FALSE LOCK IN SAMPLED-DATA PHASE LOCK LOOPS

by

Hatcher Edward Chalkley

Thesis submitted to the Graduate Faculty of the

Virginia Polytechnic Institute

in partial fulfillment for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

APPROVED:

Chairman L. L. Grigsby

W. A. Blackwell

H. L. Krauss

H. K. Ebert

H. F. VanLandingham

August 1968

Blacksburg, Virginia

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF SYMBOLS

Symbol	First Appears In	Meaning
A	(2.1.1)	initial phase of s
A'	(2.3.2)	mean square estimate of A
B	(2.1.1)	frequency of s
B'	(2.3.2)	mean square estimate of B
B_m	(2.1.2) + 5 lines	maximum magnitude of B
$\underline{C}$	(2.4.3)	position of hypercube center
c	(1.1.7)	filter output
D	(2.3.5)	minimum of distance (2.3.1)
D_f	(2.8.11) + 2 lines	special value of D
e	(1.1.1)	phase detector output
I	(2.1.4)	cycle indicator
$I_1(\cdot)$	(2.10.4)	a particular integral
$\bar{I}_1(\cdot)$	(2.10.15)	a particular integral
$\underline{J}$	(2.4.3)	integer term in $\underline{C}$
k	(2.8.2)	number of ones in $(0,\cdots,0,1,\cdots,1)$
m	(1.1.5)	clamp circuit output
N	(2.1.3) + 3 lines	index of last sample
n	(2.1.2)	noise signal
P		dummy variable
P[·]	(2.1.5)	probability or probability density
r	(2.1.3)	received signal for ideal system

Symbol	First Appears In	Meaning
s	(2.1.1)	input signal
$\tilde{s}$	(3.2.1)	a prediction of s
T	(1.1.5)	sampling period
u	(2.1.2)	signal plus noise
V	(2.10.2)	a hyperspherical volume
$\bar{V}$	(2.10.16)	the complement of V
x		dummy variable
X_A	(2.6.2)	vector describing AB subspace
X_B	(2.6.3)	vector describing AB subspace
X_1	(3.3.9)	state variable
X_2	(3.3.9)	state variable
θ	(1.1.4)	signal phase
σ^2	(2.1.1) + 2 lines	variance of the noise
ϕ	(1.1.2)	phase detector transfer characteristic

1. INTRODUCTION

Sampled-data phase lock loops have various applications in today's electronic systems [BL1][GA1][MI1].†

Phase lock loops (PLLs) are subject to an improper mode of operation known as false lock. Researchers have investigated the problem of false lock for a continuous PLL [TA1]. No references have been found in the literature to the problem of false lock in sampled-data PLLs. This thesis considers the false lock characteristics of a sampled-data PLL which is required to acquire and track the phase of a signal of fixed but unknown frequency in the presence of additive noise.

1.1 Components in the PLL

Fig. 1.1 is a block diagram of a sampled-data phase lock loop.

The block labeled PD represents a phase detector. Throughout this work it will be assumed that the phase detector output, e, is given by:

$$e = \phi(\Delta\theta) \qquad (1.1.1)$$

where for any x

$$\phi(x) = x \qquad -\pi < x \leq \pi \qquad (1.1.2)$$

---

†Bracketed citations consisting of two letters and a number refer to references listed in the bibliography.

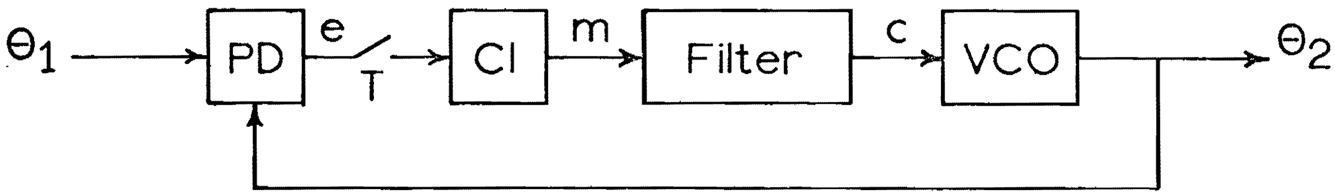

Fig. 1.1 Sampled-Data Phase Lock Loop

$$\phi(x + 2\pi) = \phi(x) \tag{1.1.3}$$

and

$$\Delta\Theta = \Theta_1 - \Theta_2 \tag{1.1.4}$$

This is the typical characteristic of multivibrator or sample-and-hold phase detectors, as those mentioned in Byrne [BY1].

The error signal, e, is sampled with uniform period, T, and its value at the sampling instants is stored by the clamp circuit, Cl, until the next sample. Thus the output, m, of the clamp is given by:

$$m(t) = e(iT) \tag{1.1.5}$$

for

$$iT < t \leq iT + T \tag{1.1.6}$$

The block labled VCO represents a voltage controlled oscillator whose output phase has the form

$$\frac{d}{dt}\,\Theta_2(t) = c(t) \tag{1.1.7}$$

Hereafter the input and output signals will be represented by their phases: Θ_1 and Θ_2.

The filter is the component which can be designed in order to obtain the required performance from the PLL.

1.2 Difficulties Caused by the Nonlinearity

The filter design would be straightforward were it not for the nonlinearity introduced by the phase detector. The periodic nature of the phase detector output as a function of the phase error makes it impossible for the PLL to distinguish between successive cycles of the input signal. The PLL must make a choice of the cycle of the input signal to which it will assign each received sample of e.

If the PLL processes the input signal so as to make a periodically increasing error in its choice of the cycle to which it assigns data, it is said to be in false lock. In this state, the PLL periodically "slips" a cycle in its tracking of the input.

Since false lock errors increase with time, they result in a much larger error in the estimate of the input phase than would be expected from the presence of the additive noise alone. Therefore it is of primary importance that false lock be avoided.

1.3 The Problem to be Investigated

It is the purpose of this work to investigate the false lock characteristics of the PLL with "ideal" false lock performance and then compare it with that of a sub-optimal system which would be practical to implement. This system is "ideal" in the sense that it has a minimum probability of false lock.

The ideal system will be derived in open-loop form. Its operation will be approximated to facilitate analysis. The approximation will be shown to introduce errors that approach zero as the variance of the noise approaches zero. The operation of the approximate system will be interpreted geometrically and this interpretation will be used to place bounds on its probability of false lock.

The ideal system will be shown to be unwieldy to implement, since its operation will require a large number of computations to be performed. Hence the performance of more practical systems is of interest. One such system will be analyzed in this thesis. The probability of its phase error exceeding one-half cycle has been found using a computer. These results will be presented.

It is assumed for all systems investigated that the phase noise on the input signal is additive Gaussian noise, independently distributed from sample to sample.

2. THE IDEAL SYSTEM

The form of the ideal open-loop estimator will be derived first and the transformation to a closed-loop system will be discussed later. The notation will be changed somewhat with this open-loop case to avoid confusion when later handling closed-loop systems. Fig. 2.1 shows the open-loop system.

2.1 The Form of the Ideal System

Assume that the system is receiving samples, s_i, at $t = i = 0,1,2,\cdots$ of a signal with phase

$$s(t) = A + Bt \tag{2.1.1}$$

which has been corrupted by additive Gaussian noise, n_i, of mean zero and variance σ^2, independently distributed for each sample, forming

$$u_i = s_i + n_i \tag{2.1.2}$$

This is a signal of fixed frequency with some initial phase. The initial signal phase, A, is assumed uniformly distributed on $[-\pi,\pi]$, i.e. the system has complete ignorance about the initial phase. The signal frequency, B, is assumed uniformly distributed on $[-B_m,B_m]$, i.e. it is known only that the

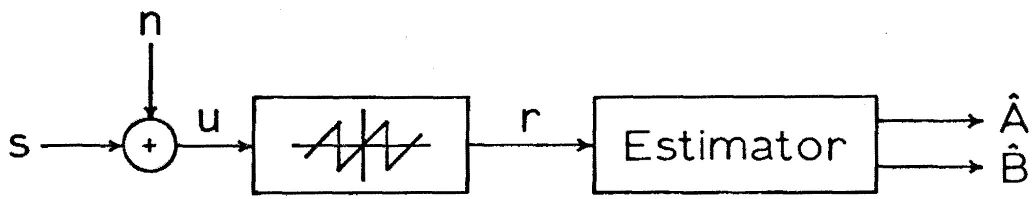

Fig. 2.1 The Open-Loop System

signal is within some frequency band. The center frequency of the band has been subtracted from the input and output signal phase for mathematical convenience. There is no loss of generality in doing this. The signal plus noise is passed through a nonlinearity which yields the received signal, r_i:

$$r_i = \phi(u_i) \tag{2.1.3}$$

This is the same as the operation of the phase detector in Fig. 1.1 when the feedback signal phase is zero.

The vector $(r_0,\cdots,r_N)$ of received signals will be denoted as $\underline{r}$.

Since no sampling system could be capable of distinguishing between signals whose values for B differed by an integral multiple of $2\pi/T$, B_m can be immediately restricted to $|B_m| < \pi$ without placing any impractical limitation on the results.

Let

$$u_i = r_i + 2\pi I_i, \tag{2.1.4}$$

where I_i is an integer indicating how many cycles of u_i have been "skipped" by the nonlinearity to yield r_i.

Because of the nonlinearity, there is not a one to one mapping from u_i to r_i. A given value of r_i could result from any one of an infinite number of u_i. Clearly, values of u_i greater in magnitude than $\pi + iB_m$ are less likely to be the correct choice than those with smaller magnitudes, since the original signal cannot be larger than this, but this restriction may still allow a large number of possible choices for I_i. Somehow the receiver must choose the correct value of I_i associated with each r_i. The possible choices of $(I_0,\cdots,I_N)$ will be denoted as $\underline{I}_j$ with elements $(I_{0j},\cdots,I_{ij},\cdots,I_{Nj})$. For this problem, the set of all possible $\underline{I}$ is countably infinite. Fortunately, all except a finite number may be eliminated from consideration without any increase in error.

It is desired that the receiver have the smallest possible probability of making an error in the selection of $\underline{I}$. Thus the receiver should select that $\underline{I}$ which, based on all available information, has the highest probability of being correct. The available information is contained in vector $\underline{r}$. The probability of $\underline{I}$ being correct is the conditional probability of the occurrence of $\underline{I}$, given the vector $\underline{r}$, and is denoted $P[\underline{I}|\underline{r}]$:

$$P[\underline{I}|\underline{r}] = \frac{P[\underline{I},\underline{r}]}{P[\underline{r}]} = \frac{\int P[\underline{I},\underline{r}|\underline{s}]\ P[\underline{s}]d\underline{s}}{P[\underline{r}]} \qquad (2.1.5)$$

The denominator of (2.1.5) is constant for given $\underline{r}$, so $P[\underline{I}|\underline{r}]$ is maximum when the numerator of (2.1.5) is maximum.

Since the noise, n_i, is independently distributed from sample to sample,

$$\int P[\underline{I},\underline{r}|\underline{s}]P[\underline{s}]d\underline{s}$$

$$= \int P[\underline{n} = \underline{r} + 2\pi\underline{I} - \underline{s}]P[\underline{s}]d\underline{s}$$

$$= \int \prod_{i=0}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp[-\frac{1}{2\sigma^2}(r_i + 2\pi I_i - s_i)]P[\underline{s}]d\underline{s}$$

$$= (\frac{1}{\sqrt{2\pi}\sigma})^{N+1} \int_{-B_m}^{B_m} \int_{-\pi}^{\pi} \exp[-\frac{1}{2\sigma^2} \sum_{i=0}^{N} (r_i + 2\pi I_i - A - iB)^2]dAdB \qquad (2.1.6)$$

Thus, the ideal receiver must decide on that $\underline{I}$ which maximizes the integral in (2.1.6) with given $\underline{r}$. This integral is discussed in the appendix and is quite involved.

In view of the difficulties involved in working directly with the double integral in (2.1.6), a different technique for selecting the most likely $\underline{I}$ will be considered. This new technique will be shown to give the same results for asymptotically small noise variance as those given by the ideal system. For larger noise variance, this approximate technique will sometimes rank the vectors $\underline{I}_j$ incorrectly compared to (2.1.6) and thus will perform worse than the ideal system. Its performance will be easier to evaluate, however, and thus can be used to place a lower bound on the performance of the ideal system. This approximate system will be called receiver R.

