AN ADAPTIVE ARRAY FOR INTERFERENCE REJECTION IN A CODED COMMUNICATION SYSTEM

K. L. Reinhard

Ohio State University
Columbus, Ohio

May 1972

RADC-TR-72-91
Technical Report
May 1972

AN ADAPTIVE ARRAY FOR INTERFERENCE REJECTION IN A CODED COMMUNICATION SYSTEM

Ohio State University Research Foundation

**Approved for public release;
distribution unlimited.**

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, New York

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Ohio State University Research Foundation 1320 Kinnear Rd Columbus, OH 43212	Unclassified 2b. GROUP

3. REPORT TITLE

AN ADAPTIVE ARRAY FOR INTERFERENCE REJECTION IN A CODED COMMUNICATION SYSTEM

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Report (6th Interim)

5. AUTHOR(S) (First name, middle initial, last name)

Reinhard, K. L.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 1972	81	7

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
F30602-69-C-0112 b. Job Order No. 45190000 d.	ElectroScience Laboratory 2738-6 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) RADC-TR-72-91

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Rome Air Development Center (CORC) Griffiss AFB, New York 13440

13. ABSTRACT

An adaptive receiving array for rejecting interference in a coded communication system is described and computer simulations are presented which illustrate basic properties of the array processor. The array automatically forms a beam in the direction of a signal bearing the proper code modulation and forms nulls in the directions of signals bearing improper modulation. The communication signal present at the array input is assumed to be biphase (0°-180°) modulated by the modulo-2 sum of a periodically-repeating code and a data sequence. The code is assumed known at the receiving site and is the means by which the array distinguishes between "desired" and "undesired" signals.

The adaptive processor minimizes the mean-square difference (error) between the array output signal and a reference signal. The reference signal contains the data and code modulations of the desired input signal and is generated from the array output signal by appropriate waveform processing. The array weighting coefficients (and hence the pattern) are adjusted by a feedback control system designed to provide a steepest-descent minimization of the error.

Digital computer simulation results illustrating the performance of a two-element array for the case of cw interference are presented. The results show that signal carrier frequency and code bit timing estimates are required at the array and that the bandwidths of the adaptive feedback loops within the processor must be restricted for proper operation. The simulation results also show that the performance of the array processor degrades gracefully as the errors in the estimates increase.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Adaptive array						
Null-steering						
Digital signalling						
Reference signal generation						
Rms error minimization						
Transient performance						
Computer simulation						

SAC--Griffiss AFB NY

AN ADAPTIVE ARRAY FOR INTERFERENCE REJECTION IN A CODED COMMUNICATION SYSTEM

K. L. Reinhard

Ohio State University Research Foundation

FOREWORD

This report was prepared by the ElectroScience Laboratory, Department of Electrical Engineering, The Ohio State University at Columbus, Ohio. Research was conducted under Contract F30602-69-C-0112, Job Order Number 45190000. Mr. John Patti (CORC) was the RADC Program Monitor. Secondary report number is ElectroScience Laboratory 2738-6.

This document has been reviewed by the Office of Information (OI) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved.

Approved: John J. Patti
JOHN J. PATTI
Project Engineer

Approved: James E. McFee
JAMES E. McFEE, Chief
Communications Techniques Branch
Communications & Navigation Division

ABSTRACT

An adaptive receiving array for rejecting interference in a coded communication system is described and computer simulations are presented which illustrate basic properties of the array processor. The array automatically forms a beam in the direction of a signal bearing the proper code modulation and forms nulls in the directions of signals bearing improper modulation. The communication signal present at the array input is assumed to be biphase (0^{o}-180^{o}) modulated by the modulo-2 sum of a periodically-repeating code and a data sequence. The code is assumed known at the receiving site and is the means by which the array distinguishes between "desired" and "undesired" signals.

The adaptive processor minimizes the mean-square difference (error) between the array output signal and a reference signal. The reference signal contains the data and code modulations of the desired input signal and is generated from the array output signal by appropriate waveform processing. The array weighting coefficients (and hence the pattern) are adjusted by a feedback control system designed to provide a steepest-descent minimization of the error.

Digital computer simulation results illustrating the performance of a two-element array for the case of cw interference are presented. The results show that signal carrier frequency and code bit timing estimates are required at the array and that the bandwidths of the adaptive feedback loops within the processor must be restricted for proper operation. The simulation results also show that the performance of the array processor degrades gracefully as the errors in the estimates increase.

TABLE OF CONTENTS

SECTION I
INTRODUCTION

Self-phased antenna arrays[1] have been considered for use in a variety of applications in recent years. A self-phased array coherently adds the signals received by the array elements regardless of the angular position of the source. An arbitrary location of the elements on any surface is acceptable provided shadowing or strong mutual-coupling effects are avoided. The ability of the self-phased array to focus in the Fresnel region and to compensate for phase disturbances in the transmission medium (wavefront distortion) are also significant features.

There are some performance limitations, however. For unequal signal amplitudes at the element inputs (resulting from aperture blockage or mutual coupling) or unequal element noise powers the SNR enhancement at the array output may be less than that realizable by other combining techniques.[2] The self-phased array is also more susceptible to external noise or interfering signal sources than ordinary directive antenna systems. Under strong interference conditions, the self-phased array has the tendency to "lock-on" and track the interference rather than the desired signal.[3]

In this report, a more general type of adaptive array processing is investigated in which both the amplitude and phase of the element signals are adjusted prior to combining. Advantages of this more general approach include the potential for diversity combining and for interference rejection. Applebaum[4] and Shor[5] were among the first to consider adaptive adjustment of element gain and phase as an optimal control problem. They chose for an optimization criterion the maximization of the ratio of output signal power to total noise power. Widrow, et al[6] later analyzed adaptive array systems which minimize the mean-square difference (error) between the array output signal and a reference signal. The angular position and modulation components of the desired signal source were assumed known a priori at the receiving array.

The processing technique discussed in this report follows that of Widrow, et al with some important differences. Knowledge of the angular position of the desired signal source is not assumed. In addition, the desired signal received at the array is assumed to have an unknown data modulation impressed. Information regarding the desired signal's carrier frequency, code modulation, and code modulation time base is required at the array, however. The most significant difference is the addition of a waveform processor at the array output. The purpose of the waveform processor is two-fold: 1) to discriminate between a properly coded desired signal and undesired signals in the array output, and 2) to provide an estimate of the unknown (data) modulation components. The waveform processor generates the required reference signal for the array.

This report is organized as follows: The adaptive array signal processing technique is described in Sec. II. This section also contains a discussion of the characteristics required of the reference signal and a description of how the reference signal can be derived from the array output. In Sec. III, the computer program used to evaluate the transient performance of the array processor is described. Computer simulation results are given in Section IV. Analytical results which augment the simulation results are included where they are available. The results are summarized and conclusions drawn in Section V.

SECTION II
THE ADAPTIVE ARRAY SIGNAL PROCESSING TECHNIQUE

A. Basic Adaptive Array Configuration

The basic configuration of an N-element adaptive array is shown in Fig. 1. Here, $x_1(t)$, $x_2(t)$, $\cdots$, $x_N(t)$ represent signals at the outputs of the array elements and $y_1(t)$, $y_3(t)$, $\cdots$, $y_{2N-1}(t)$ represent the processor input signals. The signal from each element is adjusted in magnitude and phase prior to signal combining (summing). The output signal from the k^{th} element is applied to two channels: an "in-phase" channel and a "quadrature" channel. The quadrature channel signal, $y_{2k}(t)$, is delayed in phase by 90^0 with respect to the signal, $y_{2k-1}(t)$, in the in-phase channel. The in-phase and quadrature signals associated with the k^{th} element are multiplied by weighting coefficients $w_{2k-1}(t)$ and $w_{2k}(t)$, respectively. The weighting coefficients are real valued and can be positive, negative, or zero. Consequently, the contribution of the k^{th} element to the array output,

$$(1)\quad S_k(t) = w_{2k-1}(t)\, y_{2k-1}(t) + W_{2k}(t) y_{2k}(t),$$

can have an arbitrary amplitude and phase relationship with respect to input signal $x_k(t)$. The array output signal, $S(t)$, is subtracted from a reference signal, $R(t)$, to produce error signal $E(t)$. The error signal and signals $y_1(t)$, $\cdots$, $y_{2N}(t)$ are applied as inputs to the feedback circuitry. Control voltages generated by the feedback circuits determine the values of weights $w_i(t)$.

The objective of the processing is to force the array output signal $S(t)$ to equal the reference signal $R(t)$. One measure of performance is the squared error given by

$$(2)\quad E^2(t) = \left[R(t) - \sum_{i=1}^{2N} w_i(t)\, y_i(t)\right]^2 \geq 0.$$

At any given time, E^2 is a quadratic function of the weights w_i; therefore $E^2(w_1, \cdots, w_{2N})$ defines a "bowl-shaped" surface with a well-defined minimum. The error can be forced to assume this minimum by appropriately designing the feedback circuitry. One approach is to adjust the weights according to a steepest-descent minimization procedure,[6] i.e., the w_i are moved in the maximum downhill direction on the squared-error surface. Since this direction is given by the negative of the gradient of E^2 with respect to the w_i, steepest-descent minimization requires the w_i to be changed as follows:

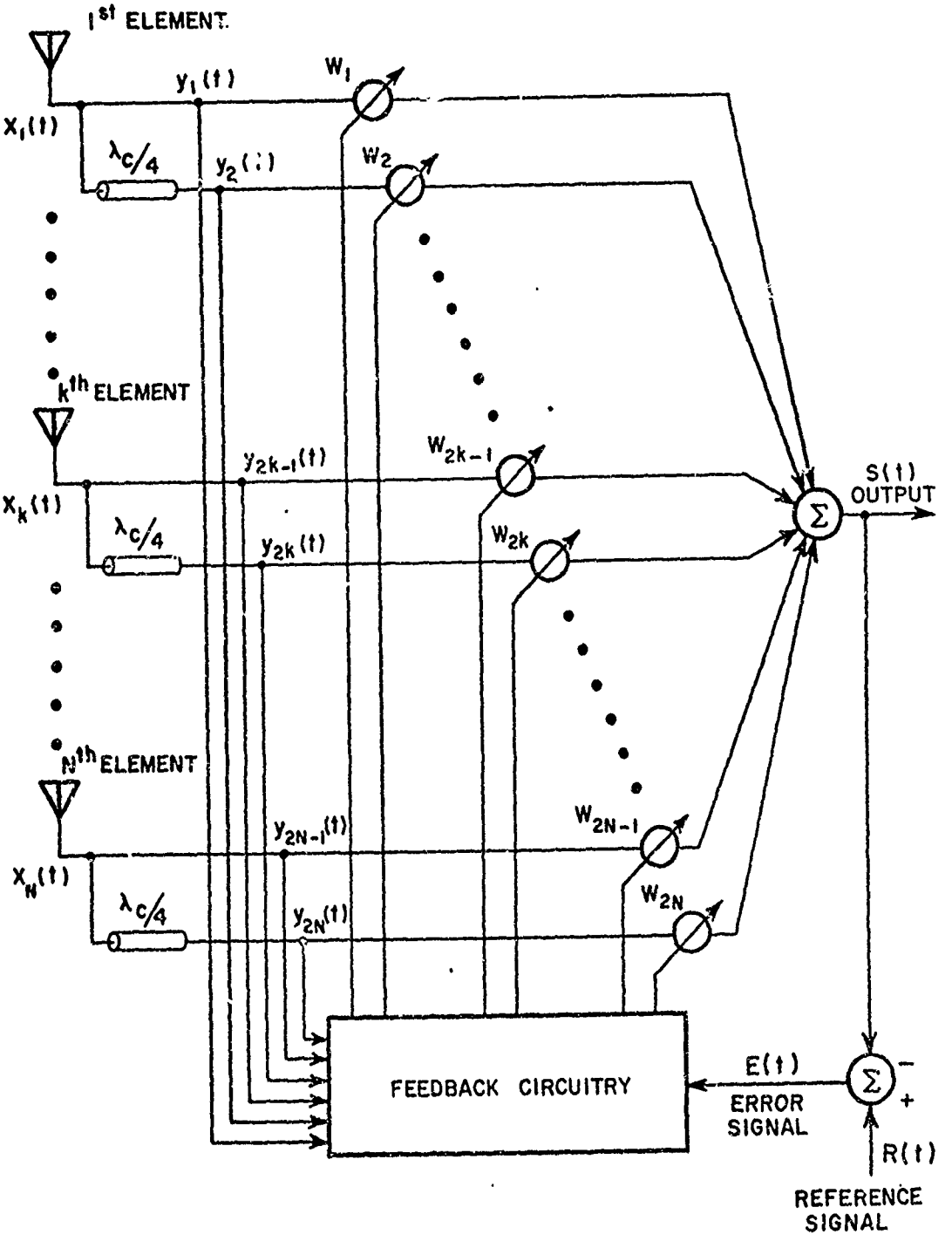


Fig. 1. The basic adaptive array.

$$(3) \qquad \frac{dw_i}{dt} = k_s \nabla_i(E^2) \; ; \quad k_s < 0.$$

The larger $\nabla_i(E^2)$, the faster w_i changes. The i^{th} component of the gradient may be evaluated as

$$(4) \quad \nabla_i(E^2) = 2E(\nabla_i E) = -2E\, y_i,$$

and the feedback rule (3) reduces to

$$(5) \quad \frac{dw_i(t)}{dt} = -2k_s\, E(t)\, y_i(t) \; ;$$

$$w_i(t) = K \int_0^t E(t')\, y_i(t')\, dt' + w_i(0)$$

where $K = -2k_s$ is the (positive) integrator gain constant. A method for implementing the feedback circuitry so that (5) is satisfied is illustrated in Fig. 2 *

B. Reference Signal Requirements

As discussed in the previous section, one approach to optimizing array performance is to minimize the mean-square error. This is equivalent to forcing the array output signal S(t) to approximate the reference signal R(t) as closely as possible in a mean-square sense. A component of S(t) which is not contained in R(t) appears as an error signal, and the feedback adjusts the weights to remove it from the array output. The result, in pattern terminology, is the formation of a pattern null in the direction from which this signal arrives. If a component of S(t) is contained in R(t), the feedback retains this signal in the output and adjusts its average amplitude and phase as closely as possible to that of R(t). Equivalently, a pattern lobe is formed in the direction of arrival of this signal. Output errors are minimized by "accepting" signals with proper amplitude and phase modulation and by rejecting signals with improper modulation. The characteristics of the reference signal, therefore, determine the ability of the adaptive array to discriminate between "desired" and "undesired" signals at the array input. For proper operation, the reference signal waveform must be highly correlated with the desired input signal waveform and have a low correlation with the input interference waveform. These conditions must

---

*The signal processing shown in Fig. 2 may be performed at IF frequencies by down-converting the received signals in separate mixers using a common local oscillator signal.

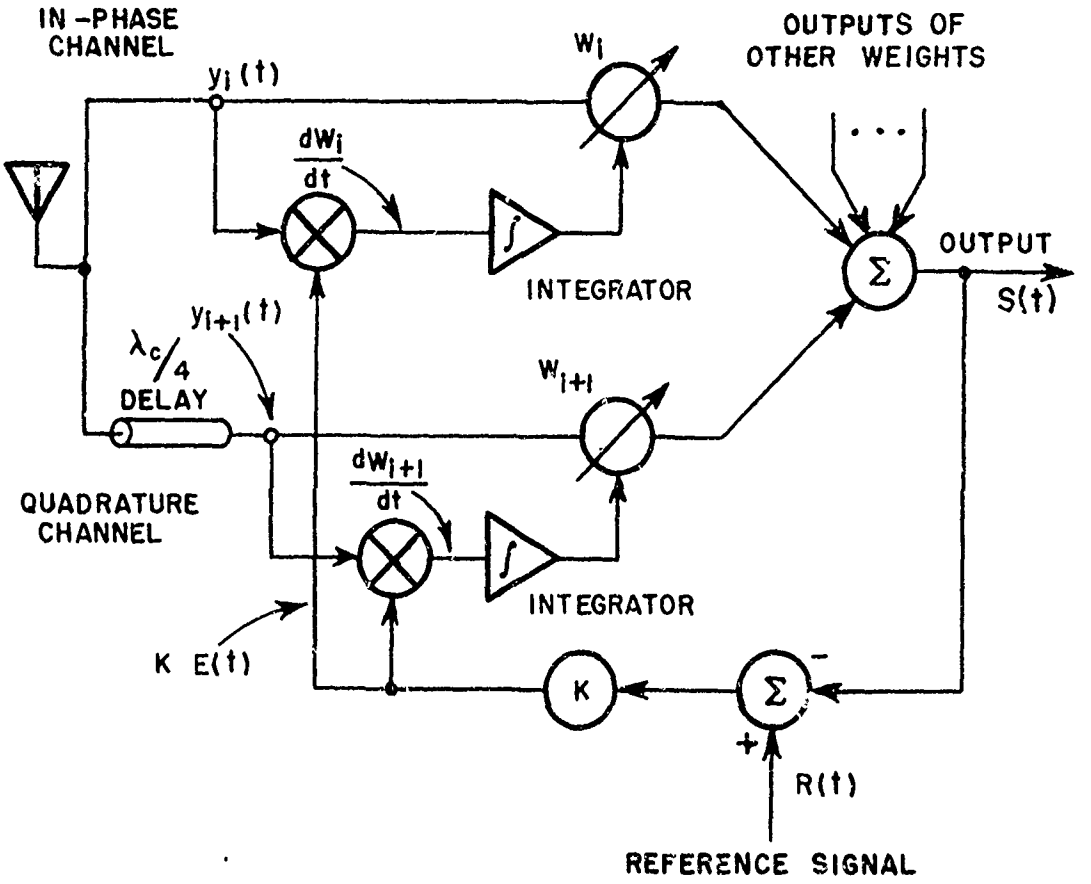


Fig. 2. Feedback circuitry for each element of the array.

be satisfied when the time interval over which correlation is measured is equal in duration to the response time of the feedback control loops and is chosen arbitrarily in time. These latter restrictions are necessary to ensure that the correlation properties of the input signals are preserved at the array output. The weighting coefficients are time-varying, in general, and are therefore capable of introducing amplitude and phase modulations having rates determined by the bandwidth of the feedback loops. The possibility exists then that the processor may alter the modulation characteristics of the input waveforms thereby affecting their correlation with the reference signal. The important point here is that although an undesired signal may be completely uncorrelated with the reference signal when the cross-correlation function is evaluated over an infinite time interval, the response of the array processor is determined by the correlation over a much smaller observation interval --- in effect, a moving "window" of width equal to the response time of the feedback loops.

One approach to satisfying the reference signal requirements presented above is to employ wideband signalling techniques. If the bandwidth occupied by the desired input and reference signals greatly exceeds the loop processing bandwidth, then these signals must possess significant amplitude and/or phase fluctuations over any time interval comparable with the response time of the loops. Undesired signals not containing the reference modulation will not correlate with the reference signal. The correlation cannot be significantly increased by weighting coefficient modulation in this case; as a result, the feedback processing removes these signals from the array output. The selection of a wideband modulation technique is influenced by the signal environment in which the array operates and by the synchronization requirement. The time base of the reference signal modulation must be aligned with the time base of the desired input signal (or vice-versa) for proper correlation. Timing error results in a correlation loss and subsequent partial suppression of the desired signal at the array output. The accuracy to which timing can be established and maintained dictates an upper bound on the signal bandwidth since the cross correlation function associated with the reference and desired input signals becomes narrower (more sharply peaked) as the bandwidth is increased. Other practical considerations influence the choice of a wideband modulation waveform, e.g., the relative ease of waveform generation and time-base adjustment, and the capability for additional processing of the received signal to obtain timing error information.

When random (data) modulation is impressed on the transmitted signal in addition to the deterministic (bandspreading) modulation, an additional problem is encountered in generating the reference signal for the array. If the reference signal does not contain the data modulation, a loss in correlation and partial rejection of the received data-carrying signal will necessarily occur. There are at least two approaches to avoiding this problem:

1) A data-carrying signal and a deterministic wideband beam-steering signal might be transmitted sequentially; the operation of the array processor would be correspondingly time-multiplexed,

or

2) Techniques for estimating the unknown (data) modulation components from the received signal itself might be incorporated into the processor. The estimates would then be used in generating the reference signal.

The latter approach is pursued here. An adaptive receiving array which employs a coded reference signal derived by waveform processing the array output signal is described in the next section.

C. Description of the Adaptive Array/Waveform Processor

A block diagram of the system under study is shown in Fig. 3. The "desired" signal at the array input is assumed to be a bi-phase modulated, digital communication signal. The binary phase modulation on this signal is the modulo-two sum of a binary code sequence and a random binary data sequence. The code is assumed known at the array and is the means by which the processor distinguishes the "desired" signal from "undesired" signals.

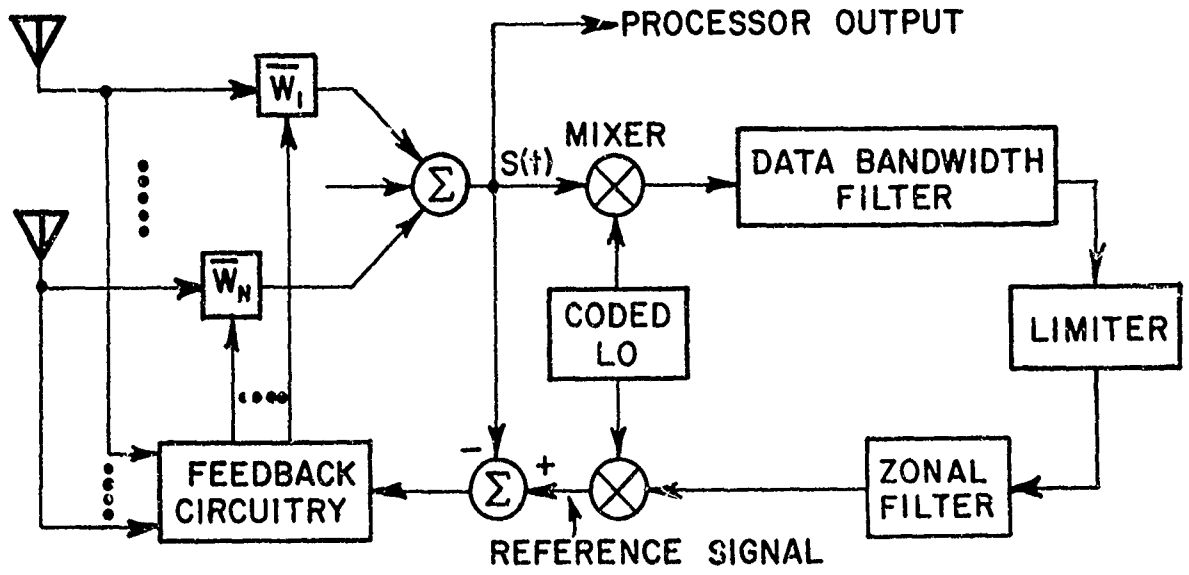


Fig. 3. Adaptive array processor with a waveform-processed reference signal.

The reference signal is derived by waveform-processing array output S(t). First, a local oscillator signal is bi-phase modulated with the same code as the desired input signal code and then mixed with S(t). The desired signal component of S(t) produces a signal at the mixer output containing only data modulation (assuming perfect synchronization of input and local oscillator codes). Since the data rate is assumed to be much less than the code rate, this signal occupies a smaller bandwidth than S(t). The undesired signals in S(t), however, do not undergo this reduction in bandwidth when multiplied by the reference code. Wideband undesired signals remain wideband at the mixer output and narrowband undesired signals are spread in bandwidth by the local oscillator phase switching. The data filter passes the data modulation and reduces the power in the undesired portions of S(t) relative to the desired part.

To a rough approximation, the reduction in power is equal to the ratio of the bandwidths. A bandpass limiter (i.e., an ideal limiter and wideband zonal filter) fixes the reference signal amplitude. Note that without limiting the weighting coefficients would be driven to zero if the reference network gain were less than unity since, in this case, the reference signal magnitude would always be smaller than the array output signal magnitude. If the reference network gain were greater than unity, the weighting coefficients would eventually be forced to their saturation limits. The use of a limiter circumvents the need to maintain unity gain. The processed signal is mixed with the coded local oscillator signal to reinsert the code modulation; only the sideband centered on the operating frequency of the array appears at the output of the second mixer (post-mixer filtering is not shown in Fig. 3). This output, i.e., the reference signal R(t) for the array, approximates the desired component in S(t). Slow phase variations of the local oscillator carrier which are introduced at the first mixer are removed in the second mixer.

In practice, the desired components in the reference signal and the array output signal will differ due to the delay introduced by the bandpass filter and code synchronization error. Due to the delay, the data transitions are not synchronized. Moreover, the phase shift through the waveform processor is frequency dependent; thus, the desired in-phase (mod. 2π) condition will not exist if the signal frequency is offset or if the filter parameters change with time due to aging or temperature variation. Imperfect code timing, i.e., synchronization error, results in a reduction of the signal to noise ratio in the waveform processor, and the generation of a noise-like component in the reference signal -- even when only desired signal is present at the array output. When interfering signals are present in the array output, they are reduced in strength and altered considerably in waveform and frequency structure by the processing operations. However, they are not eliminated completely from the reference signal. Fortunately, complete suppression of interference in the reference signal is not a prerequisite for satisfactory interference rejection by the array. The essential requirement is that the interference components in the reference signal must be smaller than the interference components in the array output signal, i.e., the effective gain of the reference waveform processor with respect to interference must be less than unity. When this condition is satisfied, the error signal contains an interference component equal to the difference in interference levels, and the feedback processing responds to this error to reduce the output interference. Reduction of the output interference continues as long as the interference component is present in the error signal, i.e., as long as interference is present in the array output and the gain of the waveform processor to interference is less than unity. The latter condition is expected to prevail even for modest values of waveform processing gain. Thus, the array processor is potentially capable of eliminating large interfering signals even when the waveform processing gain is relatively small.

The effects of waveform processing delay, synchronization error, cw interference, and system parameters on the processor's transient response were investigated by digital computer simulation. The simulation is described in the following section.

