope $x(t)$ relative to $f_c$ is [3]

$$S_{x_r}^\alpha(f) = \tfrac{1}{4}[S_{xx}^\alpha(f-f_c) + S_{xx}^\alpha(-f-f_c) + S_{xx^*}^{\alpha-2f_c}(f) + S_{xx^*}^{-\alpha-2f_c}(f)^*].$$

## II. The Time-Smoothing and Frequency-Smoothing Methods

### A. The Time-Smoothing Method

For fixed spectrum frequency $f_0$ and cycle frequency $\alpha_0$, the time-smoothing method of cross spectral analysis leads to the spectral correlation analyzer structure shown in Fig. 1 [2], [3]. The spectral components of interest are extracted from the signal via two bandpass filters, each having bandwidth $\Delta f$, centered at frequencies $f_0 + \alpha_0/2$ and $f_0 - \alpha_0/2$. Each filter output is then frequency shifted to baseband to produce the demodulates $x_+(n)$ and $x_-(n)$.

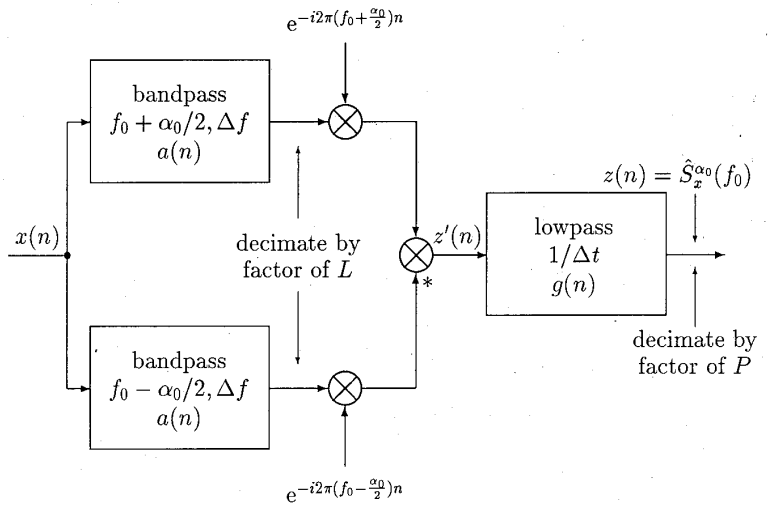



Fig. 1. The time-smoothing spectral-correlation analyzer.

The cross correlation between the demodulates is then measured by averaging the product $x_+(n)x_-^*(n)$ over a sliding time interval of approximate width $\Delta t$ samples. The output function of time $z(n)$ is the spectral correlation estimate based on an input segment centered at time index $n$ having a length of approximately $\Delta t$ samples.

In an efficient digital implementation, the bandpass filter outputs are decimated to retain one out of $L$ samples, where $L$ is chosen to yield tolerable aliasing distortion in the product signal $z'(Ln)$ within the output filter passband $|f| \leq 1/\Delta t$. Since the bandwidth of $z(n)$ is roughly $1/\Delta t$, the output can be further decimated by a factor of $1/P$ to yield $z(LPm)$, $m = 0, \pm 1, \pm 2, \cdots$, where $LP \approx \Delta t$.

The system is first studied for the case of no decimation, i.e., $L = P = 1$. The demodulates are given by

$$x_\pm(n) = \sum_k a(k)x(n-k)e^{-i2\pi(f_0 \pm \alpha_0/2)(n-k)} \quad (1)$$

where $a(k)$ is a finite-duration data-tapering window of approximate width $1/\Delta f$. The output can be expressed as

$$z(n) = \sum_m x_+(m)x_-^*(m)g(n-m) \quad (2)$$

where $g(n)$ is a finite-duration low-pass impulse response of approximate width $\Delta t$. Substituting (1) into (2) yields the following system representation:

$$z(n) = \sum_q \sum_r m(q, r)x(n-q)x(n-r)^{-i2\pi\alpha_0 n} \quad (3)$$

where

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

is referred to as the system kernel. With finite duration pulses $a(n)$ and $g(n)$ scaled such that $\Sigma_n a^2(n) = \Sigma_n g(n) = 1$ and under appropriate restrictions on $S_x^\alpha(f)$, it can be verified that [3]

$$S_x^{\alpha_0}(f_0) = \lim_{\Delta f \to \infty} \lim_{\Delta t \to \infty} z(n).$$

### B. The General Form of the System Kernel

The analyzer output can be decomposed into a sum of sine wave components directly related to the input signal SCF plus an erratic component considered noise. The output sine waves (or alternatively $E\{z(n)\}$) can be expressed as

$$\tilde{z}(n) \triangleq \sum_{-1/2 < \beta \leq 1/2} \lim_{N \to \infty} \frac{1}{2N+1} \sum_{m=-N}^{N} z(m)e^{-i2\pi\beta m}e^{i2\pi\beta n}$$

$$= \sum_{-1/2 < \beta - \alpha_0 \leq 1/2} \int_{-1/2}^{1/2} \tilde{M}(\beta, v)\, S_x^\beta(v) \cdot dv\, e^{i2\pi(\beta-\alpha_0)n} \quad (4)$$

where $\tilde{M}$ is the rotated-coordinate Fourier-series transform of $m$ defined by

$$\tilde{M}(\beta, v) \triangleq \sum_q \sum_r m(q, r)e^{-i2\pi(v+\beta/2)q}e^{i2\pi(v-\beta/2)r} \quad (5)$$

$$= G(\beta - \alpha_0)\, A\left(v - f_0 + \frac{\beta - \alpha_0}{2}\right) \cdot A^*\left(v - f_0 - \frac{\beta - \alpha_0}{2}\right) \quad (6)$$

and $G(\beta)$ and $A(v)$ are the Fourier-series transforms of $g(n)$ and $a(n)$, respectively. In (4) and in the sequel, the notation $\Sigma_{\beta \in S} f(\beta)$ refers to the countable sum over all values of $\beta \in S$ for which $f(\beta) \neq 0$.

The frequency and cycle frequency resolution properties of the system can be gleaned from the system kernel transform $\tilde{M}$. Note that $\tilde{M}$ is defined over the entire $\{v, \beta\}$ plane and it possesses the periodicity property $\tilde{M}(\beta + j, v + k + j/2) = \tilde{M}(\beta, v)$ for integers $j$ and $k$. With attention restricted to the diamond shaped region $\{|v| < 1/2, |\beta| \leq 1 - 2|v|\}$, the function $\tilde{M}(\beta, v)$ can be characterized as a two-dimensional pulse centered at $\{\beta = \alpha_0, v = f_0\}$ with most of its volume concentrated over the region $|\beta - \alpha_0| \leq 1/2\Delta t$, $|v - f_0| \leq \Delta f/2$. As quanti-

tatively described by (4), features of $S_x^\beta(v)$ within this region contribute to the output whereas features outside this region are suppressed:

$$\bar{z}(n) \approx \sum_{|\beta-\alpha_0|<1/2\Delta t} \int_{f_0-\Delta f/2}^{f_0+\Delta f/2} \tilde{M}(\beta, v)\, S_x^\beta(v)dv\, e^{i2\pi(\beta-\alpha_0)n}.$$

Thus, the width of $\tilde{M}(\beta, v)$ in its first argument determines the cycle frequency resolution of the analyzer, i.e., $\Delta\alpha = 1/\Delta t$, and the width of $\tilde{M}(\beta, v)$ in its second argument determines the frequency resolution $\Delta f$.

Although the preceding is based on the time-smoothing structure of Fig. 1, any time-invariant quadratic system with its output multiplied by $e^{-i2\pi\alpha_0 n}$ has the general system representation of (3) [3]. For an arbitrary quadratic system, the kernel may not have the specific functional form implied by (6). However, if the kernel transform for an alternative system has the properties described above then the system will function as a spectral correlation analyzer. In particular, systems based on the frequency-smoothing method and the variants of the time-smoothing method discussed in Section III all have kernels with the desired properties.

In (6), the tuning parameters $f_0$ and $\alpha_0$ do not affect the shape of the pulse but simply shift its location in the plane. A convenient representation of the kernel is therefore the shifted kernel transform

$$P(\beta, v) \triangleq \tilde{M}(\beta + \alpha_0, v + f_0) \tag{7}$$

which has the same properties as $\tilde{M}(\beta, v)$ but is centered at $\beta = v = 0$. Results concerning output signal-to-noise ratio in the sequel are expressed in terms of $P$ rather than $\tilde{M}$ for conciseness. For the time-smoothing method, $P(\beta, v) = G(\beta)A(v + \beta/2)A^*(v - \beta/2)$ is independent of the tuning parameters. For the system discussed in Section III-B, however, the shape of $P$ does depend on the tuning parameters.

Although detailed design of the kernel may require an exact expression for $P(\cdot, \cdot)$, interpretation of various formulas (such as (4) and (11)) and most performance analysis can be done by approximating $P$ as a periodic extension of a two-dimensional rectangular pulse. That is, with

$$P_0(\beta, v) \triangleq \frac{1}{\Delta f} u_{1/\Delta t}(\beta) u_{\Delta f}(v)$$

where $u_w(v)$ is the unit rectangle

$$u_w(v) \triangleq \begin{cases} 1 & -W/2 < v \le W/2 \\ 0 & \text{otherwise} \end{cases}$$

$P$ has the approximate form

$$P(\beta, v) \approx \sum_j \sum_k P_0(\beta + j, v + k + j/2). \tag{8}$$

### C. Time Smoothing with Decimation

If the bandpass filter outputs are decimated by a factor of $L$ and the output formed according to

$$z(mL) = \sum_j x_+(jL)x_-^*(jL)g_2(m - j)$$

where $g_2(n)$ has duration $P = \Delta t/L$ samples, (3) and (4) still apply with $n = mL$ and with the system kernel transform given by (6) with $G(\beta)$ replaced by $G'(\beta) = G_2(L\beta)$. Note that $G'(\beta)$ is periodic with period $1/L$.

If $L$ is too large, the replicas in $G'(\beta)$ produce undesired pulses in $\tilde{M}(\beta, v)$ centered at $\{v = f_0, \beta = \alpha_0 \pm k/L, k = 1, 2 \cdots\}$. According to (4), such pulses allow features of the SCF at cycle frequencies outside the cycle frequency resolution cell to affect the output. This occurrence is called *cycle leakage*.