2.2 Receiver R

Consider the difference between performing the integration in (2.1.6) over the entire (A,B) plane rather than only within the rectangle ($|A| < \pi, |B| < B_m$). The integrand is sharply peaked about those values of A and B which imply lines passing close to the set of points $\underline{r} + 2\pi\underline{I}$. If each u_i is inside the limits

$$|u_i| < \pi + iB_m \qquad (2.2.1)$$

by "several" multiples of σ, then that portion of the integral outside the rectangle will be small compared with the total integral. In this case, the limited integral and the unlimited integral are essentially the same. If the noise variance is asymptotically small, these two integrals will be asymptotically equal for any $\underline{u}$ with components satisfying (2.2.1).

If any of the u_i are outside the limits (2.2.1) by as much as several σ, then there will be a drastic difference between the limited and the unlimited integral. Since the integrals of these $\underline{u}$ would rank quite low in the ordering according to their limited integral, they can be disregarded as possible candidates for that $\underline{u}$ which maximizes that integral. A more explicit criterion for differentiating between admissible and inadmissible $\underline{u}$ (and hence $\underline{I}$) will be developed later.

This then will be the technique used by receiver R. It will select that $\underline{I}$ from the admissible $\underline{I}_j$ which maximizes the integral (2.1.6) taken over the entire (A,B) plane.

2.3 The Error in the Approximation

The approximation of the operation of the ideal system by receiver R will result in significant errors only when the $\underline{I}$ maximizing (2.1.6) contains one or more u_i within several σ of the limit (2.2.1). As the noise variance is decreased, the probability of an input signal having samples in this region also decreases, since the size of the region decreases. Hence the probability of errors in approximation also decrease.

Consider an (N + 1)-dimensioned Euclidean space, denoted E^{N+1}, each of whose coordinates corresponds to the value of a function at time $t = i$, $i = 0,\cdots,N$. For given $\underline{I}$, the vector $\underline{u}$ can be completely described by a single point in E^{N+1}.

The locus of points describing all possible A + Bt is represented by lines of direction $(1,\cdots,1)$ passing through the point $(0,B,2B,\cdots,NB)$. Since the locus of intersection points $(0,\cdots,NB)$ is also a straight line, the two variables describe a two-dimensional subspace of E^{N+1}, which will be called the AB subspace.

From this point of view, the quantity

$$\sum_{i=0}^{N} (r_i + 2\pi I_i - A - iB)^2 \qquad (2.3.1)$$

which appears in (2.1.6) is the square of the Euclidean distance between the point $\underline{u}$ and the point in the AB subspace implied by the given A and B.

The minimum value of (2.3.1) is obtained for A' and B' given by the well-known linear regression equations:

$$\sum_{i=0}^{N} (A' + iB') = \sum_{i=0}^{N} u_i \qquad (2.3.2)$$

$$\sum_{i=0}^{N} i(A' + iB') = \sum_{i=0}^{N} iu_i . \qquad (2.3.3)$$

These equations have the solution

$$\begin{bmatrix} A' \\ B' \end{bmatrix} = \frac{1}{(N+1)(N+2)} \begin{bmatrix} 2(2N+1) & -6 \\ -6 & 12 \end{bmatrix} \begin{bmatrix} \sum_{i=0}^{N} u_i \\ \sum_{i=0}^{N} iu_i \end{bmatrix} \qquad (2.3.4)$$

This minimum,

$$D = \sum_{i=0}^{N} (u_i - A' - iB')^2, \qquad (2.3.5)$$

is the square of the length of the perpendicular from the point $\underline{u}$ to the AB subspace. Denote the point of intersection of this perpendicular with the AB subspace as (AB)'. This perpendicular distance may be different for different $\underline{I}$. Denote the distance associated with $\underline{I}_j$ as D_j.

The following theorem gives the equivalence of a ranking of the $\underline{u}_i$ according to the unlimited integral (2.1.6) and a ranking according to D_i.

Theorem 2.3.1

For given $\underline{r}$

$$\int_{-\infty}^{\infty}\!\!\int \exp[-\frac{1}{2\sigma^2}\sum_{i=0}^{N}(r_i + 2\pi I_{ij} - A - iB)^2]\,dA\,dB$$

$$> \int_{-\infty}^{\infty}\!\!\int \exp[-\frac{1}{2\sigma^2}\sum_{i=0}^{N}(r_i + 2\pi I_{ik} - A - iB)^2]\,dA\,dB \qquad (2.3.6)$$

if and only if

$$D_j < D_k \qquad (2.3.7)$$

Proof:

Set up a polar coordinate system (ρ,θ) in the AB subspace whose origin is at the point (AB)'. Thus each choice of I_j will have its associated $(AB)_j'$ and $(\rho,\theta)_j$. The distance from $\underline{u}_j$ to some point $(\rho,\theta)_j$ in the AB subspace is $D^2 + \rho^2$. The integrals above may be expressed in the form:

$$2\pi\int_0^\infty \exp[-\frac{1}{2\sigma^2}(D_j^2 + \rho^2)]\rho d\rho \qquad (2.3.8)$$

For given ρ, $D_j < D_k$ implies

$$\exp[-\frac{1}{2\sigma^2}(D_j^2 + \rho^2)] > \exp[-\frac{1}{2\sigma^2}(D_k^2 + \rho^2)] \qquad (2.3.9)$$

and since this is true over the whole range of integration the theorem is proved.

Theorem 2.3.1 allows a considerable simplification in the mechanics of implementing receiver R. Rather than evaluating the necessary integrals directly, it is only necessary to calculate the distances D in order to select the proper $\underline{I}$. This calculation also offers us a simple way of eliminating those $\underline{u}$ which will have a small integral (2.1.6) because some or all of the u_i are outside of the A,B rectangle. If, for a given $\underline{I}_j$, A' or B' exceeds the limits on the original signal, i.e. if $|A'| > \pi$ or $|B'| > B_m$, then the distance D_j will not be computed for the perpendicular distance from $\underline{u}_j$ to the AB subspace, but will be computed from $\underline{u}_j$ to the closest point in the AB subspace which meets the limits. For example, if for $\underline{I}_{89}$, A' = 0.3 and B' = 3.0, but $B_m = 2.5$, then D_j will be taken as the distance to (AB)'' where B'' = 2.5, and A'' satisfies

$$\left. \frac{\partial \sum_{i=0}^{N} (u_i - A - iB)^2}{\partial A} \right|_{B=2.5} = 0 \qquad (2.3.10)$$

which is the same as (2.3.2) with B = 2.5.

It is now clear that many values of I_i need never be considered at all. Let I_{imax} be the smallest I_i such that

$$2\pi I_i - \left| r_i \right| - \pi - iB_m > 0. \qquad (2.3.11)$$

Then there is no vector $\underline{I}_j$ with $\left| I_{ij} \right| > I_{imax}$ for which

$$\sum (r_i + 2\pi I_i - A' - iB')^2 < \sum (r_i + 2\pi I_{imax} - A' - iB')^2. \qquad (2.3.12)$$

Therefore the receiver need only consider the

$$\prod_{i=0}^{N} (2I_{imax} + 1) \qquad (2.3.13)$$

possible $\underline{I}_j$ which meet this limitation.

2.4 Geometrical Interpretation of Operation

The representation of the signals in E^{N+1} will again be used. The reader may find it helpful to refer to Fig. 2.2 where the representation is illustrated for $N = 2$, $B_m = \pi$.

The original signal at any given sample time is limited to

$$|s_i| \leq \pi + iB_m \qquad (2.4.1)$$

Therefore the original signal must be within the edges of the rectangular hyperspace

$$|x_i| \leq \pi + iB_m \qquad i = 0,1,\cdots,N. \qquad (2.4.2)$$

This hyperspace may be divided into hypercubes with edge length 2π and centers at

$$\underline{C}_j = 2\pi(0,J_{1j},J_{2j},\cdots,J_{Nj}) \qquad (2.4.3)$$

where J_{ij} is an integer and

$$|J_{ij}| \leq \frac{iB_m + \pi}{2\pi}\;. \qquad (2.4.4)$$

In the example illustrated in Fig. 2.2 $B_m = \pi$ and there are nine such cubes.

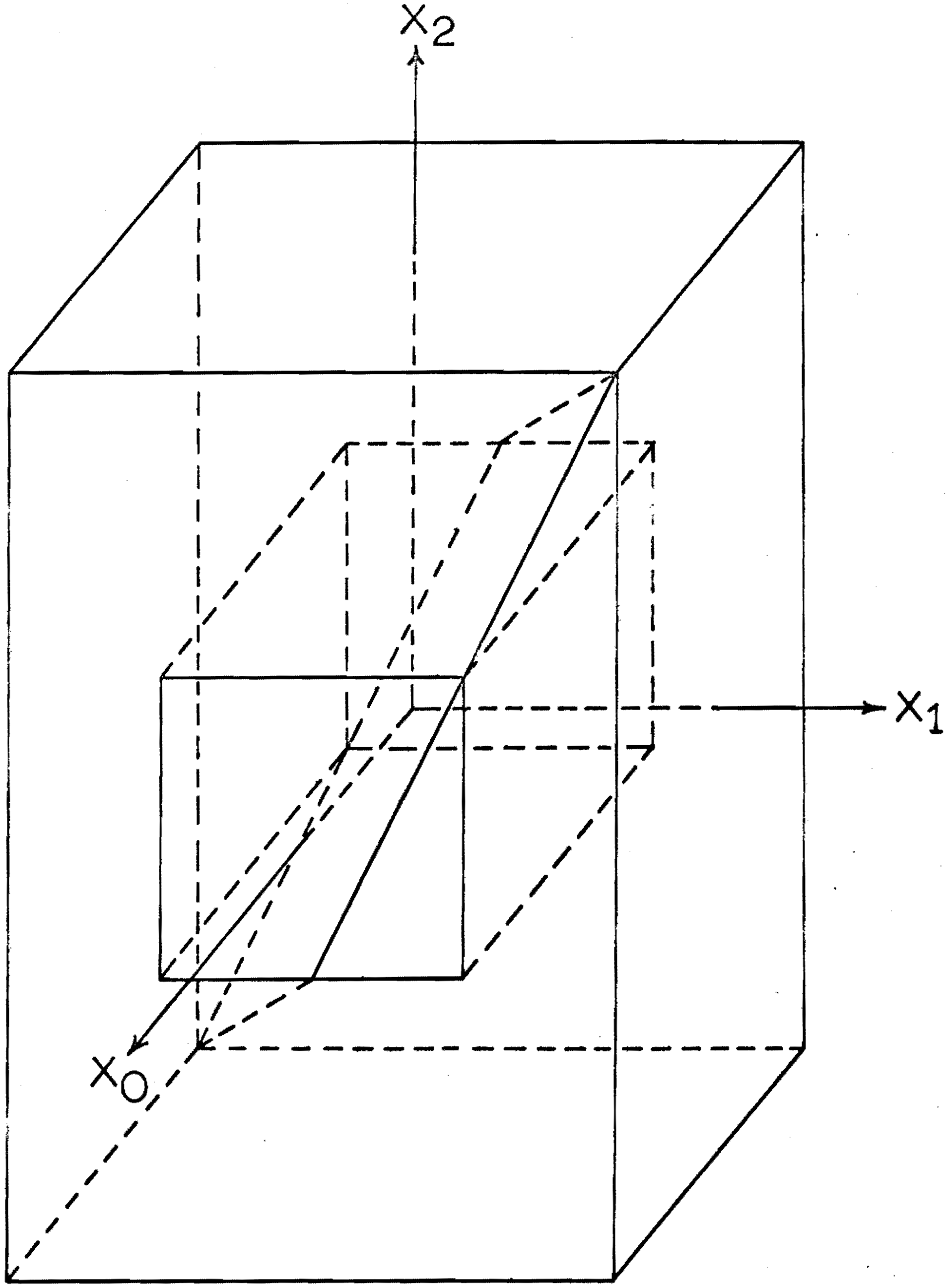

Fig. 2.2 N=2. The AB-plane within a rectangular space defined by the signal range for each of the three samples with $B_m=\pi$. The center hypercube is shown.