SECTION III
COMPUTER SIMULATION OF THE ADAPTIVE ARRAY/WAVEFORM PROCESSOR

A. Simulation of the Weighting Coefficient Loop Equations

A computer program was written to solve the set of 2N differential equations (4) for the weighting coefficients. In sampled form these equations are approximated by

$$(6) \qquad \frac{dw_i(t_j)}{dt} = \frac{w_i(t_j+T) - w_i(t_j)}{T} = K\, E(t_j)\, y_i(t_j)$$

or

$$(7) \quad w_i(t_{j+1}) = w_i(t_j) + KT\, E(t_j)\, y_i(t_j)\ ;$$

$$i = 1, 2, \cdots, 2N,$$

where

$$(8) \quad E(t_j) = R(t_j) - S(t_j) = R(t_j) - \sum_{i=1}^{2N} w_i(t_j) y_i(t_j)$$

is the sampled error signal and

$$(9) \qquad T = \frac{1}{F} = t_{j+1} - t_j$$

is the sampling period corresponding to a sampling frequency F. The time reference $t_o = 0$ was assumed so that

$$(10) \qquad t_j = jT \qquad j = 0,1,2,\cdots .$$

The $y_i(t_j)$ are the sampled signals (including interference) at the inputs to the weighting coefficients. The computer program performed the following sequence of computations given the values of the weighting coefficients $w_i(t_j)$ at time t_j:

a) Computation of the signal samples $y_i(t_j)$, $i=1,\cdots,2N$ from appropriate analytical expressions,

b) Computation of the array output signal sample $S(t_j)$ from the $w_i(t_j)$ and $y_i(t_j)$,

c) Computation of the reference signal sample $R(t_j)$ from $S(t_j)$ by means of appropriate digital processing used to simulate the reference generation network.

d) Computation of the error signal sample $E(t_j)$ from (8), and

e) Computation of subsequent values for the weighting coefficients $w_i(t_{j+1})$ from the recursion equations (7).

The signal equations $y_i(t)$ used in step (a) and the digital processing used in step (c) are discussed in the following two sections.

B. Input Signal Simulation

A linear (one-dimensional) array composed of an even number N of equispaced elements was assumed in formulating the equations for signals $y_i(t)$ at the weighting coefficient inputs. The elements and weighting coefficients were numbered as shown in Fig. 4 with odd-numbered elements to the right of the array phase center and even-numbered elements to the left.

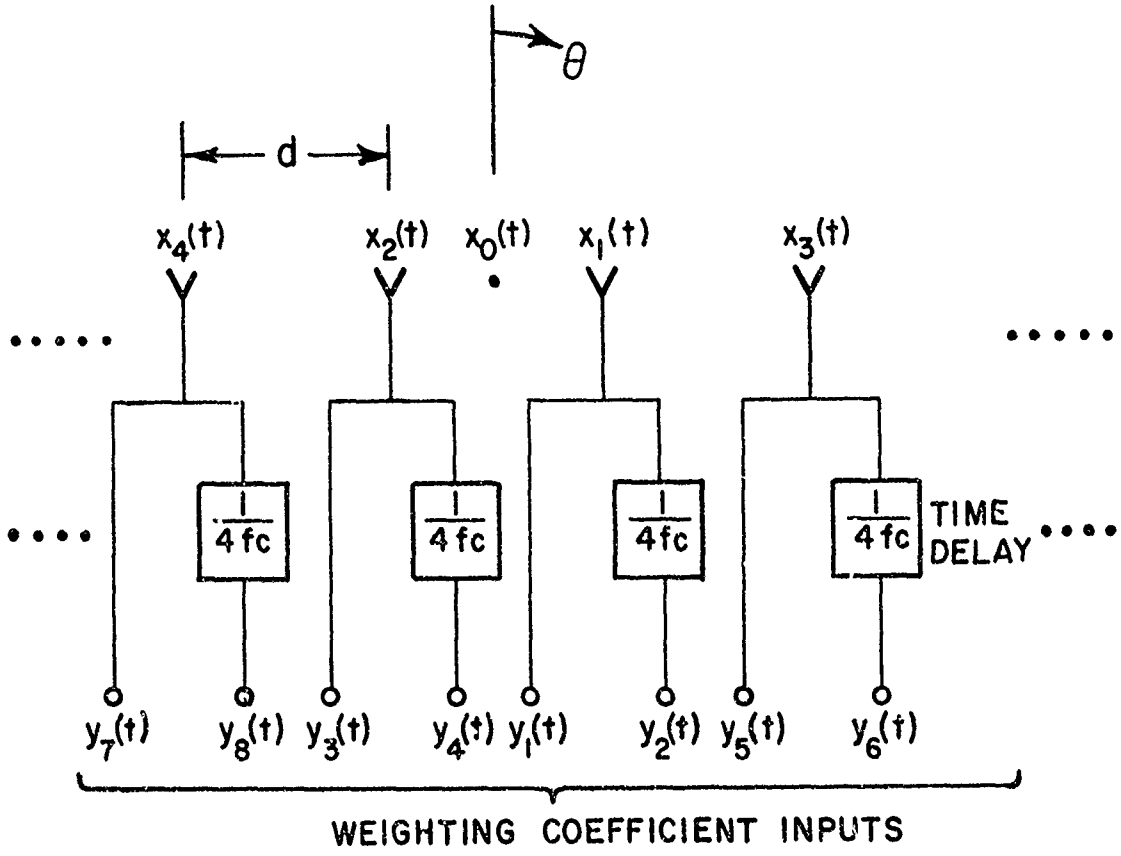


Fig. 4. Linear array geometry.

Signal $x_o(t)$ which would be received by an element placed at the array phase center was assumed to be given by

$$(11) \qquad x_o(t) = s_o(t) + n_o(t),$$

with

$$(12) \qquad s_o(t) = A_s\, C(t)\, D(t) \sin\{2\pi(f_c + \Delta f_s)t + \alpha_{so}\}$$

$$(13) \qquad n_o(t) = A_n \sin\{2\pi(f_c + \Delta f_n)t + \alpha_{no}\}$$

where

A_s = desired signal amplitude

A_n = interference signal amplitude

f_c = array center frequency

Δf_s = desired-signal frequency offset

Δf_n = interference-signal frequency offset

α_{so} = desired-signal initial (t=0) phase

α_{no} = interference-signal initial phase

$C(t)$ = desired-signal bi-phase code (+1, -1 amplitude) modulation

$D(t)$ = desired-signal bi-phase data (+1, -1 amplitude) modulation.

The signal received by the kth element, $x_k(t)$, was expressed as

$$(14) \qquad x_k(t) = s_k(t) + n_k(t)$$

$$= \begin{cases} s_o(t + \frac{k}{2}\frac{d}{c}\sin\theta_s) + n_o(t + \frac{k}{2}\frac{d}{c}\sin\theta_n) \\ \qquad\qquad k = 1, 3, \cdots, N-1 \\ s_o(t - \frac{k-1}{2}\frac{d}{c}\sin\theta_s) + n_o(t - \frac{k-1}{2}\frac{d}{c}\sin\theta_n) \\ \qquad\qquad k = 2, 4, \cdots, N. \end{cases}$$

where

$\frac{d}{c}$ = propagation time delay between elements

θ_s = desired-signal arrival angle

θ_n = interfering-signal arrival angle.

The propagation time delay across the array was assumed negligibly small compared to the code and data bit periods of the desired signal modulation, i.e.,

$$(15) \qquad N \frac{d}{c} << f_{cd}^{-1} < f_d^{-1}$$

where

f_{cd} = desired-signal code bit rate

f_d = desired-signal data bit rate.

This assumption of an "array bandwidth" much larger than the input signal modulation bandwidth justified the approximations

$$(16) \qquad C(t \pm \frac{k}{2} \frac{d}{c} \sin \theta_s) \doteq C(t),$$

$$(17) \qquad D(t \pm \frac{k}{2} \frac{d}{c} \sin \theta_s) \doteq D(t).$$

The array elements were assumed to be ideal (noise-free) point receptors. Accordingly, the output signal from the kth element, $y_{2k-1}(t)$, was expressed as

$$(18) \qquad y_{2k-1}(t) =$$

$$A_s C(t)D(t) \sin \{2\pi\left(1 + \frac{\Delta f_s}{f_c}\right) f_c t + k\pi d_{\lambda_c} \left(1 + \frac{\Delta f_s}{f_c}\right) \sin \theta_s + \alpha_{so}\}$$

$$+ A_n \sin \{2\pi\left(1 + \frac{\Delta f_n}{f_c}\right) f_c t + k\pi d_{\lambda_c} \left(1 + \frac{\Delta f_n}{f_c}\right) \sin \theta_n + \alpha_{no}\}$$

when $k = 1,3,5,\cdots, N-1$

and

$$(19) \qquad y_{2k-1}(t) =$$

$$A_s C(t)D(t) \sin \left\{2\pi \left(1 + \frac{\Delta f_s}{f_c}\right) f_c t - (k-1)\pi \, d_{\lambda_c} \left(1 + \frac{\Delta f_s}{f_c}\right) \sin \theta_s + \alpha_{so}\right\}$$

$$+ A_n \sin \left\{2\pi \left(1 + \frac{\Delta f_n}{f_c}\right) f_c t - (k-i)\pi \, d_{\lambda_c} \left(1 + \frac{\Delta f_n}{f_c}\right) \sin \theta_n + \alpha_{no}\right\}$$

when $\quad k = 2,4,\cdots,N$

where $\quad d_{\lambda_c} = \frac{d}{\lambda_c} = \left(\frac{d}{c}\right) f_c$

is the element spacing in wavelengths at the array center frequency. The quadrature channel inputs, $y_{2k}(t)$, were assumed to be given by expressions identical to (18) and (19) except for the addition of quadrature phase shifts to the sine arguments of the desired and interfering signal terms.*

The desired signal code modulation C(t) consisted of a length-127 pseudo random sequence generated by a seven-stage feedback shift register having the configuration shown in Fig. 5. The initial contents of the register were chosen to provide a representative switching pattern** over the first twenty bits of the sequence.

The data modulation, D(t), was assumed to consist of an alternating (+1, -1) sequence. This "square-wave" pattern maximizes the number of bi-phase data transitions which occur during the initial transient period thus allowing the effects of data modulation delay in the reference processing network to be readily observed.

---

*The simulation program actually introduced time delays of $1/4f_c$ seconds which gave rise to phase shifts of $90^0(1 + \Delta f/f_c)$ where Δf is the input signal frequency offset from center frequency. However, the frequency offsets employed in the simulation were small in comparison with f_c so that the phase shifts were approximately equal to 90^0.

**Such a pattern is considered to have approximately equal occurrence of "1"s and "0"s and no long consecutive runs of either symbol.

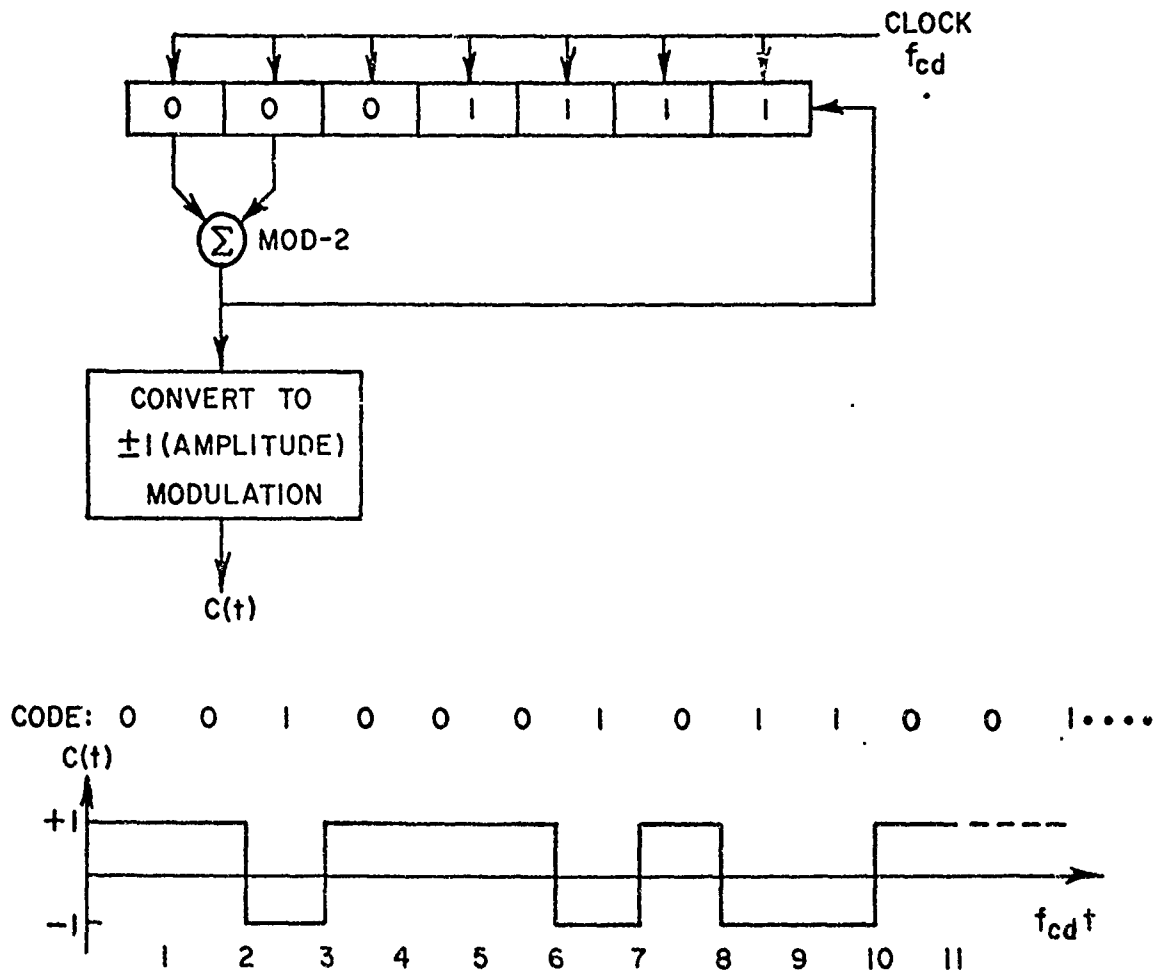


Fig. 5. Pseudorandom code modulation generation.

C. Simulation of the Reference Network Signal Processing

The digital processing used to simulate the reference network is illustrated in Fig. 6. Mixer simulation consisted of multiplication of the input samples by +1 or -1 depending on the value of the reference code $C_r(t_j)$. The up/down frequency conversions performed by the mixers were omitted for simplicity. The reference code was a delayed version of the input code

$$C_r(t_j) = C(t_j - \varepsilon) \tag{20}$$

where delay parameter ε could be varied to simulate synchronization offsets ε having values from zero to one code bit period ($\varepsilon_{max} \cdot f_{cd} = 1$). Lerner, digital bandpass filters were chosen to simulate the analog bandpass filters in the reference network because of their excellent phase linearity and reasonably selective passbands. The fourth-order Lerner filters used in the simulation are characterized by the z-transform

$$H(z) = \sum_{i=1}^{4} \frac{a_i \left[1 - \left(e^{-\pi/2\ BT} \cos b_i T\right) Z^{-1}\right]}{1 - \left(2e^{-\pi/2\ BT} \cos b_i T\right) Z^{-1} + e^{-\pi BT} Z^{-2}} \tag{21}$$

where $a_1 = -a_4 = 0.5$

$-a_2 = a_3 = 1$

$b_{1,4} = 2\pi \left(f_c \mp \frac{B}{2}\right)$

$b_{2,3} = 2\pi \left(f_c \mp \frac{B}{4}\right)$

B = Bandwidth, Hz.

The above transfer function was obtained by using the criteria of impulse invariance: the impulse response of the digital filter equals the sampled impulse response of the Lerner analog filter. Details of the design procedure are available in Reference 7. Figure 7 shows the pole-zero configurations of the Lerner filter in both the analog (s-plane) and digital (z-plane) domains. The sampled output of the filter at time $t = nT$ is given by a sum of four sub-outputs:

$$V(nT) = \sum_{i=1}^{4} a_i V_i(nT), \tag{22}$$

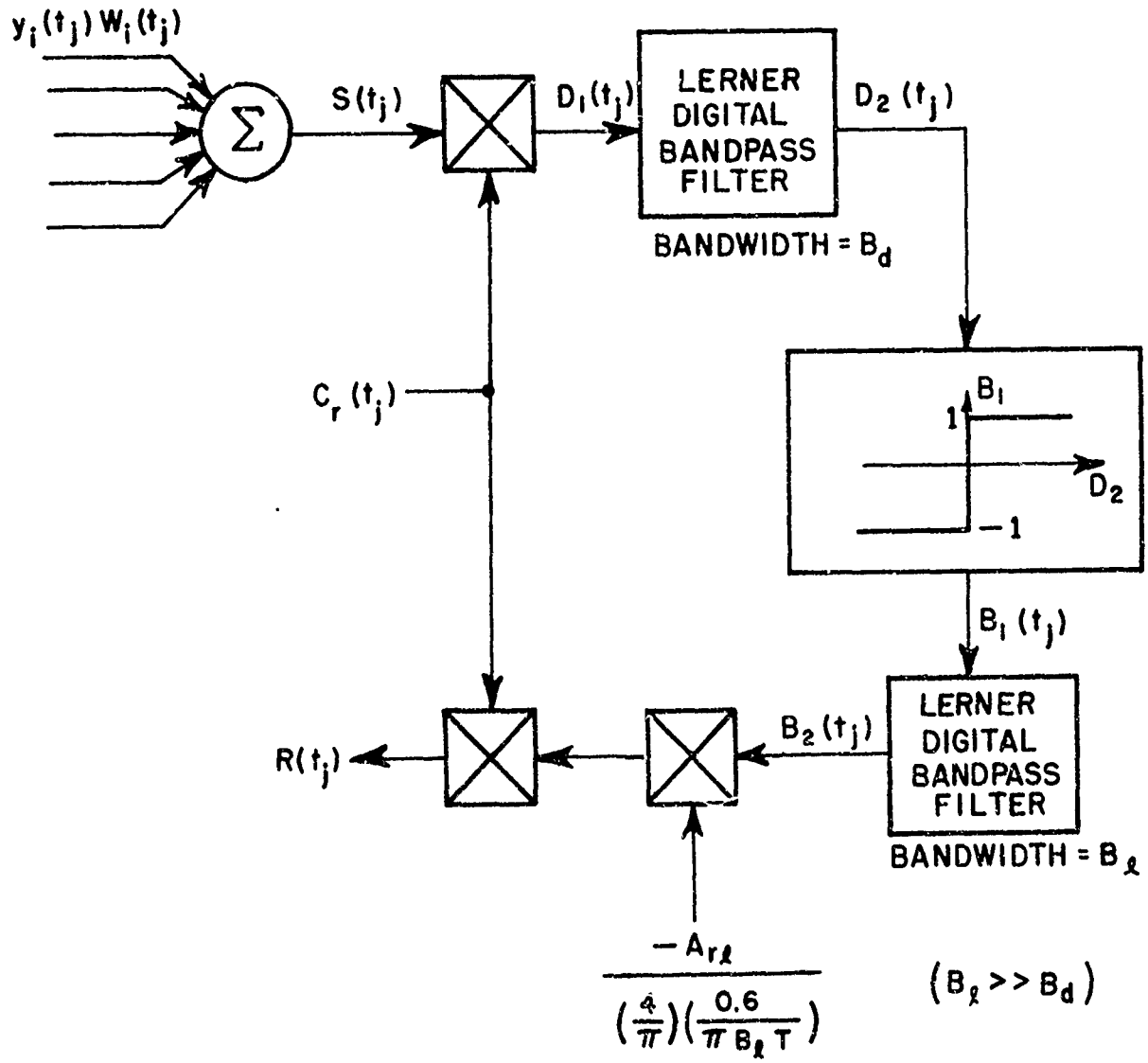
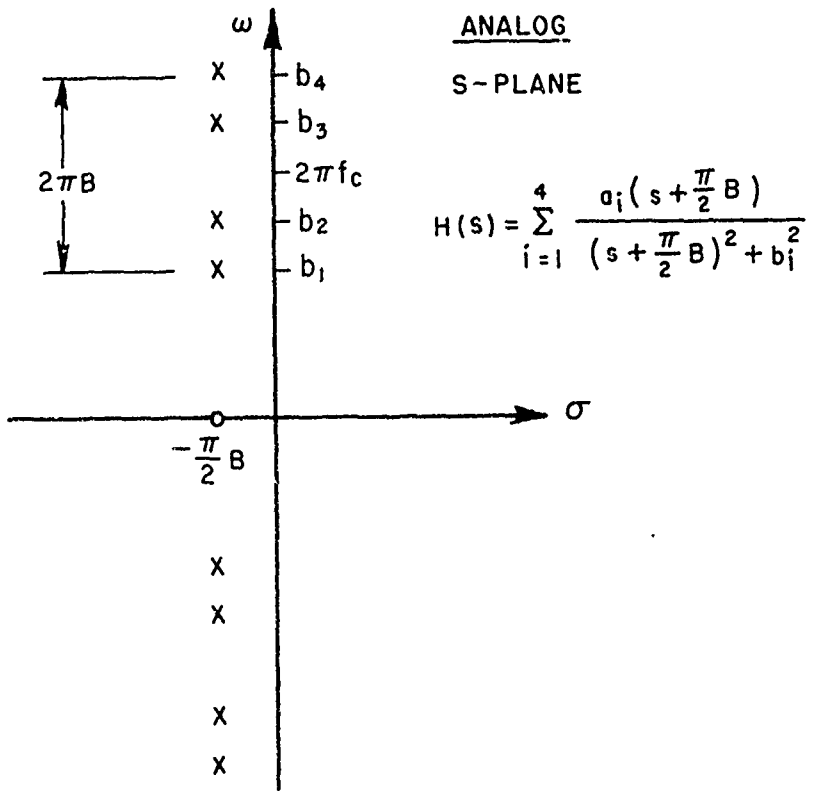
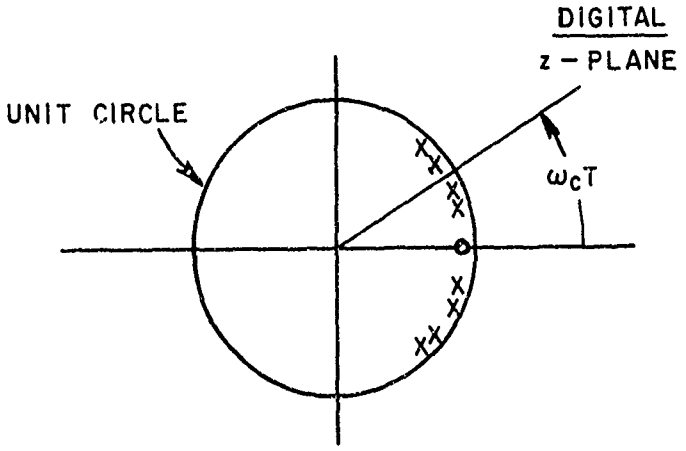


Fig. 6. Reference network digital processing.





$$H(z) = \frac{a_i\left[1 - e^{-\frac{\pi}{2}BT}\cos b_i T \;\; z^{-1}\right]}{1 - 2e^{-\frac{\pi}{2}BT}\cos b_i T \;\; z^{-1} + e^{-\pi BT} z^{-2}}$$

Fig. 7. Pole-zero configuration of 4th order Lerner filter.

where the sub-outputs are expressed by

$$(23) \qquad V_i(nT) = e^{-\pi/2\,BT} \cos b_1 T\,[2V_i(nT-T) - U(nT-T)]$$

$$-e^{-\pi BT}\,V_i(nT-2T) + U(nT).$$

Function U(nT) represents the sampled input at time t = nT. The frequency response of the Lerner filter was calculated from (21) with $z = \exp(j\omega T)$ and is shown in Fig. 8. Important characteristics to be noted are the ninety degree phase shift at center frequency and the change in phase shift over the bandwidth B of nominally 360 degrees. At the bandedge, i.e., at $f = f_c \pm B/2$, the amplitude response is 5.32 dB smaller than the mid-band response and the phase shift differs 12.5 degrees from linear. Envelopes of the impulse response, the step amplitude response, and the response to a 180^0 step in the carrier phase are illustrated in Figs. 9, 10, and 11, respectively. The latter figure shows that the transition in the output signal's phase occurs approximately 1/B seconds after the phase step is applied to the input signal.

The bandpass limiter in the reference network was simulated by an ideal limiter characteristic followed by a Lerner digital filter having a passband centered on the carrier frequency and a bandwidth B_ℓ much larger than the data filter bandwidth B_d. The purpose of this second filter was to reject frequency components outside the first zone from the output waveform. The changes in amplitude experienced by the signal at the data filter's output following a 180^0 phase step due to the data modulation (see Fig. 11) are "squared-up" by the limiter and the zonal filter introduces a relatively small additional delay of $1/B_\ell$ seconds. The zonal filter's output is multiplied by a constant to result in a limited reference signal having the desired value: A_{r_ℓ}. A negative constant is employed for the scaling factor so that the two ninety degree phase advances introduced by the digital filters (at the center frequency) are effectively cancelled. This forces the desired signal components at the waveform processor and array outputs to be in phase alignment when the frequency offset equals zero. When the frequency offset, $f-f_c$, is sufficiently small so that the filter phase characteristics are approximately linear, the total phase shift through the reference network is given approximately by

$$(24) \qquad \gamma(f) \doteq -2\pi\left(\frac{1}{B_d} + \frac{1}{B_\ell}\right)(f-f_c) \doteq \frac{-2\pi(f-f_c)}{B_{composite}} \quad .$$

The scaled output of the zonal filter is multiplied by the reference code to simulate the second mixer. This completes the digital simulation of the waveform processing network.