The situation is depicted graphically in Fig. 2 which shows the approximate support diagrams for $G'(\beta - \alpha_0)$, $A(v - (f_0 + \alpha_0/2) + \beta/2)$, and $A^*(v - (f_0 - \alpha_0/2) - \beta/2)$. From the diagram, the replicas in $G'(\beta - \alpha_0)$ will not intersect the *channel pair region* defined as the support of $A(v - (f_0 + \alpha_0/2) + \beta/2)A^*(v - (f_0 - \alpha_0/2) - \beta/2)$ provided that $1/L > \Delta f + 1/2\Delta t$. This provides only a rough guideline for selection of $L$ since it does not take into account the skirts and sidelobes of $A(v)$ and $G'(\beta)$.

The sine-wave components of the output associated with cycle leakage experience aliasing if their frequencies exceed the decimated sampling rate. This effect is called *cycle aliasing*.

### D. The Frequency-Smoothing Method

This approach to cross spectral analysis also leads to a system representation of the form of (3) [3]. In this case input signal segments of length $N = \Delta t$ samples are Fourier transformed and down-converted to obtain the set of demodulates.

$$x_k(n) = \sum_p b(p)x(n - p)\, e^{-i2\pi k(n-p)/N}$$

where $b(p)$ is a tapering window of width $N$. Averaging over frequency is accomplished by

$$z(n) = \sum_{m=-M_1}^{M_2} x_{j+m}(n)x_{k+m}^*(n)W(m)$$

where the integers $j$ and $k$ determine the tuning according to

$$\alpha_0 = (j - k)/N$$

$$f_0 = (j + k + M_2 - M_1)/2N$$

and $W(m)$ is the frequency-smoothing function centered at $m = 0$ of width

$$M = M_1 + M_2 + 1 \approx \Delta t\Delta f$$

samples. Note that $f_0$ is always a multiple of $1/2N$ but not necessarily a multiple of $1/N$. The kernel transform takes the form

$$\tilde{M}(\beta, v) = \sum_m B\left(v - f_0' + \frac{\beta - \alpha_0}{2} - \frac{m}{N}\right) \cdot B^*\left(v - f_0' - \frac{\beta - \alpha_0}{2} - \frac{m}{N}\right) W(m)$$

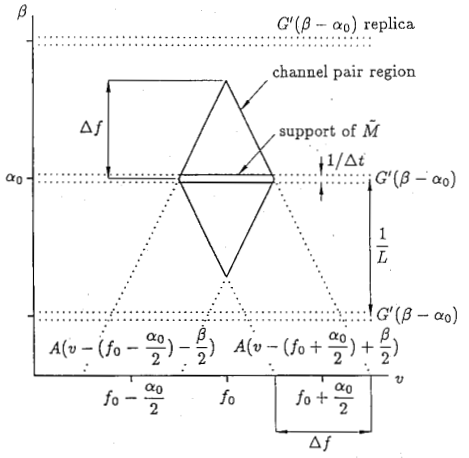



Fig. 2. Region-of-support diagram for the factors of $\tilde{M}(\beta, v)$.

where

$$f_0' = f_0 - \frac{M_2 - M_1}{2N}.$$

Since $B(v + \beta/2)\, B^*(v - \beta/2)$ has the bulk of its volume concentrated in $\{|v| < 1/2\Delta t, |\beta| < 1/2\Delta t\}$, the sum above synthesizes a pulse over the region $\{|v - f_0| < M/2\Delta t = \Delta f/2, |\beta - \alpha_0| < 1/2\Delta t\}$. For $\Delta t \Delta f >> 1$ this function has characteristics essentially identical to the kernel transform for the time-smoothing method. This verifies the equivalence (which was first established in [3]) between time smoothing and frequency smoothing for spectral correlation analyzers.

### E. Output Signal-to-Noise Ratio

The reliability of the spectral correlation estimate is limited by the noise component $r(n)$ present in the output:

$$r(n) \triangleq z(n) - \tilde{z}(n).$$

Since the last stage in the analyzer of Fig. 1 is a low-pass filter with bandwidth $1/\Delta t$, the noise is necessarily low pass with bandwidth no greater than $1/\Delta t$. A measure of performance sufficient for most purposes is the output signal-to-noise ratio defined by

$$\rho_{\text{out}} \triangleq \frac{P_{\tilde{z}}}{P_r}$$

where $P_{\tilde{z}}$ is the average power of the desired component of the output

$$\begin{aligned} P_{\tilde{z}} &= \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |\tilde{z}(n)|^2 \\ &= \sum_{-1/2 < \beta - \alpha_0 \le 1/2} \left| \int_{-1/2}^{1/2} \tilde{M}(\beta, v)\, S_x^{\beta}(v) dv \right|^2 \\ &\approx \left| \int_{-1/2}^{1/2} P(0, f)\, S_x^{\alpha_0}(f + f_0) df \right|^2 \end{aligned} \tag{9}$$

and $P_r$ is the average power of the noise component

$$P_r = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |z(n) - \tilde{z}(n)|^2. \tag{10}$$

In general, the noise power depends on the system kernel and on the fourth-order statistical moments of the input waveform. In order to simplify the analysis it is assumed that the input waveform is a zero-mean Gaussian cyclostationary time series, which is, with probability one, equivalent to a sample function of an ergodic zero-mean Gaussian cyclostationary random process. This allows the noise power to be simply expressed in terms of the second order statistics, namely the spectral correlation function, of the input waveform. Furthermore, although many cyclostationary communication signals are not Gaussian, the performance predictions based on the Gaussian assumption are reasonably accurate, especially if the input waveform is dominated by a strong Gaussian noise component. The analysis can be extended to the nonzero-mean case as shown in [4].

Substitution of the expressions for $z$ and $\tilde{z}$ into (10) and use of the fourth-order moment formula for Gaussian waveforms yield [3]

$$P_r = P_1 + P_2 \tag{11a}$$

$$\begin{aligned} P_1 = \sum_{-1/2 < \beta \le 1/2} \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} & P\left(u - v, \frac{u+v}{2} + \frac{\beta}{2}\right) \\ & \cdot P^*\left(u - v, \frac{u+v}{2} - \frac{\beta}{2}\right) \\ & \cdot S_x^{\beta}\left(u + f_0 + \frac{\alpha_0}{2}\right) S_x^{\beta}\left(v + f_0 - \frac{\alpha_0}{2}\right)^* du dv \end{aligned} \tag{11b}$$

$$\begin{aligned} P_2 = \sum_{-1/2 < \beta \le 1/2} \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} & P\left(u - v, \frac{u+v}{2} + \frac{\beta}{2}\right) \\ & \cdot P^*\left(u - v, -\frac{u+v}{2} + \frac{\beta}{2}\right) \\ & \cdot S_x^{\beta+2f_0}\left(u + \frac{\alpha_0}{2}\right) S_x^{\beta+2f_0}\left(v - \frac{\alpha_0}{2}\right)^* du dv \end{aligned} \tag{11c}$$

where $P(\cdot, \cdot)$ is defined by (7). These equations are exact under the zero-mean Gaussian assumption and are useful for detailed design of the kernel. They also provide a starting point for approximations leading to simpler expressions more suitable for system performance analysis. Since the equations are expressed in terms of the general kernel representation $P(\beta, v) \triangleq \tilde{M}(\beta + \alpha_0, v + f_0)$, they are applicable to both the time- and frequency-smoothing methods and to the computationally efficient methods described in Section III.

Considerable simplification occurs if at most one term dominates the sum over $\beta$ in (11b) and (11c) and if the observation time is large compared with the reciprocal resolution width of the input SCF. Suppose that $S_x^{\beta}(f) \equiv$

0 for $0 < \beta \leq \Delta f$ and further suppose that $\Delta t$ is sufficiently large that $S_x(f + 1/\Delta t) \approx S_x(f)$. Then the expression for $P_1$ simplifies to

$$P_1 \approx \int_{-1/2}^{1/2} \Psi(f)\, S_x\left(f + f_0 + \frac{\alpha_0}{2}\right) S_x\left(f + f_0 - \frac{\alpha_0}{2}\right) df \tag{12}$$

where

$$\Psi(f) = u_1(f) \int_{-1+2|f|}^{1-2|f|} |P(v, f)|^2 dv \approx \frac{1}{\Delta t \Delta f^2}\, u_{\Delta f}(f).$$

A similar simplification of the expression for $P_2$ is obtained as follows. Suppose there is at most one value of $\beta$ within the range $|\beta| \leq \Delta f$ for which $S_x^{\beta + 2f_0}(f) \not\equiv 0$. Let $\beta_0$ denote this value and assume that $S_x^{\beta_0 + 2f_0}(f + 1/\Delta t) \approx S_x^{\beta_0 + 2f_0}(f)$. Then the expression for $P_2$ simplifies to

$$P_2 \approx \int_{-1/2}^{1/2} \Psi_2(f, \beta_0)\, S_x^{\beta_0 + 2f_0}\left(f + \frac{\alpha_0}{2}\right) S_x^{\beta_0 + 2f_0}\left(f - \frac{\alpha_0}{2}\right)^* df \tag{13}$$

where $|\beta_0| \leq \Delta f$ and

$$\begin{aligned}\Psi_2(f, \beta) &= u_1(f) \int_{-1+2|f|}^{1-2|f|} P(v, f + \beta/2) \\ &\quad \cdot P^*(v, -f + \beta/2) dv \\ &\approx \frac{1}{\Delta t \Delta f^2}\, u_{\Delta f - |\beta|}(f) u_{2\Delta f}(\beta).\end{aligned}$$

An important application of the spectral correlation analyzer is the detection of weak cyclostationary signals buried in broad-band stationary noise and interference. In this case the spectral correlation features for nonzero cycle frequency are small compared with the power spectral density. Therefore, provided $|f_0| > \Delta f/2$, $P_2$ is small compared with $P_1$. However, if $|f_0| \leq \Delta f/2$ then the expression (13) involves the power spectral density

$$P_2 \approx \int_{-1/2}^{1/2} \Psi_2(f, -2f_0)\, S_x\left(f + \frac{\alpha_0}{2}\right) S_x\left(f + \frac{\alpha_0}{2}\right) df$$

and, moreover, if $f_0 = 0$ and $P(v, f)$ is symmetric in $f$ then $P_2 = P_1$. The situation is summarized by the following rule of thumb: For low input signal-to-noise ratio, the average power of the output noise is

$$P_r \approx \begin{cases} 2P_1 & \text{for } |f_0| \leq \Delta f/2 \\ P_1 & \text{otherwise.} \end{cases}$$