The original signal lies somewhere in the AB subspace, a two dimensional subspace of E^{N+1}. This subspace intersects some, but not all, of the hypercubes mentioned above. The noise added to the signal may drive the point $\underline{u}$ anywhere in the rectangular hyperspace or even outside of the rectangular boundaries. The action of the periodic nonlinearity then delivers to the receiver the coordinates, $\underline{r}$, of the point $\underline{u}$ with respect to the center of the hypercube in which it lies, but no information about in which cube it lies. Thus the space of the received signal, r, is a single hypercube of N+1 dimensions.

The receiver makes its decision selecting the most likely hypercube by placing a point of the given coordinates in every possible hypercube, and selecting that hypercube in which the point lies closest to the AB subspace. The probability of the receiver's making an error in this selection is the probability that the noise will drive the point $\underline{u}$ into a region which is closer to the AB subspace in the wrong hypercube.

Consider how sections of the AB subspace are oriented with respect to the hypercube they intersect. This will be their orientation in the space of the received signal, r. They are, of course, all parallel and spaced in some regular pattern. For example, in the case N = 2, the five different sections of the AB plane (within the limit $|A| \leq \pi, |B| \leq \pi$) intersect their respective cubes so as to form three evenly spaced parallel planes when all shifted to the same cube. This is illustrated in Fig. 2.3. The pattern is not as simple for N greater than 2. Fig. 2.4 illustrates this for N = 3. It shows the positions of shifted sections of the AB plane in the two-dimensional subspace orthogonal to the AB plane.

2.5 False Lock in Receiver R

The probability that the receiver will false lock is related to the probability that the component of the noise orthogonal to the AB plane will drive the signal to a position closer to the wrong section of the AB plane in the space of the received signal, $\underline{r}$. In this case the receiver will select $\underline{I}$ improperly. It will be shown that not all errors in the selection of $\underline{I}$ should be classified as false lock. A simple criterion of distinction will be developed. This will allow bounds to be placed on the distance from the AB plane to the nearest shifted region causing false lock. This in turn will be used to place bounds on the probability to false lock.

2.6 The Distance D

In this section the perpendicular distance from the AB subspace to the shifted regions of the AB subspace will be determined.

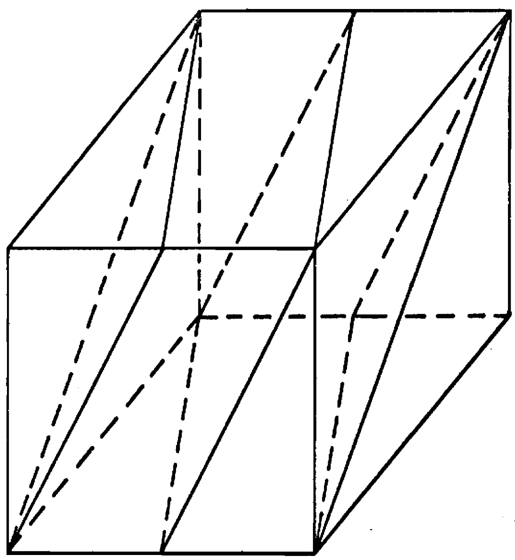

Fig. 2.3 N=2. The locations of segments of the AB-plane within the space of received signals.

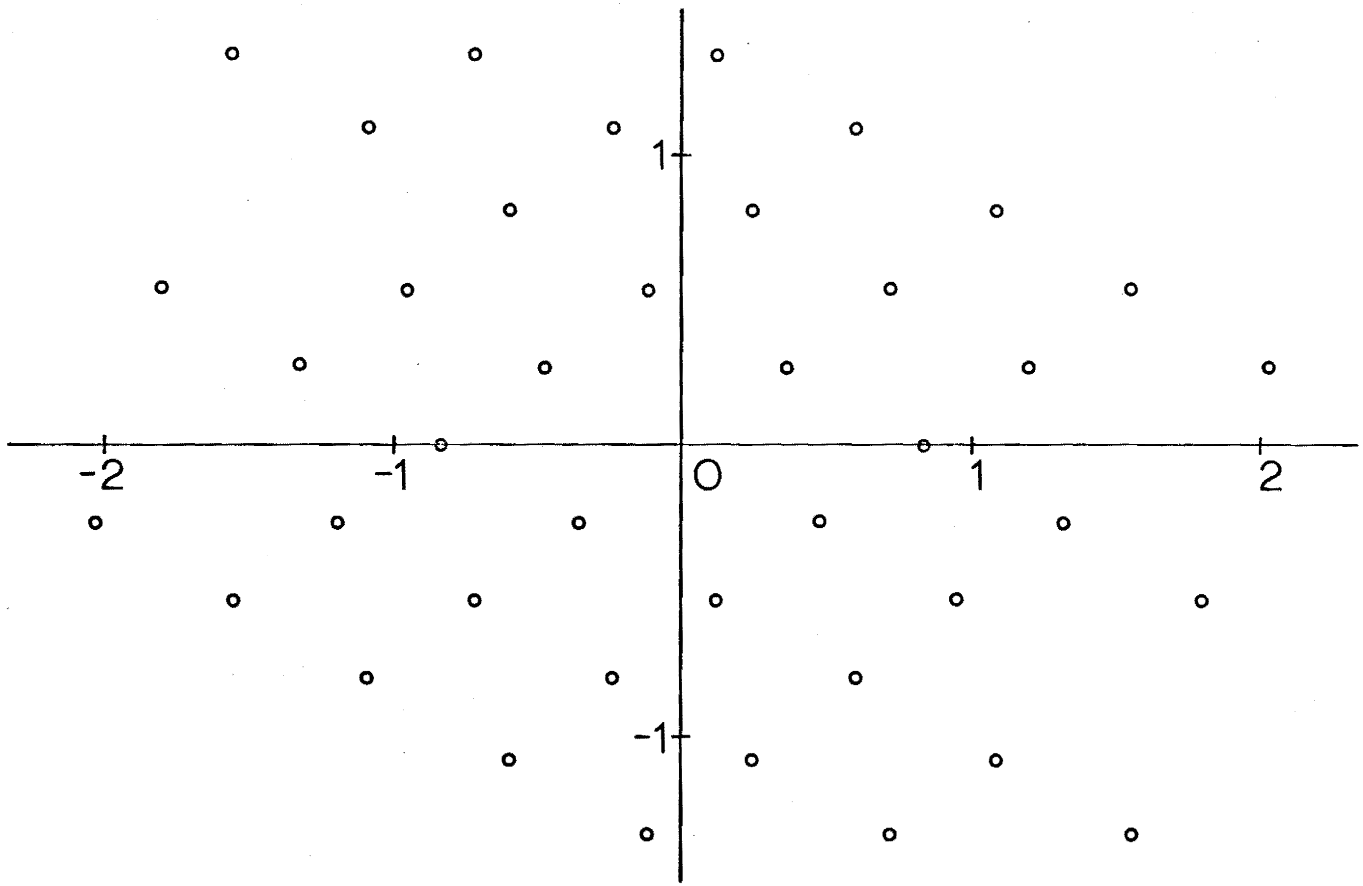

Fig. 2.4 N=3. Projection of the hypercube centers on the two-dimensional subspace orthogonal to the AB plane.

Since the position of a point within the AB subspace does not affect its perpendicular distance to any other point, the point A = 0, B = 0, i.e. the origin of the space can, for convenience, be used in all calculations. Its perpendicular distance from the center of each hypercube in the space will be calculated.

The coordinates of the center of a hypercube with respect to the origin of the space was given in (2.4.3) as:

$$\underline{C}_j = 2\pi(0, J_{1j}, J_{2j}, \cdots, J_{Nj})$$

All possibly different distances will be considered if the integers J_{ij} have the maximum magnitude:

$$|J_{ij}| \leq \frac{1 + i}{2} \qquad (2.6.1)$$

For every $\underline{C}$ outside this range, there is another $\underline{C}$ inside this range with exactly the same distance.

Two orthonormal vectors describing the AB subspace are:

$$\underline{X}_A = \frac{1}{\sqrt{N + 1}}(1, 1, \cdots, 1) \qquad (2.6.2)$$

and:

$$\underline{X}_B = \sqrt{\frac{12}{N(N+1)(N+2)}}\left[(0, 1, 2, \cdots, N) - \frac{N}{2}(1, 1, \cdots, 1)\right] \qquad (2.6.3)$$

The component of $\underline{C}_j$ perpendicular to the AB subspace is:

$$\underline{D}_j = \underline{C}_j - (\underline{C}_j, \underline{X}_A)\underline{X}_A - (\underline{C}_j, \underline{X}_B)\underline{X}_B \qquad (2.6.4)$$

where $(\underline{X}, \underline{Y})$ denotes the inner product operation:

$$(\underline{X}, \underline{Y}) = \sum_{i=0}^{N} X_i Y_i \qquad (2.6.5)$$

This reduces to:

$$\begin{aligned}(\pi/2)\underline{D}_j = {} & (0, J_{1j}, J_{2j}, \cdots, J_{Nj}) \\ & - \{(2N+1)\sum_{i=1}^{N} J_{ij} - 3\sum_{i=1}^{N} iJ_{ij}\}\frac{3}{(N+1)(N+2)}(1,1,\cdots,1) \\ & + \{\frac{N}{2}\sum_{i=1}^{N} J_{ij} - \sum_{i=1}^{N} iJ_{ij}\}\frac{12}{N(N+1)(N+2)}(0,1,2,\cdots,N)\end{aligned} \qquad (2.6.6)$$

The square of the length of this vector is:

$$\begin{aligned}|(\pi/2)\underline{D}|^2 = {} & \sum J_{ij}^2 - \frac{1}{(N+1)(N+2)}\{(\sum J_{ij})^2(4N+2) \\ & - 12(\sum J_{ij})(\sum iJ_{ij}) + \frac{12}{N}(\sum iJ_{ij})^2\}\end{aligned} \qquad (2.6.7)$$

It would be desirable to find the value of D for every $\underline{J}$ which represents an error in the selection of $\underline{I}$ resulting in false lock. However, because of the large number of such different $\underline{J}$, the work presented here has been directed toward finding a lower bound on such D.

2.7 Restrictions on $\underline{J}$

Since the probability of false lock is related only to the orthogonal distance from $\underline{C}$ to the AB plane, the location in the plane of original signal is irrelevant. For simplicity, it will henceforth be assumed that the original signal is zero for all time.

For every point (A,B) in the AB plane, that value of $\underline{J}$ can be found which characterizes the hypercube center closest to (A,B). This can be done simply by minimizing each individual component of the difference vector:

$$\min_{J_i} (J_i - A - iB). \qquad (2.7.1)$$

This will yield a $\underline{J}$ which increases (or decreases) monotonically in a periodic or almost periodic manner. The qualification "almost" in the last sentence is used to allow those cases in which, for some i, the minimization of (2.7.1) will occur for two values of J_i, and either may used in $\underline{J}$. Since this is the property expected of vectors $\underline{I}$ resulting in false lock, the distance D associated with these vectors are the distances that should be used to bound the probability of false lock. It is not known which of these vectors has the smallest value of D, but it is known that there is no vector outside this class with a value of D smaller than those inside this class.

For reasons which will soon become apparent, it is necessary to place an additional qualification on those $\underline{J}$ which will be considered as leading to false lock. This qualification will be made on the basis of the amount of phase error caused by the error in slope for a simple linear minimum mean-square estimator whose input is $\underline{J}$. The unbiased minimum mean-square-error estimate of B is given in (2.3.4). The total phase error caused by the error in B is:

$$NB' = \frac{2\pi}{(N+1)(N+2)} (-6N\sum J_i + 12\sum iJ_i). \qquad (2.7.2)$$

It will be said that for $\underline{J}$ to be considered as resulting in false lock,

$$NB' > \pi. \qquad (2.7.3)$$

It is not meant to be implied that the PLL must actually use a minimum MSE estimator in processing the signals $\underline{u}$

to estimate A and B. This work deals only with the selection of $\underline{I}$ and places no restrictions on the other processing steps.

Since D in (2.7.2) is independent of the sign of $\underline{J}$, investigation may be limited to positive J_i without loss of generality. In addition, since the addition of a constant to all terms of $\underline{J}$ does not change its orthogonal distance to the AB plane, the first term of $\underline{J}$ may be taken to be zero, or any other convenient value.