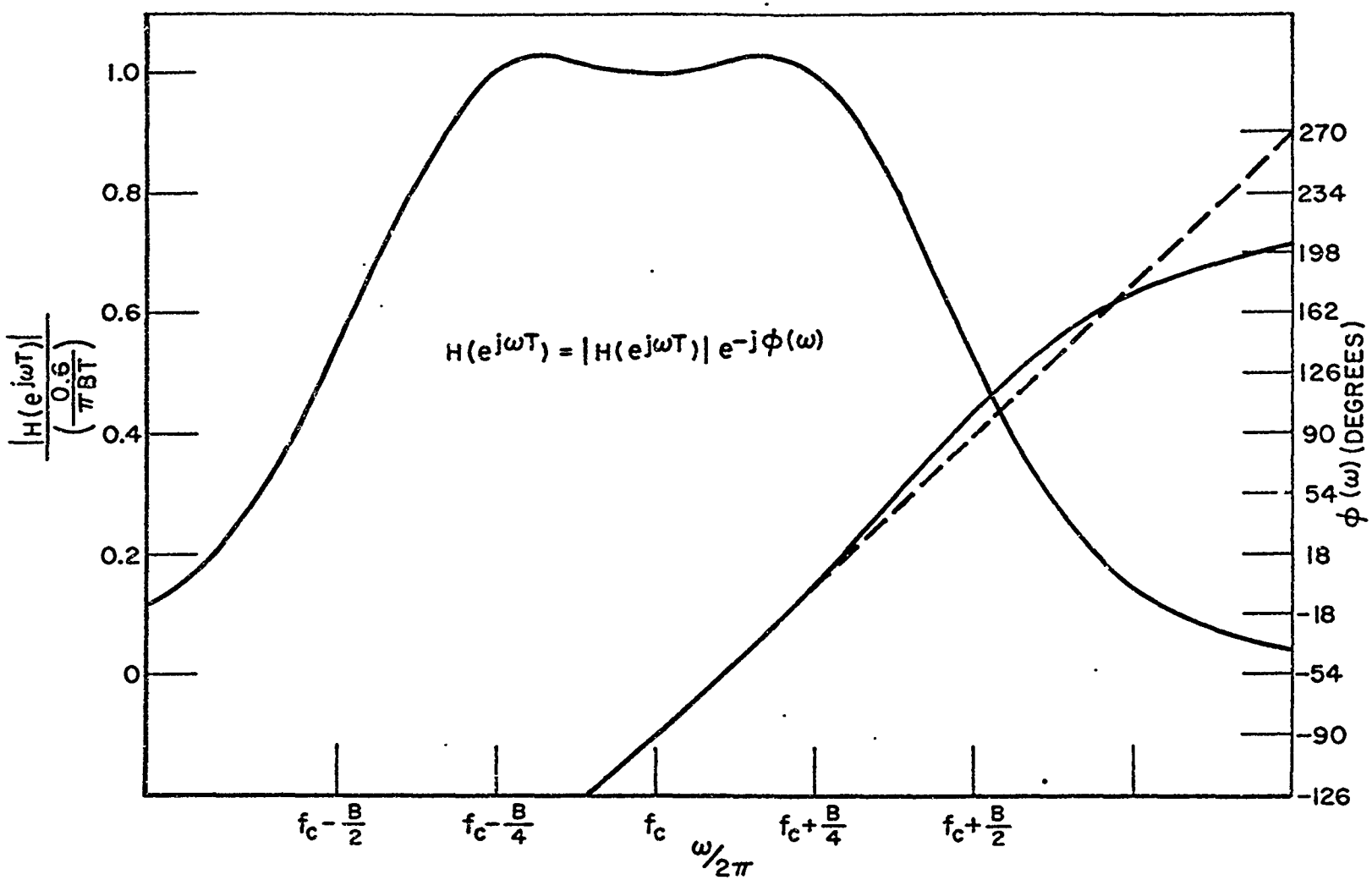


Fig. 8. Lerner digital filter frequency response.

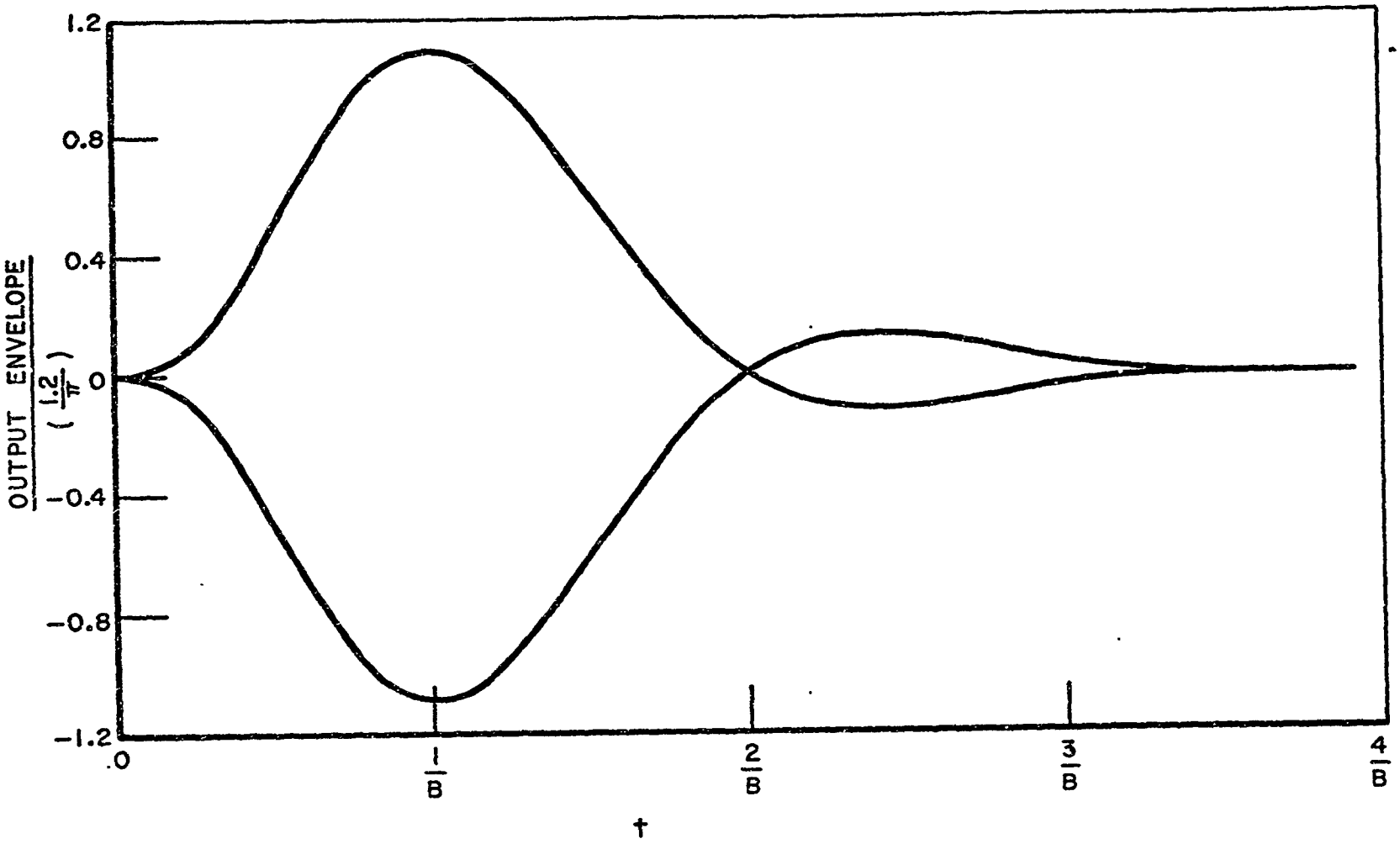


Fig. 9. Lerner digital filter unit impulse response.

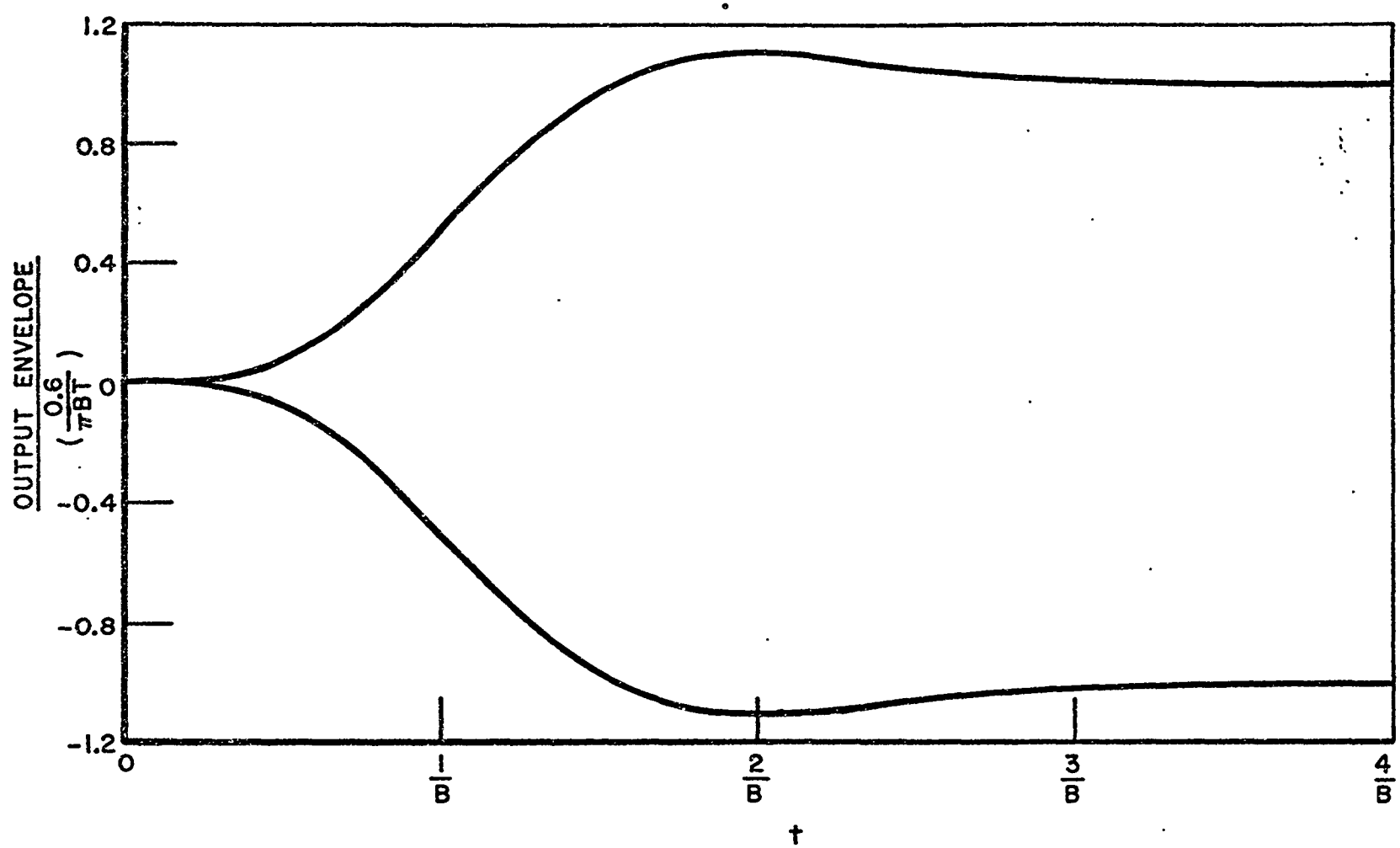


Fig. 10. Lerner digital filter unit RF step response.

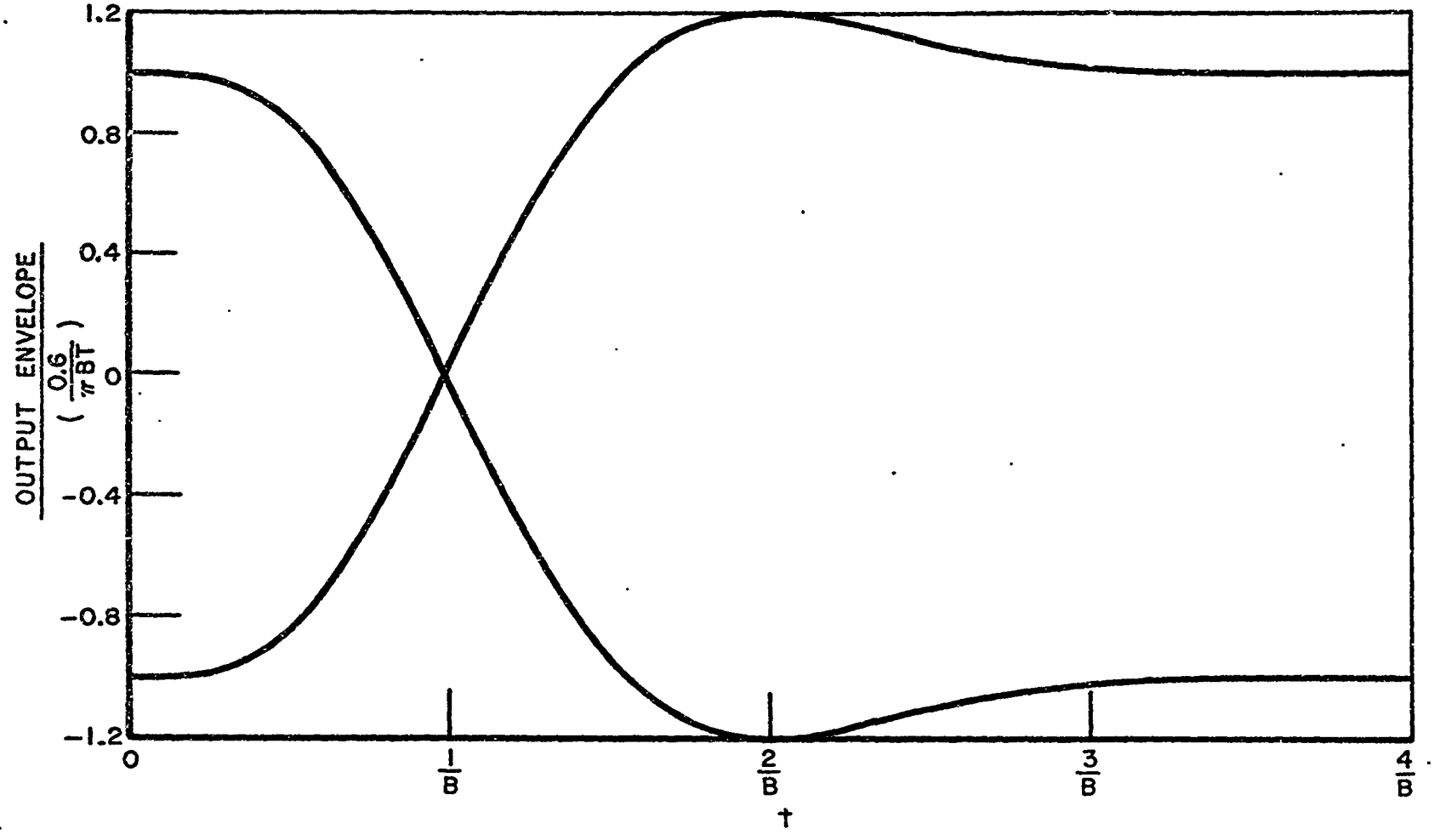


Fig. 11. Lerner digital filter bi-phase step response.

D. Sampling Rate Considerations

The selection of a sampling rate, F, was influenced by several factors. Adequate representation of the input (bandpass) signals in sampled form required that the sampling rate exceed $2f_c$: the Nyquist sampling criterion. Another factor to consider was the cumulative truncation errors resulting from the approximate integration in (7) of the weighting coefficient differential equations. These errors cause the computer solutions to differ from the correct analog solutions. They are dependent upon the rate-of-change of the weighting coefficients and therefore, from (6), are related to input signal magnitude. Approximation errors are reduced by increasing the sampling rate. Another factor arose from the limiter nonlinearity in the reference network which produced frequency components in zones centered about odd harmonics of f_c. The frequency response of the second digital filter is periodic, as illustrated in Fig. 12, with bandpass repetitions centered on the frequencies

$$f = nF \pm f_c \tag{25}$$

where n is any integer. To ensure rejection of frequency components outside the fundamental zone (f_c), i.e., to minimize frequency aliasing effects in the second digital filter, the sampling rate was chosen as an odd multiple of f_c as in Fig. 12. The responses to components having frequencies around twice the sampling frequency (2F) are expected to be negligibly small when the sampling rate is sufficiently large. The combination of the limiter non-linearity and a finite sampling rate also resulted in reference signal phase quantization as will be discussed in the next section. All of these factors suggest that the sampling rate should be made much larger than f_c.

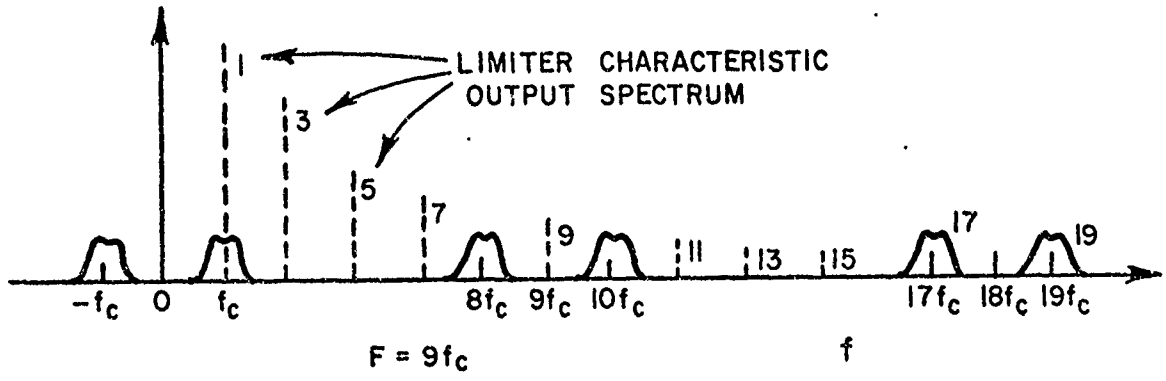


Fig. 12. Second digital filter input spectrum and periodic frequency response characteristics.

In practice, the sampling rate which can be utilized is restricted by the length of time required to execute the computer program. In the study, the transient decay times encountered were generally on the order of a few thousand carrier cycles for realistic values of the parameters. Program run time considerations limited the total number of samples processed to a few ten thousand. As a result, it was necessary to restrict the sampling rate to nine times the carrier frequency: $F = 9f_c$.

E. Reference-Signal Phase Quantization

The phase of the sampled reference signal was restricted to a discrete set of values as a result of using a finite sampling rate and an ideal limiter characteristic. This phase quantization is perhaps best illustrated by an example. Figure 13 illustrates the sampled waveforms at various points in the reference processing network of Fig. 6 when the array output signal is given by

$$S(t_j) = \sin(\omega_c t_j + 0^0). \tag{26}$$

The ninety degree phase advance in the data filter output waveform (b) results in a limiter output waveform (c) composed of five positive and four negative samples per cycle. The fundamental component (d) of this sampled square wave at the second filter output is changed in sign and amplitude-scaled in (e) to form the sampled reference waveform. Shifting the phase of the input sinusoid in (a) by any amount up to plus or minus ten degrees changes the values, but not the signs, of the samples in (b). Therefore, the sampled waveforms in (c), (d), and (e) are unchanged. The reference signal phase is constant at zero degrees for this case. For input phase angles between ten and thirty degrees, the polarity of one sample (per cycle) will change in (b) leading to a four positive, five negative repetitive sample "pattern" in (c) and a reference phase of 20^0. More generally, when the phase of the input signal lies in the interval $[(n \cdot 20^0 - 10^0), (n \cdot 20^0 + 10^0)]$ where n is an integer or zero, the phase of the reference signal equals $n \cdot 20^0$. Note that the five positive, four negative sample pattern in (c) or its negative - a five negative, four positive pattern - is the only sample pattern possible when the input to the reference generation network is a sinusoid having a frequency f_c. All discrete values of the reference signal's phase are obtained from the response shown in (d) and its negative by shifting the time origin in increments equal to the sampling interval. The sample values listed below (e) apply for a unit amplitude reference signal and were obtained from simulation results.

When the array output signal is a sinusoid and its frequency is offset from the center frequency, the phase of the limited signal (c) at the second filter's input increments in discrete 20^0 jumps forming a stairstep pattern in time. The second filter responds very rapidly to these phase steps -- on the order of six carrier cycles -- due to this filter's wide bandwidth. As a result, the phase of the reference signal is essentially a stairstep function of time also.

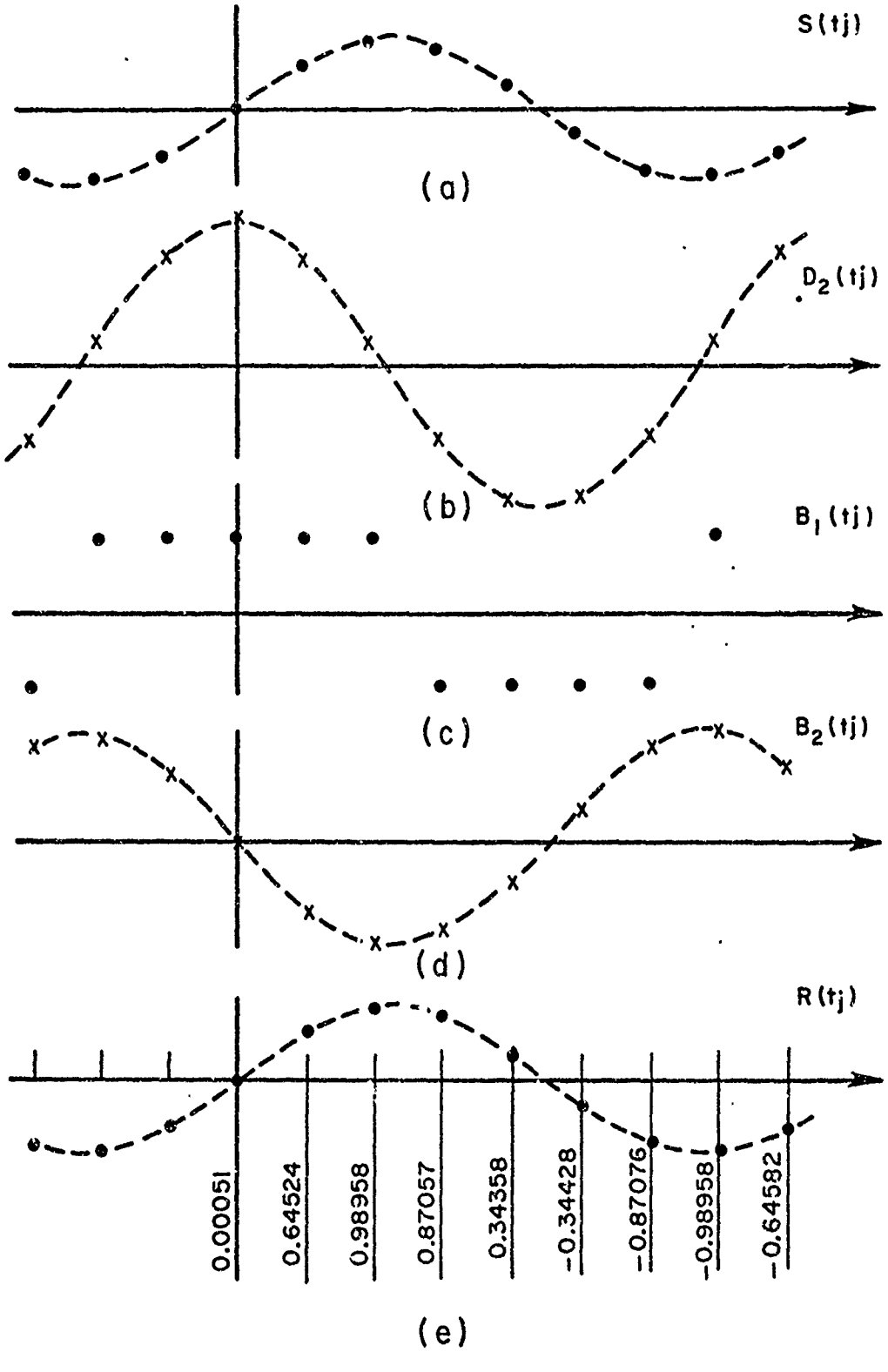


Fig. 13. Reference network sampled-data waveforms.

F. Performance Measure: CW Pattern

The performance measure selected for evaluating the simulation results was the cw power pattern of the array. The power pattern was computed at selected instants of time during adaption from the set of weighting coefficients which occured at those instants. The power pattern at a frequency $f_c + \Delta f$ was defined as the array output response (magnitude squared in dB and phase) to a unit amplitude cw signal of frequency offset Δf and variable arrival angle θ:

$$(27)\quad \overline{P}_\Sigma(\theta,\Delta f) = 10 \log_{10}|\overline{V}_\Sigma(\theta,\Delta f)|^2 \exp[j \text{ Arg } \overline{V}_\Sigma(\theta,\Delta f)]$$

where $\overline{P}_\Sigma(\theta,\Delta f)$ = cw power pattern phasor

$\overline{V}_\Sigma(\theta,\Delta f)$ = array output (voltage) phasor.

The expression for the array output phasor in terms of the set of weighting coefficients is given in Appendix I. The computer simulation program is listed in Appendix II; subroutine REFRN performs the reference network digital processing.

SECTION IV
COMPUTER SIMULATION RESULTS

A. Introduction

The simulation parameters described in the previous section were assigned values as shown in Table I. The parameters are listed in columns; each line in the table represents a different simulation run. The parameters listed under the heading "Modifications" apply to modified computer simulation models and will be discussed in context as the modifications are introduced. All parameters having dimensions of frequency (or time) are normalized to the array center frequency f_c (or its reciprocal T_c). In all cases the weighting coefficients were initially set to unity to establish a broadside pattern. Because of processing time restrictions, it was necessary to limit the simulation to the case of a two-element array. The important response characteristics - especially those attributed to the reference waveform processor - do not appear to be dependent on the number of array elements. These response characteristics are presented in the following sections.

B. Effect of Reference Network Time Delay

When the code and data rates are fixed, the selection of the data filter bandwidth involves a tradeoff between the processing gain and the time delay introduced by the reference network. Figure 14 shows the effect of time delay when the data filter has a bandwidth equal to four times the data rate. The pattern magnitude in the desired signal direction (0^o) and in the interference direction (40^o) versus adaption time in carrier periods (T_c) are illustrated. The time delay results in a 180^o phase difference between the desired components in the array output and reference signals for the first fifty carrier periods (50 T_c) of each data bit period (200 T_c) since the data is an alternating (worst-case) sequence. After the initial adaptive transient reduces the desired signal's output level to the correct value (0 dB pattern magnitude), the character of the signal response changes. The feedback system attempts to reverse the sign of the weighting coefficients during the delay error periods as shown in Fig. 15. As the weights decrease in the process of sign reversal the pattern magnitude in the signal direction decreases. During the remaining three-quarters of each data bit the phasing is correct and the feedback system attempts to recover the gain loss in the signal direction. The pattern gain variations in the desired signal direction can be reduced by decreasing the gain (bandwidth) of the feedback loops as shown in Figs. 16 and 17. The loop gain factor K has been reduced by a factor of five here. If the loop gain is reduced even further to the point where the response time of the loops is much larger than the data bit period, then the output signal amplitude is essentially constant in the steady-state. The steady-state amplitude can be calculated as shown in Appendix III as the value which minimizes the mean-square error.