In this case the output signal-to-noise ratio can be approximated by

$$\rho_{\text{out}} \approx \frac{\left| \int_{-1/2}^{1/2} P(0, f)\, S_x^{\alpha_0}(f + f_0) df \right|^2}{(1 + u_{\Delta f}(f_0)) \int_{-1/2}^{1/2} \Psi(f)\, S_x\left(f + f_0 + \frac{\alpha_0}{2}\right) S_x\left(f + f_0 - \frac{\alpha_0}{2}\right) df}. \tag{14}$$

If, in addition, the frequency resolution of the analyzer resolves the signal SCF in the sense that $S_x^{\alpha}(f + \Delta f) \approx S_x^{\alpha}(f)$, then $P_{\tilde{z}} \approx |S_x^{\alpha_0}(f_0)|^2$, $P_1 \approx S_x(f_0 + \alpha_0/2)\, S_x(f_0 - \alpha_0/2)/\Delta t \Delta f$, and (14) becomes

$$\rho_{\text{out}} \approx \frac{\Delta t \Delta f}{(1 + u_{\Delta f}(f_0))} \frac{|S_x^{\alpha_0}(f_0)|^2}{S_x\left(f_0 + \frac{\alpha_0}{2}\right) S_x\left(f_0 - \frac{\alpha_0}{2}\right)}$$

which agrees with the results in [2]. Thus, under these conditions the output signal-to-noise ratio is proportional to the time-frequency resolution product of the analyzer and to the square of the autocoherence magnitude. A specific example illustrating application of the output signal-to-noise ratio formula is given in Section V.

## III. Fourier Transform Based Time-Smoothing Methods

### A. Theoretical Basis

As is shown in Section IV, considerable computational savings are obtained in variants of the smoothing-in-time method in which the output averaging operation is transformed (by incorporating a variable-frequency shifting operation) into a Fourier transform operation performed using a fast Fourier transform algorithm. Since these systems also possess system representations of the form of (3), their tuning and resolution parameters, $\{f_0, \alpha_0, \Delta f, \Delta\alpha, \Delta t\}$, and their cycle frequency leakage and aliasing performance can be ascertained from the kernel transform $\tilde{M}$. Consider the structure shown in Fig. 3, which is the time-smoothing analyzer of Fig. 1 but with the output formed as

$$z(n) = \sum_m x_{f_1}(m)\, x_{f_2}^*(m)\, g(n - m)\, e^{-i2\pi\epsilon_1 m} \tag{15}$$

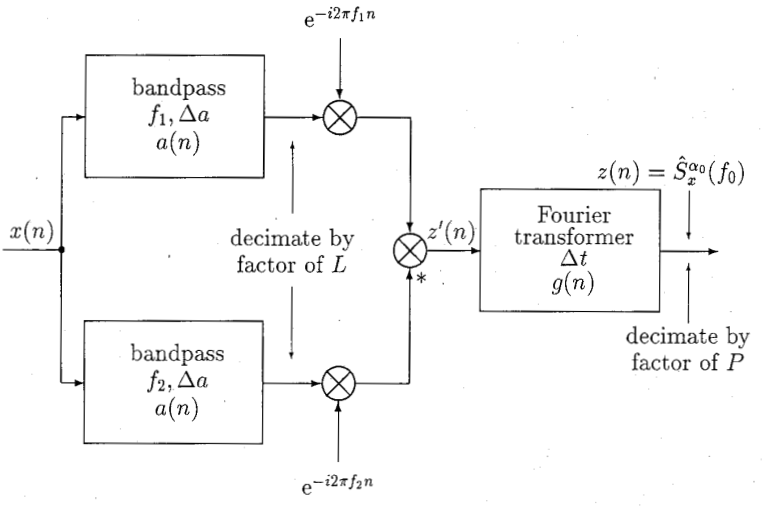



Fig. 3. Spectral-correlation analyzer employing an output Fourier transformer.

where

$$x_{f_j}(n) = \sum_k a(k)\, x(n-k)\, e^{-i2\pi f_j (n-k)} \tag{16}$$

or if the bandpass filter outputs are decimated by a factor of $L$ then the output is formed according to

$$z(nL) = \sum_m x_{f_1}(mL)\, x^*_{f_2}(mL)\, g_2(n-m)\, e^{-i2\pi\epsilon_2 Lm}. \tag{17}$$

The basic characteristics of the system are most easily determined from the full sampling rate system of (15). This case is studied first and then the results are extended to include the effects of $L > 1$. Substitution of (16) into (15) yields the system representation of (3) with $\alpha_0 = f_1 - f_2 + \epsilon_1$ and

$$m(q, r) = \sum_p g(p) a(q-p)\, a(r-p)\, e^{i2\pi\epsilon_1 p}\, e^{i2\pi f_1 q} \cdot e^{-i2\pi f_2 r}.$$

The kernel transform becomes

$$\tilde{M}(\beta, v) = G(\beta - \alpha_0)\, A(v - f_1 + \beta/2) \cdot A^*(v - f_2 - \beta/2). \tag{18}$$

The support diagram of Fig. 4 shows that this function is of the appropriate form, i.e., a pulse centered at $\{f_0 = (f_1 + f_2)/2, \alpha_0\}$ with width $\Delta\alpha = 1/\Delta t$ in its first argument and width $\Delta f = \Delta a - |\epsilon_1|$ (for $|\epsilon_1| < \Delta a$) in its second argument. The quantity $\Delta a$ is defined as the bandwidth of $A(f)$.

The primary distinction between this kernel and the ordinary time-smoothing kernel is that the frequency resolution varies here with $|\epsilon_1|$ and therefore with $\alpha_0$. If the output Fourier transform is performed as a discrete Fourier transform then $\epsilon_1$ takes on the discrete value $\epsilon_1 = \pm s/\Delta t$, $s = 0, 1, \cdots, \Delta a \Delta t$. A complete scan across the bins of the output transformer provides estimates of the SCF covering the entire channel pair region;

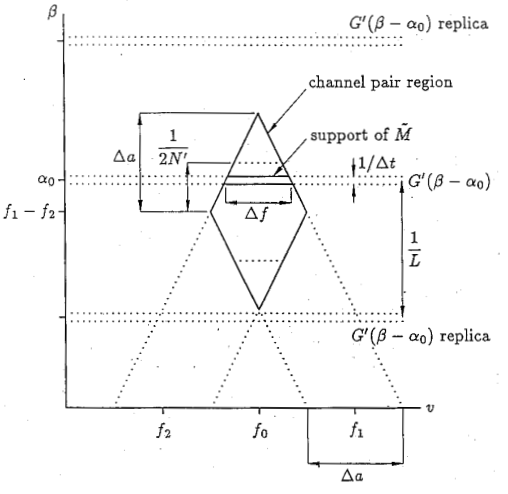



Fig. 4. Region-of-support diagram for the FFT accumulation method.

$\tilde{S}_x^{f_1 - f_2 + s/\Delta t}((f_1 + f_2)/2)$, where $|s| \le \Delta a \Delta t$. For complete coverage of the plane, a bank of bandpass filters provides a complete set of demodulates $x_{f_j}(n)$ with center frequencies $f_j = j/N'$ where $N'$ is the length of the tapering window $a(n)$. The channel pair regions generated by the set of center frequency pairs $\{f_j, f_k, |j| \le N'/2, |k| \le N'/2\}$ are sufficient to completely cover the region of the $\{\beta, v\}$ plane of interest. The set of channel pair regions can be visualized with the aid of Fig. 5.

The variation of $\Delta f$ with the relative location of $\alpha_0$ within the channel pair region is problematic. It is usually preferable for all estimates to be obtained with the same or nearly the same frequency resolution. The variation of $\Delta f$ can be reduced by limiting the output transformer frequency variable to

$$|\epsilon_1| \le 1/2N'$$

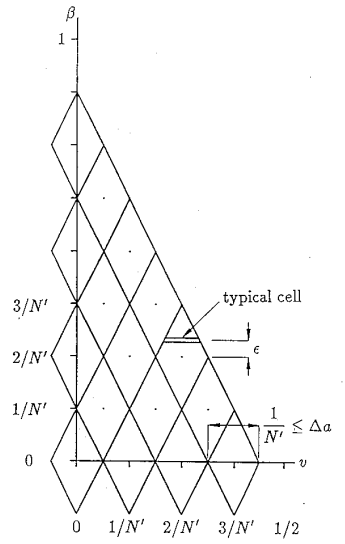



Fig. 5. Channel-pair-region-converge diagram.

which assures that

$$\Delta a - 1/2N' \leq \Delta f \leq \Delta a.$$

It can be shown that the resulting gaps in the coverage diagram are eliminated if the input filter passbands overlap sufficiently, in particular, if $\Delta a \geq 3/2N'$. For the critical value of overlap, $\Delta a = 3/2N'$, the frequency resolution range is $1/N' \leq \Delta f \leq \Delta a$.

### B. FFT Accumulation with Decimation

The bandwidth of the demodulate product waveform $x_{f_1}(n)x^*_{f_2}(n)$ is nominally $2\Delta a$ which is typically much less than unity. This suggests that significant computational savings can be achieved by decimating the bandpass filter outputs as indicated in (17) and thereby reducing the size of the output FFT to $P = \Delta t/L$ samples. We refer to the resulting algorithm as the FFT accumulation method. With the output formed as in (17), $z(mL)$ and $\tilde{z}(mL)$ are given by (3) and (4), respectively, with $n = mL$, $\alpha_0 = f_1 - f_2 + \epsilon_2$, and kernel transform $\tilde{M}(\beta, v)$ given by (18) with $G(\beta)$ replaced by $G'(\beta) = G_2(L\beta)$.