2.8 The Case $\underline{J} = (0,\cdots,1)$

The simplest form the vector $\underline{J}$ could take is:

$$\underline{J} = (0,0,\cdots,0,1,1,\cdots,1) \tag{2.8.1}$$

The number of components is N + 1. Let k = number of ones. Then:

$$\sum_{i=0}^{N} J_i = \sum_{i=0}^{N} J_i^2 = k \tag{2.8.2}$$

and

$$\sum_{i=0}^{N} iJ_i = \frac{1}{2}(2kN - k^2 + k). \tag{2.8.3}$$

Substitution into (2.6.7) yields:

$$N(N+1)(N+2)(D/2\pi)^2 = -3k^4 + 6(N+1)k^3 - (4N^2 + BN + 3)k$$

$$+ (N^3 + 3N^2 + 2N)k \qquad (2.8.4)$$

Examination of the first and second partial derivatives with respect to k shows a local minimum at

$$k = (N + 1)/2 \qquad (2.8.5)$$

and local maxima at

$$k = \frac{N+1}{2} \pm \frac{\sqrt{3}}{6}\sqrt{N^2 + 4N + 3} \qquad (2.8.6)$$

The fact that k must be an integer leads to the division of this problem into two cases:

For N odd, (N+1)/2 is an integer, and D has a local minimum at

$$k = \frac{N + 1}{2} \qquad (2.8.7)$$

At this value of k,

$$(D/2\pi)^2 = \frac{(N-1)(N+1)(N+3)}{16N(N+2)} \qquad (2.8.8)$$

For N even, the local minima are at the adjacent points:

$$k = N/2 \tag{2.8.9}$$

and

$$k = (N + 2)/2 \tag{2.8.10}$$

At both these values of k,

$$(D/2\pi)^2 = \frac{N(N + 2)}{16(N + 1)} \tag{2.8.11}$$

Let the values of D given in (2.8.8) and (2.8.11) be denoted D_f. Substitution of D_f into (2.8.4) shows that D also equals D_f when:

For N odd

$$k = \frac{N+1}{2} \pm \frac{1}{6}\sqrt{N^2 + 2N + 3} \tag{2.8.12}$$

and for N even

$$k = \frac{N+1}{2} \pm \frac{\sqrt{3}}{6}\sqrt{2N^2 + 4N + 3} \tag{2.8.13}$$

The restriction

$$1 \leq k \leq N \tag{2.8.14}$$

can be met by (2.8.12) and (2.8.13) when

$$N \geq 10 \tag{2.8.15}$$

with equality at both limits for N = 10. Thus, for $N \geq 11$, there are some values of $\underline{J}$ with D less than D_f.

Substitution of (2.8.12) and (2.8.13) into (2.7.2) shows that for the values of k given by these two equations, the corresponding NB' is:

for N odd

$$NB' = \pi \frac{N^2 + 2N - 3}{N^2 + 2N + 3} < \pi \tag{2.8.16}$$

for N even

$$NB' = \pi \frac{N}{(N+1)} < \pi \tag{2.8.17}$$

Values of $\underline{J}$ with smaller corresponding values for D also have smaller corresponding values of NB'.

Thus it can be seen why it is desirable to place the restriction of (2.7.3) on $\underline{J}$. To not do so would be to use an unduly conservative value for D, when that value corresponds to a $\underline{J}$ which causes a total tracking error of less than one-half cycle.

2.9 Other Vectors

No general algebraic expression has been found which expresses the values of D and NB' of a vector more complicated than $(0,\cdots,1)$ when N is not fixed.

A computer program has been written to search the class of all possible vectors for any which simultaneously satisfy the conditions:

$$D < D_f \tag{2.9.1}$$

$$NB' > \pi \tag{2.9.2}$$

For the cases $2 \leq N \leq 20$ no such vectors have been found. It is hypothesized that no such vectors exist for any N.

It has been found that for two vectors:

$$(0,1,1,2,2,3,3,\cdots) \tag{2.9.3}$$

and

$$(0,0,1,1,2,2,3,3,\cdots) \tag{2.9.4}$$

$D = D_f$. It can be shown algebraically that this is true for all N.

2.10 Probability of Noise Within a Hypersphere

In order to evaluate the probability of false lock, it is necessary to evaluate integrals of the form:

$$I_1(n,\sigma,P) = \iint_V \cdots \int (2\pi\sigma^2)^{-\frac{n+1}{2}} \exp[-\frac{1}{2\sigma^2} \sum_{i=0}^{n} x_i^2]$$

$$dx_0 dx_1 \cdots dx_n \qquad (2.10.1)$$

where the integration is over the hyperspherical volume:

$$V: \quad \sum_{i=0}^{n} x_i^2 \leq P. \qquad (2.10.2)$$

Making the change of variables:

$$\begin{aligned}
X_0 &= \rho \sin(\alpha_1) \sin(\alpha_2) \sin(\alpha_3) \cdots \sin(\alpha_n) \\
X_1 &= \rho \cos(\alpha_1) \sin(\alpha_2) \sin(\alpha_3) \cdots \sin(\alpha_n) \\
X_2 &= \rho \cos(\alpha_2) \sin(\alpha_3) \cdots \sin(\alpha_n) \\
X_3 &= \rho \cos(\alpha_3) \cdots \sin(\alpha_n) \\
&\vdots \\
X_{n-1} &= \rho \cos(\alpha_{n-1}) \sin(\alpha_n) \\
X_n &= \rho \cos(\alpha_n)
\end{aligned}$$

yields the form:

$$I_1(n,\sigma,P) = \int_0^{\pi}\cdots\int_0^{\pi}\int_0^{2\pi}\int_0^{P} (2\pi\sigma^2)^{-\frac{n+1}{2}} \exp[-\frac{\rho^2}{2\sigma^2}]\,\rho^n \sin\alpha_2 \sin^2\alpha_3$$

$$\cdots \sin^{n-1}\alpha_n\, d\rho\, d\alpha_1\, d\alpha_2 \cdots d\alpha_n \qquad (2.10.4)$$

Using the relation

$$\int_0^{\pi} \sin^n x dx = \frac{1\cdot 3\cdot 5\cdots(n-1)}{2\cdot 4\cdot 6\cdots n}\,\pi \qquad n \text{ even integer} \qquad (2.10.5)$$

$$= \frac{2\cdot 4\cdot 6\cdots(n-1)}{1\cdot 3\cdot 5\cdots n}(2) \qquad n \text{ odd integer} \qquad (2.10.6)$$

this may be reduced to the forms

n odd:

$$I_1(n,\sigma,P) = 2\frac{(\pi)^{\frac{n+1}{2}}}{(\frac{n-1}{2})!}\int_0^{P} (2\pi\sigma^2)^{-\frac{n+1}{2}} \exp(-\frac{\rho^2}{2\sigma^2})\,\rho^n d\rho \qquad (2.10.7)$$

n even:

$$I_1(n,\sigma,P) = (2)^n(\pi)^{\frac{n}{2}}\,\frac{(\frac{n}{2}-1)!}{(n-1)!}\int_0^{P} (2\pi\sigma^2)^{-\frac{n+1}{2}} \exp(-\frac{\rho^2}{2\sigma^2})\,\rho^n d\rho$$

$$(2.10.8)$$

The form

$$\int \rho^n \exp[-\frac{\rho^2}{2\sigma^2}]\, d\rho \qquad (2.10.9)$$

may be integrated by parts to yield the recursion formula:

$$\int \rho^n \exp\left[-\frac{\rho^2}{2\sigma^2}\right] d\rho = -\sigma^2 \rho^{n-1} \exp\left(-\frac{\rho^2}{2\sigma^2}\right) + \sigma^2 (n-1)$$

$$\cdot \int \rho^{n-2} \exp\left(-\frac{\rho^2}{2\sigma^2}\right) d\rho \qquad (2.10.10)$$

Successive application of this formula yields:

$$\int \rho^n \exp\left[-\frac{\rho^2}{2\sigma^2}\right] d\rho$$

for n even

$$= -\frac{1}{n+1} \exp\left(-\frac{\rho^2}{2\sigma^2}\right) \sum_{i=1}^{\frac{n}{2}} \sigma^{2i} \rho^{n+1-2i} \prod_{j=1}^{i} (n+3-2j)$$

$$+ \sigma^n \frac{(n-1)!}{2^{\frac{n}{2}-1} (\frac{n}{2}-1)!} \sqrt{\frac{\pi\sigma^2}{2}} \operatorname{erf}\left(\frac{\rho}{\sqrt{2}\sigma}\right) \qquad (2.10.11)$$

for n odd

$$= -\frac{1}{n+1} \exp\left(-\frac{\rho^2}{2\sigma^2}\right) \left[\sum_{i=1}^{\frac{n+1}{2}} \sigma^{2i} \rho^{n+1-2i} \prod_{j=1}^{i} (n+3-2j) \right] \qquad (2.10.12)$$

Inserting the limits 0 and P yields:

n even:

$$- \frac{1}{n+1} \exp(- \frac{P^2}{2\sigma^2}) \sum_{i=1}^{\frac{n}{2}} \sigma^{2i} P^{n+1-2i} \prod_{j=1}^{i} (n+3-2j)$$

$$+ \sigma^n \frac{(n-1)!}{2^{\frac{n}{2}-1} (\frac{n}{2}-1)!} \sqrt{\frac{\pi\sigma^2}{2}} \operatorname{erf}(\frac{P}{\sqrt{2}\sigma}) \qquad (2.10.13)$$

n odd:

$$\frac{1}{n+1} \sigma^{n+1} \prod_{j=1}^{\frac{n+1}{2}} (n+3-2j) - \frac{1}{n+1} \exp(- \frac{P}{2\sigma^2})$$

$$\cdot [\sum_{i=1}^{\frac{n+1}{2}} \sigma^{2i} P^{n+1-2i} \prod_{j=1}^{i} (n+3-2j)] \qquad (2.10.14)$$

The probability that the noise will lie outside the hypersphere is found by substituting the limits $]_P^\infty$ instead of $]_0^P$. Use of this form is sometimes preferable for numerical reasons.

Let

$$\overline{I}_1(n,\sigma,P) = 1 - I(n\ \sigma,P) = \int\int_{\overline{V}}\cdots\int (\text{same as before}) \qquad (2.10.15)$$

where

$$\overline{V}: \quad \sum_{i=0}^{n} x_i^2 \geq P \qquad (2.10.16)$$

This changes only the term:

$$\int_{P}^{\infty} \rho^{n} \exp[-\frac{\rho^{2}}{2\sigma^{2}}]d\rho \qquad (2.10.17)$$

which then has the value:

n even

$$\frac{1}{n+1} \exp(-\frac{P^{2}}{2\sigma^{2}}) \sum_{i=1}^{\frac{n}{2}} \sigma^{2i} P^{n+1-2i} \prod_{j=1}^{i} (n+3-2i)$$

$$+ \sigma^{n} \frac{(n-1)!}{2^{\frac{n}{2}-1}(\frac{n}{2}-1)!} \sqrt{\frac{\pi\sigma^{2}}{2}} \operatorname{erfc}(\frac{P}{\sqrt{2}\sigma}) \qquad (2.10.18)$$

n odd

$$\frac{1}{n+1} \exp(-\frac{P^{2}}{2\sigma^{2}}) \sum_{i=1}^{\frac{n+1}{2}} \sigma^{2i} P^{n+1-2i} \prod_{j=1}^{i} (n+3-2j) \qquad (2.10.19)$$

2.11 Probability of False Lock

Using previous results, an upper bound may now be placed on the probability that the ideal system will false lock. If the component of the noise orthogonal to the AB subspace has a magnitude less than $D_f/2$, the ideal receiver will not be in false lock. For

$$N \leq t < N + 1 \tag{2.11.1}$$

the system has N + 1 samples to process and the above mentioned component of the noise is normally distributed in an N - 1 dimensioned space. Hence the probability of false lock must be bounded above by:

$$P[FL,N] \leq \bar{I}_1 (N-2,\sigma,D_f/2) \tag{2.11.2}$$

2.12 Feedback in the Ideal Receiver

At this point it should be remarked that the use of feedback within the ideal receiver, or even within receiver R, would not simplify the calculations that these receivers make. Referring to Fig. 1.1 and Fig. 2.1, if, at t = i, any θ_{2i} other than zero were subtracted from θ_{1i} before it was operated on by the nonlinearity, ϕ, a like factor would have to be added to r_i in (2.1.6) and all equations following from (2.1.6). This would, if anything, make the calculations more cumbersome.