TABLE I. Simulation Parameter Assignments

Figure Reference	d_{λ_c}	θ_s	θ_n	A_s	A_n	Δf_s	Δf_n	f_{cd}	$\varepsilon \cdot f_{cd}$	f_d	K	Modifications: T_h	Modifications: τ_{agc}	Modifications: Nonlinear Error Processing
14,15	0.5	0	40^0	1	1	0	0	0.05 f_c	0	0.005 f_c	0.0025 f_c			
16,17	0.5	0	40	1	1	0	0	0.05	0	0.005	0.0005			
18	0.5	0	40	1	1	0	0	0.05	0	0.005	0.0025	60 T_c		
19	0.5	0	40	1	1	0	0	0.05	0.25	0.005	0.0025	60		
19	0.5	0	40	1	1	0	0	0.05	0.5	0.005	0.0025	60		
21,22	0.5	0		1	0	0.0025 f_c		0.05	0	0.0005	0.01	100		
23	0.5	0	40	1	1	0	0	0.05	0	0.005	0.0025	60		
23	0.5	0	40	1	1	0.0005	0	0.05	0	0.005	0.0025	60		
23	0.5	0	40	1	1	0.0025	0	0.05	0	0.005	0.0025	60		
24	0.5	0	40	1	1	0	0	0.01	0	0.005	0.0025	60		
24	0.5	0	40	1	1	0	0	0.05	0	0.005	0.0025	60		
24	0.5	0	40	1	1	0	0	0.20	0	0.005	0.0025	60		
25,26	10	2.85	0	1	10	0	0	0.05	0	0.005	0.0005	60		
28	10	2.85	0	1	100	0	0	0.05	0	0.005	0.0005	60		
	10	2.85	0	1	100	0	0.01 f_c	0.05	0	0.005	0.0005	60		
29	0.5	0	40	1	100	0	0	0.05	0	0.005	0.0005	60		
30	0.5	0	40	1	100	0	0	0.05	0	0.005	0.0005	60	250 T_c	
32	0.5	0	40	1	100	0	0	0.05	0	0.005	0.001	60		*

Fixed Parameters:

$N = 2$ $\quad A_{r_\ell} = 1$ $\quad \alpha_{so} = \alpha_{no} = 0^0$ $\quad B_d = 0.02\ f_c$ $\quad B_\ell = 0.5\ f_c$

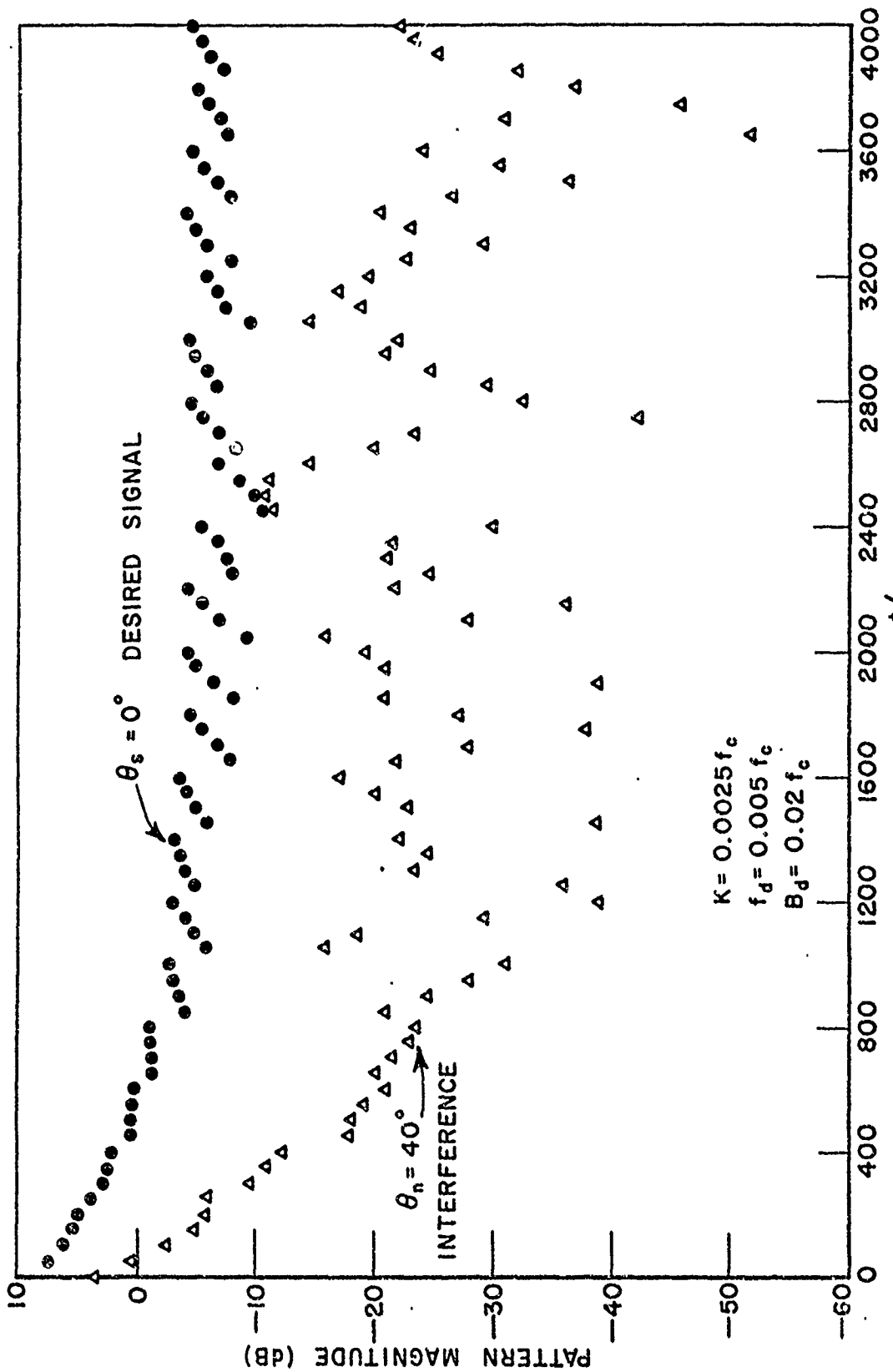


Fig. 14. The effect of data modulation time delay in the reference network.

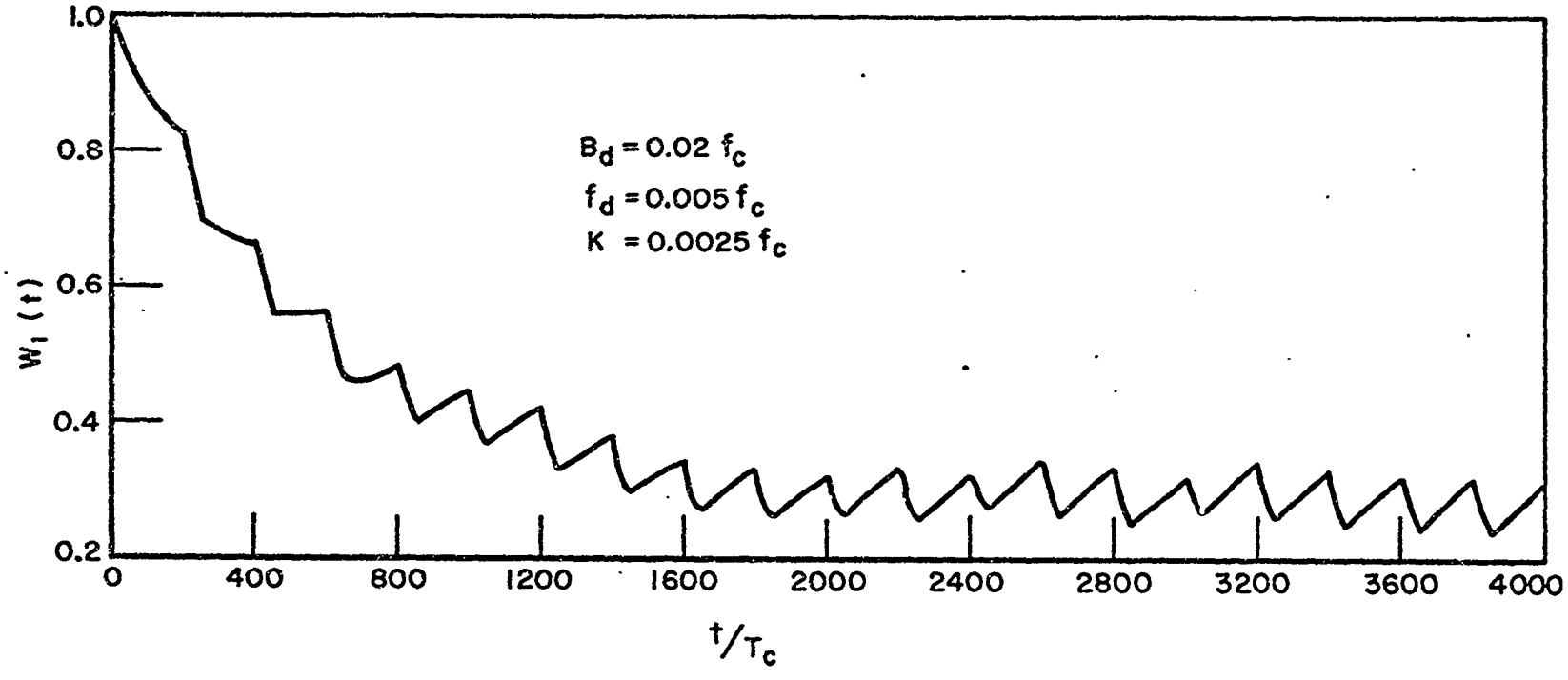


Fig. 15. Weighting coefficient response for result in Fig. 14.

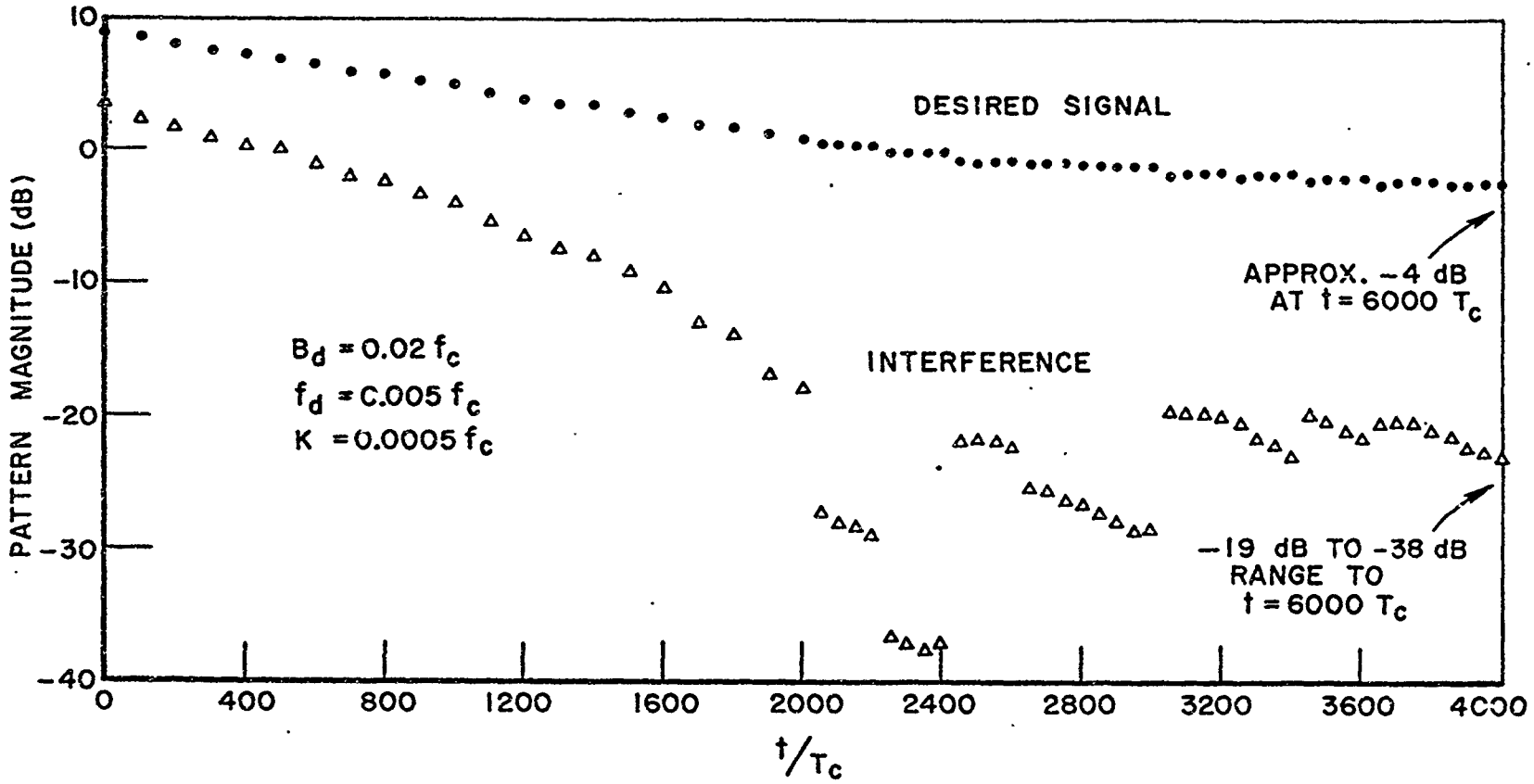


Fig. 16. The effect of data modulation time delay in the reference network.

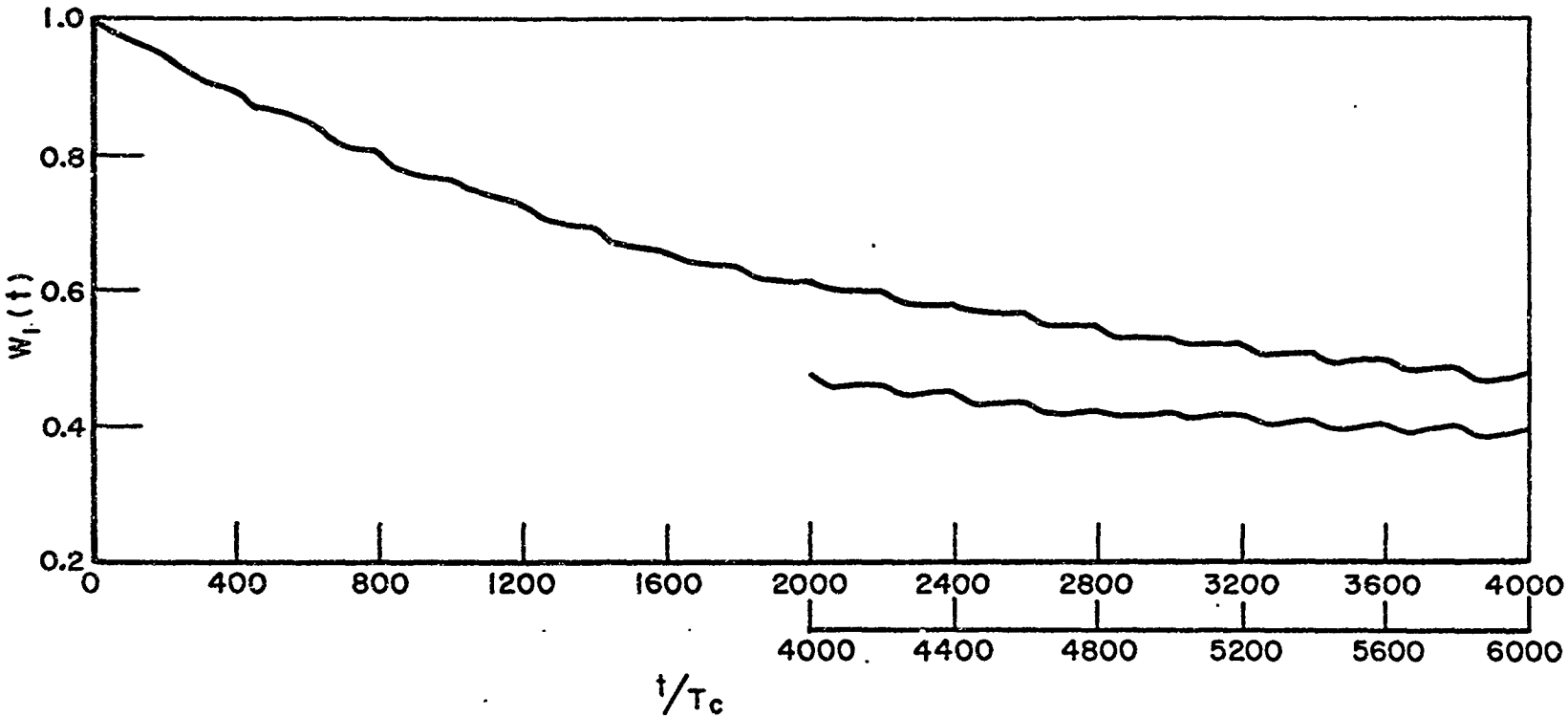


Fig. 17. Weighting coefficient response for result in Fig. 16.

This steady-state amplitude becomes smaller as the percentage of time when bi-phase errors are present increases, i.e. as the reference processor time delay is increased. For delays of one-eighth, two-eighths, and three-eighths the data bit period, the suppression of the desired output signal below the constant amplitude (0 dB) reference signal is 2.5 dB, 6 dB, and 12 dB respectively. These figures apply for the worst-case (square-wave) data sequence. For a random data sequence the rate of occurence of bi-phase errors is less by a factor of two and the corresponding values of suppression are 1.16 dB, 2.5 dB, and 4.1 dB, respectively. Pattern adaption in the interference direction in Figs. 14 and 16 is very erratic as a result of the corrections caused by the desired signal.

It is possible to equalize the reference processor delay by employing wideband delay lines to delay 1) the signal inputs to the error by signal multipliers, 2) the array output signal applied to the subtractor which forms the error signal, and 3) the local oscillator signal at the input to the second mixer in the reference processor (or the corresponding PN sequence could be delayed). This approach allows other types of data modulation to be accommodated, however, it complicates the task of maintaining proper phasing conditions in the feedback loops of the processor. An alternative to equalization which is applicable for the case of bi-phase data modulation is simply to inhibit processing (i.e., hold the weights constant) during periods when the error signal is incorrect. Figure 18 illustrates the effect of holding the weights constant for the first three-tenths of each data bit period ($T_h = 60\ T_c$). This was accomplished in the simulation program by setting the error signal samples equal to zero at the appropriate times:

$$(28)\quad E(t) = \begin{cases} 0 & ;\ nf_d^{-1} < t < nf_d^{-1} + 60\ T_c \\ & \qquad n = 0,1,2,\cdots \\ R(t)-S(t); & \text{elsewhere,} \end{cases}$$

where f_d is the data rate. The pattern adapts smoothly in the desired signal direction to the proper magnitude and the rejection of interference is considerably improved compared to the result shown in Fig. 14.

C. Effects of Code Synchronization Error

When the received signal code in S(t) and the reference code are not properly aligned in time, the signal at the output of the first mixer in the reference network is distorted. For small timing errors, the smoothed signal at the data filter output may be represented by two component signals: a data-carrying signal identical to that obtained when no timing errors are present except for a reduction in amplitude and a low-level, noiselike signal resulting from the distortion components of the codes. The bandpass limiter removes the amplitude fluctuations of the composite signal; the phase fluctuations due the noiselike component, however, are retained at the limiter output and therefore appear on the reference signal. In the computer simulation, the phase of the reference

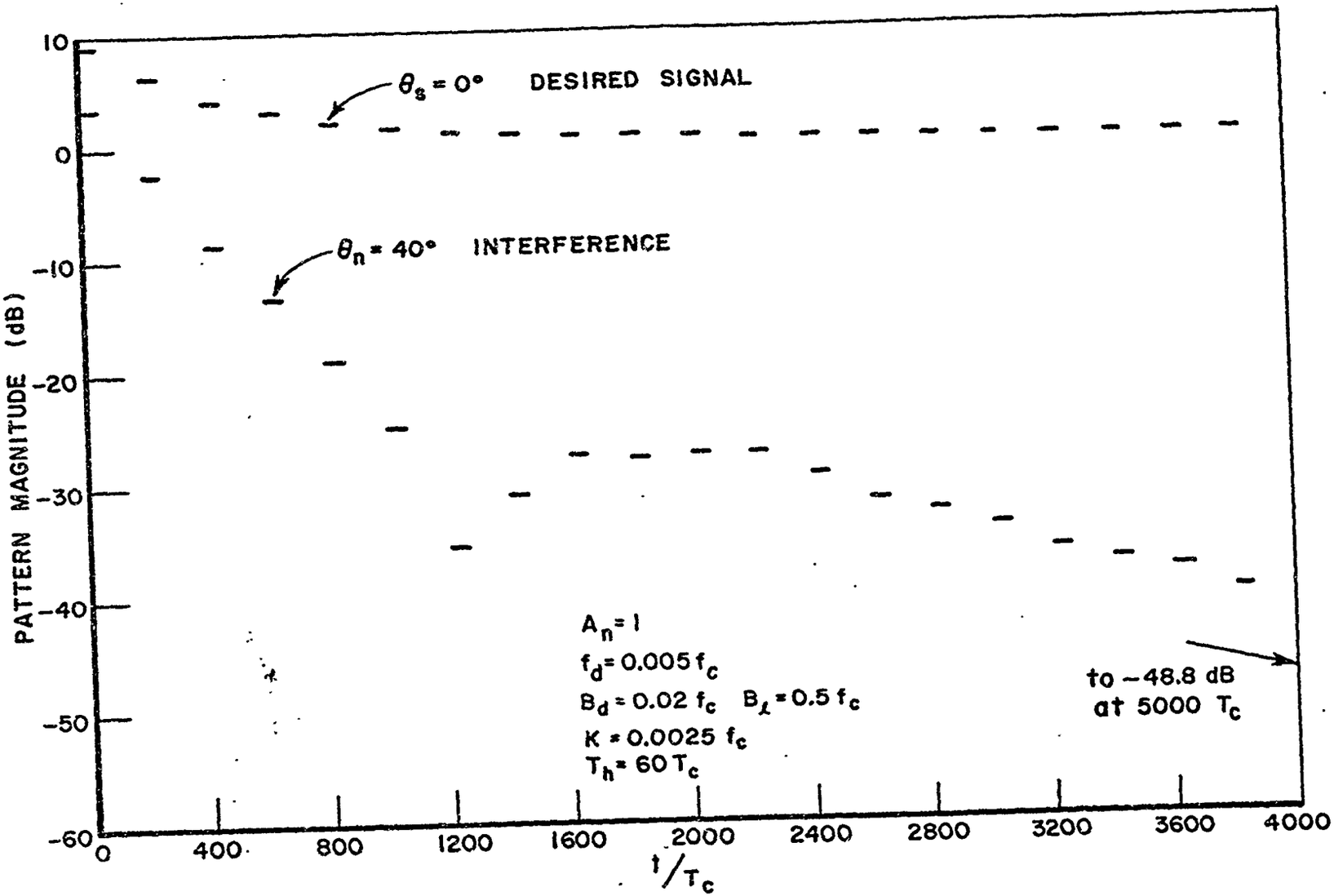


Fig. 18. The effect of non-processing (hold) intervals at the beginning of each data bit.

signal was quantized so these small phase variations were also removed.* Consequently, the reference signal in the simulation differed from the desired component of S(t) only in code timing (and data modulation delay, of course).

The degradations caused by imperfect code timing and a non-zero delay in the reference processing network are expected to be similar since the reference signal is out of phase by 180^{0} with respect to the desired reference signal over discrete time intervals in both cases. However, the intervals are distributed differently in time and their durations are different in the two cases. When code timing is in error, the intervals over which the reference signal is inverted must be much shorter than the reference network processing delay -- i.e., on the order of one-half of a code bit period or less -- if the adaptive array processor is to operate as intended. The rate at which the inversions occur has an average value equal to one-half the code rate when the code is a (pseudo) random sequence. The reference signal inversions caused by reference network processing delay occur at a much slower rate: one-half the data rate on the average when the data bit stream is a random sequence. Code timing errors equaling one-quarter and one-half of a code bit period cause the responses to change from those shown in Fig. 18 to the responses shown in Fig. 19. When the error equals one-half the code bit period, the reference signal is inverted approximately one-fourth of the time. This is approximately the same percentage of time the reference signal is inverted as a result of the delay introduced by the reference processing network. Thus, it is appropriate to compare the signal responses shown in Fig. 14 and in Fig. 19 for the one-half code bit timing error case. The signal response in Fig. 19 is more erratic because the reference signal inversions are distributed randomly. It is apparent from Fig. 19 that the feedback circuitry cannot change the weights (pattern) significantly over the short inversion intervals. Nevertheless the amplitude of the desired signal at the array output is reduced relative to the reference level by nominally the same amount in the two cases.

Comparing the interference responses in Figs. 18 and 19 shows that the interference is suppressed less when the code timing is in error -- at least for times t less than 4000 T_c. In concept, the weights could be held constant from an instant preceding each transition of the code generated in the array processor to an instant following each transition to reduce the degradation in response caused by code timing error. However, switching speed limitations of practical integrate/hold circuits restricts the usefulness of such an approach to cases where the code rate is relatively low. In many applications, the code timing error should be maintainable at less than one-fourth of a code bit length and the array performance will be acceptable without resorting to holding the weights constant during the inversion intervals.

---

*This minor defect of the simulation model is expected to result in only second-order effects since the distortion components are usually small.

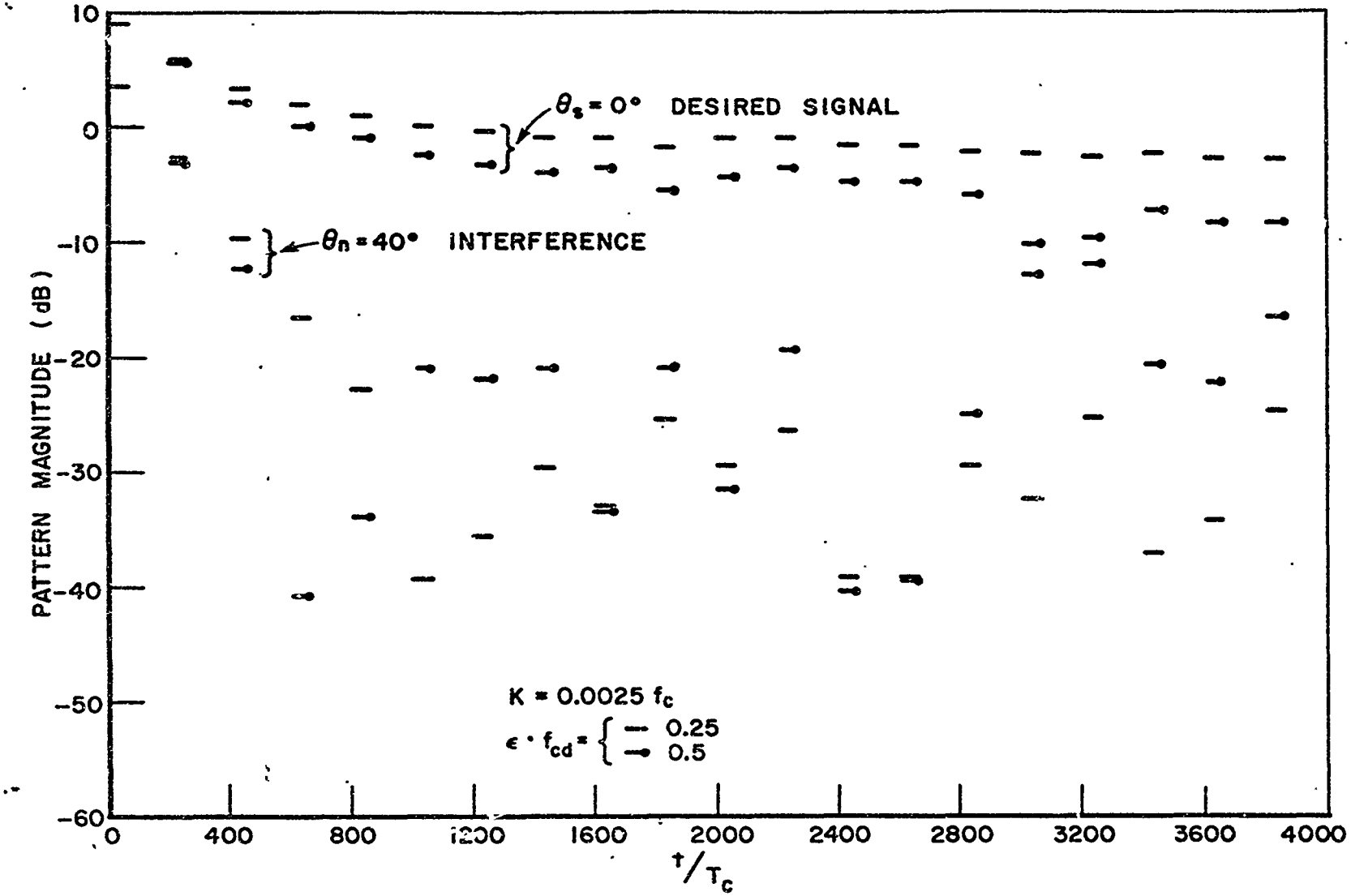


Fig. 19. The effect of code timing error.