As indicated in Fig. 4, if the replicas in $G'(\beta - \alpha_0)$, centered at $\beta = \alpha_0 \pm k/L$, $k = 1, 2, \cdots$, do not intersect the channel pair region then the kernel transform is of the desired form. Otherwise $\tilde{M}(\beta, v)$ will have extraneous passbands associated with cycle frequency leakage and aliasing as determined by (4). A guideline for selection of the decimation factor is obtained from Fig. 4. Assume that the output transformer frequency is limited to $|\epsilon_2| \leq 1/2N'$ for the reasons given in the preceding section. Then the replicas of $G'(\beta - \alpha_0)$ do not intersect the channel pair region if $1/L > 1/2N' + \Delta a + 1/2\Delta t$. If coverage gaps are eliminated by designing the data-tapering window such that $\Delta a = 3/2N'$, then this becomes $1/L > 2/N' + 1/2\Delta t$ or approximately $L < N'/2$. The convenience of using radix-2 FFT's and allowance for a safety factor to account for the skirts and sidelobes of $A(v)$ and $G'(\beta)$ suggests the design rule

$$L = N'/4.$$

### C. The Strip Spectral Correlation Analyzer

Comparable computational savings are obtained at the expense of minor degradation in output signal-to-noise ratio as explained in [4] by eliminating one of the bandpass filters in Fig. 3 to form the output

$$z(n) = \sum_m x_{f_k}(m)x(m)g(n - m)e^{-i2\pi\epsilon_3 m}$$

as shown in Fig. 6. Equations (3) and (4) apply to this system with

$$\alpha_0 = f_k + \epsilon_3$$

$$f_0 = f_k - \alpha_0/2$$

and

$$\tilde{M}(\beta, v) = G(\beta - \alpha_0)A(v - f_0 + (\beta - \alpha_0)/2).$$

For $\Delta t\Delta f >> 1$, this kernel transform is of the desired form. For this system, $g(n)$ is scaled as before such that $G(0) = 1$ and $a(n)$ is scaled such that $a(0) = 1$ to preserve proper scaling in the limit, i.e.,

$$\lim_{\Delta f \to 0} \lim_{\Delta t \to \infty} z(n) = S_x^{\alpha_0}(f_k - \alpha_0/2).$$

The channel pair region for this system is the support region of $A(v - f_0 + (\beta - \alpha_0)/2)$ which is an infinite strip of width $\Delta a$ centered on the line $\beta = -2v + 2f_k$. Therefore, the frequency resolution of this analyzer does not vary with $\epsilon_3$ or $\alpha_0$ but is constant; $\Delta f = \Delta a$. A complete scan of the output transformer frequency variable, $\epsilon_3(s) = \pm s/\Delta t$, $s = 0, 1, \cdots \Delta t/2 - 1$, produces estimates covering the entire strip given by $\tilde{S}_x^{f_k + s/\Delta t}(f_k/2 - s/(2\Delta t))$. The set of input filter center frequencies $\{f_k = k/N', k = 0, 1, \cdots N'/2 - 1, N' \approx 1/\Delta f\}$ generates strips covering the $\{v, \beta\}$ plane as shown in Fig. 7.

## IV. Digital Realizations

### A. FFT Realizations

A fundamental building block of our various realizations is the $N$-point FFT. For our discussions, we will use the radix-2 FFT, although similar developments can be made for higher radix ones.

Our objective is to consider ways to implement these algorithms in near real time. As an aid to characterizing the closeness to real-time computation, we define the *factor of real time* $F_T$ as

$$F_T = \frac{\text{Computation time}}{\text{Collect time}} \tag{19}$$

where the computation time $= C_u \cdot T_c$.

We will find it useful to estimate the complexity or number of hardware units $p_{hu}$ needed to operate at a par-

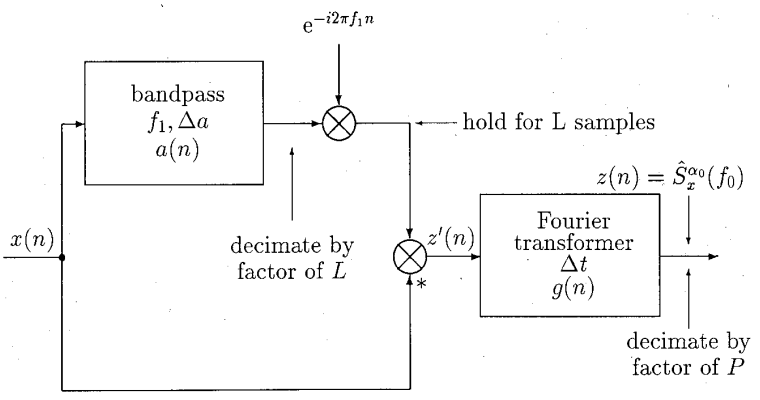



Fig. 6. The strip spectral-correlation analyzer.

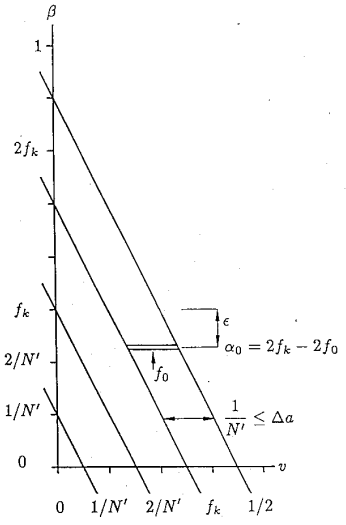



Fig. 7. Converge diagram for the strip spectral-correlation analyzer.

ticular factor of real time $F_T$. This is simply the number of operations performed by that unit $C_u$, divided by the total number of clock periods available to do the operations, $F_T \cdot \Delta t$, where $\Delta t$ is the total number of samples processed. Thus,

$$p_{hu} = \frac{C_u}{F_T \Delta t} = \frac{C_u}{F_T N}.$$

We will use the complexity product, $p_{hu} \cdot F_T$, as a measure of the hardware complexity of a particular architecture in our subsequent analysis of different schemes for computing the spectral correlation function.

If an algorithm requires $C$ operations to calculate the desired function, and we have $F_T \cdot N$ samples in which to do the calculation, the complexity product for rate-1 operators will be

$$p_1 \cdot F_T = \frac{C}{N}.$$

If we use the rate-$R$ operator, it will require $1/R$ as many hardware units, so

$$p_R \cdot F_T = \frac{C_b}{R \cdot N}. \tag{20}$$

The fundamental building block of the FFT, in turn, is the decimation-in-time complex butterfly (abbreviated DIT CBF), which is presented in a number of basic digital signal processing references such as Oppenheim and Schafer [14]. In general, the radix-$R$ butterfly can be constructed which accepts the $R$-input complex numbers in $R$ clock pulse periods and produces in a pipeline fashion the $R$ complex output numbers. This unit can be constructed from rate $1/R$ pipeline multiply and add modules and is shown in block diagram form in Fig. 8(a). The detailed structure of one possible realization for the radix-2 DIT CBF is illustrated in Fig. 8(b). The delays $d_{cm}$ and $d_{ca}$ account for the $d_{cm}$ stages in the pipeline complex-multiply units and $d_{ca}$ stages in the pipeline complex-adder units. Each of the complex arithmetic modules are in turn defined in terms of rate-1/2 pipeline real multipliers and real adders, such that $d_{cm} = d_{rm} + d_{ra}$ and $d_{ca} = d_{ra}$. This structure can be easily generalized for radix $R$. One can also create a pipeline radix-$R$ butterfly from rate-1 multipliers and adders; such a unit would have a considerably different architecture from that of Fig. 8(b).

Our example pipeline radix-2 DIT CBF operates at 1/2 rate, that is, the unit produces one CBF calculation per two sample periods. The adders and multipliers that make up the CBF also operate at 1/2 rate. Our input and output data memories will be each operating at one operand per clock period, furnishing the two operands needed for the CBF calculations in two clock periods. Use of rate-1 multiplier and adder modules is of course possible, but would require different architectures.

The complexity for the $N$-point, radix-2, FFT, in number of complex butterflies, is

$$C_b = (N/2) \log_2 N.$$

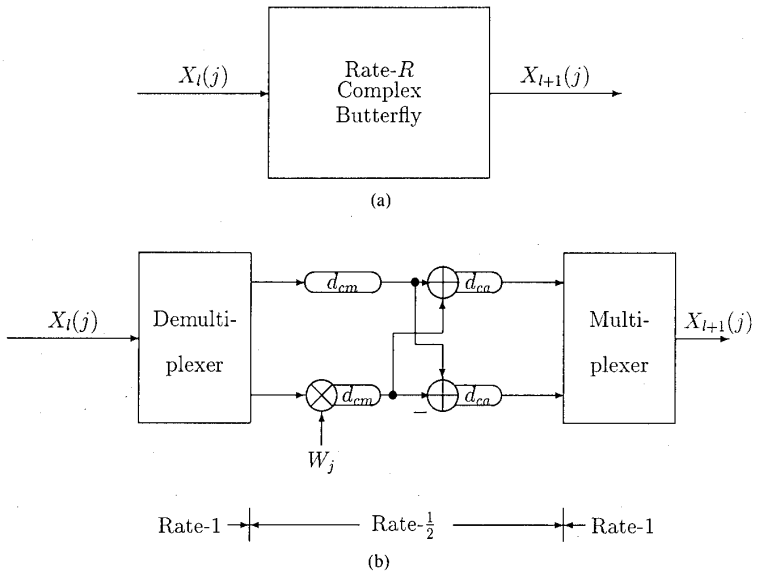



Fig. 8. (a) Rate-$1/R$ radix-$R$ pipelined complex butterfly. (b) Rate-$1/2$ radix-2 pipeline DIT complex butterfly.

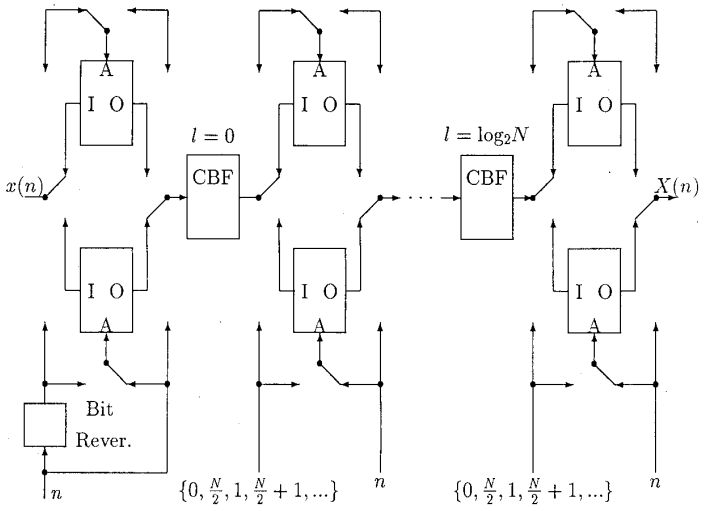



Fig. 9. Real-time pipeline DIT FFT.