3. SUBOPTIMAL SYSTEMS

3.1 Motivation

The number of possible values for $\underline{I}$ that must be considered by receiver R is approximately

$$\left(\frac{NB_m}{\pi}\right)! \tag{3.1.1}$$

For N other than very small, this requires very large numbers of calculations. Apparently receiver R would not be a very practical device to implement. Methods of reducing the required number of calculations are of primary interest.

3.2 Systems Using Feedback

A system involving less computation for the selection of $\underline{I}$ is one which considers only one possibility for I_N based on the received signal at $t = 0,\cdots,N-1$ but not considering r_N. A prediction, $\tilde{s}_N$, of the values of s_N based on the previous N samples, is made and I_N is chosen so that

$$-\pi < (\tilde{s}_N - r_N) \leqq \pi. \tag{3.2.1}$$

This scheme is quite easy to implement using feedback. Referring to Fig. 1.1, if θ_2 is the predicted value $\tilde{s}_N$

at the instant that $\theta_i - \theta_2$ is sampled, the difference samples will be the desired value of $\tilde{s}_N - r_N$, without further computation. This latest sample and the value of $\tilde{s}_N$ can then be used to form the prediction, $\tilde{s}_{N+1}$, and the estimates of A and B.

In this chapter one such feedback system will be investigated. This system will use a time-varying linear filter to generate the feedback signal.

It will be shown that the operation of this system is described by a two-dimensional Markov process. Because of the nonlinearity of the phase detector, it is impossible to analytically derive the probability distribution of the error of this system.

A computer program has been written to calculate the probability that the phase error of the system in tracking the input signal is less than π. The system was approximated in order to facilitate computation. This approximation consisted of adding an absorbing barrier or trapping state to the original two-state process. This barrier was placed at NB' = Z. Thus, if NB' ever reached Z, the phase error could never become smaller. In the actual system, there would be some probability of the phase error returning to a lower value.

3.3 The Ideal Linear Estimator

The minimum mean-square-error, linear, unbiased extrapolator of a sampled polynomial, corrupted by additive Gaussian noise, has been derived by Lees [LE1]. He has shown that for a first order polynomial this estimator forms:

$$\tilde{s}_{N+1} = A' + (N+1)B' \tag{3.3.1}$$

where A' and B' are given by (2.3.4). This is, of course, a time-varying linear estimator, since the values of the matrix in (2.3.4) change with each additional sample. This can be simplified to:

$$\tilde{s}_{N+1} = \frac{-2}{N+1} \sum_{i=0}^{N} u_i + \frac{6}{N(N+1)} \sum_{i=0}^{N} i u_i \tag{3.3.2}$$

This equation holds for $N > 0$. For $N = 0$, i.e. when only one sample has been received, the set of equations leading to (2.3.4) is singular. In this case, the best estimate of s_1 is:

$$\tilde{s}_1 = A' = u_0 \tag{3.3.3}$$

This estimator can be used for the filter unit in the system illustrated in Fig. 1.1. In this system

$$u_N = m_N + \theta_{2N} \tag{3.3.4}$$

yielding

$$\begin{bmatrix} \sum u_i \\ \\ \sum iu_i \end{bmatrix}_N = \begin{bmatrix} \sum u_i \\ \\ \sum iu_i \end{bmatrix}_{N-1} + \begin{bmatrix} 1 \\ \\ N \end{bmatrix} (m_N + \theta_{2N}) \qquad (3.3.5)$$

where

$$\theta_{2N} = \frac{1}{N} \left[-2 \qquad \frac{6}{N-1}\right] \begin{bmatrix} \sum u_i \\ \\ \sum iu_i \end{bmatrix}_{N-1} \qquad (3.3.6)$$

and

$$m_N = \phi[\theta_{1N} - \theta_{2N}] \qquad (3.3.7)$$

Equations (3.3.5) and (3.3.6) can be combined to form:

$$\begin{bmatrix} \sum u_i \\ \\ \sum iu_i \end{bmatrix}_N = \begin{bmatrix} \frac{N-2}{2} & \frac{6}{N(N-1)} \\ \\ -2 & \frac{N+5}{N-1} \end{bmatrix} \begin{bmatrix} \sum u_i \\ \\ \sum iu_i \end{bmatrix}_{N-1} + \begin{bmatrix} 1 \\ \\ N \end{bmatrix} m_N \qquad (3.3.8)$$

Making the substitution:

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_N = \frac{1}{(N+1)(N+2)} \begin{bmatrix} -6N & 12 \\ 4N+2 & -6 \end{bmatrix} \begin{bmatrix} \sum u_i \\ \sum iu_i \end{bmatrix}_N , \qquad (3.3.9)$$

yields

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_N = \begin{bmatrix} \frac{N}{N-1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_{N-1} + \frac{1}{(N+1)(N+2)} \begin{bmatrix} 6N \\ -2(N-1) \end{bmatrix} m_N , \qquad (3.3.10)$$

where $$m_N = \phi[\Theta_{1N} - \Theta_{2N}], \qquad (3.3.11)$$

and $$\Theta_{2N} = [\tfrac{N}{N-1} \quad 1][X]_{N-1} \qquad (3.3.12)$$

The diagonal form of (3.3.10) simplifies calculations.

Comparison of (3.3.9) with (2.3.4) shows that

$$X_{1N} = NB' \qquad (3.3.13)$$

and

$$X_{2N} = A' \qquad (3.3.14)$$

Note that during the interval $N < t < N + 1$ the state of the system is completely described by the two-element vector $\underline{X}_N$. Hence the probability of the system being in any state is given by the joint probability distribution which will be denoted:

$$P[\underline{X}_N].$$

Hence the process describing the operation of this system is a two-dimensional Markov process. An analogy may be drawn to a two-dimensional diffusion process.

3.4 Errors with the Ideal Linear System

After the sample at $t = 0$, the output of the system is:

$$\theta_{20} = m_0 = [A + n_0] \qquad (3.4.1)$$

Upon receipt of the sample at $t = 1$, the system forms:

$$\theta_{21} = 2(m_1 + \theta_{20}) - m_0$$

$$= 2\phi[A + B + n_1 - \phi[A + n_0]] + \phi[A + n_0] \qquad (3.4.2)$$

Since for all x and y

$$\phi[x + \phi[y]] = \phi[x + y] \tag{3.4.3}$$

$$\Theta_{21} = 2\phi[B + n_1 - n_0] + \phi[A + n_0] \tag{3.4.4}$$

Similarly, A enters into the equations for all subsequent Θ_{2i} only in a term of the form (3.4.1).

Since the interest here is in the manner in which the system tracks changes in Θ_1, and not in the size of any constant error, it may be assumed that

$$A = 0 \tag{3.4.5}$$

Under this condition, the state vector at t = 1 is:

$$\underline{X}_1 = \begin{bmatrix} \phi[B + n_1 - n_0] \\ \\ \phi[n_0] \end{bmatrix} \tag{3.4.6}$$

and thus the probability distribution on $\underline{X}$ is:

$$P[\underline{X}_1] = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} P[n_0 = X_{21} + 2\pi i,\ n_1 = X_{11} + X_{21} - B + 2\pi j]$$

$$\text{for} \quad |X_{11}|,\ |X_{21}| \leq \pi$$

$$= 0 \qquad \text{otherwise} \tag{3.4.7}$$

Since the two variables n_0 and n_1 are independently normally distributed,

$$P[\underline{X}_1] = \sum_{i,j} \frac{1}{2\pi\sigma^2} \exp[-\frac{1}{2\sigma^2}\{(X_{21} + 2\pi i)^2 + (X_{11} + X_{21} - B + 2\pi j)^2\}]$$

$$\text{for} \quad |X_{11}|, \ |X_{21}| \leq \pi$$

$$= 0 \qquad \text{otherwise} \qquad (3.4.8)$$

The distribution of $\underline{X}_2$, the state vector after the next sample, may be found by performing the integration:

$$P[\underline{X}_2] = k \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} P[\underline{X}_1] P[m_2: \ \underline{X}_1 \rightarrow X_2] dX_{11} dX_{22} \qquad (3.4.9)$$

where $P[m_2: \ \underline{X}_1 \rightarrow \underline{X}_2]$ symbolizes: the probability density of that m_2 such that $\underline{X}_1$ is mapped to $\underline{X}_2$, and the constant k includes any Jacobians required to adjust the probability densities to take into account the effects of the constants in (3.3.10). The probability density of m_{N+1} given $\underline{X}_N$ is

$$P[m_{N+1}|\underline{X}_N] = \sum_{i=-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp[-\frac{1}{2\sigma^2}(m_2 + \frac{N+1}{N} X_{1N} + X_{2N} + 2\pi i)^2]$$

$$|m_{N+1}| < \pi$$

$$= 0 \qquad \text{otherwise} \qquad (3.4.10)$$

Referring to (3.3.10) it is seen that for given $\underline{X}_2$, $\underline{X}_1$ is restricted to those values which satisfy the equations

$$\begin{bmatrix} \underline{X} \end{bmatrix}_{N-1} = \begin{bmatrix} \frac{N-1}{N} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{X} \end{bmatrix}_{N} + \frac{(N-1)}{(N+1)(N+2)} \begin{bmatrix} -6 \\ 2 \end{bmatrix} m \qquad (3.4.11)$$

Thus the integration in (3.4.9) is only over the line:

$$(X_{11} - \tfrac{1}{2}X_{12}) = -3(X_{21} - X_{22}) \qquad (3.4.12)$$

It is possible to express (3.4.9) as a triple summation of integrals, each of which may be integrated in closed form. However, the extension of this calculation past the next sample involves the quadruple sum of integrals of the form:

$$\int_a^b \operatorname{erf}[cx - d] \exp[-e(x - f)^2]dx \qquad (3.4.13)$$

which may not be integrated in closed form.

Because of this difficulty with analytical characterization of the system performance, a computer has been used to directly perform the integration in equations of the form (3.4.9). The program evaluated the probability that

$$X_{1N} = NB' < \pi. \qquad (3.4.14)$$

3.5 Approximations in the Computer Program

Three approximations were used in the computer program. The first and unavoidable approximation was the use of a field described only at grid points as a representation of a probability density which was actually a continuous function of two variables.

The second approximation was the addition of a trapping state to the original process. This was done for the following reason: While the state variable X_{1N} is restricted to a small region at the beginning of the process, it soon has a range of many π. Even though the probability of X_{1N} having a large value is small, straightforward analysis would require storing the description of the field at these extreme values, and this in turn would require very large data storage in order to preserve accuracy in the region corresponding to small X_{1N}.

The trapping state alters the process so that if the magnitude of X_{1N} ever exceeds a fixed boundary, Z, it may never return. Hence, since the interest is in values of X_{1N} less than π, no consideration need be given these states with X_{1N} greater than Z in the subsequent calculations of the probability density $P[\underline{X}]$.

This approximation will be in error by the probability that X_{1N} could be larger than Z at some time and then return to a value less than Z at the sample being evaluated. Z was chosen experimentally so that

$$P[X_{1N} > Z] < (.01)P[X_{1N} > \pi] \qquad (3.5.1)$$

Thus the error this approximation causes in the results of interest must be less than 1 percent.

The third approximation was the assumption that for the original signal

$$B = 0 \qquad (3.5.2)$$

The value of B has some effect on the distribution of m for the sample at

$$t = 1 \qquad (3.5.3)$$

and this influences the distribution of the error for all subsequent samples. If B is always small compared to π, which would be reasonable in practice, then this approximation will not produce appreciable errors. This approximation provides a symmetry in the probability distribution $P[\underline{X}_N]$ which reduces computation time significantly.

4. RESULTS AND CONCLUSIONS

4.1 Numerical Results

Figs. 4.1 through 4.3 show graphically the bound on the probability of false lock for the ideal system obtained by evaluating (2.11.2) and the probability that X_{1N} is greater than pi for the system described by (3.3.10). This second quantity was only calculated for $N \leq 10$ because of limitations on computer time available.

4.2 Comparison of the Two Systems

Although both of the systems investigated are sampled-data phase lock loops (or at least would be after the open loop ideal system was transformed to a closed loop system), there are some strong differences between them.