D. Effect of Desired-Signal Frequency Offset

As discussed previously, phase alignment between the desired signals in S(t) and R(t) occurs when the phase shift through the reference network is a multiple of 360^0. This condition is not satisfied, in general, when the carrier frequency of the desired signal is offset from the center frequency since the reference network phase shift is frequency dependent.* An analysis of the effects of small frequency offsets is presented below and is followed by results of the computer simulation.

A few assumptions were made to simplify the analysis:

a) $C(t) = D(t) = 1$ (cw desired signal)

b) $A_n = 0$ (No interference)

c) $\frac{\Delta f_s}{f_c} << 1$ (small frequency offset).

The signals at the input to the weighting coefficients were expressed as

$$(29)\quad y_{2k-\delta}(t) = A_s \sin\left[\omega_c t + \Delta\omega_s t + \phi_k - (1-\delta)\frac{\pi}{2}\right]$$

$$k = 1,2,\cdots,N\ ; \qquad \delta = 0,1$$

where inphase and quadrature signals from the kth element correspond to δ equaling one and zero, respectively. Signal phase angle ϕ_k is an arbitrary constant.

The first step in the analysis was to hypothesize that the array output signal and the reference signal can be represented in the steady-state as

$$(30)\quad S(t) = A_\Sigma \sin(\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma);\quad t \to \infty$$

and

$$(31)\quad R(t) = A_{r_\ell} \sin[\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma + \gamma(\Delta\omega_\Sigma)]\ ;\ t \to \infty$$

*If delay equalization is employed as discussed in Section IVB, then signals at the error subtractor may be maintained in phase over a range of input frequencies.

where

$\Delta\omega_\Sigma$ = array output signal frequency offset

ϕ_Σ = array output signal phase

$\gamma(\Delta\omega_\Sigma)$ = reference network phase shift, $(\gamma(0) = 0)$

and

$\Delta\omega_\Sigma \neq \Delta\omega_s$.*

The steady-state error signal, given by (31) minus (30), is processed by the feedback loops to produce steady-state weighting coefficient variations given by

$$(32)\quad w_{2k-\delta}(t) = \int K\, E(t')\, y_{2k-\delta}(t')dt'$$

$$= \frac{KA_s}{2(\Delta\omega_s-\Delta\omega_\Sigma)} \{A_{r_\ell} \sin[(\Delta\omega_s-\Delta\omega_\Sigma)t - \phi_\Sigma - \gamma(\Delta\omega_\Sigma) + \phi_k-(1-\delta)\tfrac{\pi}{2}\ ]$$

$$- A_\Sigma \sin\ [(\Delta\omega_s -\Delta\omega_\Sigma)t - \phi_\Sigma + \phi_k - (1-\delta)\tfrac{\pi}{2}\ ]\}$$

$$t \to \infty;\quad k = 1, 2, \cdots, N;\ \delta = 0,1.$$

The weighting coefficients modulate the input signals (29) resulting in the steady-state output signal

$$(33)\quad S(t) = \sum_{k=1}^{N} \sum_{\delta=0}^{1} w_{2k-\delta}(t)\, y_{2k-\delta}(t)$$

$$= \frac{KA_s^2}{2(\Delta\omega_s-\Delta\omega_\Sigma)} \sum_{k=1}^{N} \sum_{\delta=0}^{1} \{\ \}$$

where

*Note that $\Delta\omega_\Sigma$ and $\Delta\omega_s$ are equal prior to adaption ($t \leq 0$) but are assumed here to be unequal in the steady-state ($t\to\infty$).

$$\{\ \} = \frac{1}{2}\ A_{r_\ell} \cos[\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma + \gamma(\Delta\omega_\Sigma)]$$

$$- \frac{1}{2}\ A_{r_\ell} \cos[\omega_c t + (2\ \Delta\omega_s - \Delta\omega_\Sigma)t - \phi_\Sigma - \gamma(\Delta\omega_\Sigma) + 2\phi_k - (1-\delta)\pi\ ]$$

$$- \frac{1}{2} A_\Sigma \cos\ [\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma]$$

$$+ \frac{1}{2} A_\Sigma \cos[\omega_c t + (2\Delta\omega_s - \Delta\omega_\Sigma)t - \phi_\Sigma + 2\phi_k - (1-\delta)\pi]\ .$$

When this bracketed expression is summed over the two values of δ, the second and fourth terms cancel leaving

$$(34)\quad \sum_{\delta=0}^{1} \{\ \} = A_{r_\ell} \cos[\omega_c t + \Delta\omega_\Sigma\ t + \phi_\Sigma + \gamma(\Delta\omega_\Sigma)]$$

$$- A_\Sigma \cos\ [\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma].$$

The expression (34) represents the contribution of the kth element to the array output signal. It is independent of the index k; therefore, the element output signals are all aligned in phase, i.e., they are phase-coherent in the steady-state. The cancellation of terms involving the input phase angles ϕ_k does not occur if the quadrature channel phase shifts are different from ninety degrees. A more general analysis is necessary in this case with anticipated results being a degradation in signal coherence and the existence, in the steady-state, of additional frequency components in the array output signal. In the present case the expression in (33) reduces to the form

$$(35)\quad S(t) = \frac{NK\ A_s^2}{2(\Delta\omega_s - \Delta\omega_\Sigma)} \{-A_{r_\ell} \sin\gamma(\Delta\omega_\Sigma) \sin[\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma]$$

$$+ [A_{r_\ell} \cos\gamma(\Delta\omega_\Sigma) - A_\Sigma] \cos\ [\omega_c t + \Delta\omega_\Sigma t + \phi_\Sigma]\}$$

by use of a trigonometric identity in (34). The conditions that must be imposed in order that (35) agree with the original hypothesis (30) are

$$(36)\quad A_\Sigma = A_{r_\ell} \cos\gamma(\Delta\omega_\Sigma)$$

and

$$A_\Sigma = \frac{-NK A_s^2 A_{r_\ell} \sin \gamma(\Delta\omega_\Sigma)}{2(\Delta\omega_s - \Delta\omega_\Sigma)} \quad . \tag{37}$$

The first condition shows that the amplitude of the output signal is reduced below the reference signal amplitude by a factor equal to the cosine of the steady-state phase error. The two conditions may be combined to obtain a transcendental equation for the output signal frequency offset:

$$\tan [-\gamma(\Delta\omega_\Sigma)] = + \frac{2(\Delta\omega_s - \Delta\omega_\Sigma)}{NK A_s^2} \quad . \tag{38}$$

The two sides of this equation are graphed in Fig. 20. The straight line representing the right side of (38) has a slope inversely related to loop gain factors and shifts with constant slope as the input frequency offset is varied. The equation is satisfied, i.e., the curves intersect, at a value of output frequency offset which is smaller than the input frequency offset ($\Delta\omega_\Sigma < \Delta\omega_S$). The frequency offset of the array output signal cannot be reduced to zero since the feedback loops require a steady-state error signal to maintain the cyclic variations of the weighting coefficients. If the loop gain is small, the line in Fig. 20 is nearly vertical and the reduction in frequency offset is small. For increasing input frequency offset, the point of intersection moves up the tangent curve, the steady-state phase error increases toward ninety degrees, and the output signal amplitude approaches zero.

The computer simulation results given in Figs. 21 and 22 confirm the analysis. The significant parameters obtained from these figures are the following:

1) $A_\Sigma = 0.886$ (-1.06 dB)

2) $\Delta f_s - \Delta f_\Sigma = 0.00085\ f_c$ (weight oscillation frequency).

These results agree almost exactly with those computed analytically from (24), (36), and (38) using the parameter values for this run, listed in Table I. The phase shift of the reference network is 28^0 as computed from (36). Note that the assumptions of the analysis are not completely satisfied here since the desired signal contains bi-phase PN coding and low-rate data modulation. The weighting coefficients are held constant for a period of time after the data bit bi-phase transition at $t = 2000\ T_c$.

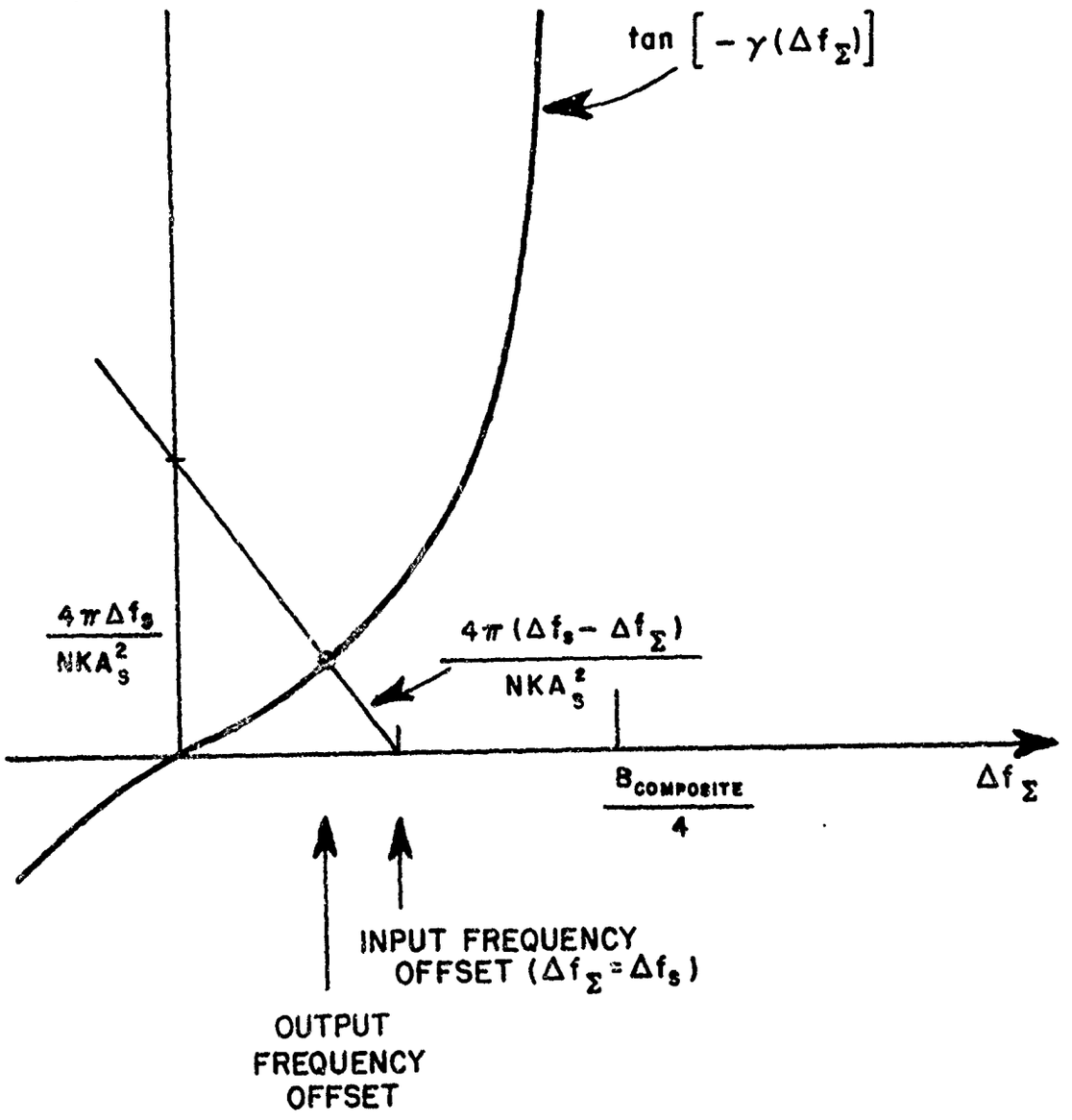


Fig. 20. Results of desired signal frequency offset analysis.

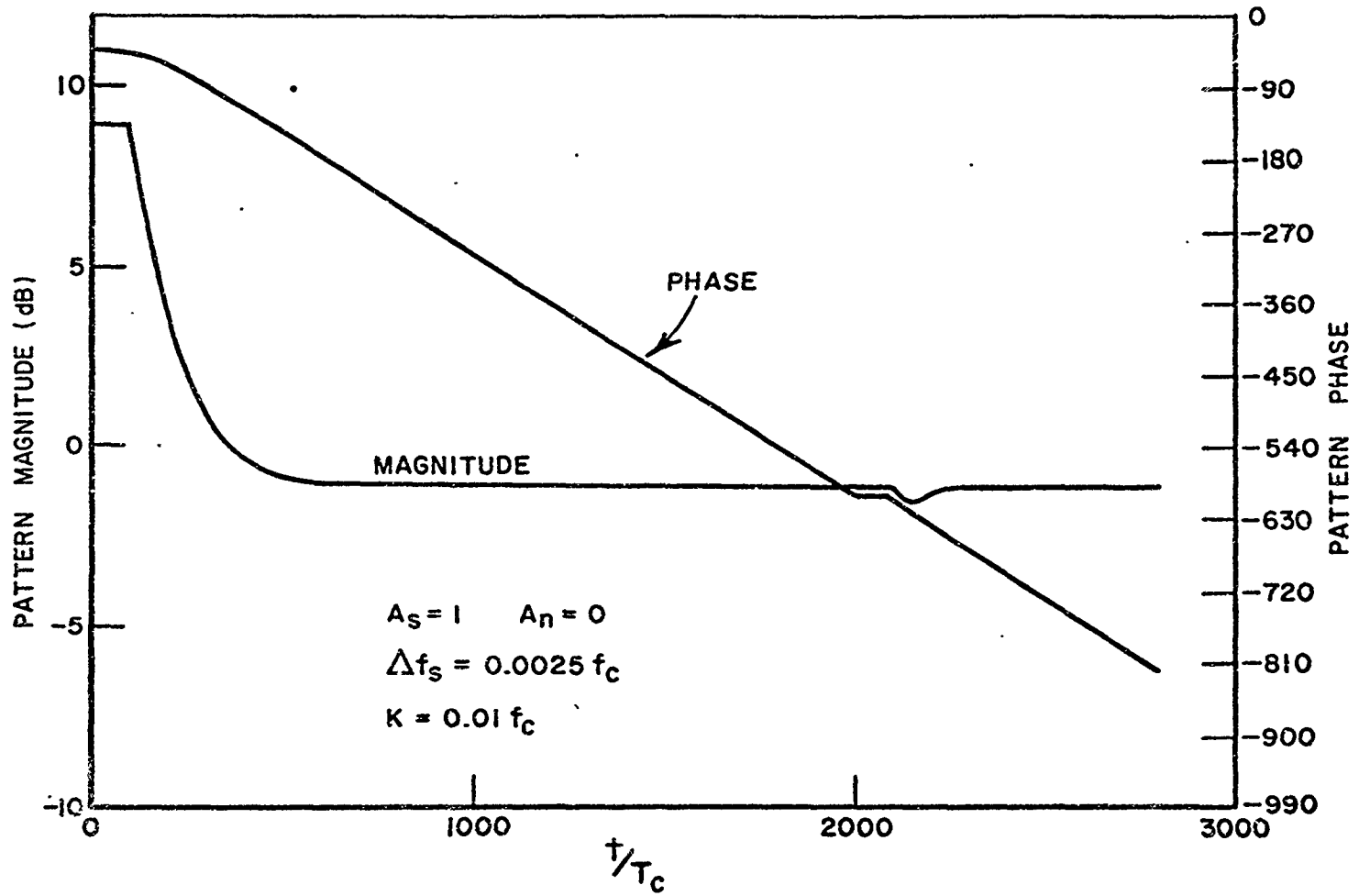


Fig. 21. The effect of desired signal frequency offset.

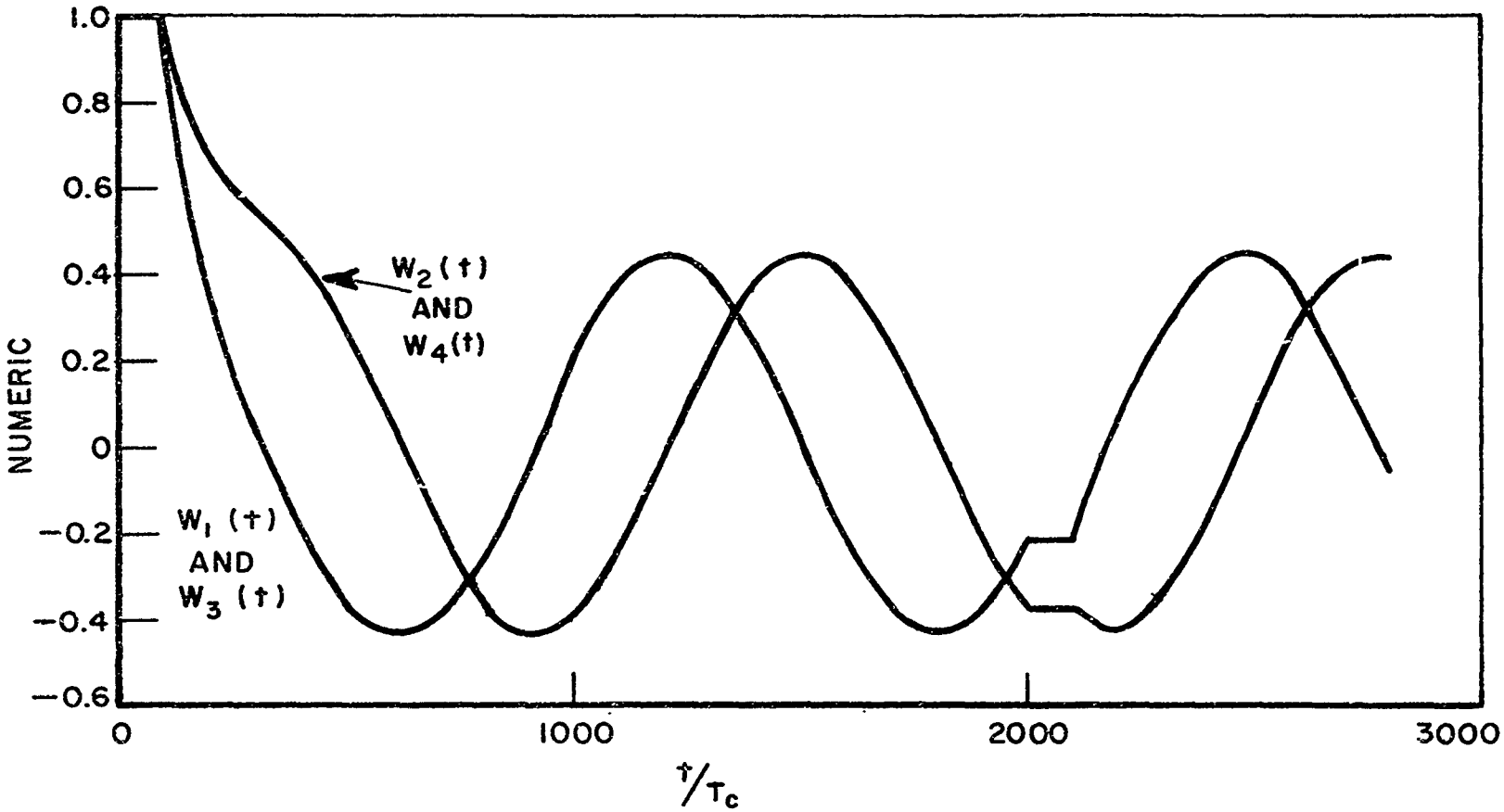


Fig. 22. Weighting coefficient response for result in Fig. 21.

During this hold period, the signal input and output frequencies are identical. Upon resuming processing, the cyclic variation of the weighting coefficients is restored as shown in Fig. 22. When the loop gain (K) was reduced by a factor of four, the results obtained were as expected: a lower output amplitude (-1.95 dB), a smaller weight oscillation frequency (0.00022 f_c), and a larger phase error in steady-state (37°). Figure 23 shows simulation results for three values of desired signal frequency offset: zero, 0.0005 f_c, and 0.0025 f_c. In this result, cw interference has been added, and the data rate has been increased significantly. The pattern response in the desired signal direction is essentially unaffected by small offsets in frequency and is reduced 3 to 4 dB when the frequency offset equals 0.0025 f_c: one-half the data rate. The addition of equal amplitude cw interference at center frequency in this case did not significantly alter the response to desired signal. The amount of interference suppression, however, does appear in Fig. 23 to be dependent on the desired-signal frequency offset. The 0.0025 f_c offset case in Fig. 23 was repeated with data modulation removed from the desired signal and with continuous loop processing (i.e., no data "hold" intervals). No significant changes in performance were observed. The code rate was increased from 0.05 f_c to 0.20 f_c in a third run and again there was no appreciable change in interference rejection over the initial 4000 T_c adaption period.

E Effect of Reference-Network Processing Gain

The effect of waveform processing gain in the reference network was assessed by comparing results for different code rates with the data filter bandwidth held fixed at 0.02 f_c. Figure 24 illustrates performance for code rates equaling 0.01 f_c, 0.05 f_c, and 0.20 f_c: (one-half, five-halves, and ten times the data filter bandwidth, respectively). The pattern response in the interference direction has been sketched in between values computed at the hold instants to make the three cases more distinguishable. The essential features to be noted in Fig. 24 are the following: 1) for very small processing gain (the .01 f_c case), the rejection of interference is very erratic, 2) for moderate processing gains (f_{cd} = .05 f_c and 0.20 f_c), the response to interference is much smoother with slightly better rejection at the higher code rate. The erratic performance for very low code rates is to be expected since the data filter does not average the bi-phase coded interference over several bits of the code. The response to interference in this case is strongly dependent on the detailed structure of the code sequence being used. Initial (transient) behavior, in particular, is influenced by the starting point in the code, i.e. by the number and distribution of transitions in the first several bits of the code. The starting point for the code employed in the simulation (see Fig. 5) produced the rather surprising result of a faster response and more rejection with the 0.01 f_c code rate than for the higher code rates in the time interval $t < 4000\ T_c$. However, it is believed that different code starting points would result in dramatically different initial responses for the 0.01 f_c code rate case, some of which would be

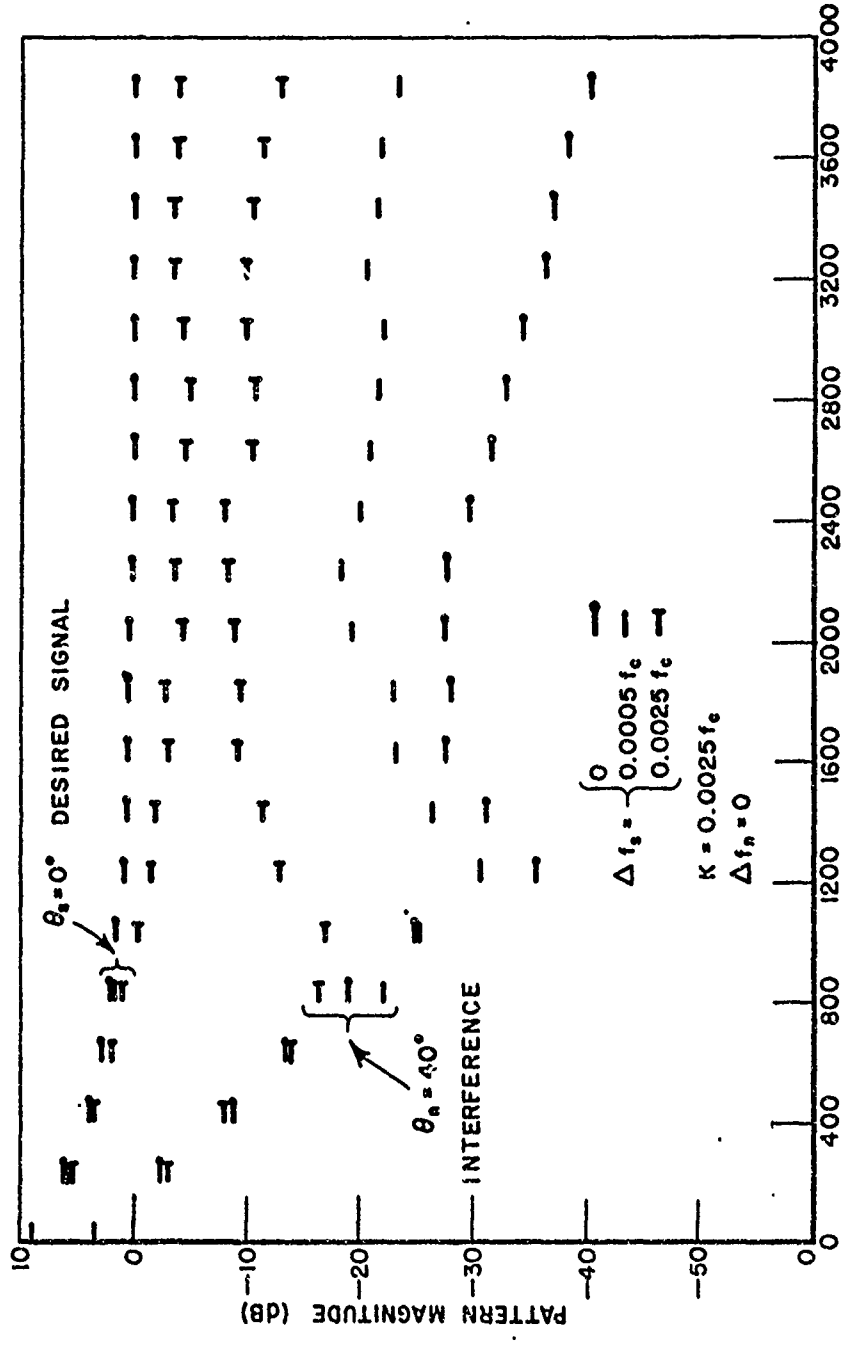


Fig. 23. The effect of desired signal frequency offset (with interference present).