Therefore, the complexity product for rate-$1/2$ butterflies is

$$P_{b2} \cdot F_T = \frac{2C_b}{N} = \log_2 N.$$

Two extremes are possible (as well as several intermediate situations). In the case where one pipeline rate-$1/2$ butterfly unit is used, a factor of real time of $\log_2 N$ is achieved. On the other hand, full real time operation is secured with $\log_2 N$ butterflies, one to each of the $\log_2 N$ levels. These levels can be operated in a pipeline sense, with the 0th hardware level computing the 0th level of the FFT for the current $N$ samples, the 1st hardware level computing the 1st level result for the previous $N$-sample block, etc. This full real-time decimation-in-time structure is shown schematically in Fig. 9.

While this arrangement will take approximately $N \log_2 N$ clock periods to compute the first $N$-point result, each successive result is computed every $N$ periods, and the throughput is therefore 1. The latency is however approximately $N \log_2 N$. The actual latency of each CBF stage is $2(d_{rm} + 2d_{ra}) + 4$, where the multiplication factor of 2 accounts for the fact that the CBF is operating at $1/2$ rate, and where the added delays of 4 account for the input demultiplexer and output multiplexer. Thus, the overall latency of the real-time pipeline FFT will be $(N + 2d_{rm} + 4d_{ra} + 4 + N) \cdot \log_2 N$.

Also, it should be noted that this FFT needs a pipeline

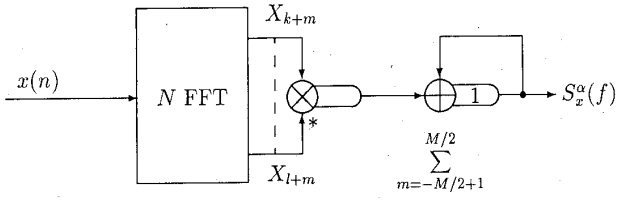



Fig. 10. Frequency-smoothing-method architecture.

bit-reverse address calculation for the input to the 0th level memory, and each stage needs two memories, one to act as the input for the following CBF and one to act as the output for the previous CBF.

This architecture for the computation of real-time FFT's can be implemented using commercially available ASIC chips introduced by Honeywell, Inc. [15].

Next, we consider three basic rationales for computing the spectral correlation function and examine high speed FFT-based architectures appropriate to each.

### B. Frequency Smoothing Method (FSM) Realizations

The digital realization based on the FSM naturally follows from Section II-D when a rectangular smoothing function is used. This realization is shown schematically in Fig. 10. The values of $f$ and $\alpha$ for this realization are $f = (k + l)/(2N)$ and $\alpha = (k - l)/N$.

Since $\Delta\alpha = 1/N$ and $\Delta f = M/N$, $N^2/4M$ values of $S_x^\alpha(f)$'s are calculated. Table I summarizes the complexity analysis for the major sections of the FSM realization.

The following equations summarize the complexity in number of real multiplies ($C_{rm}$), and the number of real adds ($C_{ra}$):

$$C_{rm} = N^2 + 2N \log_2 N \tag{21a}$$

$$C_{ra} = N^2 + 3N \log_2 N. \tag{21b}$$

In practice, we will pick a value for $\Delta f = M/N$ determined by the region of frequency containing the broadband signal of interest. Thus, $M/N$ is fixed for a particular signal.

The time-bandwidth product, $\Delta t\Delta f = M$, on the other hand, must be increased sufficiently to reduce variance effects in the output of the correlator. This suggests an alternate form for the complexity as a function of $\Delta t\Delta f$ and $\Delta f$. Expressing (21a) in this form yields:

$$C_{rm} = \frac{(\Delta t\Delta f)^2 \cdot}{(\Delta f)^2} + \frac{2(\Delta t\Delta f)}{\Delta f} \log_2 \left[\frac{\Delta t\Delta f}{\Delta f}\right]. \tag{22}$$

Figs. 11 and 12 show $C_{rm}$ as a function of $\Delta t\Delta f$ for two different values of $\Delta f$. Now let us consider the computation of the spectral correlation using the FSM method in real time. The complexity product for rate-1 real multipliers is

$$p_{rm1} \cdot F_T = \frac{\Delta t\Delta f}{\Delta f} + 2 \log_2 \left(\frac{\Delta t\Delta f}{\Delta f}\right). \tag{23}$$

TABLE I
FSM COMPLEXITY

| | (1 required) | $N^2/4M$ Required | |
|---|---|---|---|
| | FFT | Correlator | Summer |
| CBF | $(N/2) \log_2 N$ | — | — |
| Cpx mpy | $(N/2) \log_2 N$ | $M$ | — |
| Cpx add | $N \log_2 N$ | — | $M$ |
| Real mpy | $2N \log_2 N$ | $4M$ | — |
| Real add | $3N \log_2 N$ | $2M$ | $2M$ |

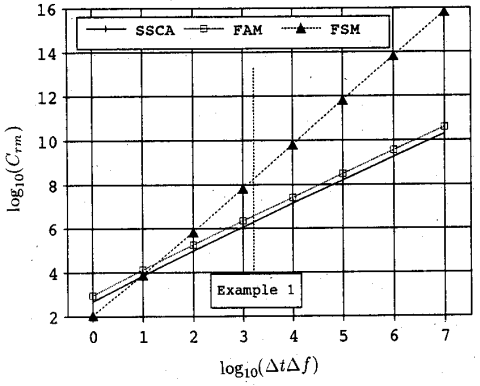



Fig. 11. Complexity of three realizations, $\Delta f = \frac{1}{8}$.

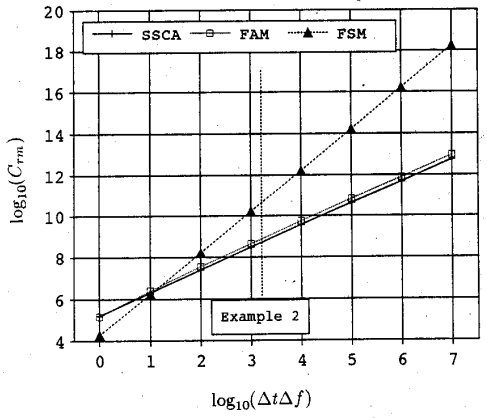



Fig. 12. Complexity of three realizations, $\Delta f = \frac{1}{128}$.

Figs. 13 and 14 show that complexity product as a function of $\Delta t\ \Delta f$ for two typical values of $\Delta f$ for the FSM architecture as well as two other architectures to be considered below.

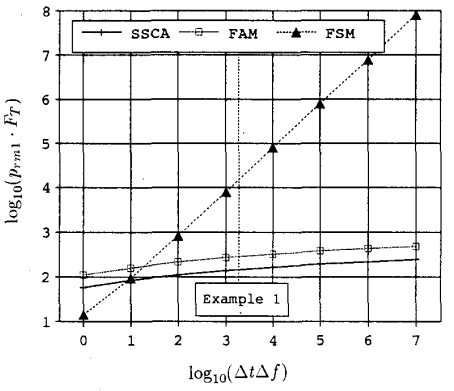



Fig. 13. Complexity product for three realizations, $\Delta f = \frac{1}{8}$.

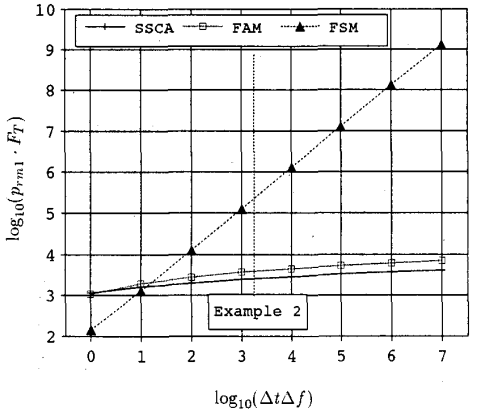



Fig. 14. Complexity product for three realizations, $\Delta f = \frac{1}{128}$.

Because we have elected to use rate-1/2 in the complex butterflies, further analysis is appropriate. First consider the hardware associated with the FFT's:

$$C_b = (\Delta t/2) \log_2 \Delta t \tag{24a}$$

and

$$p_{b2} \cdot F_T = \log_2 \Delta t. \tag{24b}$$

If we have $\log_2 \Delta t$ rate-1/2 complex-butterfly units operating as a real-time $\Delta t$-point FFT, we would take care of the calculations enumerated by the second term of (22). This would require 4 $\log_2 \Delta t$ rate-1/2 real multipliers.

The first term in (22) shows the following hardware requirement product:

$$p_{rm1} \cdot F_T = \frac{\Delta t \Delta f}{\Delta f} \text{ correlator rate } - 1 \text{ real multiplies.} \tag{24c}$$

One extreme in realization occurs when $F_T$ is 1, which corresponds to real-time operation. This occurs when we have a pipeline, real-time FFT architecture as discussed above with $\log_2 \Delta t$ complex, rate-1/2 butterfly units (and, of course the necessary memory) and $(\Delta t \Delta f/\Delta f)$ correlator rate-1 real multiply units.

There is another extreme, the case when there is one real multiply unit available to do all the work of (22). In this case

$$F_T = 2 \log_2 \left(\frac{\Delta t \Delta f}{\Delta f}\right) + \left(\frac{\Delta t \Delta f}{\Delta f}\right).$$

Only certain values of $F_T$ between these extremes have straightforward architectures.

Because of the computational complexity, the FSM is not the method of choice for computing $S_x^\alpha(f)$ for the entire quarter-bifrequency plane. Instead, the time-smoothing methods considered below are much more efficient. Nevertheless, the FSM is appropriate for computing the spectral correlation function for a small range of $\alpha$. The details of these tradeoffs as well as enhancements possible for the FSM architectures are considered in [16].