The ideal system does not have a fixed number of state variables. It must store separately the value of each sample received. In addition, the closed loop version would need to store the value of its own output, Θ_2, at each sampling instant in order to properly perform the operation discussed in 2.12. The characteristics of receiver R are such that the operation of this system can be easily predicted by examining the signal plus noise before they are operated on by the nonlinearity. This examination involves a measurement of geometric distances

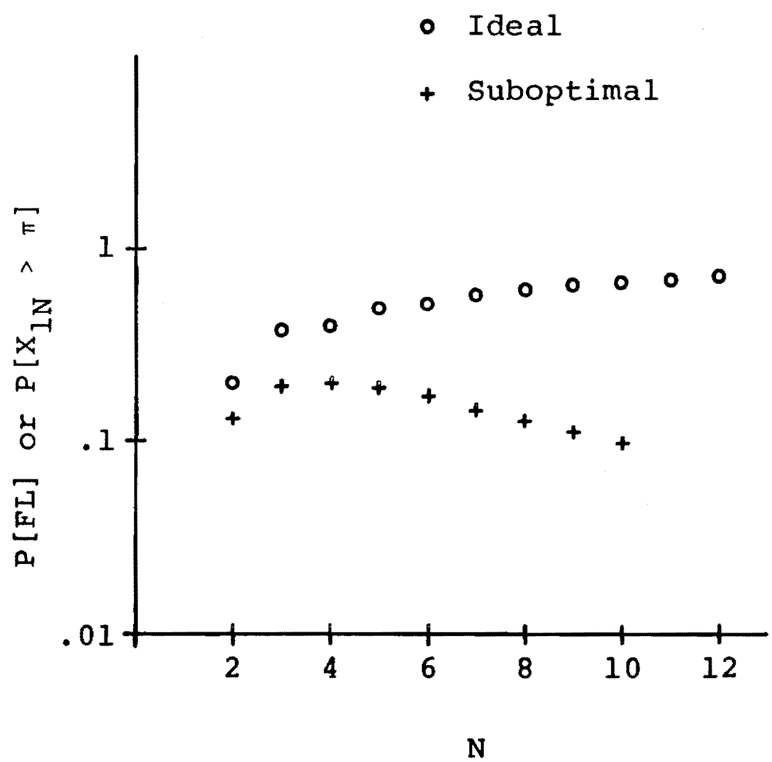

Fig. 4.1 $\sigma = 1.0$ Performance of the Two Systems

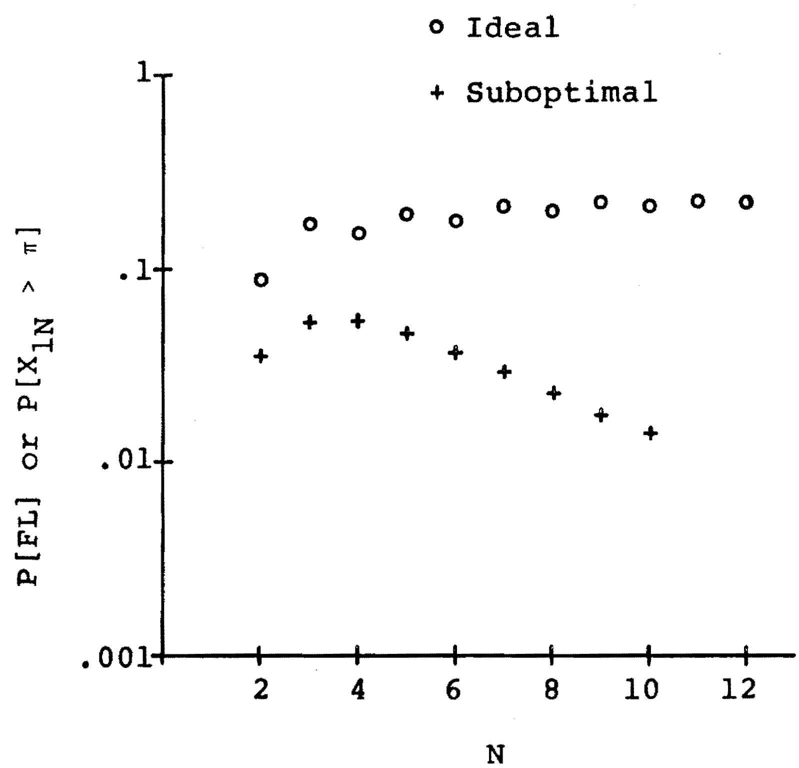

Fig. 4.2 $\sigma = .75$ Performance of the Two Systems

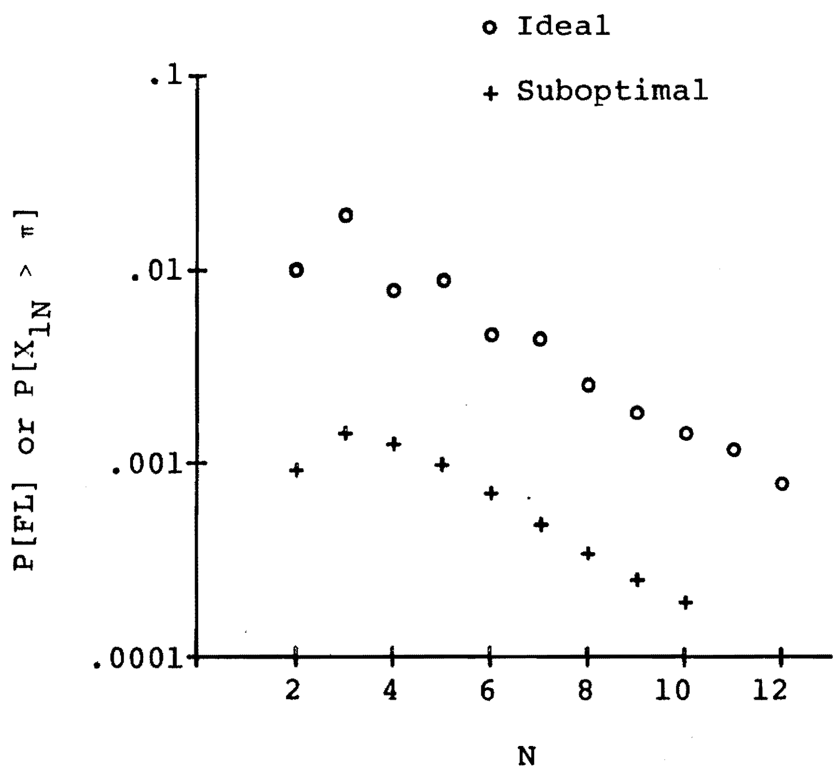

Fig. 4.3 σ = .5 Performance of the Two Systems

in a multi-dimensioned Euclidean space. It was natural to separate the errors caused by the noise into those caused by components of the noise in the plane of the original signal and those caused by components of the noise in the subspace orthogonal to the plane of the signal. Some of these latter errors could be identified as false lock errors, independent of the components of the noise in the plane of the signal.

The system with the linear estimator (3.3.10) has only two state variables. The only known way of finding the system output for a given input, is that of step-by-step evaluation of (3.3.10). Hence no way is known of identifying errors due to false lock apart from other errors.

4.3 Conclusions

Examination of Figs. 4.1 through 4.3 show that in all cases investigated, the probability that $X_{1N} = NB'$ exceeds pi for the system with the linear estimator is less than the bound which has been derived on the probability that false lock in the ideal system will cause an error in the estimate of NB'greater than pi. Thus the performance of the sub-optimal system in this respect is in these cases better than the bound which has been placed on the performance of the ideal system. Hence, it can be concluded that the bound in (2.11.2) is a rather conservative one, at least for the cases considered.

4.4 Suggestions for Further Study

If the solutions of (2.6.7) could be characterized more precisely, it might be possible to derive several interesting results. Both a less conservative upper bound on the probability of false lock and lower bound on the probability of false lock for the ideal system would be of interest.

The extension of the results of this study to simple, easy to implement systems would have practical significance.

BIBLIOGRAPHY

[BL1] Blachowicz, L. F., "Dial Any Channel to 500 Mhz," Electronics, Vol. 39, No. 9, pp. 60-69, May 2, 1966.

[BY1] Byrne, C. J., "Properties and Design of the Phase-Controlled Oscillator with a Sawtooth Comparator," Bell System Technical Journal, Vol. 41, No. 2, pp. 559-602, March 1962.

[GA1] Gardner, F. M., Phaselock Techniques, John Wiley New York, 1966.

[LE1] Lees, A. B., "Interpolation and Extrapolation of Sampled Data," IRE Transactions on Information Theory, Vol. IT-2, No. 1, pp. 12-17, March 1956.

[MI1] Millar, R. A., "Digital Control of Shaft Speed and Position," IEEE Spectrum, Vol. 5, No. 1, pp. 90-95, Jan. 1966.

[TA1] Tausworthe, R. C., "Acquisition and False-Lock Behavior of Phase-Locked Loops With Noisy Inputs," JPL Space Progress Summary 37-46, Vol. IV, pp. 226-234, Aug. 31, 1967.

APPENDIX A

The Integral in (2.1.6)

$$\int_{-b}^{b}\int_{-a}^{a} \exp[-c\sum_{i=0}^{N}(r_i - A - iB)^2]dAdB$$

$$= \int_{-b}^{b}\int_{-a}^{a} \exp[-c\sum_{i}^{N} r_i^2 + (2c\sum_{i}^{N} r_i)A + (2c\sum ir_i)B$$

$$-cN(N+1)AB - \frac{N(N+1)2N+1)}{6} B^2]dAdB \qquad (A.1)$$

By performing a rotation of the axes A and B, completing the square in each of the new variables, and then shifting the new axes by appropriate constants, this integral may be reduced to a sum of five integrals, each of the form:

$$\int_{d_0}^{d_1}\int_{d_2}^{d_3+d_4x_2} \exp[-k_1^2x_1^2 - k_2^2x_2^2]dx_1dx_2$$

$$= \int_{d_0}^{d_1} \exp(-k_2x_2^2)\ \frac{\sqrt{\pi}}{2}\ k_1\{\mathrm{erf}[k_1(d_3 + d_4x_2)]$$

$$- \mathrm{erf}[k_1d_2]\}dx_2 \qquad (A.2)$$

For one of the five integrals, $d_4 = 0$ and the evaluation of the integral is trivial, as is always the case for the second term inside the $\{\cdot\}$. The other four integrals are composed of terms of the form:

$$\frac{\sqrt{\pi}}{2}\int_0^D \exp(-\gamma^2x^2)\,\mathrm{erf}(\alpha + \beta x)\,dx \tag{A.3}$$

This integral has no closed form expression, but it can be evaluated by resorting to integration of the Cauchy product of the infinite series for the two transcendental functions.

$$\int_0^D \Big[\sum_{i=0}^{\infty}(-1)^i\frac{(\gamma x)^{2i}}{i!}\Big]\Big\{\sum_{k=0}^{\infty}(\beta x)^{2k}\sum_{n=k}^{\infty}\frac{\alpha^{2(n-k)}}{(2n+1)n!}\Big[\binom{2n+1}{2k+1}\alpha + \binom{2n+1}{2k+2}\beta x\Big] \tag{A.4}$$

$$= \int_0^D \sum_{i=0}^{\infty} x^{2i}\sum_{k=0}^{i}(-1)^{i-k}\frac{(\gamma)^{2(i-k)}\beta^{2k}}{(i-k)!}\sum_{n=k}^{\infty}(-1)^n$$

$$\frac{\alpha^{2(n-k)}}{(2n+1)n!}\Big[\binom{2n+1}{2k+1}\alpha + \binom{2n+1}{2k+2}\beta x\Big] \tag{A.5}$$

$$= \sum_{i=0}^{\infty} D^{2i}\sum_{k=0}^{i}(-1)^{(i-k)}\frac{\gamma^{2(i-k)}\beta^{2k}}{(i-k)!}\sum_{n=k}^{\infty}(-1)^n\frac{\alpha^{2(n-k)}}{(2n+1)n!}$$

$$\Big[\binom{2n+1}{2k+1}\frac{\alpha}{2i+1} + \binom{2n+1}{2k+2}\frac{\beta D}{2i+2}\Big] \tag{A.6}$$