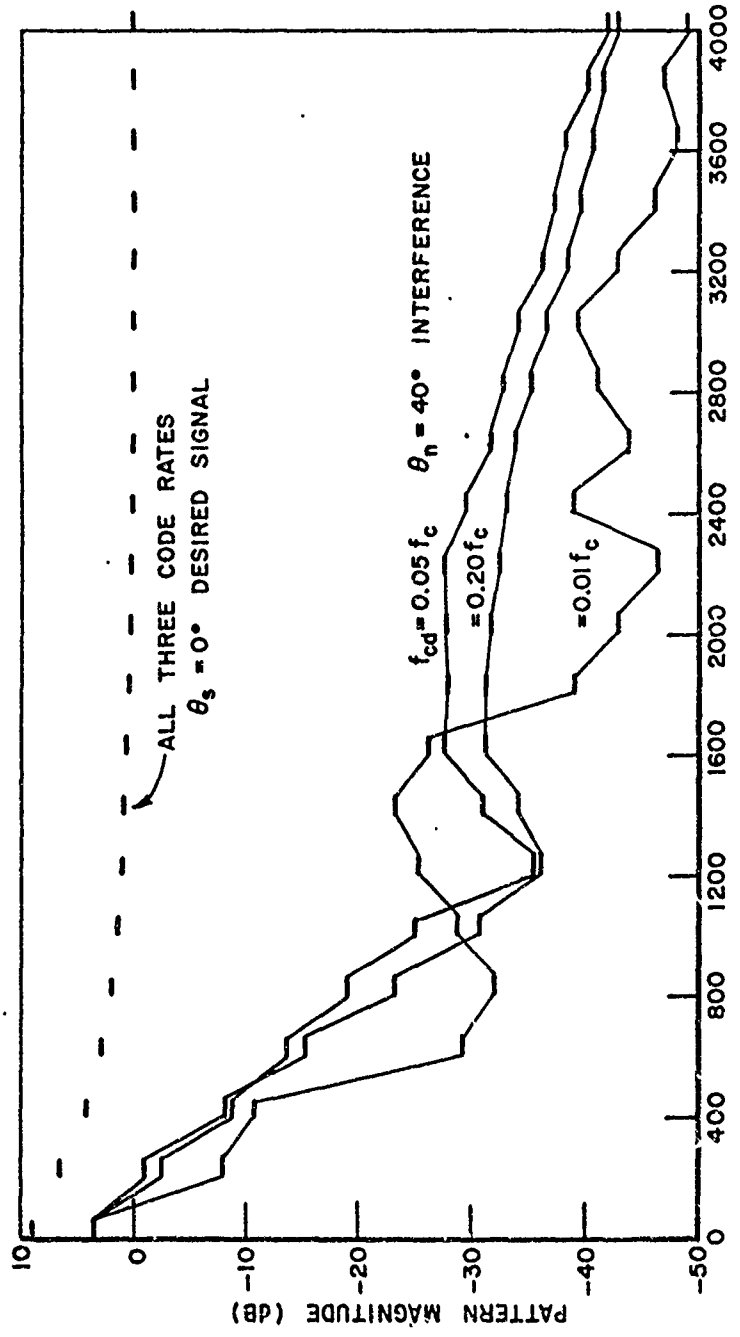


Fig. 24. The effect of code rate.

much less desirable than the response shown.* In addition, it is believed that the response for $f_{cd} = 0.01\ f_c$ is not uniformly better than the responses shown for higher code rates for $t > 4000\ T_c$. Thus, although the information presented is incomplete due to limits on computation time, it appears that the ratio of the code rate to the data filter bandwidth must exceed approximately three in a practical system.

F. Performance for Large Input Interference

The simulation results presented to this point have been limited to the case where the desired and interfering signals are equal in magnitude. The performance of the processor for larger levels of input interference will now be examined. Before proceeding with the results, it is desirable to discuss the characteristics of the processor which affect its speed of response. The gain of the feedback loops (and thus their response time) is dependent upon the amplitude(s) of signal(s) present at the array inputs. This dependence can be shown from the defining equations (5) for the weighting coefficients:

$$(39)\quad \dot{w}_i = K \{y_i(t)\ [R(t) - \sum_{j=1}^{2N} w_j(t)\ y_j(t)]\}_{\substack{\text{Baseband} \\ \text{Terms}}}$$

Upon rearrangement, these equations become

$$(40)\quad \dot{w}_i + K \{y_i^2(t)\}_{\text{Baseband}}\ w_i + K \sum_{\substack{j=1 \\ j\neq i}}^{2N} w_j(t) \{y_i(t)\ y_j(t)\}_{\text{Baseband}}$$

$$= K \{y_i(t)\ R(t)\}_{\text{Baseband}}.$$

The coefficient multiplying ith weight w_i in (40) is proportional to the square of (i.e., the power of) the signal $y_i(t)$ at the input to the ith feedback loop. By analogy with the simple, constant-coefficient equation

$$(41)\quad \dot{w} + \lambda w = \gamma\ ;\quad \lambda, \gamma \text{ constants},$$

whose solution is of exponential type, $\exp[-\lambda t]$, it is to be expected that the response time of the ith feedback loop is strongly dependent on this coefficient.

---

*Variation in the response with code starting point is also expected at higher code rates, although the amount of variation should decrease as the code rate is increased.

The system of equations (40) can be solved analytically for the special case of a single cw signal incident upon the array and a cw reference signal. This is, of course, the constant coefficient case; however the system of equations is still coupled. Details of the solution are given in Appendix IV. The weighting coefficients decay exponentially in this case with a time constant inversely proportional to signal power:

$$\tau = \left[NK \left(\frac{A_s^2}{2} \right) \right]^{-1} \quad . \tag{42}$$

This time constant also appears in the result which shows the effects of frequency offset (38).

When interfering signals are present and a processed reference signal is used, the weighting coefficient responses are not necessarily given by simple exponentials. Consequently, the notion of a single time constant as a measure of convergence may not be appropriate. The above results, however, indicate that the weighting coefficients become more responsive as the power level of either the desired signal or interfering signals increases. This does not imply that the rates of change of the various signals in the array output are uniformly affected by some measure of the total input power. On the contrary, the different output signals, or equivalently, the array pattern in different directions, may change at differing rates. A possible interpretation here is that the rate of response of the array to a given input signal is primarily determined by the power contained in that signal and to a much lesser extent by the characteristics of the other input signals. That is, there is some evidence* which supports the conjecture that the array responds to each signal more or less independently except when 1) the number of interfering sources exceed the number of independent pattern nulls (the overconstrained-array case), 2) the angular separation between two or more sources is very small or zero, and 3) an interfering source is sufficiently large so as to produce non-linear effects (saturation and limiting) in the array circuitry. Another situation leading to non-independence of desired and interfering signal responses occurs when the response times of the feedback loops are decreased - as a result of increased interference power - to values less than a code bit period. The simulation results which follow illustrate this case.

Pattern adaption results for a 20 dB interference-to-signal ratio at the array inputs are shown in Fig. 25. The element spacing was increased to ten wavelengths here to illustrate the increase in angular resolution with wider spacing. The initial (broadside) pattern is maximum (+9 dB) in the interference direction (0^o) and nearly zero (-32 dB) in the desired signal direction (2.85^o). Corresponding initial output levels are +29 dB

*Results obtained by experimental testing with a four-element adaptive array processor lend support to this interpretation.

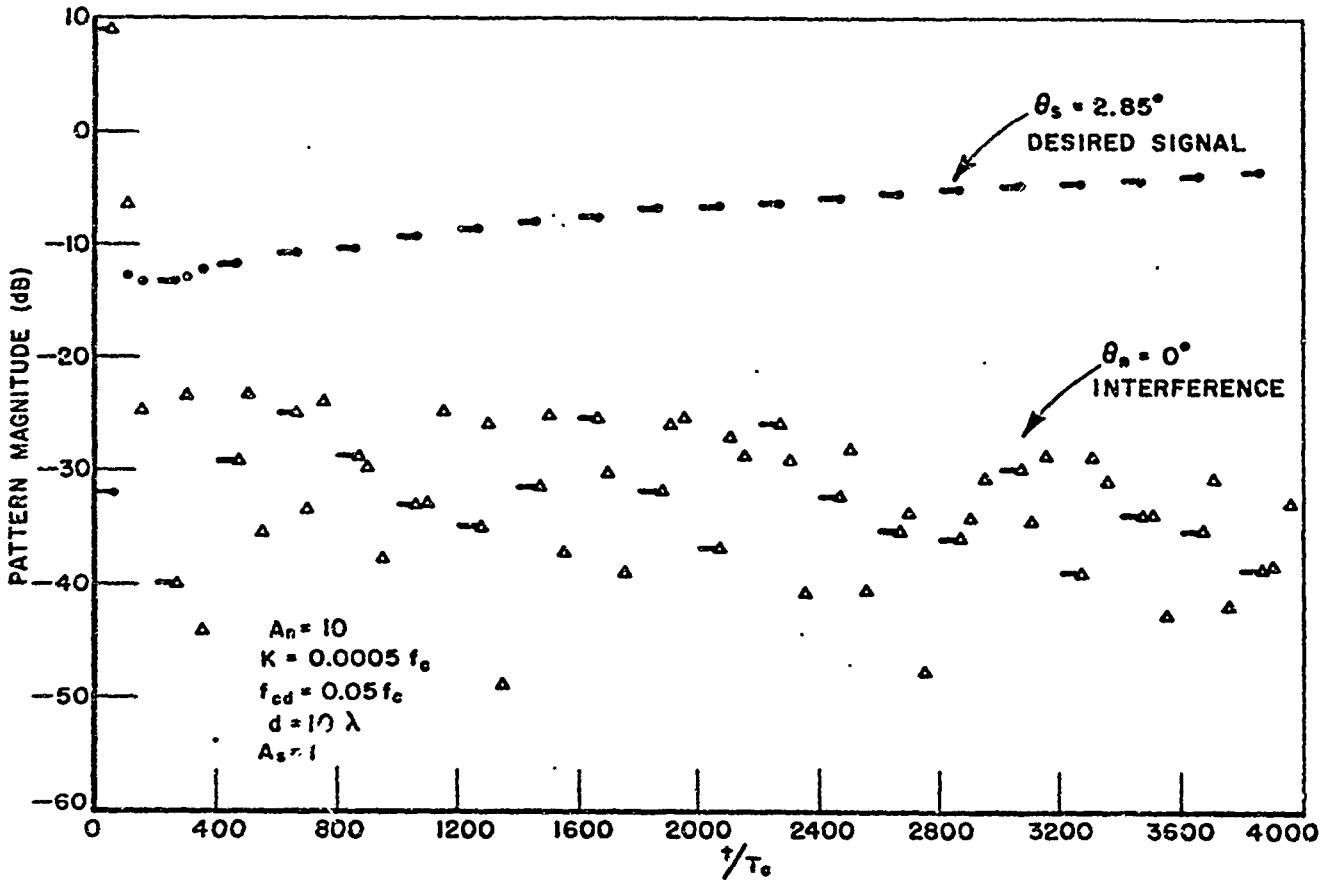


Fig. 25. The effect of processing loop bandwidths approaching the code rate.

for interference and -32 dB for desired signal: a 61 dB interference-to-signal ratio. The frequency offsets of both signals are zero. The processor adapts quickly to reduce the large output interference level to approximately -3 dB (note the -23 dB pattern response in the interference direction. Simultaneously, the desired output signal increases from -32 dB to approximately -13 dB. Thereafter, the pattern magnitude in the desired signal direction increases slowly with a corresponding slow decrease in the maximum value of the pattern in the interference direction. The pattern variations in the interference direction and the small changes in the weighting coefficients in Fig. 26 represent the response of the processor to errors produced by bi-phase code transitions of the desired signal. The essential characteristics of this response are illustrated by phasor diagrams of the array output signals in Fig. 27. The desired signal and interference phasors are approximately equal in magnitude (-6 dB) at a time near $t = 2400\ T_c$ as evidenced by the 20 dB pattern differential in the two directions. Before a code transition, the sum of output desired signal and interference (nearly) equals the reference signal and the error signal is (nearly) zero. This condition is represented by the dotted phasors of Fig. 27a. Immediately after a code transition, an error results which is removed primarily by a (near) bi-phase transition of the output interference phasor as shown in Fig. 27b. As the output interference phasor undergoes this transition, its magnitude first decreases and then increases as the transition is completed. The "up-and-down" pattern magnitude variations in Fig. 25 in the interference direction illustrate this behavior over many code transitions. Only a very small change occurs in the desired signal phasor of Fig. 27. Changes in the array output interference dominate over desired signal changes for two reasons:

1) the error x input interference components at the outputs of the error multipliers in all the feedback loops are much larger (collectively) than the error x input signal components, and

2) only small weight changes are required to produce large changes in the output interference component when the input interference is large.

Thus, an error signal can be eliminated more rapidly by changing the interference at the array output rather than the desired signal. Full 180° phase transitions in the output interference phasor are completed only when several code bits of the same sign follow a code bit transition. Since this code structure seldom occurs in pseudorandom codes, the error signal is not completely nulled between code transitions. Processing of the residual error signal over many code transitions results in the cummulative effects observed: a steady increase in the desired signal phasor and a corresponding decrease in the interference phasor (see Fig. 27b). The rate at which the changes occur in Fig. 25 is strongly influenced by the ratio of the feedback-loop bandwidths to the code modulation bandwidth. The behavior of the phasor diagrams with time in this example indicate that this ratio is approximately equal to one.

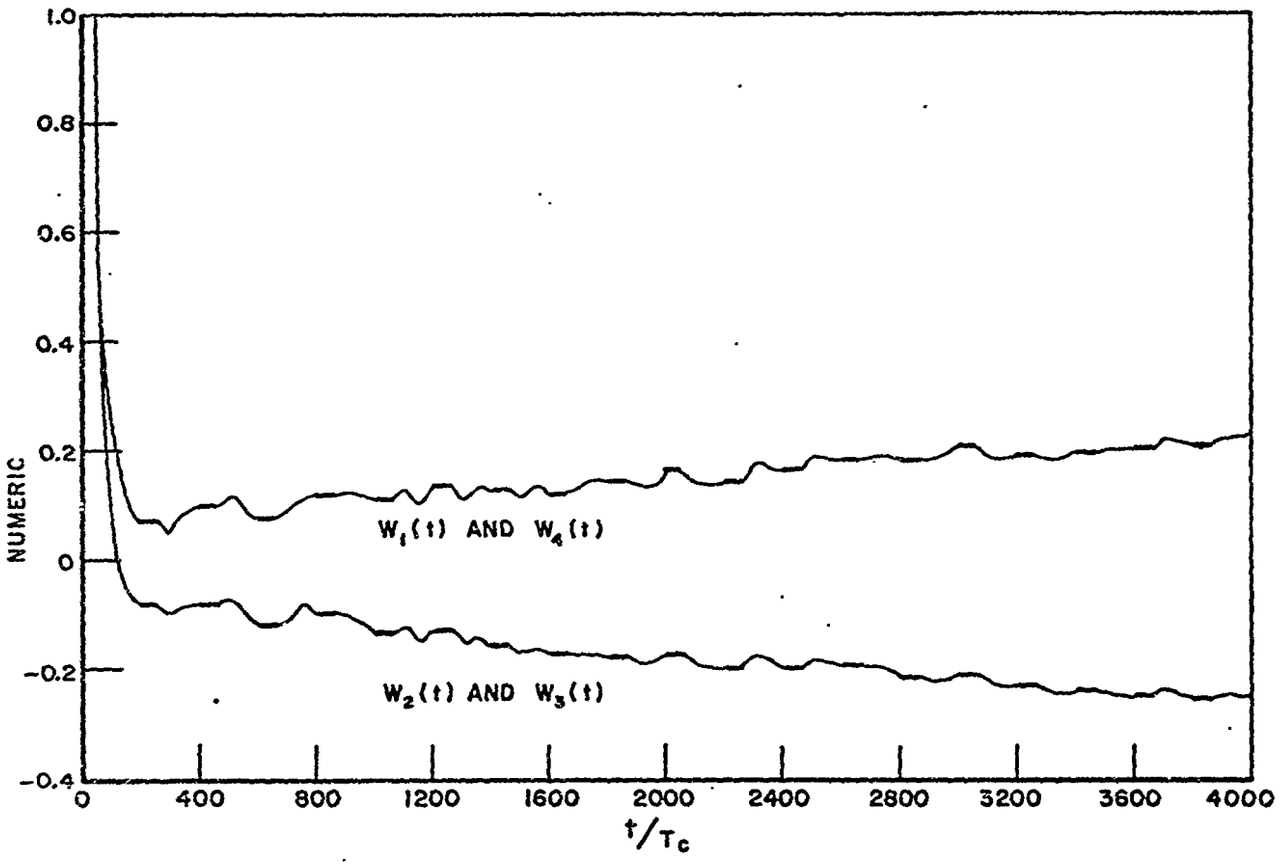


Fig. 26. Weighting coefficient response for result in Fig. 25.

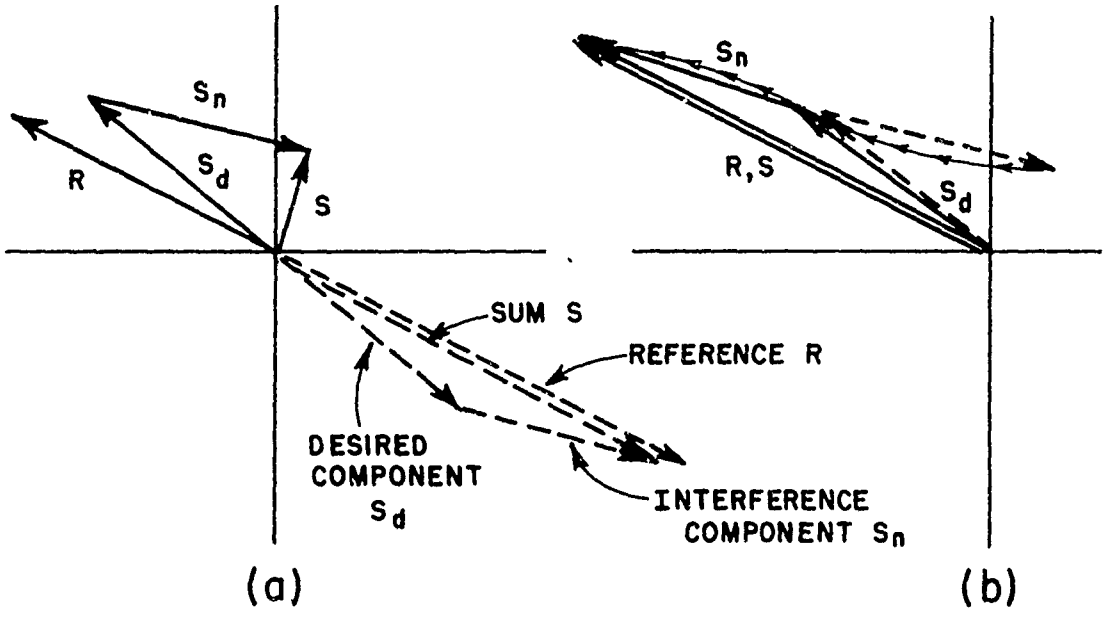


Fig. 27. Array output conditions: (a) immediately after a code transition and (b) after processing.

A 40 dB interference-to-desired signal input ratio produces the pattern response shown in Fig. 28. All parameters except interference level are unchanged from the previous case.* For the present case, the phase of the interference at the array output is switched rapidly by 180^0 as a result of changes in the weights following a 180^0 transition in the reference signal's phase. Initially, the processor responds rapidly to reduce the output interference level to approximately 0 dB (note the -40 dB pattern response). After the decay of this initial transient, there is no tendency for the processor to increase the level of the desired signal at its output to the desired value. The loops respond, i.e., the weights are changed, so that the cw interference at the processor inputs is bi-phase modulated by the reference code. As a result, the error signal can be made small on the average even though the level of the desired signal at the output is much smaller than the desired value. The "coded" output interference in S(t) is processed by the reference network to produce a "clean" coded reference signal. Since the desired signal component at the output is down approximately 30 dB the data modulation produces only small amplitude and phase changes on the composite output signal S(t), i.e., the

*This level of interference (A_n=100) is close to the maximum permissible value for the chosen gain constant and sampling rate. Larger levels of interference produce instability in the digital feedback loops of the simulation (see Sec. III.D).

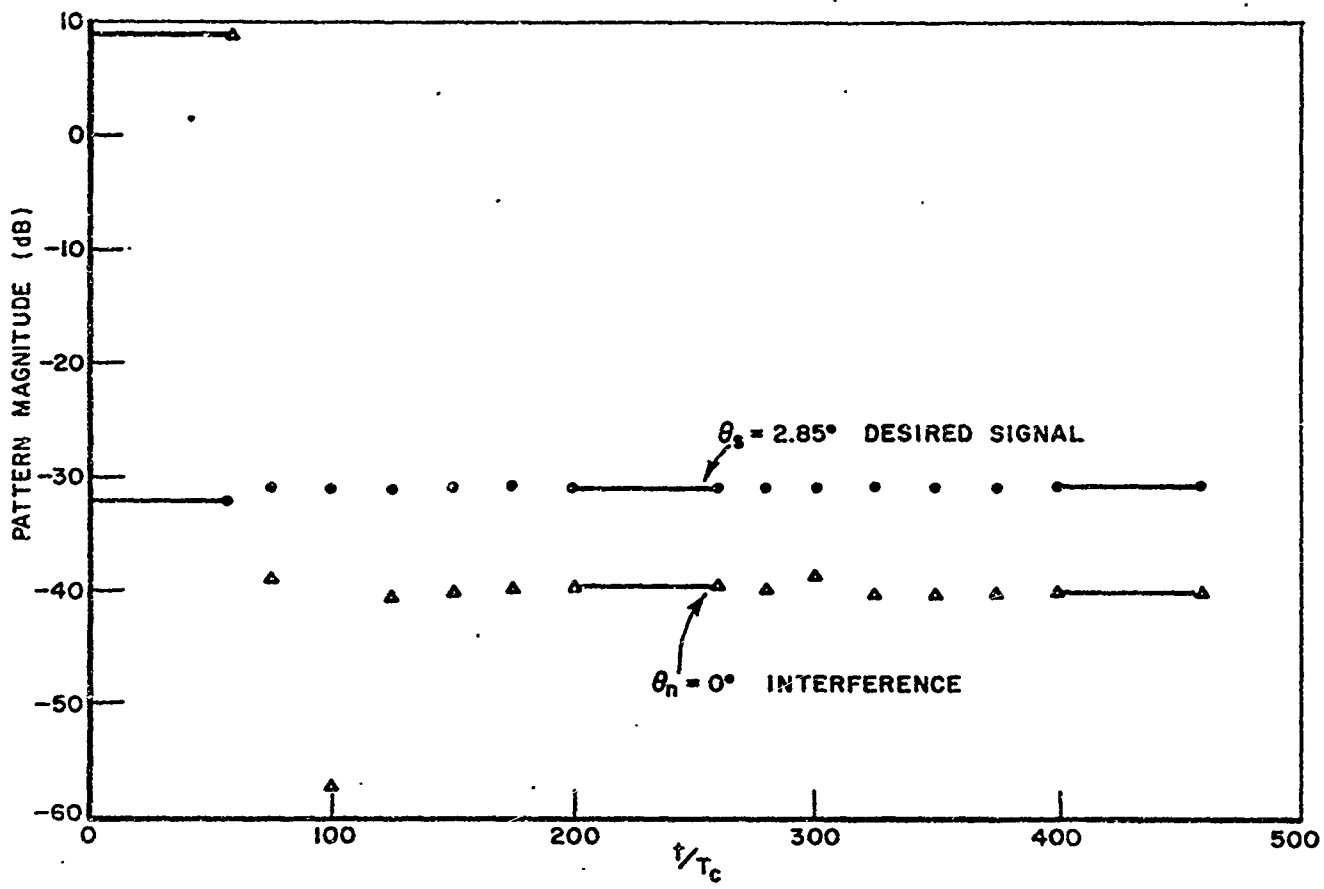


Fig. 28. The effect of processing loop bandwidths exceeding the code rate.

data is essentially removed in this case. To determine if this behavior is unique to the case where the frequency of the cw interference equals the center frequency of the processor, the frequency of the interference was offset by 0.01 f_c and the simulation repeated.* The pattern magnitude responses remained essentially unchanged; however, the pattern phase angle in the interference direction decreased linearly except at code transition instants where a 180^0 step change was observed. This result implies that the processor tends to remove the frequency offset of the interfering signal in addition to bi-phase modulating it with the code rather than responding properly to the much smaller desired signal.

The pattern response when the initial pattern is broadside, the desired signal arrives at an angle of 0^0, and the interference at 40^0 is shown in Fig. 29. The element spacing here is one-half wavelength. The pattern response in the direction of the desired signal reduces from 9 dB to 6 dB as a 40 dB null is formed in the interference direction. At the array output, the desired and interfering signals have steady-state magnitudes of two and one, respectively. They are 180^0 out-of-phase except during data hold intervals when processing is interrupted. Bi-phase coding of the output interference component occurs as in the previous example. In this case, however, the bi-phase data modulation is retained on the composite array output signal since the desired signal component at the output is larger than the interference.

It is apparent from the results shown in Figs. 28 and 29 that the bandwidths of the processing loops have increased to the point where they greatly exceed the code modulation bandwidth as a result of increasing the power in the interfering signal. This condition occurs in the simulation because the analytically-derived feedback loops of Fig. 2 contain no band-limited elements** to limit loop response time as the levels of the input signals are increased. An unlimited bandwidth in the ideal processor is, of course, consistent with the theoretical objective: to minimize the squared error. As the processor becomes more responsive with higher signal levels, the composite array output signal is changed more quickly to make it conform to the reference signal and smaller errors occur as a result. The responses shown in Figs. 28 and 29 confirm this behavior: the error signal is maintained at a very small value throughout the processing interval. It should be noted from these figures, however, that the output interference is not reduced to a small value; also, the amplitude error of the desired signal is not reduced to zero. The important conclusion

---

*The phase shift through the reference network at this frequency (1.01 f_c) is approximately 200^0. The phase shift equals 0^0 when the frequency offset equals zero.