### C. FFT Accumulation Method (FAM) Realization

The structure that follows naturally from the time-smoothing method of obtaining the smoothed cyclic periodogram is shown in Fig. 1. In that figure, replace the down converter frequencies $f_1$ and $f_2$ with $f_1 + \epsilon_1$ and $f_2 - \epsilon_1$. As long as $\epsilon_1$ is not too large ($\epsilon_1 < \Delta f/2$), the frequencies down converted to zero will be within the passband of the filters.

The exponential factors involving $\epsilon_1$ can be factored through the correlation multiplier and combined with the $\Sigma_{\Delta t^*}$ to yield an output FFT, as shown in Fig. 3. The purely digital realization of this structure is illustrated in Fig. 15.

The correspondence between the FFT indices and the values of $f_0$ and $\alpha_2$ for the output $\hat{S}_x^{\alpha_0}(f_0)$ are given as follows:

$$S_s \approx S_x^{\alpha_0}(f_0)$$

where

$$f_0 = \frac{k + l}{2N'}$$

and

$$\alpha_0 = \left(\frac{k - l}{N'} + \frac{p}{LP}\right)$$

and

$$\Delta t = L \cdot P; \qquad \Delta f \approx \frac{1}{N'}; \qquad \Delta \alpha = \frac{1}{LP}.$$

The complexity analysis for the FAM system follows. We again calculate $S_x^\alpha(f)$ in only the first quadrant for real signals $x(t)$. A closeup showing the plane and the relationships of $f$, $\alpha$, and the $P$-point FFT outputs is given in Fig. 4.

There will be a subsampled sequence produced for subsequent $P$-point FFT's for approximately $(N'/2) \cdot (N') \cdot (1/2)$—channel pairs to input to a multiplier and final FFT. However, before doing that, there are a number of

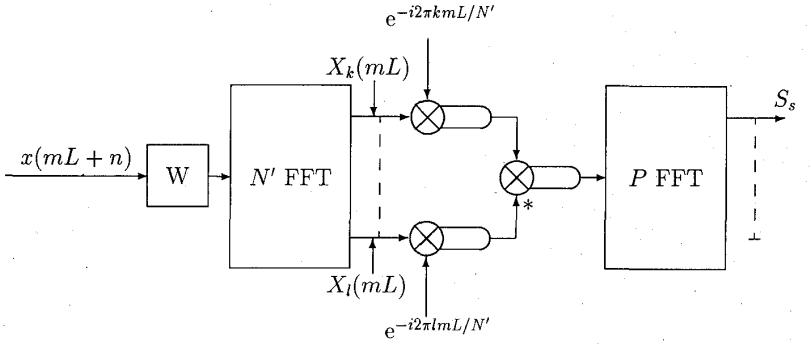



Fig. 15. FFT-accumulation-method architecture.

computations that are performed for each channel. These are, $PN'$ real multiplies for the window, $P$ $N'$-point FFT's for the input channelizer, and $PN'$ complex multiplies for the down converter. These are then followed by $(N')^2/4$ $P$-point FFT's.

These calculations are summarized in Table II.

In order to put the complexity relations on the same footing as in the FSM case, recall that, $LP = \Delta t$ and $N = \Delta t$. Therefore $N = LP$. Also, $1/N' = \Delta f$ and $M/N = \Delta f$. Therefore, $N/M = N'$. Finally, $L = N'/4$. Therefore, $L = N/(4M)$ and $P = 4M$.

Thus,

$$C_{rm} = \frac{2N^2}{M}\log_2(4M) + 8N\log_2\frac{N}{M} + \frac{4N^2}{M} + 20N \tag{25a}$$

$$C_{ra} = \frac{3N^2}{M}\log_2(4M) + 12N\log_2\frac{N}{M} + \frac{2N^2}{M} + 8N. \tag{25b}$$

Noting that $\Delta t\Delta f = M$ and $\Delta f = (M/N)$ yields (25a) as a function of $(\Delta t\Delta f)$:

$$C_{rm} = \frac{2(\Delta t\Delta f)}{(\Delta f)^2}\log_2[4(\Delta t\Delta f)] + \frac{8(\Delta t\Delta f)}{(\Delta f)}\log_2(1/\Delta f) + \frac{4(\Delta t\Delta f)}{(\Delta f)^2} + \frac{20(\Delta t\Delta f)}{(\Delta f)}. \tag{26}$$

The value of $C_{rm}$ as a function of $\Delta t\Delta f$ is plotted in Figs. 11 and 12 for all three realization architectures. The associated complexity product for the three architectures is shown in Figs. 13 and 14.

In order to achieve near-real-time operation, we must introduce parallelism in each of the sections of the realization associated with each column of Table II (or term of (26)). The corresponding factor of real time associated with each section is given in the following equations.

TABLE II
COMPLEXITY SUMMARY FOR FAM

| | ($P$ Required) | | | ($(N')^2/4$ Required) | |
|---|---|---|---|---|---|
| | Wndw | $N'$ FFT | Down Convt. | Correl. Multi. | $P$ FFT |
| CBF | — | $(N'/2)\log_2 N'$ | — | — | $(P/2)\log_2 P$ |
| Cpx mpy | — | $(N'/2)\log_2 N'$ | $N'$ | $P$ | $(P/2)\log_2 P$ |
| Cpx add | — | $N'\log_2 N'$ | — | — | $P\log_2 P$ |
| Real mpy | $N'$ | $2N'\log_2 N'$ | $4N'$ | $4P$ | $2P\log_2 P$ |
| Real add | — | $3N'\log_2 N'$ | $2N'$ | $2P$ | $3P\log_2 P$ |

First, for the rate-1/2 complex butterflies:

$$p_b \cdot F_T = \frac{PN'\log_2 N' + [P(N')^2/4]\log_2 P}{N} = (N/M)\log_2(4M) + 4\log_2(N/M) = (1/\Delta f)\log_2(4\Delta t\Delta f) + 4\log_2(1/\Delta f). \tag{27a}$$

The complex butterflies enumerated in (27a) are associated with the first two terms in (26). The remaining terms in (26) yield the following rate-1 real multiplies associated with the down-converters, correlators and window multipliers.

$$p_{rm} \cdot F_T = 4(N/M) + 20 = 4(1/\Delta f) + 20. \tag{27b}$$

Again, not all values of $p_{rm}$ and $p_b$ have reasonable structures to insure that all processors are kept busy with the right data. It should be emphasized that all of these must be pipelined units, with the units in the complex butterflies operating at 1/2 rate and the rest at the full sample rate. The memories needed for the pipeline FFT's to support the complex butterflies must operate at the same rate when reading or writing.

### D. Strip Spectral Correlation Analyzer (SSCA)

The structure of this spectral correlation algorithm was developed from Fig. 3 at the end of Section III. In that configuration, the output of the $k$th Fourier transformer

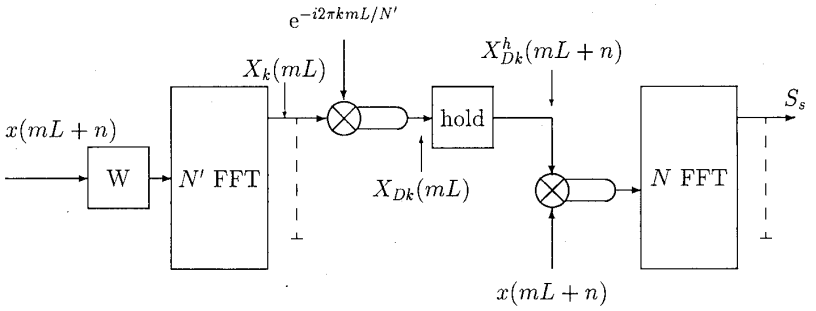



Fig. 16. Strip spectral-correlation-analyzer architecture.

will be a diagonal strip in the $f$-$\alpha$ plane of $S_x^\alpha(f_k - \alpha/2)$. The zero frequency component of the output Fourier transform will correspond to $\alpha = f_k$ and $f = f_k/2$. The $s$th component output of the Fourier transformer will correspond to $\alpha_0 = f_k + s/\Delta t$ or $S_x^{f_k + s/\Delta t}(f_k/2 - s/2\,\Delta t)$.

The digital implementation of the algorithm is shown in Fig. 16. The input filter bank is implemented using a sliding, windowed, coarse FFT, which produces a set of outputs, each of which appears to be the output of a bandpass filter of approximate bandwidth $1/N'$.

The question naturally arises, since we have downconverted the output of the narrow-band filtered channelized signal $X_k$, can we decimate in time by factor $L$ as in FAM? While we can for the downconverted $X_k$, we cannot for the output of the correlation multiplier. In contrast to the FAM approach, one of the correlation multiplier inputs is the unfiltered signal $x_n$, which requires that the multiplier output be sampled at least the same rate as $x_n$. Thus, the subsampling parameter $L$ is $N'/4$ as in FAM, and our sliding windowed input channelizer advances its time origin by $N'/4$ samples for each block of $N'$ inputs to the channelizer. However, a simple hold operation fills in the values of $X_{Dk}(mL + n)$ as $X^h_{Dk}(mL)$ for $1 \le n \le L - 1$, so that the output of the correlation multiplier will be sampled at the full sample rate.

For this implementation, $LP = \Delta t$, $L = N'/4 = N/(4M)$, $P = 4M$, $\Delta f = 1/N' = M/N$, $\Delta\alpha = 1/\Delta t = 1/N$, and $\Delta t\,\Delta f = N/N' = M$.

A complexity analysis is next performed on the major functional parts of this realization in a manner similar to the previous two realizations, resulting in Table III.