APPENDIX B

COMPUTER PROGRAMS USED IN THIS RESEARCH

```
C       THIS PROGRAM SEARCHES FOR VECTORS WHICH SIMULTANEOUSLY
C       MEET CONDITIONS (2.9.1) AND (2.9.2)
C
10010 FORMAT (I10)
10020 FORMAT (I1,3HN =,I3,9X,1HM,5X,14HVECTORS TESTED/)
10030 FORMAT (3HOD=,E16.8,6H     B=,E16.8,6H     K=,28I3)
10040 FORMAT (I17,I13)
10050 FORMAT (19H COMPUTATION TIME =,F6.1,4H SEC)
      INTEGER OLD
      DIMENSION K(200)
   10 READ (5,10010) N
      IPAGE=1
      WRITE (6,10020) IPAGE,N
      IF (N.EQ.0) STOP
      CALL TIME(OLD)
      M=N/2
      IF (MOD(N,2).EQ.1) GO TO 20
C     N EVEN
      DF=FLOAT(N*(N+2))/(16.*FLOAT(N+1))
      GO TO 30
C     N ODD
   20 DF=FLOAT((N-1)*(N+1)*(N+3))/(16.*FLOAT(N*(N+2)))
   30 KZMAX=N/M+1
      NT=0
      DO 110 KZ=1,KZMAX
      KMMAX=(N-KZ)/M
      IF (KMMAX.LE.0) GO TO 110
      DO 100 KM=1,KMMAX
      LENGTH=N+1-KZ-KM
      MINUS=M-1
      IF (M.GE.3) GO TO 35
      K(1)=LENGTH
      GO TO 55
   35 KMAX=(N-KZ-KM+2)/(M-1)+1
      KMIN=(N-KZ-KM-1000*(M-1))/(M-1)+999
C     START VECTOR
      DO 40 I=1,MINUS
   40 K(I)=KMIN
C     CHECK LENGTH
   45 L=0
      DO 50 I=1,MINUS
   50 L=L+K(I)
```

```
      IF (L.NE.LENGTH) GO TO 80
C     FIND D
   55 ISUM=M*KM
      ISUMSQ=M*M*KM
      ISUMI=0.
      L=KZ
      DO 60 I=1,MINUS
      ISUM=ISUM+I*K(I)
      ISUMSQ=ISUMSQ+I*I*K(I)
      ISUMI=ISUMI+I*(K(I)*(K(I)+1+2*L))/2
   60 L=L+K(I)
      ISUMI=ISUMI+M*(KM*(KM+1+2*L))/2
      D=ISUM*ISUM*(4*N+2)-12*ISUM*ISUMI
      D=FLOAT(ISUMSQ)-(D+FLOAT(ISUMI*ISUMI)*12./FLOAT(N))/FL
     1OAT((N+1)*(N+2))
      NT=NT+1
      IF (D.GT.DF) GO TO 80
      B=FLOAT(-6*N*ISUM+12*ISUMI)/FLOAT((N+1)*(N+2))
      IF (B.GT..5) GO TO 80
      D=SQRT(D)
      WRITE (6,10030) D,B,KZ,(K(I),I=1,MINUS),KM
   80 IF (MINUS.EQ.1) GO TO 100
      K(MINUS)=K(MINUS)+1
      DO 90 I=2,MINUS
      J=M-I+1
      IF (K(J).LE.KMAX) GO TO 45
      K(J)=KMIN
   90 K(J-1)=K(J-1)+1
      IF (K(1).LE.KMAX) GO TO 45
  100 CONTINUE
  110 CONTINUE
      WRITE (6,10040) M,NT
      M=M-1
      IF (M.GT.1) GO TO 30
      CALL TIME(NEW)
      T=.001*FLOAT(OLD-NEW)
      WRITE (6,10050) T
      OLD=NEW
      IPAGE=IPAGE+7
      IF (IPAGE.GT.8) IPAGE=1
      GO TO 10
      END
```

```
      SUBROUTINES HALFDF AND LONER WERE USED TO EVALUATE
      THE BOUND GIVEN IN (2.11.2) ON THE PROBABILITY OF
      FALSE LOCK FOR THE IDEAL SYSTEM


      SUBROUTINE HALFDF(N,D)
C
C     THIS SUBROUTINE RETURNS HALF THE LENGTH OF THE NORMAL
C     VECTOR FROM THE AB SUBSPACE TO THE NEAREST HYPERCUBE
C     CENTER CAUSING FALSE LOCK
C
      PI=3.141593
      IF (MOD(N,2).EQ.0) GO TO 10
C     N IS ODD
      D=FLOAT((N-1)*(N+1)*(N+3))/FLOAT((N+2)*N*16)
      GO TO 30
C     N IS EVEN
   10 D=FLOAT((N+2)*N)/FLOAT((N+1)*16)
   30 D=SQRT(D)*PI
      RETURN
      END




      SUBROUTINE  LONER(N,SIGMA,P,PROB)
C
C     THIS SUBROUTINE RETURNS THE PROBABILITY THAT A
C     GAUSSIAN VECTOR WILL FALL OUTSIDE A HYPERSPHERE OF
C     RADIUS P IN AN N+1 DIMENSIONED SPACE
C
      PI=3.141593
      IF (MOD(N,2).EQ.0) GO TO 100
C     N IS ODD
      CONST=PI**((N+1)/2)*2.
      N2=(N-1)/2
      AN2=FLOAT(N2)+.5
      AN1=1.
   10 CONST=CONST/AN1
      AN1=AN1+1.
      IF (AN1.LE.AN2) GO TO 10
      PROD=1.
      SUM =0.
      N2=N2+1
      DO 20 I=1,N2
      PROD=PROD*FLOAT(N+3-2*I)
```

```
      SUM=SUM+SIGMA**(2*I)*P**(N+1-2*I)*PROD
   20 CONTINUE
      SUM=SUM*EXP(-P*P/(2.*SIGMA*SIGMA))
      SUM=SUM/FLOAT(N+1)
      PROB=CONST*SUM
      PROB=PROB/(2.*PI*SIGMA*SIGMA)**((N+1)/2)
      RETURN
C     N IS EVEN
  100 CONST=2.**N*PI**(N/2)
      IF (N.NE.0) GO TO 105
      PROB=ERFC(P/(SQRT(2.)*SIGMA))
      RETURN
  105 CONTINUE
      N1=N/2
      AN1=N1
      AN2=FLOAT(N)-.5
  110 CONST=CONST/AN1
      AN1=AN1+1.
      IF (AN1.LE.AN2) GO TO 110
      PROD=1.
      SUM =0.
      DO 120 I=1,N1
      PROD=PROD*FLOAT(N+3-2*I)
  120 SUM=SUM+SIGMA**(2*I)*P**(N+1-2*I)*PROD
      SUM=SUM*EXP(-P*P/(2.*SIGMA*SIGMA))
      SUM=SUM+          SQRT(PI*SIGMA*SIGMA/2.)*ERFC(P/SQRT(2
     1.)/SIGMA)*PROD*SIGMA**N
      SUM=SUM/FLOAT(N+1)
      PROB=SUM*CONST
      PROB=PROB/(2.*PI*SIGMA*SIGMA)**(FLOAT(N+1)/2.)
      RETURN
      END
```

```
C     THIS MAIN PROGRAM AND THE ACCOMPANYING SUBROUTINES
C     CALCULATE THE PROBABILITY THAT ABS(X1) IS GREATER
C     THAN PI FOR THE SYSTEM DESCRIBED IN SECTION 3.3
C
C     SUBROUTINES REQUIRED: POINT
C                           VOLUME
C                           PHI
C                           PR
C                           LOOK
C
10010 FORMAT (3I10,F10.5,E20.8,F10.5)
10015 FORMAT (1H1,3I10,F10.5,E20.8,F10.5)
10020 FORMAT (  3HON=,I3,7H      P(,I3,1H,,I3,11H).GE.BOTTOM)
10025 FORMAT (7HOX1MAX=,F10.6,8H  X2MIN=,F10.6,8H  X2MAX=,F1
     10.6)
10030 FORMAT (3HON=,I4,10H    REGION,I2,10H    PROB =,E16.8)
10040 FORMAT (7H1TIME =,I6,4H SEC)
10050 FORMAT (22H0TRAPPED THIS SAMPLE =,E16.8,20H      TOTAL
     1TRAPPED =,E16.8)
10060 FORMAT (13H0TOTAL PROB =,F10.7,22H    P(ABS(X1).GT.PI)
     1 =,E16.7)
10080 FORMAT (28H0LOWER LIMIT FOR TRAP IS I =,I3)
      DIMENSION P(101,101),PN(101,101)
     0COMMON PI,TWOPI,C2,CAN,CBN,CN,D2H,D2L,N1,X2MIN,DELX1,D
     1ELX2,X1N,X2N,I,J,P,PN
      REAL ML,MEAN
      INTEGER D1,D2,D2H,D2L,REGION
      PI=3.141593
      TWOPI=2.*PI
   10 READ (5,10010) D1,D2,NMAX,SIGMA,BOTTOM,FACTOR
      WRITE (6,10015) D1,D2,NMAX,SIGMA,BOTTOM,FACTOR
      IF (NMAX.EQ.0) CALL EXIT
      D2H=D2
      D2L=1
      BNDRY=FACTOR*PI
C
C     INITIALIZE PROBABILITY MATRIX
C
      N=1
      X1MAX=PI
      X2MAX=PI
      X2MIN=-PI
      DELX1=X1MAX/FLOAT(D1-1)
      DELX2=(X2MAX-X2MIN)/FLOAT(D2-1)
      C1=1./(TWOPI*SIGMA*SIGMA)
      C2=1./(2.*SIGMA*SIGMA)
      CD=SQRT(C2)
      DO 40 I=1,D1
```

```
      DO 30 J=D2L,D2H
      X1=FLOAT(I-1)*DELX1
      X2=FLOAT(J-1)*DELX2+X2MIN+X1
      X1=X1*X1
     0PROB=C1*(EXP(-C2*(X1+X2*X2))+EXP(-C2*(X1+(X2+TWOPI)*(X
     12+TWOPI)))  +EXP(-C2*(X1+(X2-TWOPI)*(X2-TWOPI))))
      IF (PROB.GT.BOTTOM) GO TO 20
      P(I,J)=0.
      GO TO 30
   20 P(I,J)=PROB
   30 CONTINUE
   40 CONTINUE
      PTRAPT=0.
C
C     DISCARD ZERO SECTION OF MATRIX
C
      X2MAXN=X2MAX
      X2MINN=X2MIN
C     TRANSFER BELOW
      GO TO 125
C
C     FIND SIZE OF NEW DISTRIBUTION
C
   60 CONTINUE
      CN=FLOAT(N)/FLOAT((N+2)*(N+3))
      CA=TWOPI*CN
      CB=6.*PI*CN
      CAN=2.*CN
      CBN=1./(6.*CN)
      CN=FLOAT(N)/FLOAT(N+1)
      CP=SQRT(C1)*DELX1*CBN*CN/3.
      X1MAXN=(X1MAX+CB)/CN
      IF (X1MAXN.GT.BNDRY) X1MAXN=BNDRY
      X2MAXN=X2MAX+CA
      X2MINN=X2MIN-CA
      DELX1N=X1MAXN/FLOAT(D1-1)
      DELX2N=(X2MAXN-X2MINN)/FLOAT(D2-1)
      N=N+1
C
C     CALCULATE NEW DISTRIBUTION
C
      DO 110 I=1,D1
      DO 110 J=1,D2
C
C     FIND RANGE IN OLD MATRIX
C
      X1N=FLOAT(I-1)*DELX1N
      X2N=X2MINN+FLOAT(J-1)*DELX2N
```

```
      N1H=(X1N*CN+CB)/DELX1+1.
      X=(X1N*CN-CB)/DELX1
      IF (X.GE.0.) GO TO 72
      N1L=X-1.
      X=FLOAT(N1L+1)-X
      GO TO 74
   72 N1L=INT(X-999.)+1000
      X=FLOAT(N1L-1)-X
   74 IF (N1H.GT.D1) N1H=D1
C     N1L.LT.(-D1)=.FALSE. ALWAYS
      IF (N1L.EQ.(-1)) N1L=1
C
C     INTEGRATE OVER RANGE
C
      PN(I,J)=0.
      IF (N1H.LE.N1L) GO TO 110
      N1=N1L
      IF (X.LE.0.) GO TO 78
      CALL POINT(3.*X)
   78 IF (MOD(IABS(N1H-N1L),2).NE.1) GO TO 80
C     LAST INTERVAL
      N1=N1H
      CALL POINT (1.5)
      N1H=N1H-1
      N1=N1H
      CALL POINT(1.5)
   80 IF (N1H.EQ.N1L) GO TO 110
      N1=N1L
C     FIRST POINT
      CALL POINT(1.)
      N1L=N1L+1
      IF (N1L.EQ.(-1)) N1L=1