**The ith error multiplier output is the product $E(t)\ y_i(t)$ rather than the convolution of this product with a low-pass impulse response as would be the case for an error multiplier having a limited output bandwidth.

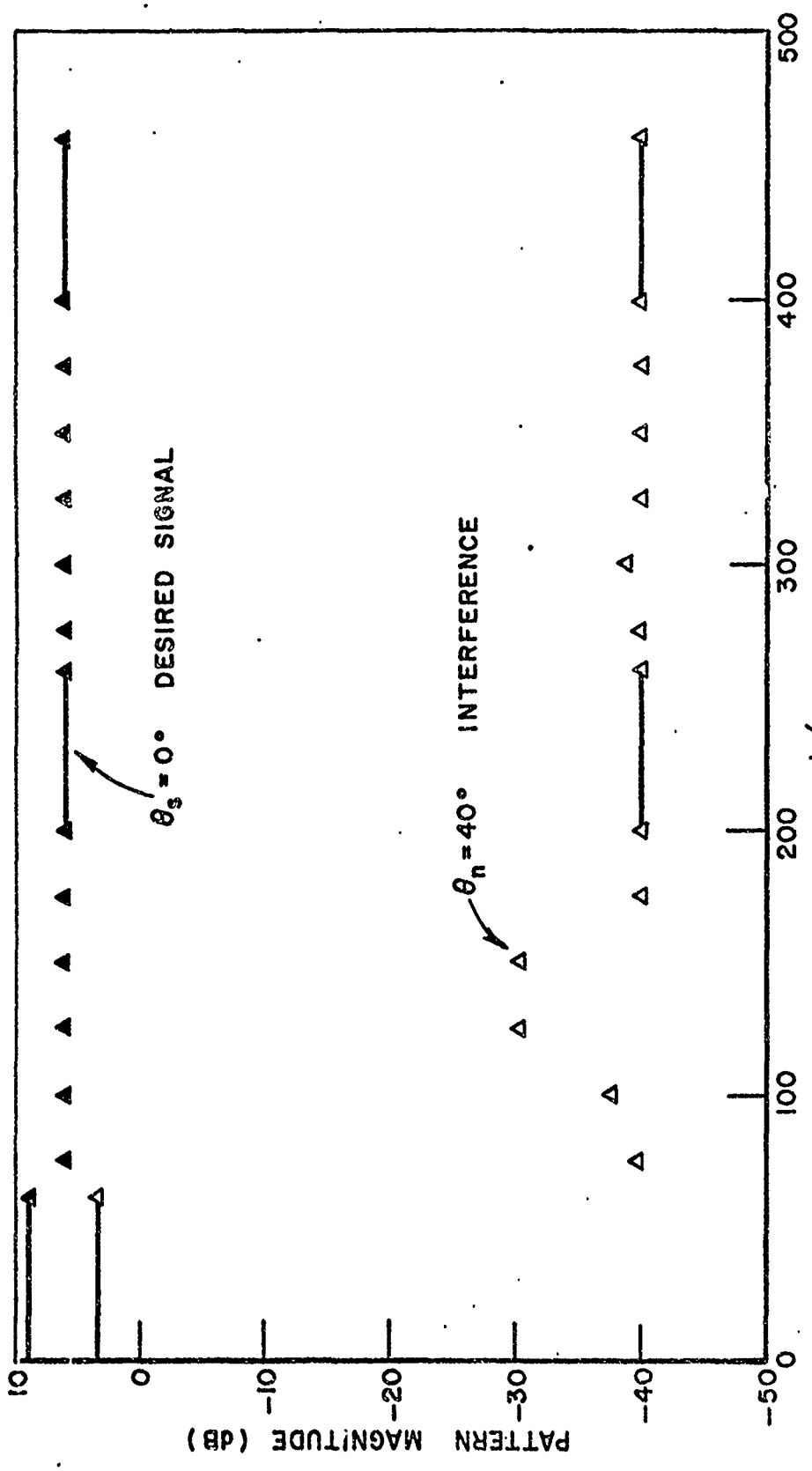


Fig. 29. The effect of processing loop bandwidths exceeding the code rate.

to be drawn from these results is that minimization of the error signal is equivalent to minimization of interference and proper adjustment of desired signal at the array output only when the loop bandwidths are less than the code modulation bandwidth. This conclusion is consistent with the discussion in Sec. IIB regarding the correlation properties of the input and reference signals required for the array to operate properly. It was noted there that the spectral width of the reference signal should greatly exceed loop bandwidths so that the correlation between the reference signal and the interference over time intervals having a duration equal to the reciprocal of the loop bandwidth is small.

The fact that loop bandwidths in the simulation become excessively large as the interference level is increased does not necessarily imply that such will be the case in practical array processors. A fairly large value was used for loop gain constant K in the simulation to restrict program run time; this value may not be achievable in a practical processor. Moreover, a non-zero error multiplier rise time and integrator slew-rate limitations may well impose upper limits on processing (loop) bandwidths which are less than the code modulation bandwidth. The simulation results for high-level interference would not apply in this case since the simulation model is incomplete. The limitations in dynamic range of practical array processors must also be considered if an extreme interference condition is to be tolerated. The occurrence of amplifier saturation, intermodulation product generation, and non-linearities in the feedback circuits will undoubtedly alter the responses from those given in the simulation. The simulation results, however, do illustrate the performance characteristics expected from an array processor ideally implemented in the minimum mean-square error configuration. In the following section, the possibility of modifying the processor's configuration to restrict the loop bandwidth in the simulation for the high-level interference case is examined.

G. Results for Modified Simulation Models

1. Loop Gain Control (AGC)

A time-varying loop gain factor G(t) was introduced into each feedback loop in the simulation to determine the feasibility of automatic gain control (AGC). Factor G(t) had an initial (t=0) value of unity and decayed exponentially to a final value equal to the ratio of desired signal power to total input power: $A_s^2/(A_s^2+A_n^2)$. Figure 30 shows the results for 40 dB input interference to signal ratio and a G(t) decay time constant of 250 T_c. The response for $t < 400\ T_c$ is, as expected, very similar to the response shown in Fig. 29. As the loop bandwidths become smaller than the code bandwidth due to the decay of G(t), the interference is increasingly rejected at the array output. However, the pattern magnitude in the desired signal direction does not change significantly indicating that a 40 dB loop gain reduction to achieve satisfactory interference rejection results in an unsatisfactory (slow) response to desired signal. This result is to be expected since both desired and interference components at the error

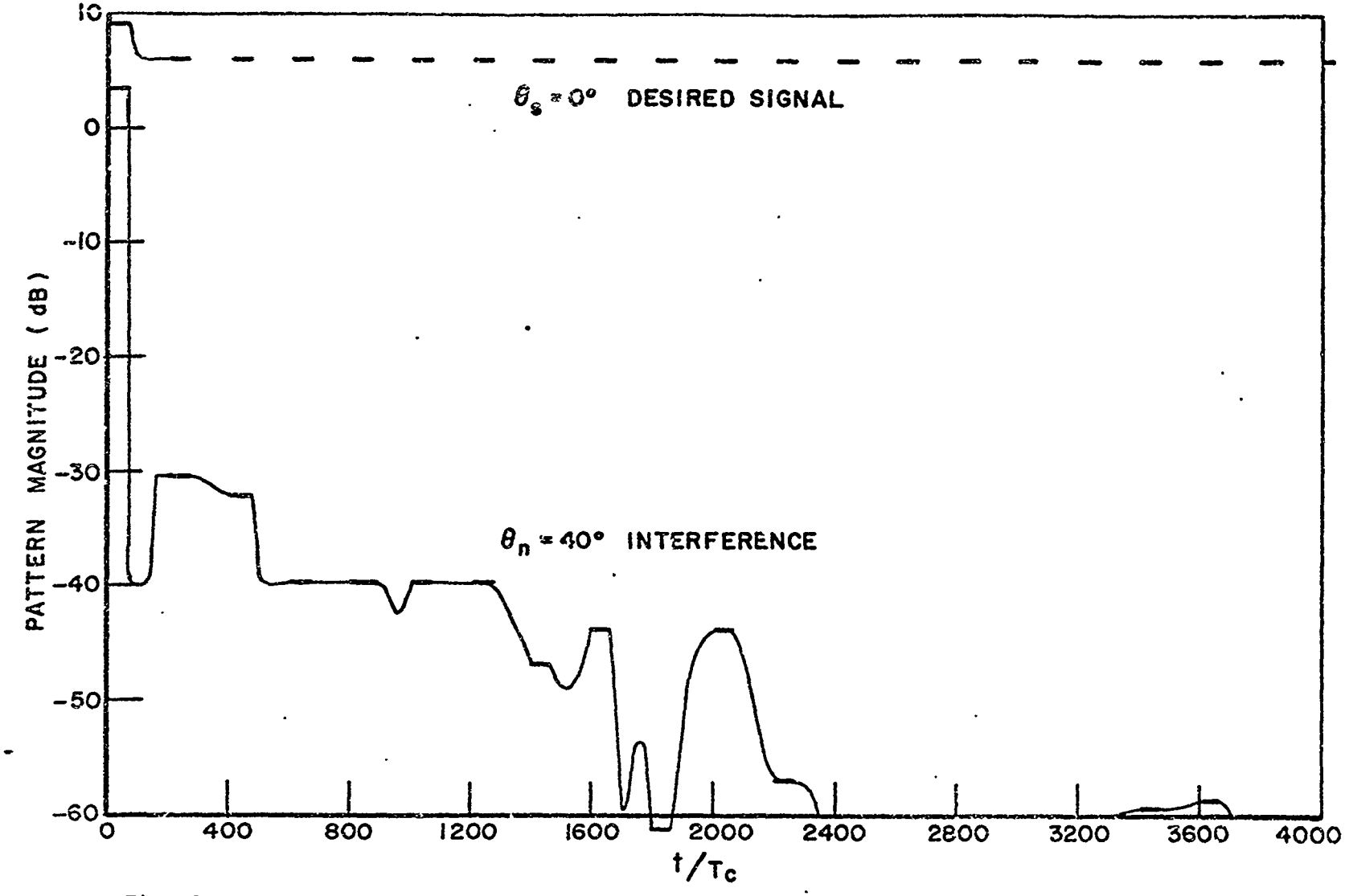


Fig. 30. The effect of loop gain reduction under high-level interference conditions.

multiplier outputs are reduced equally by G(t). Although the gain control is in the feedback paths here, essentially equivalent results are anticipated with input (array element) AGC methods since the gain reduction is again common to both desired signal and interference. The results in Fig. 30 suggest that AGC methods might be effective provided 1) the initial level of desired signal in the array output is adequate, i.e., the pattern does not have a null in the direction of desired signal initially, and 2) the rapid changes in the pattern occuring initially when the interference null is formed do not produce a null in the direction of the desired signal. The use of a moderate amount of gain control in the array element input circuitry might be desirable to prevent non-linearities (saturation) in the weighting coefficient multipliers when the input signal levels are large.

2. Error-Multiplier Output Non-linearity

The effect of amplitude-limiting (saturation) in the error multipliers under high-level interference conditions was investigated in the simulation. The output samples $E(t_j)y_j(t_j)$ of the error multipliers were adjusted in amplitude in accordance with the non-linear characteristic shown in Fig. 31.

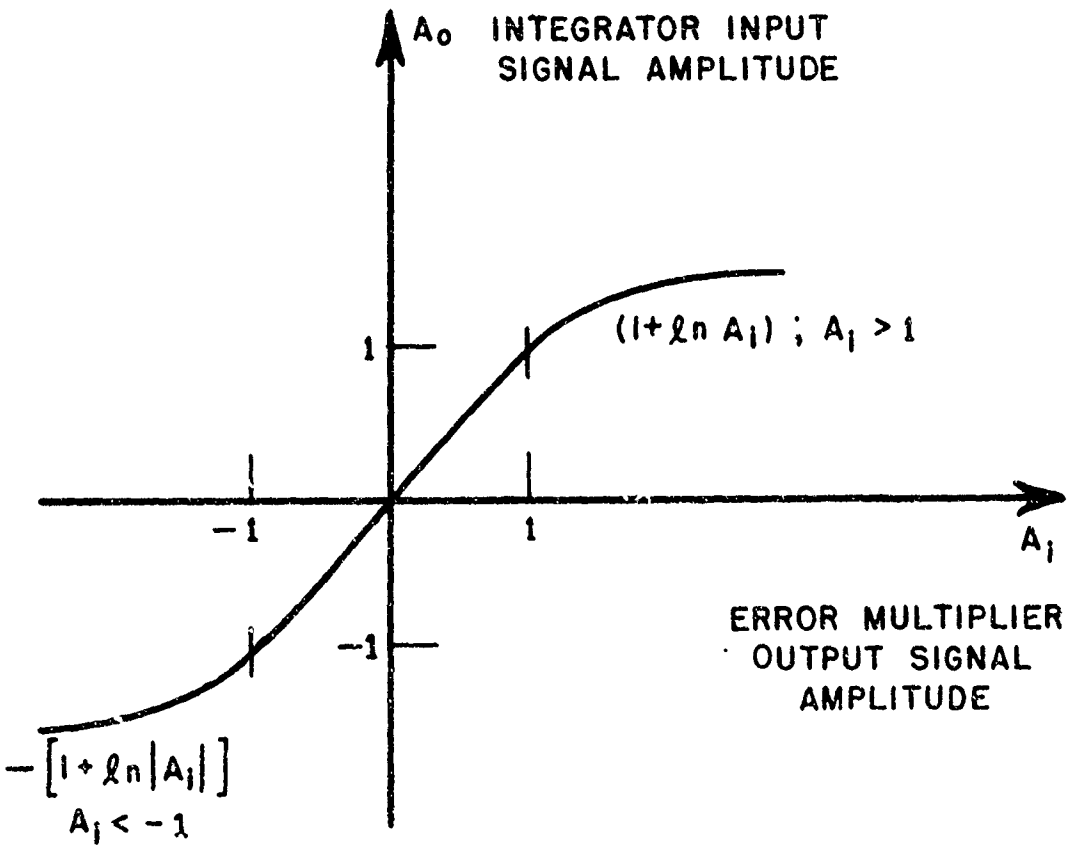


Fig. 31. Simulation characteristic for error multiplier output amplitude saturation.

For small inputs, the characteristic is linear and the gain equals one. For large inputs, the output amplitudes are compressed according to a logarithmic function. Simulation results for a 40 dB input interference to signal ratio are shown in Fig. 32. These results show that the non-linearity effectively reduces loop gain initially when the output interference (and thus the error signal) is very large. The rates of pattern response shown for $t < 200\ T_c$ are considerably smaller than those without the non-linearity (see Fig. 29). As the output error becomes small, however, and linear operating conditions are restored, the gains (and bandwidths) of the feedback loops increase. For $t > 200\ T_c$, the character of the response is identical to that obtained previously, i.e., bi-phase switching of the interference occurs to remove errors resulting from the reference-code phase transitions. The responsiveness of the feedback loops when the error is small is not affected by the non-linearity.

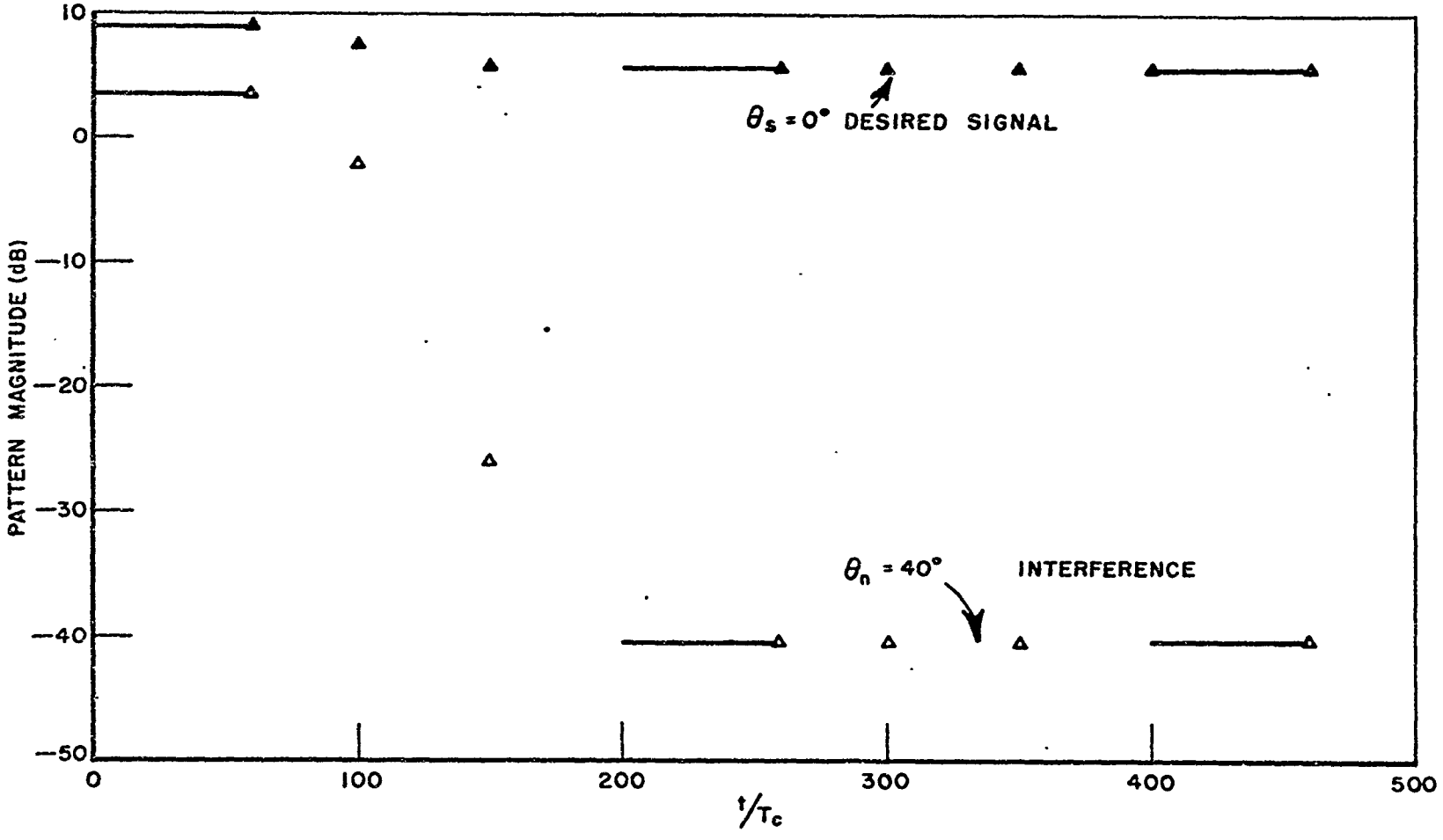


Fig. 32. The effect of error multiplier output saturation.

SECTION VI
SUMMARY

A receiving array capable of adaptive pattern shaping has been described. The signals at the outputs of the array elements are (effectively) adjusted in amplitude and phase by a feedback control system designed to minimize the difference between the array output signal and a reference signal. The reference signal is derived from the array output signal by a waveform processing network. The particular waveform processor employed accepts digital communication signals which are bi-phase modulated with the modulo-2 sum of a code and a random data sequence.

The initial (transient) adaption performance of a two-element adaptive array processor has been simulated on a digital computer for the case of cw input interference. The simulation results show that the waveform processor's time delay (or equivalently, its bandwidth) is a significant parameter affecting performance. The waveform processor delays the desired signal's data modulation and, in addition, shifts its carrier phase by an amount dependent on the carrier frequency. These alterations reduce the correlation between the reference signal and the desired signal at the array output. As a consequence, the desired signal is reduced in amplitude (i.e., partially suppressed) at the array output to a level below that of the constant-amplitude reference signal. The worst-case suppression due to data modulation delay is shown to be approximately 2.5 dB, 6 dB, and 12 dB for time delays of one-eighth, two-eighths, and three-eighths of the data bit period, respectively. The suppression factor due to waveform processor phase shift (with no data modulation present on the desired signal) is equal to the cosine of the phase shift. The simulation results show that the weighting coefficients undergo periodic variations in the steady-state in an attempt to shift the carrier frequency of the desired signal to a value where the waveform processor's phase shift is a multiple of 360°, the phase error is reduced by an amount dependent on the gain of the feedback loops. Complete elimination of phase error is not possible since a residual error signal is necessary to force the weighting coefficients to exhibit a periodic behavior.

The delay introduced by the waveform processor also influences the interference rejection capabilities of the array. The simulation results show that suppression of desired signal during the initial adaption period is accompanied by a decrease in interference rejection. A technique which compensates for data delay errors by processing only during appropriate time intervals was described and its effectiveness demonstrated. A delay equalization technique which could be used to improve the processor's performance was also described.

The correlation between the reference signal and desired signal is also reduced by imperfect synchronization of the code generated within the processor. The simulation results show that timing errors of one-quarter and one-half a code bit period result in approximately 3 dB and 6 dB signal suppression, respectively. The results indicate that the "steady-state" performance of the array processor is relatively independent of the waveform processor's processing gain - provided reasonable processing gains (code rates) are employed. Increasing the code rate does improve the initial rate of interference rejection, however.

The simulation results show that the bandwidths of the feedback loops in the array processor increase as the power in the input signals is increased. Unsatisfactory performance was observed when the input interference power levels were increased to the point where processing loop bandwidths exceeded the bandwidth of the pseudonoise code modulation. The array processor responds only to the interfering signal in this case and alters its amplitude and phase modulation to match the reference signal. The level of the desired signal at the array output under these conditions is shown to depend on the initial values of the weighting coefficients. The performance of practical array processors under high-level interference conditions is expected to differ from the simulation results because of limitations in the bandwidth and dynamic range of components used in the implementation.

The introduction of automatic gain control into the feedback loops of the simulation model resulted in smaller processing bandwidths and satisfactory interference rejection but unsatisfactory (slow) response to desired signal. It was shown that amplitude limiting in the error multipliers under large (initial) error signal conditions reduces the processing bandwidths only until the error becomes small.

The simulation results presented in this report provide basic information regarding the transient performance of adaptive array processors having a waveform-processed reference signal. The coverage, however, is incomplete in several respects. Further investigation is required to determine how processor performance degrades as the angular separation between the sources of the desired and interfering signals is reduced to small values, i.e., to determine the spatial resolution properties of the array. Processor performance when receiver noise is non-negligible and when the interfering signal has a non-zero spectral width should also be determined. Currently, experiments are being conducted to obtain information to augment the simulation results contained in this report. An experimental four-element adaptive array processor has been instrumented and initial feasibility tests performed. The data to be obtained from the experiments will provide a more complete basis for designing practical adaptive array processors.