In a manner similar to the analysis of the FAM method, we can derive complexity figures in terms of $M$ and $N$ rather than $N'$, resulting in (28a–c).

$$C_b = \frac{N^2}{(4M)}\log_2 N + 2N\log_2\left(\frac{N}{M}\right) \tag{28a}$$

$$C_{rm} = \frac{N^2}{M}\log_2 N + \frac{2N^2}{M} + 8N\log_2(N/M) + 12N \tag{28b}$$

$$C_{ra} = \frac{3N^2}{2M}\log_2 N + \frac{N^2}{M} + 12N\log_2(N/M) + 4N. \tag{28c}$$

TABLE III
COMPLEXITY SUMMARY FOR SSCA

| | ($P$ Required) | | | ($N'/2$ Required) | |
|---|---|---|---|---|---|
| | Wndw | $N'$ FFT | Down Convt. | Corr. Mult. | $N$ FFT |
| CBF | — | $N'/2\log_2 N'$ | — | — | $N/2\log_2 N$ |
| Cpx mpy | — | $N'/2\log_2 N'$ | $N'/2$ | $N$ | $N/2\log_2 N$ |
| Cpx add | — | $N'\log_2 N'$ | — | — | $N\log_2 N$ |
| Real mpy | $N'$ | $2N'\log_2 N'$ | $2N'$ | $4N$ | $2N\log_2 N$ |
| Real add | — | $3N'\log_2 N'$ | $N'$ | $2N$ | $3N\log_2 N$ |

Equations (28a) and (28b), when rewritten in terms of $(\Delta t\Delta f)$ and $\Delta f$ become

$$C_b = \frac{\Delta t\Delta f}{4\Delta f^2}\log_2\left[\frac{\Delta t\Delta f}{\Delta f}\right] + \frac{2\Delta t\Delta f}{\Delta f}\log_2(1/\Delta f) \tag{29a}$$

$$C_{rm} = \frac{\Delta t\Delta f}{\Delta f^2}\log_2\left[\frac{\Delta t\,\Delta f}{\Delta f}\right] + \frac{8\,\Delta t\Delta f}{\Delta f^2}\log_2(1/\Delta f) + \frac{2\Delta t\Delta f}{\Delta f^2} + \frac{12\Delta t\Delta f}{\Delta f} \tag{29b}$$

Next, we can compute the hardware complexity product for the real multiply units:

$$p_{rm}\cdot F_T = (1/\Delta f)\log_2\left(\frac{\Delta t\Delta f}{\Delta f}\right) + 8\log_2(1/\Delta f) + 2(1/\Delta f) + 12. \tag{30}$$

The value of $C_{rm}$ as a function of $\Delta t\Delta f$ is plotted in Figs. 11 and 12 for all three realization architectures. The associated complexity product for the three architectures is shown in Figs. 13 and 14.

Equations (29) and (30) can be broken down into parts corresponding to the FFT's and the other multipliers, and, when combined with (20), yield the following hardware complexity products:

$$p_b\cdot F_T = \left(\frac{1}{2\Delta f}\right)\log_2\left(\frac{\Delta t\Delta f}{\Delta f}\right) + 4\log_2(1/\Delta f) \tag{31a}$$

for the rate-1/2 complex butterflies, and

$$p_{rm} \cdot F_T = 2(1/\Delta f) + 12 \tag{31b}$$

for the rate-1 real multiplies in the window and down converter and correlation multipliers.

Equation (31) suggests that balanced real-time operation can be achieved by having two real-time input $(1/\Delta f)$-point FFT's, $[1/(4\Delta f)]$ real-time output $\Delta t$-point FFT's, 12 window and down-convert real multipliers, and $(2/\Delta f)$ correlation multipliers.

## V. Example

In this section, we develop an application involving the estimation of the spectral correlation function for a real-valued signal and for all $f$ and $\alpha$ satisfying $0 \le f \le 0.5$ and $0 \le \alpha \le 1 - 2|f|$. The spectral-correlation analyzer is to be used to detect the presence of a high-data-rate binary-phase-shift-keyed (BPSK) signal obscured by stationary white Gaussian noise. We examine the complexity required to achieve a desired output signal-to-noise ratio (SNR), $\rho_{\text{out}}$, for the cyclic feature associated with the carrier frequency that occurs at $f = 0$ and $\alpha = 2f_c$. While this example is developed around the use of our analyzer to detect the particular feature associated with the carrier frequency, other features would be sought for the detection and parameter measurement for other signal types and parameters, e.g., the keying-rate feature at $f = f_c$, $\alpha = f_b$ for BPSK or QPSK. Thus the general SCA needs to have the ability to compute the SCF for the entire first quadrant of the $f$-$\alpha$ plane.

If the BPSK signal is present, the observed waveform has the form

$$x(n) = s(n) + w(n)$$

where $w$ is white Gaussian noise with average power $N_0$ and $s$ is the BPSK signal of interest:

$$s(n) = \sum_{k=-\infty}^{\infty} \sqrt{2P_s} b_k u_{T_b}(n - kT_b) \cos(2\pi f_c n).$$

The parameters $P_s$, $f_c$, and $T_c$ are the average power, IF carrier frequency, and bit interval, respectively, and $b_k \in \{\pm 1\}$ is a white random message sequence. Assume that the sampling rate is large compared with the BPSK signal bandwidth $(1/T_b \ll 1)$ and that the carrier frequency is not near the edges of the analysis band $(2/T_b < f_c < 0.5 - 2/T_b)$ so that aliasing effects can be ignored. The spectral correlation function for the observed waveform is

$$S_x^\alpha(f) = \begin{cases} S_s(f) + N_0 & \text{for } \alpha = 0 \\ S_s^\alpha(f) & \text{otherwise,} \end{cases}$$

and, as reported in [8], the spectral correlation function for the BPSK signal is

$$S_s^\alpha(f) = \begin{cases} \dfrac{P_s}{2T_b}\left(Q(f - f_c + \alpha/2)Q(f - f_c - \alpha/2)\right. \\ \qquad \left. + Q(f + f_c + \alpha/2)Q(f + f_c - \alpha/2)\right) \\ \qquad \text{where } -1 < \alpha = k/T_b < 1 \\ \dfrac{P_s}{2T_b} Q(f - k/2T_b)Q(f + k/2T_b) \\ \qquad \text{where } 0 < \alpha = 2f_c + k/T_b < 1 \\ \dfrac{P_s}{2T_b} Q(f - k/2T_b)Q(f + k/2T_b) \\ \qquad \text{where } -1 < \alpha = -2f_c + k/T_b < 0 \end{cases} \tag{32}$$

where $k = 0, \pm 1, \pm 2, \cdots$ and

$$Q(f) \triangleq \frac{\sin \pi f T_b}{\pi f}.$$

With attention restricted to nonzero cycle frequencies, the spectral feature with the highest peak value is the one associated with the carrier at $\alpha = \pm 2f_c$,

$$S_s^{\pm 2f_c}(f) = \frac{P_s}{2T_b} Q^2(f). \tag{33}$$

This feature is studied in detail to illustrate the relationship between SNR and computational complexity. Let the analyzer be tuned to $f_0 = 0$, $\alpha_0 = 2f_c$, and assume that the analyzer is well characterized by the rectangular kernel (8). The input SNR is defined as the average signal power divided by the average noise power within half the null-to-null bandwidth of the BPSK signal:

$$\rho_{\text{in}} = \frac{P_s T_b}{2N_0}. \tag{34}$$

Substitution of (32)–(34) into the expression for output SNR (14) and use of the low-input-SNR condition $\rho_{\text{in}} \ll 1$ yields

$$\rho_{\text{out}} \approx \frac{\Delta t \Delta f}{2} \left| \frac{1}{T_b \Delta f} \int_{-T_b\Delta f/2}^{T_b\Delta f/2} \frac{\sin^2 \pi v}{(\pi v)^2} dv \right|^2 \rho_{\text{in}}^2. \tag{35}$$

Note that the output SNR is directly proportional to the time-frequency resolution product of the analyzer, $\Delta t \Delta f$, and to the square of the input SNR.

The optimal cycle detector described in [9] is essentially a spectral correlation analyzer with the kernel matched to the signal spectral correlation function, i.e., $\tilde{M}(\alpha_0, f) = S_r^{\alpha_0}(f)^*$. The optimal cycle detector operat-

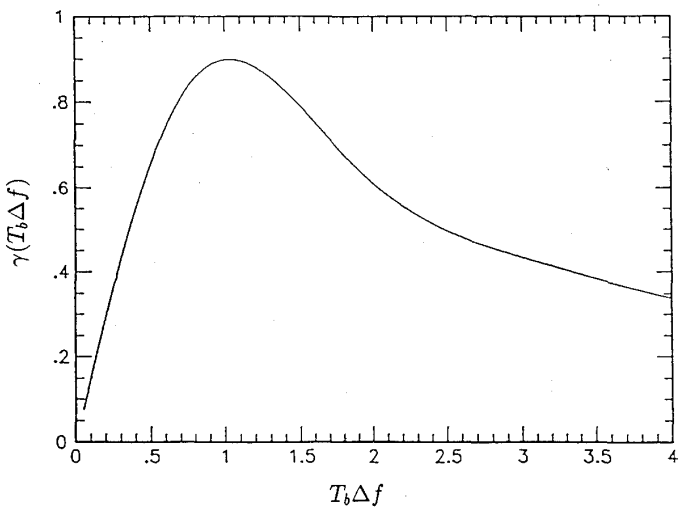



Fig. 17. Analyzer-performance function ($\gamma$).

ing on an observation interval of duration $\Delta t$ produces the output SNR [9]

$$\rho_0 = \frac{\Delta t}{2N_0^2} \int_{-1/2}^{1/2} |S_s^{\alpha 0}(f)|^2 \, df.$$

Substitution of (33) into the expression for $\rho_0$ yields [10]

$$\rho_0 = \frac{1}{3}\left(\frac{P_s T_b}{2N_0}\right)^2 \frac{\Delta t}{T_b} = \frac{1}{3}\rho_{\text{in}}^2 \frac{\Delta t}{T_b}. \tag{36}$$

The performance of the ordinary analyzer relative to the optimal cycle detector is characterized by

$$\gamma(T_b\Delta f) \triangleq \frac{\rho_{\text{out}}}{\rho_0} \approx \frac{3}{2T_b\Delta f}\left| \int_{-T_b\Delta f/2}^{T_b\Delta f/2} \frac{\sin^2 \pi v}{(\pi v)^2} \, dv \right|^2. \tag{37}$$

This ratio is evaluated numerically and plotted in Fig. 17. For a fixed observation interval, the output SNR is highest if the frequency-resolution width of the analyzer is matched to the reciprocal of the bit interval, i.e., $\Delta f = 1/T_b$, and in this case the output SNR is

$$\max_{\Delta f} \{\rho_{\text{out}}\} = \max_{\Delta f} \{\gamma(T_b\Delta f)\rho_0\} = 0.90\,\rho_0.$$

Thus, 0.46-dB loss in output SNR is incurred by using the spectral correlation analyzer with $\Delta f = 1/T_b$ rather than the optimal cycle detector. So, for cases where $\Delta f = 1/T_b$,

$$\rho_{\text{out}} = 0.3\rho_{\text{in}}^2 \frac{\Delta t}{T_b}. \tag{38}$$

For $\Delta f$ small compared to $2/T_b$, (37) can be shown to lead to the approximation

$$\gamma(T_b\,\Delta f) = \tfrac{3}{2}\,T_b\Delta f$$

or

$$\rho_{\text{out}} = \tfrac{1}{2}\,\rho_{\text{in}}^2\,\Delta t\Delta f \qquad \Delta f < \frac{0.5}{T_b}. \tag{39}$$

A general purpose spectral correlation analyzer would most likely have a relatively small $\Delta f$, matching the $f_b = 1/T_b$ of the most narrow-band PSK signal expected. Such an SCA would also be able to find features for signals with higher $f_b$. For this situation, (38) applies to the case where $f_b = \Delta f$, and (39) to cases where $f_b > 2\Delta f$. Note that for $\rho_{\text{in}}$ fixed,

$$\rho_{\text{out}}(f_b)|_{f_b=\Delta f} < \rho_{\text{out}}(f_b)|_{f_b>2\Delta f}.$$

In general, if the $\rho_{\text{in}}$s are not the same, both (38) and (39) will have to be checked, and the one requiring the higher $\Delta t\Delta f$ will govern.