      N1=N1H
C     LAST POINT
      CALL POINT(1.)
      N1H=N1H-1
      N1=N1H
C     NEXT TO LAST POINT
      CALL POINT(4.)
      N1H=N1H-1
      IF (N1L.GE.1) GO TO 90
C     NEGATIVE VALUES
      N1L=-N1L
      NX=1
      IF (MOD(N1L,2).EQ.1) NX=2
      DO 84 K=NX,N1L,2
      N1=-K
      CALL POINT(2.)
```

```
      N1=N1-1
   84 CALL POINT(4.)
      N1L=3-NX
C     POSITIVE VALUES
   90 IF (N1H.LE.N1L) GO TO 110
      DO 100 K=N1L,N1H,2
      N1=K
      CALL POINT(4.)
      N1=N1+1
  100 CALL POINT(2.)
  110 PN(I,J)=PN(I,J)*CP
C
C     FIND PROB OF BECOMING TRAPPED
C
      PTRAP=0.
      IF (X1MAXN.LT.BNDRY) GO TO 117
C     LOWER LIMIT
      N1L=INT((BNDRY-PI/CBN)*CN/DELX1-999.)+1000
      WRITE (6,10080) N1L
      DO 115 I=N1L,D1
      DO 115 J=D2L,D2H
C     EACH POINT'S CONTRIBUTION
      X1=FLOAT(I-1)*DELX1
      X2=FLOAT(J-D2L)*DELX2+X2MIN
C     THETA=X1/CN+X2
      MEAN=PHI(-X1/CN-X2)
      ML=(BNDRY*CN-X1)*CBN
  115 PTRAP=PTRAP+P(I,J)*(ERF((PI-MEAN)*CD)-ERF((ML-MEAN)*CD
     1))
      PTRAP=PTRAP*DELX1*DELX2
      PTRAPT=PTRAPT+PTRAP
  117 CONTINUE
C
C     REPLACE OLD DISTRIBUTION WITH NEW
C
      DO 120 I=1,D1
      DO 120 J=1,D2
  120 P(I,J)=PN(I,J)
      DELX1=DELX1N
      DELX2=DELX2N
      X1MAX=X1MAXN
C
C     DISCARD ZERO SECTION
C
  125 CALL TIME(IDUM)
      WRITE (6,10040) IDUM
      DO 130 J=1,D2
      DO 130 I=1,D1
```

```
      N1=D1+1-I
      IF (P(N1,J).GE.BOTTOM) GO TO 140
  130 CONTINUE
  140 D2L=J
      WRITE (6,10020)   N,N1,J
      X=FLOAT(J-1)*DELX2
      X2MIN=X2MINN+X
      DO 150 J=1,D2
      N2=D2+1-J
      DO 150 I=1,D1
      IF (P(I,N2).GE.BOTTOM) GO TO 160
  150 CONTINUE
  160 D2H=N2
      WRITE (6,10020)   N,I,N2
      X=FLOAT(J-1)*DELX2
      X2MAX=X2MAXN-X
      WRITE (6,10025) X1MAX,X2MIN,X2MAX
C
C     FIND PROBABILITY THAT MAG(X1).LE.PI
C
      REGION=1
      N1H=PI/DELX1+1.
      N1L=1
      N2H=D2H
      N2L=D2L
      PROB=2.*VOLUME(N1L,N1H,N2L,N2H)
      WRITE (6,10030) N,REGION,PROB
      SAVE=PROB
      IF (X1MAX.LE.PI) GO TO 270
C
C     FIND PROBABILITY THAT PI.LE.MAG(X1).LT.BNDRY
C
      REGION=2
      N1L=N1H
      N1H=D1
      N2L=D2L
      N2H=D2H
      PROB=2.*VOLUME(N1L,N1H,N2L,N2H)
  260 WRITE (6,10030) N,REGION,PROB
      WRITE (6,10050) PTRAP,PTRAPT
      PROB=PROB+PTRAPT
      SAVE=SAVE+PROB
      WRITE (6,10060) SAVE,PROB
  270 CONTINUE
      CALL LOOK(D1,D2)
      IF (N.LT.NMAX) GO TO 60
      GO TO 10
      END
```

```
      SUBROUTINE POINT(WEIGHT)
      DIMENSION P(101,101),PN(101,101)
     0COMMON PI,TWOPI,C2,CAN,CBN,CN,D2H,D2L,N1,X2MIN,DELX1,D
     1ELX2,X1N,X2N,I,J,P,PN
      LOGICAL POS
      INTEGER D2H,D2L
      REAL M
      POS=N1.GE.0
      X1=FLOAT(N1-ISIGN(1,N1))*DELX1
      M=-CBN*(X1-X1N*CN)
      X2=X2N+CAN*M
      IF (POS) GO TO 10
      N1=-N1
      X1=-X1
      X2=-X2
      M=-M
   10 DELN=(X2-X2MIN)/DELX2
      N2L=DELN
      DELN=DELN-FLOAT(N2L)
      N2L=N2L+D2L
      N2H=N2L+1
      IF (N2H.GT.D2H) GO TO 20
      IF (N2L.GT.D2L) GO TO 15
      IF (N2L.LT.D2L) GO TO 20
      IF (DELN.LT.0.) GO TO 20
   15 CONTINUE
C     THETA=X1/CN+X2
      X=PR(PHI(M+X2+X1/CN))
      Y=(P(N1,N2H)*DELN+P(N1,N2L)*(1.-DELN))*WEIGHT
      IF (Y.EQ.0..OR.X.EQ.0.) GO TO 20
      IF (ALOG(Y)+ALOG(X).LT.(-115.)) GO TO 20
      PN(I,J)=PN(I,J)+X*Y
   20 IF (POS) RETURN
      N1=-N1
      RETURN
      END
```

```
      FUNCTION VOLUME(N1L,N1H,N2L,N2H)
C
C     INTEGRATION IN TWO DIMENSIONS
C
      COMMON DUMMY(10),DELX1,DELX2,DUM(4),P(101,101)
      NB=N1H-N1L
      NA=N2H-N2L
      PROB=0.
C     CORNERS
      PROB=PROB+P(N1L,N2L)*2.
      PROB=PROB+P(N1L,N2H)*(2.-FLOAT(MOD(NA,2)))
      PROB=PROB+P(N1H,N2L)*(2.-FLOAT(MOD(NB,2)))
      PROB=PROB+P(N1H,N2H)*(2.-FLOAT(MOD(NA+NB,2)))
C     EDGE (N1L,*)
      NX=N2H-1
      N2=N1L+1
      IF (MOD(NA,2).EQ.1) GO TO 170
      PROB=PROB+P(N1L,N2H-1)*2.
      PROB=PROB+P(N1L+1,N2H)*2.
      N2=N2+1
      NX=NX-1
  170 N1=N2L+1
      DO 180 I=N1,NX,2
      J=I
      PROB=PROB+P(N1L,J)*2.
      J=J+1
  180 PROB=PROB+P(N1L,J)*4.
C     EDGE (*,N2L)
      NX=N1H-1
      N3=N2L+1
      IF (MOD(NB,2).EQ.1) GO TO 190
      PROB=PROB+P(N1H-1,N2L)*2.
      PROB=PROB+P(N1H,N2L+1)*2.
      N3=N3+1
      NX=NX-1
  190 N1=N1L+1
      DO 200 I=N1,NX,2
      J=I
      PROB=PROB+P(J,N2L)*2.
      J=J+1
  200 PROB=PROB+P(J,N2L)*4.
C     EDGE (*,N2H)
      NX=N1H-1
      IF (MOD(NX-N2,2).EQ.1) GO TO 210
      PROB=PROB+P(NX,N2H)*4.
      NX=NX-1
  210 DO 220 I=N2,NX,2
      J=I
```

```
      PROB=PROB+P(J,N2H)*4.
      J=J+1
  220 PROB=PROB+P(J,N2H)*2.
C     EDGE (N1H,*)
      NX=N2H-1
      IF (MOD(NX-N3,2).EQ.1) GO TO 230
      PROB=PROB+P(N1H,NX)*4.
      NX=NX-1
  230 DO 240 I=N3,NX,2
      J=I
      PROB=PROB+P(N1H,J)*4.
      J=J+1
  240 PROB=PROB+P(N1H,J)*2.
C     INTERIOR
      NA =N1L+1
      NB =N1H-1
      N1 =N2L+1
      N2 =N2H-1
      C=0.
      DO 250 I=NA,NB
      C=C+4.
      IF (C.GT.6.) C=0.
      CJ=C
      DO 250 J=N1,N2
      CJ=CJ+4.
      IF (CJ.GT.10.) CJ=4.
  250 PROB=PROB+P(I,J)*CJ
      VOLUME=PROB*DELX1*DELX2/6.
      RETURN
      END
```

```
      SUBROUTINE LOOK(D1,D2)
10010 FORMAT (11H1      MAX =,E16.8//)
10020 FORMAT(10X,101I1)
      DIMENSION DUMMY(16),P(101,101),LINE(101)
      COMMON DUMMY,P
      REAL MAX,MIN
      INTEGER D1,D2
      MAX=0.
      DO 10 I=1,D1
      DO 10 J=1,D2
   10 IF (P(I,J).GT.MAX) MAX=P(I,J)
      MIN=MAX/100.
      DEL=MAX*0.11
      WRITE (6,10010) MAX
      DO 30 N=1,D2
      J=D2+1-N
      DO 20 I=1,D1
   20 LINE(I)=(P(I,J)-MIN)/DEL+1.
   30 WRITE (6,10020) (LINE(I),I=1,D1)
      RETURN
      END
```

```
      FUNCTION PHI(X)
      COMMON PI,TWOPI
      LOGICAL NEG
      NEG=X.LT.0.
      IF (NEG) X=-X
      PHI=AMOD(X+PI,TWOPI)-PI
      IF (.NOT.NEG) RETURN
      PHI=-PHI
      X=-X
      RETURN
      END
```

```
      FUNCTION PR(X)
C     THIS SUBROUTINE USES SIMPLIFICATIONS GOOD
C     FOR SIGMA.LE.1.
      COMMON PI,TWOPI,C2
      PR=0.
      Y=X*X*C2
      IF (Y.GT.87.) GO TO 10
      PR=PR+EXP(-Y)
   10 Y=X+TWOPI
      Y=Y*Y*C2
      IF (Y.GT.87.) GO TO 20
      PR=PR+EXP(-Y)
   20 Y=X-TWOPI
      Y=Y*Y*C2
      IF (Y.GT.87.) RETURN
      PR=PR+EXP(-Y)
      RETURN
      END
```

INPUT CARDS FOR THIS PROGRAM

D1	D2	NMAX	SIGMA	BOTTOM	FACTOR
45	25	10	1.	1.E-30	4.
45	25	10	.75	1.E-30	4.
45	25	10	.50	1.E-30	4.

The vita has been removed from the scanned document

FALSE LOCK IN SAMPLED-DATA
PHASE LOCK LOOPS

by

Hatcher Edward Chalkley

Abstract

The false lock characteristics of a sampled-data phase lock loop containing a phase detector with a sawtooth characteristic are investigated.

The ideal processor of data operated on by such a phase detector nonlinearity is derived in open-loop form. A second system is proposed which is shown to approximate the operation of the ideal system with increasing accuracy for decreasing noise variance. The operation of the approximate system is interpreted in geometric terms. This geometric interpretation is used to place a lower bound on the probability of false lock of the ideal system.

A suboptimal system which uses feedback and a time-varying linear filter is analyzed. It was necessary to use a computer to perform the integration leading to the probability distribution of the error of this system.

The bound on the probability of false lock for the ideal system is compared with the probability of a similar error for the suboptimal system. It is concluded that this bound is a conservative one.