APPENDIX I
CW PATTERN COMPUTATION

The array geometry and notation of Fig. 4 were used in computing the cw pattern. Antenna elements having ideal (isotropic) patterns were assumed; the received signal from each element was assumed to be a unit amplitude sinusoid of frequency $f_c + \Delta f$. The array output signal was represented as

$$(46)\quad V_\Sigma(t,\theta,\Delta f) = \mathrm{Re}\{\, \overline{V}_\Sigma(\theta,\Delta f)\, e^{j(\omega_c+\Delta\omega)t} \,\}$$

where the array output (sum) phasor is expressed as

$$(47)\quad \overline{V}_\Sigma(\theta,\Delta f) = \sum_{n=1}^{N/2} \left(\overline{W}_n\, e^{j\phi_n(\theta)} + \overline{W}_{-n}\, e^{-j\phi_n(\theta)}\right) \quad .$$

Each term in the summation corresponds to the phasor contribution of a pair of elements spaced equidistant from the array phase center. Coefficient $\overline{W}_n$ is the complex weighting coefficient at frequency $f_c + \Delta f$ of the nth element to the right of the array phase center; phase $\phi_n(\theta)$ represents the phase shift due to path delay to that element. When (47) is expressed in the notation used in Sec. IIIB the result is as follows:

$$(48)\quad \overline{V}_\Sigma(\theta,\Delta f) = \sum_{\ell=2,6,10,\cdots}^{2N-2} \left\{ \left[W_{\ell-1} - W_\ell \sin\left(\frac{\pi}{2}\frac{\Delta f}{f_c}\right) - j\, W_\ell \cos\left(\frac{\pi}{2}\frac{\Delta f}{f_c}\right)\right] x \right.$$

$$e^{j\left(\frac{\ell-2}{2}+1\right)\pi\, d_{\lambda_c}\left(1+\frac{\Delta f}{f_c}\right)\sin\theta}$$

$$+ \left[W_{\ell+1} - W_{\ell+2} \sin\left(\frac{\pi}{2}\frac{\Delta f}{f_c}\right) - j\, W_{\ell+2} \cos\left(\frac{\pi}{2}\frac{\Delta f}{f_c}\right)\right] x$$

$$\left. e^{-j\left(\frac{\ell-2}{2}+1\right)\pi\, d_{\lambda_c}\left(1+\frac{\Delta f}{f_c}\right)\sin\theta} \right\}.$$

APPENDIX II
COMPUTER SIMULATION PROGRAM

```
FTN,B,L,T
C                  CODED ADAPTIVE ARRAY SIMULATION
C
C
C      SPC=NO. OF SAMPLES PER CYCLE OF CARRIER AT CENTER FREQ.
C
C
C***ARRAY PROCESSOR PARAMETERS;
C
C      NUM=NO. OF ELEMENTS (EVEN INTEGER)
C      DIS=ELEMENT SEPARATION IN WAVELENGTHS AT CENTER FREQ.
C      GAIN=RATIO OF LOOP GAIN (K) TO ARRAY CENTER FREQ.
C      W(I)=CHANNEL I WEIGHTING COEFFICIENT
C
C***INPUT SIGNAL PARAMETERS;
C
C      THS=SIGNAL ARRIVAL ANGLE FROM BROADSIDE (SPATIAL DEG.)
C      THM=INTERFERENCE ARRIVAL ANGLE
C      AS=SIGNAL AMPLITUDE
C      AM=INTERFERENCE AMPLITUDE
C      FFS=SIGNAL FRACTIONAL FREQUENCY OFFSET
C      FFM=INTERFERENCE FRACTIONAL FREQUENCY OFFSET
C      ALS0=SIGNAL INITIAL PHASE AT ARRAY PHASE CENTER (DEG.)
C       ALM0=INTERFERENCE INITIAL PHASE AT ARRAY PHASE CENTER
C      CODR=RATIO OF CODE CLOCK FREQ. TO ARRAY CENTER FREQ.
C       DATR=RATIO OF DATA CLOCK FREQ. TO ARRAY CENTER FREQ.
C
C***REFERENCE SIGNAL/NETWORK PARAMETERS;
C
C      IDREF=TYPE OF REFERENCE SIGNAL GENERATION DESIRED
C                    1. FIXED
C                    2. LINEAR WAVEFORM PROCESSING
C                    3. LIMITED WAVEFORM PROCESSING
C
C      BWF1=RATIO OF 5.3-DB BANDWIDTH OF FILTER 1 TO CEN. FREQ.
C      BWF2=RATIO OF 5.3-DB BANDWIDTH OF FILTER 2 TO CEN. FREQ.
C      CDOFF=RATIO OF LOCAL CODE TIME DELAY TO CODE CLK. PERIOD
C      HOLD=NON-PROCESSING INTERVAL DURATION IN CARRIER CYCLES
C      AR=REFERENCE AMPLITUDE (IDREF=1)
C      FFR=REFERENCE FRACTIONAL FREQ. OFFSET
C      ALR0=REFERENCE INITIAL PHASE (DEG.)
C      GREF=TOTAL REFERENCE NETWORK GAIN (IDREF=2)
C      ARL=LIMITED REFERENCE AMPLITUDE (IDREF=3)
C
C***PROGRAM CONTROL PARAMETERS;
C
C      JWGHT=NO. OF SAMPLES PER WEIGHTING COEFF. PRINTOUT
C       JCNT=NO. OF SAMPLES PER STOP/CONTINUE CONTROLS
C      JPTRN=NO. OF SAMPLES PER PATTERN PRINTOUT
C      PFF=FRACTIONAL FREQ. OFFSET FOR PATTERN COMPUTATION
C       PTMIN=PATTERN MINIMUM ANGLE
C      PTMAX=PATTERN MAXIMUM ANGLE
C      PTINC=PATTERN INCREMENT ANGLE
C
```

```
C
C
C
      PROGRAM CAAS
      DIMENSION W(16),Y(16),FBK(16),PHSG(8),PHM(8),IC(10)
      COMMON FJ,IDREF,JHLD,JDLAY,SPC,PI,CODR
      COMMON AR,ARC,CODLO,SUM,REF,ERR
900   WRITE(1,10)
10    FORMAT(24H ENTER  SPC NUM DIS GAIN)
      READ(1,*)SPC,NUM,DIS,GAIN
      NUM1=NUM-1
      NUM2=2*NUM
      TK=GAIN/SPC
      WRITE(1,14)
14    FORMAT(23H ENTER  INITIAL WEIGHTS)
      READ(1,*)(W(I),I=1,NUM2)
      WRITE(1,16)
16    FORMAT(21H ENTER  AS AM THS THM)
      READ(1,*)AS,AM,THS,THM
      WRITE(1,18)
18    FORMAT(25H ENTER  FFS FFM ALS0 ALM0)
      READ(1,*)FFS,FFM,ALS0,ALM0
      PI=3.1415926
      PD1=PI*(1.+FFS)*DIS*SIN(THS*PI/180.)
      PD2=PI*(1.+FFM)*DIS*SIN(THM*PI/180.)
      DO 20 K=1,NUM1,2
      ZK=K
      PHSG(K)=ZK*PD1+ALS0*PI/180.
      PHSG(K+1)=-ZK*PD1+ALS0*PI/180.
      PHM(K)=ZK*PD2+ALM0*PI/180.
20    PHM(K+1)=-ZK*PD2+ALM0*PI/180.
      Q1=PI*(1.+FFS)/2.
      Q2=PI*(1.+FFM)/2.
      PSI=2.*PI*(1.+FFS)/SPC
      PMI=2.*PI*(1.+FFM)/SPC
      WRITE(1,22)
22    FORMAT(17H ENTER  CODR DATR)
      READ(1,*)CODR,DATR
      JCMAX=SPC/CODR+.5
      JDMAX=SPC/DATR+.5
      JCODE=JCMAX
      JDATA=JDMAX
      DO 24 I=1,4
      IC(I)=1
24    IC(I+4)=0
      IDATA=1
      IDT=-1
      ICODL=1
70    WRITE(1,72)
72    FORMAT(24H ENTER  JWGHT JCNT JPTRN)
```

```
      READ(1,*)JWGHT,JCNT,JPTRN
      WRITE(1,74)
74    FORMAT(29H ENTER  PFF PTMIN PTMAX PTINC)
      READ(1,*)PFF,PTMIN,PTMAX,PTINC
      PPO=PI*DIS*(1.+PFF)
      ESN=SIN(PI*PFF/2.)
      ECS=COS(PI*PFF/2.)
      KKM=((PTMAX-PTMIN)/PTINC)+1.5
      JW=0
      JCN=0
      JPAT=0
      FJ=-1.
      CALL REFRN
      CALL PHS(2,12)
      WRITE(1,80)SPC,NUM,DIS,GAIN,THS,THM,AS,AM,FFS,FFM,ALSO,
     CALMO,CODR,JCMAX,DATR,JDMAX,PFF
80    FORMAT(4HSPC=,F5.1,/,4HNUM=,I2,10X,4HDIS=,F7.3,5X,5HGAIN=,
     CE10.4,/,4HTHS=,F7.3,5X,4HTHM=,F7.3,/,3HAS=,F8.4,5X,3HAM=,
     CF8.4,/,4HFFS=,E10.4,2X,4HFFM=,E10.4,/,5HALSO=,F7.3,4X,
     C5HALMO=,F7.3,/,5HCODR=,E10.4,17X,6HJCMAX=,I6,/,5HDATR=,
     CE10.4,17X,6HJDMAX=,I6,/,4HPFF=,E10.4,///,"INITIAL WEIGHTS: _")
      CALL PHS(2,NUM2)
      WRITE(1,90)(W(I),I=1,NUM2)
90    FORMAT(32(E12.5,/,17X))
118   CALL PHS(4,0)
      CALL PHS(2,3)
      WRITE(1,100)
100   FORMAT(16X,6HW(J+1),8X,6HSUM(J),9X,6HREF(J),9X,6HERR(J),//)
C
C***********START OF LOOP HERE*********************
C
120   IF(JCODE-JCMAX)160,125
125   JCODE=0
      JDL=0
      ICODE=IC(6)+IC(7)
      DO 130 I=2,7
      II=9-I
130   IC(II)=IC(II-1)
      IF(ICODE-1)140,140,135
135   ICODE=0
140   IC(1)=ICODE
      IF(JDATA-JDMAX)150,145
145   JDATA=0
      JHLD=0
      IDATA=IDATA+IDT
      IDT=-IDT
150   IMOD=ICODE+IDATA
```

```
      IF(IMOD-1)155,157,155
155   ASC=AS
      GO TO (156,160,160),IDREF
156   ARC=AR
      GO TO 190
157   ASC=-AS
      GO TO (158,160,160),IDREF
158   ARC=-AR
      GO TO 190
160   IF(JDLAY-JDL)170,165,170
165   ICODL=ICODE
170   IF(ICODL)175,175,180
175   CODLO=1.
      GO TO 190
180   CODLO=-1.
190   DO 200 K=1,NUM
      I=2*K
      Y(I-1)=ASC*SIN(PHSG(K))+AM*SIN(PHM(K))
      Y(I)=ASC*SIN(PHSG(K)-Q1)+AM*SIN(PHM(K)-Q2)
      PHSG(K)=PHSG(K)+PSI
      IF(PHSG(K)-6.2831853)195,192
192   PHSG(K)=PHSG(K)-6.2831853
195   PHM(K)=PHM(K)+PMI
      IF(PHM(K)-6.2831853)200,198
198   PHM(K)=PHM(K)-6.2831853
200   CONTINUE
      SUM=0.
      DO 205 I=1,NUM2
205   SUM=SUM+Y(I)*W(I)
      CALL REFRN
320   DO 330 I=1,NUM2
      FBK(I)=Y(I)*ERR
330   W(I)=W(I)+TK*FBK(I)
      IF(JW-JWGHT)350,340
340   JW=0
      CALL PHS(2,1)
      WRITE(1,342)FJ,W(1),SUM,REF,ERR
342   FORMAT(2HJ=,F6.0,4(3X,E12.5))
      DO 344 I=2,NUM2
      CALL PHS(2,1)
      WRITE(1,346)W(I)
344   CONTINUE
346   FORMAT(11X,E12.5)
350   IF(JPAT-JPTRN)370,500
370   IF(JCN-JCNT)400,380
380   CALL PHS(2,1)
      WRITE(1,384)
384   FORMAT("CONTINUE?  ENTER 1=YES  0=NO  _")
      READ(1,*)JDEC
      IF(JDEC)799,799,390
```

```
390     JCN=0
        WRITE(1,72)
        READ(1,*)JWGHT,JCNT,JPTRN
400     FJ=FJ+1.
        JCODE=JCODE+1
        JDATA=JDATA+1
        JDL=JDL+1
        JHLD=JHLD+1
        JW=JW+1
        JCN=JCN+1
        JPAT=JPAT+1
        GO TO 120
C
C*********************END OF LOOP HERE; PATTERN PROG. BELOW******
C
500     JPAT=0
        PTH=PTMIN
        CALL PHS(3,2)
        CALL PHS(2,1)
        WRITE(1,502)FJ
502     FORMAT(2HJ=,F6.0,3X,7HPATTERN)
        DO 530  KK=1,KKM
        SMR=0.
        SMI=0.
        NUM4=NUM2-2
        DO 510 K=2,NUM4,4
        I=(K-2)/2+1
        DO 510 L=1,3,2
        LL=2-L
        ZZK=I*LL
        Z1=ZZK*PP0*SIN(PTH*PI/180.)
        Z1R=COS(Z1)
        Z1Q=SIN(Z1)
        LK=K-LL
        RR=W(LK)-W(LK+1)*ESN
        QQ=W(LK+1)*ECS
        SMR=SMR+Z1R*RR+Z1Q*QQ
510     SMI=SMI+Z1Q*RR-Z1R*QQ
        POWR=4.34295*ALOG(SMR*SMR+SMI*SMI)
        ARGU=SMI/SMR
        ANGH=57.2958*ATAN(ARGU)
        CALL PHS(2,1)
        WRITE(1,525)PTH,POWR,ANGH
525     FORMAT(8X,F7.2,5X,F9.4,5X,F9.4)
530     PTH=PTH+PTINC
        CALL PHS(3,2)
        GO TO 370
799     CALL PHS(5,0)
        WRITE(1,800)
800     FORMAT(32H ANOTHER RUN   ENTER 1=YES  0=NO)
        READ(1,*)JGO
        IF(JGO)999,999,900
999     STOP
        END
C
```

```
C
C     USE WITH PROGRAM CAAS: CODED ADAPTIVE ARRAY SIMULATION
C
C
C
C
      SUBROUTINE REFRN
      DIMENSION O1(4),O1T(4),O1TT(4),B1(4),EXCO1(4)
      DIMENSION O2(4),O2T(4),O2TT(4),B2(4),EXCO2(4)
      COMMON FJ,IDREF,JHLD,JDLAY,SPC,PI,CODR
      COMMON AR,ARC,CODLO,SUM,REF,ERR
      IF(FJ)25,207
25    WRITE(1,26)
26    FORMAT(13H ENTER  IDREF)
      READ(1,*)IDREF
      GO TO (30,40,40),IDREF
30    WRITE(1,32)
32    FORMAT(19H ENTER  AR FFR ALRO)
      READ(1,*)AR,FFR,ALRO
      PHR=ALRO*PI/180.
      PRI=2.*PI*(1.+FFR)/SPC
      GO TO 97
40    WRITE(1,42)
42    FORMAT(28H ENTER  BWF1 BWF2 HOLD CDOFF)
      READ(1,*)BWF1,BWF2,HOLD,CDOFF
      TAU1=1./BWF1
      TAU2=1./BWF2
      BT1=PI*BWF1/(2.*SPC)
      BT2=PI*BWF2/(2.*SPC)
      B1(1)=(4.*TAU1-2.)*BT1
      B1(2)=(4.*TAU1-1.)*BT1
      B1(3)=(4.*TAU1+1.)*BT1
      B1(4)=(4.*TAU1+2.)*BT1
      B2(1)=(4.*TAU2-2.)*BT2
      B2(2)=(4.*TAU2-1.)*BT2
      B2(3)=(4.*TAU2+1.)*BT2
      B2(4)=(4.*TAU2+2.)*BT2
      EX1=EXP(-2.*BT1)
      EX2=EXP(-2.*BT2)
      DO 44 I=1,4
      EXCO1(I)=EXP(-BT1)*COS(B1(I))
      EXCO2(I)=EXP(-BT2)*COS(B2(I))
      O1T(I)=0.
      O1TT(I)=0.
      O2T(I)=0.
44    O2TT(I)=0.
      DINT=0.
      VINT=0.
      JHOLD=HOLD*SPC+.5
      JHLD=0
      JDLAY=(CDOFF*SPC/CODR)+.5
```

```
      IF(IDREF-2)97,50,60
50    WRITE(1,52)
52    FORMAT(12H ENTER  GREF)
      READ(1,*)GREF
      SLOPE=GREF*PI*PI*BWF1*BWF2/(0.36*SPC*SPC)
      GO TO 97
60    WRITE(1,62)
62    FORMAT(11H ENTER  ARL)
      READ(1,*)ARL
      GLIM=ARL*PI*PI*BWF2/(2.4*SPC)
97    FJ=0.
      CALL TTYLF(5)
      CALL PHS(1,0)
      GO TO (98,104,110),IDREF
98    CALL PHS(2,2)
      WRITE(1,100)IDREF,AR,FFR,ALR0
100   FORMAT(6HIDREF=,I2,8X,15HFIXED REFERENCE,/,3HAR=,F8.4,5X,
     C4HFFR=,E10.4,2X,5HALR0=,F7.3)
      GO TO 118
104   CALL PHS(2,5)
      WRITE(1,106)IDREF,GREF
106   FORMAT(6HIDREF=,I2,8X,16HLINEAR REFERENCE,/,5HGREF=,F8.4)
      GO TO 114
110   CALL PHS(2,5)
      WRITE(1,112)IDREF,ARL
112   FORMAT(6HIDREF=,I2,8X,17HLIMITED REFERENCE,/,4HARL=,F8.4)
114   WRITE(1,116)BWF1,BWF2,CDOFF,JDLAY,HOLD,JHOLD
116    FORMAT(5HBWF1=,E10.4,17X,5HBWF2=,E10.4,/,6HCDOFF=,F8.4,
     C2X,6HJDLAY=,I6,/,5HHOLD=,F8.4,3X,6HJHOLD=,I6)
118   CALL PHS(3,10)
      GO TO 315
207   GO TO (310,210,210),IDREF
210   DIN=CODLO*SUM
      DO 220 I=1,4
220   O1(I)=EXCO1(I)*(2.*O1T(I)-DINT)-EX1*O1TT(I)+DIN
      OUT1=0.5*(O1(1)-O1(4))-O1(2)+O1(3)
      DINT=DIN
      DO 230 I=1,4
      O1TT(I)=O1T(I)
230   O1T(I)=O1(I)
      GO TO (310,240,250),IDREF
240   VIN=SLOPE*OUT1
      GO TO 280
250   VIN=SIGN(GLIM,OUT1)
280   DO 290 I=1,4
290   O2(I)=EXCO2(I)*(2.*O2T(I)-VINT)-EX2*O2TT(I)+VIN
      OUT2=0.5*(O2(1)-O2(4))-O2(2)+O2(3)
      VINT=VIN
      DO 300 I=1,4
      O2TT(I)=O2T(I)
300   O2T(I)=O2(I)
```

```
      REF=-COOLD*OUT2
      ERR=REF-SUM
      IF(JHLD-JHOLD)302,315
302   ERR=0.
      GO TO 315
310   REF=ARC*SIN(PHR)
      PHR=PHR+PRI
      IF(PHR-6.2831853)313,312
312   PHR=PHR-6.2831853
313   ERR=REF-SUM
315   RETURN
      END
      ENDS
```

APPENDIX III

In this appendix, the steady-state amplitude of the array output signal which minimizes the mean-square error when the reference signal is inverted for a fraction of the time is calculated. The steady-state array output signal is assumed to be of the form

$$S(t) = q\, M(t) \sin(\omega_c t + \phi) \tag{49}$$

where constant q is to be calculated and M(t) represents a +1, -1 amplitude modulation which is equivalent to the actual bi-phase (0^0,180^0) modulation. The reference signal is assumed to have unity amplitude and (+1,-1) amplitude coding $\hat{M}(t)$:

$$R(t) = \hat{M}(t) \sin(\omega_c t + \phi). \tag{50}$$

The error and squared-error are therefore given by

$$E(t) = [\hat{M}(t) - q\, M(t)] \sin(\omega_c t + \phi) \tag{51}$$

$$E^2(t) = [\hat{M}(t) - q\, M(t)]^2 \sin^2(\omega_c t + \phi).$$

The sine-squared factor has a constant time-average value of one-half independent of q. To proceed further, the modulations $\hat{M}(t)$ and M(t) will be assumed to differ in sign a fraction δ of the time. Figure 33 illustrates one possible distribution of the reference signal inversion periods contributing to the fraction δ; the distribution does not affect the average square error which is given by

$$\overline{E^2} = \frac{1}{2}\{[-(1+q)]^2\delta + (1-q)^2(1-\delta)\} \tag{52}$$

$$= \frac{1}{2}[4q\delta + (1-q)^2].$$

The value of q which minimizes the mean-square error is found from the equation

$$\frac{\partial \overline{E^2}}{\partial q} = 0 = \frac{1}{2}[(4\delta - 2(1 - q_{mse})]\ ; \tag{53}$$

$$q_{mse} = 1 - 2\delta.$$

The application of this result is illustrated by the following three examples.

Example 1. Square-wave Data Modulation

Assume that M(t) is an alternating +1, -1 (square-wave) data sequence which occurs at a rate f_d and that the data filter delay equals one-fourth the data bit period. The delayed sequence is $\hat{M}(t)$. The signs of M(t) and $\hat{M}(t)$ differ during the first quarter of each data bit; therefore, $\delta = 0.25$ and $q_{mse} = 0.5$. The suppression is 6 dB.

Example 2. Random Data Modulation

Assume that M(t) and $\hat{M}(t)$ are random binary data sequences and that the data filter delay is the same as in Example 1: one-fourth a data bit period. Since the rate of occurrence of the sign differences is one-half the data rate, it follows that $\delta = (0.5)(0.25) = 0.125$. Therefore, $q_{mse} = 0.75$ which corresponds to a 2.5 dB suppression.

Example 3. Random Code Timing Error

Assume that M(t) and $\hat{M}(t)$ are random binary code sequences which are identical except for a misalignment of their time bases by one-half of a code bit period: $\varepsilon = 0.5\ f_{cd}^{-1}$. Now, $\delta = (0.5\ f_{cd}^{-1})(0.5\ f_{cd}) = 1/4$ since the rate at which the reference signal is inverted equals one-half the code rate. Therefore, $q_{mse} = 1/2$; a 6 dB reduction.

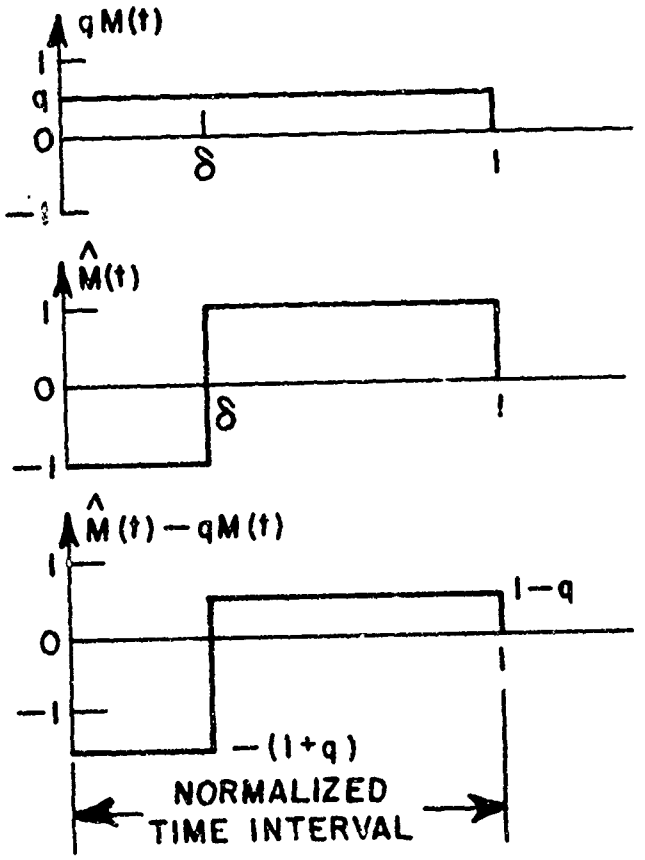


Fig. 33. Example waveforms for calculating mean-square error.

APPENDIX IV
ADAPTIVE ARRAY RESPONSE FOR A CW INPUT SIGNAL AND CW REFERENCE

For the special case of cw input and reference signals, i.e., when signals $y_i(t)$ and $R(t)$ are expressible as

$$y_i(t) = A_s \cos(\omega_c t + \theta_{si}); \quad i = 1, \cdots, 2N \tag{54}$$

$$R(t) = A_r \cos(\omega_c t + \theta_r) \tag{55}$$

the system of Eqs. (39) can be written in matrix form as

$$\left[\frac{dW(t)}{dt}\right] = \frac{K}{2} A_s A_r \begin{bmatrix} \cos(\theta_r - \theta_{s1}) \\ \vdots \\ \cos(\theta_r - \theta_{s2N}) \end{bmatrix} - \frac{K}{2} A_s^2 \left[C_{ik}\right] \left[W(t)\right] \tag{56}$$

Vector $[W(t)]$ is a column vector composed of the 2N weighting coefficients and $[C_{ik}]$ is a 2N x 2N matrix with elements given by

$$C_{ik} = \cos(\theta_{si} - \theta_{sk}) \, . \tag{57}$$

If the incident signal arrives broadside to the linear array then

$$\theta_{si} = \begin{cases} 0 & \text{i-odd} \\ -\frac{\pi}{2} & \text{i-even} \end{cases} \tag{58}$$

and the matrix C_{ik} is degenerate:

$$[C_{ik}] = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 1 & \cdots \\ 1 & 0 & 1 & 0 & 1 & 0 & \cdots \\ 0 & 1 & 0 & 1 & 0 & 1 & \cdots \\ \vdots & & & & \vdots & & \\ \vdots & & & & \vdots & & \\ \vdots & & & & \vdots & & \end{bmatrix} \, . \tag{59}$$

Clearly, from (56)

$$\dot{W}_1 = \dot{W}_3 = \cdots = \dot{W}_{2N-1} \tag{60}$$

$$\dot{W}_2 = \dot{W}_4 = \cdots = \dot{W}_{2N}$$

so that

$$W_3(t) = W_1(t) - W_1(0) + W_3(0) \qquad W_4(t) = W_2(t) - W_2(0) + W_4(0) \tag{61}$$

$$W_5(t) = W_1(t) - W_1(0) + W_5(0) \qquad W_6(t) = W_2(t) - W_2(0) + W_6(0)$$

$$\vdots \qquad\qquad\qquad \vdots$$

$$\text{etc.} \qquad\qquad\qquad \text{etc.}$$

Substitution of (61) into (56) gives the two equations of interest:

$$\dot{W}_1(t) = \frac{K}{2} A_s A_r \cos\theta_r - \frac{K}{2} A_s^2 \{N W_1(t) - N W_1(0) + \sum_{\substack{k=1 \\ k\text{-odd}}}^{2N-1} W_k(0)\} \tag{62}$$

$$\dot{W}_2(t) = \frac{K}{2} A_s A_r \cos(\theta_r + \tfrac{\pi}{2}) - \frac{K}{2} A_s^2 \{N W_2(t) - N W_2(0) + \sum_{\substack{\ell=2 \\ \ell\text{-even}}}^{2N} W_\ell(0)\} \; .$$

These equations have solutions given by

$$W_1(t) = \left[\frac{-A_r}{NA_s} \cos\theta_r + \frac{1}{N} \sum_{\substack{k=1 \\ k\text{-odd}}}^{2N-1} W_k(0) \right] e^{-\frac{NKA_s^2}{2} t} \tag{63}$$

$$+ \frac{A_r}{NA_s} \cos\theta_r - \frac{1}{N} \sum_{\substack{k=1 \\ k\text{-odd}}}^{2N-1} W_k(0) + W_1(0)$$

and

$$(64)\quad W_2(t) = \left[\frac{-A_r}{NA_s} \cos\left(\theta_r + \frac{\pi}{2}\right) + \frac{1}{N} \sum_{\substack{\ell=2 \\ \ell\text{-even}}}^{2N} W_\ell(0) \right] e^{-\frac{NKA_s^2}{2} t}$$

$$+ \frac{A_r}{NA_s} \cos\left(\theta_r + \frac{\pi}{2}\right) - \frac{1}{N} \sum_{\substack{\ell=2 \\ \ell\text{-even}}}^{2N} W_\ell(0) + W_2(0).$$

Equations (61), (63), and (64) represent the complete solution when the signal arrives broadside to the array.

Difficulty is encountered in attempting to obtain solutions applicable for arbitrary angle of incidence due to dependencies among the weighting coefficient derivatives. However, solutions have been obtained for off-broadside incidence at angles corresponding to 45^0 phase shift/element and 90^0 phase shift/element. The matrix C_{ik} is different from (59) in each of the two cases, but it is again degenerate of rank two. Relationships between the weighting coefficient derivatives can again be found and system (56) reduced to two equations. Their exponential solutions contain the same time constant as in (63) and (64) above. It would appear that the time constant is independent of the angle-of-incidence from these results.

REFERENCES

1. Special issue on active and adaptive arrays; IEEE Trans. on Antennas and Propagation, vol AP-12, December, 1964.

2. Brennan, D.G., "Linear Diversity Combining Techniques," Proc. IEEE, pp. 1075-1102, June, 1959.

3. Svoboda, D.E., "A Phase-Locked Receiving Array for High-frequency Communications Use," IEEE Trans. on Antennas and Propagation, pp. 207-215, March, 1964.

4. Applebaum, S.P., "Adaptive Arrays," Syracuse University Research Corp., SPL-769, June, 1964.

5. Shor, S.W.W., "Adaptive Technique to Discriminate Against Coherent Noise in a Narrow-band System," J. Acoust. Soc. Amer., January, 1966.

6. Widrow, B., Mantey, P.E., Griffiths, L.J., Goode, B.B., "Adaptive Antenna Systems," Proc. IEEE, December, 1967.

7. Gold, B. and Radar, C.M., Digital Processing of Signals, McGraw-Hill Book Co., 1969, pp. 51-54.