This output SNR, $\rho_{\text{out}}$, can be used to estimate the probability of detection and of false detection of the feature being estimated by a spectral-correlation analyzer.

Now to evaluate the complexity of our example. We assume that the BPSK signal of interest has a $\rho_{\text{in}}$ in the bandwidth $1/T_b$ of $-9$ dB, or

$$\rho_{\text{in}} = \frac{P_s T_b}{2N_0} = \frac{1}{8}.$$

We will also consider a moderately broadband signal, one with a relative data rate of $f_b = (1/T_b) = \frac{1}{8}$. This would correspond to 4 MHz, $(\frac{1}{8}) \times f_s$ where $f_s$, the sample rate, is 32 MHz. This sample rate is one that is a challenge to the state of the art in analog-to-digital conversion and digital IC technology, but a rate achievable with careful design in one-micron CMOS technology. This rate will allow the analyzer to process signals with bandwidths of up to approximately 15 MHz. Finally, we assume that the front end of our analyzer has down converted the center

frequency of the band being processed to an IF of 8 MHz ($f_c = \frac{1}{4}$).

To design our analyzer to obtain the maximum $\rho_{out}$, we use a $\Delta f = 1/T_b$, yielding a $\gamma(1) = 0.90$. Furthermore, let us assume that it is desired to have a $\rho_{out} \geq 8$ (+9 dB). Therefore, we need to satisfy (38),

$$8 \leq \rho_{out} = 0.3(\tfrac{1}{8})^2 \Delta t \Delta f.$$

Consequently, $\Delta t\, \Delta f \geq 1.7 \times 10^3$, and $\Delta t \geq 1.4 \times 10^4$ samples. Thus, if we use radix-two FFT's as building blocks, we should pick $\Delta t = 16K$, and therefore $\Delta t \Delta f = 2K$. Then, with $\Delta t \Delta f = 2048$, the actual $\rho_{out}$ is 9.6. These values appear in Figs. 11 and 13 as the value of $\Delta t \Delta f$ marked "example 1." Thus we see that about $2.7 \times 10^6$, $4.7 \times 10^6$, and $2.70 \times 10^9$ real multiplies are required to compute one quadrant of the spectral-correlation function with an output SNR of 8 (+9 dB) or greater using the SSCA, FAM, and FSM architectures, respectively.

Instead, if we want to make a more detailed analysis of the bifrequency plane, attempting to locate signals with a variety of bandwidths, down to an $f_b = \frac{1}{128}$, we would pick $\Delta f = \frac{1}{128}$, which would be appropriate for the detection of a BPSK signal of that bit frequency. Now the question is, what $\Delta t \Delta f$ is required to obtain at least the desired $\rho_{out}$ on that signal as well as one with $f_b = 1/T_b = \frac{1}{8}$?

Equation (38) applies again, since $\Delta f = \frac{1}{128}$, and the smallest $1/T_b = \frac{1}{128}$. Assuming a $\rho_{out}$ of 8 is desired, (38) yields:

$$8 = 0.3(\tfrac{1}{8})^2 \Delta t \Delta f.$$

So,

$$\Delta t \Delta f \geq 1.7 \times 10^3$$

and

$$\Delta t \geq 2.18 \times 10^5$$

or

$$\Delta t = 2^{18}.$$

Then with $\Delta t = 2^{18}$ and $\Delta f = \frac{1}{128}$, for a BPSK signal whose $f_b = \frac{1}{128}$, $\rho_{out}$ would be 9.6. For the other signal, whose $f_b = \frac{1}{8}$, (39) applies, and

$$\rho_{out} = \tfrac{1}{2}(\tfrac{1}{8})^2 \cdot 2^{11} = 16.$$

The complexity curves that apply to this example are shown in Figs. 12 and 14. The values appropriate are marked "example 2."

## VI. Conclusions

A variety of quadratic system implementations of spectral correlation analyzers using pipeline digital techniques has been studied. The functionality of these types of spectral correlation analyzers is simply related to properties of the system kernel as described by (4).

The most straightforward implementation, the FSM architecture was found to have a complexity on $o(N^2)$, and would require a parallelism factor of $\log_2(N)$ complex butterflies and $N/4$ complex correlators in order to be computed in real time. Or, viewed as a function of $(\Delta t \Delta f)$ for fixed $\Delta f$, the complexity is on $o((\Delta t \Delta f)^2)$.

The two computationally efficient structures based on time smoothing, however, have complexity on $o((N^2/M)\log_2(N))$. Furthermore, when the complexity is viewed as a function of $(\Delta t \Delta f)$, for fixed $\Delta f$, the complexity is on $o((\Delta t \Delta f)\log_2(\Delta t \Delta f))$.

Figs. 12–14 show curves of log of complexity as a function of log $(\Delta t \Delta f)$ for three values of $\Delta f$. The improved complexity dependence for the FAM and SSCA methods relative to the FSM architecture is clearly shown by these curves.

An example illustrates the complexity required for the computation of the first quadrant of the bifrequency plane with a $\Delta f$ of $\frac{1}{128}$. Carrier features from BPSK signals with $f_b \geq \frac{1}{128}$ and $\rho_{in} = \frac{1}{8}$ will be produced by the analyzer with a $\rho_{out} \geq 8$, provided that $\Delta t\, \Delta f \geq 2^{11}$. This complexity is shown as "Example 2" on Figure 12 and the complexity product in Figure 14.

This paper has presented the theoretical justification for three SCA architectures and has analyzed the complexity of these architectures. Also, a measure of feature detectability, the analyzer output SNR, $\rho_{out}$, has been developed and its use in designing an SCA was illustrated.

## References


[1] W. A. Gardner, "The spectral correlation theory of cyclostationary time series," *Signal Processing*, vol. 11, pp. 13–36, July 1986.

[2] W. A. Gardner, "Measurement of spectral correlation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1111–1123, Oct. 1986.

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

[4] W. A. Brown, "On the theory of cyclostationary signals," Ph.D. dissertation, Univ. California, Davis, Sept. 1987.

[5] W. A. Brown and H. H. Loomis, Jr., "Digital implementations of spectral correlation analyzers," in *Proc. IEEE Fourth Annu. ASSP Workshop Spectrum Estimation Modeling*, Aug. 1988, pp. 264–270.

[6] R. S. Roberts, W. A. Brown, and H. H. Loomis, Jr., "Computationally efficient algorithms for cyclic spectral analysis," *IEEE Signal Processing Mag.*, vol. 8, no. 2, pp. 38–49, Apr. 1991.

[7] W. A. Gardner, "Spectral correlation of modulated signals: Part I—Analog modulation," *IEEE Trans. Commun.*, vol. COM-35, pp. 584–594, June 1987.

[8] W. A. Gardner, W. A. Brown, and C. K. Chen, "Spectral correlation of modulated signals: Part II—Digital modulation," *IEEE Trans. Commun.*, vol. COM-35, pp. 595–601, June 1987.

[9] W. A. Gardner, "Signal interception: A unifying theoretical framework for feature detection," *IEEE Trans. Commun.*, vol. 36, pp. 897–906, Aug. 1988.

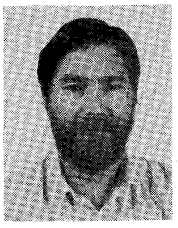
[10] W. A. Gardner and C. M. Spooner, "Signal interception: Performance advantages of cyclic feature detectors," *IEEE Trans. Commun.*, vol. 40, pp. 149–159, Jan. 1992.

[11] W. A. Gardner and C. K. Chen, "Signal-selective time-difference-of-arrival estimation for passive location of manmade signal sources in highly corruptive environments: Part I–Theory and method," *IEEE Trans. Signal Processing*, vol. 40, no. 5, pp. 1168–1184, May 1992.

[12] C. K. Chen and W. A. Gardner, "Signal-selective time-difference-of-arrival estimation for passive location of manmade signal sources in highly corruptive environments: Part II—Algorithms and performance," *IEEE Trans. Signal Processing*, vol. 40, no. 5, pp. 1185–1197, May 1992.

[13] B. G. Agee, S. V. Schell, and W. A. Gardner, "Spectral self-coherence restoral: A new approach to blind adaptive signal extraction using antenna arrays," *Proc. IEEE*, vol. 78, pp. 753–767, Apr. 1990.

[14] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1975.
[15] "Digital signal processing chip set," Honeywell, Inc., Publ. DSP-6008/489, Rev. A, 3/88, Colorado Springs, CO.
[16] R. S. Roberts, "Architectures for digital cyclic spectral analysis," Ph.D. dissertation, Univ. California, Davis, Sept. 1989.
[17] S. V. Schell and W. A. Gardner, "High resolution direction finding," in *Handbook of Statistics*, N. K. Bose and C. R. Rao, Eds. Amsterdam: Elsevier, 1992.



**William A. Brown, III** (S'74–M'77) 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 the 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'76) was born in Wilmington, DE, 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. degree in electrical engineering from the University of Maryland in 1959, and the Ph.D. degree from the 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 (UC), Davis, 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 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.