COLORADO STATE
UNIVERSITY
FORT COLLINS, COLORADO
80523

department of electrical engineering

A Dynamic Programming Algorithm for Phase Estimation and Data Decoding on Random Phase Channels

Odile Macchi

Louis L. Scharf

Technical Report No. DEC79-3

December 1979

A Dynamic Programming Algorithm for Phase Estimation and Data Decoding on Random Phase Channels[1]

by

Odile Macchi[2]

Louis L. Scharf[3]

Manuscript Submitted to

IEEE Transactions on Information Theory

December 1979

---

[1]This work supported in part by the Centre National de la Recherche Scientifique; the Office of Naval Research, Statistics and Probability Branch, Arlington, VA; and the Army Research Office, Research Triangle Park, NC

---

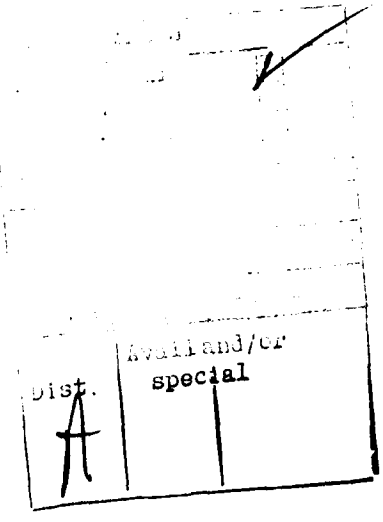
[2]C.N.R.S. Laboratoire des Signaux et Systèmes, Plateau du Moulon, 91190 Gif-sur-Yvette, France

---

[3]Electrical Engineering Department, Colorado State University, Fort Collins, CO 80523, U.S.A. Associated with the C.N.R.S. Laboratoire des Signaux et Systèmes, Plateau du Moulon, 91190 Gif-sur-Yvette, France, in academic year 1977-1978.

Abstract

The problem of simultaneously estimating phase and decoding data symbols from baseband data is posed. The phase sequence is assumed to be a random sequence on the circle and the symbols are assumed to be equally-likely symbols transmitted over a perfectly equalized channel. A dynamic programming algorithm (Viterbi algorithm) is derived for decoding a maximum a posteriori (MAP) phase-symbol sequence on a finite dimensional phase-symbol trellis. A new and interesting principle of optimality for simultaneously estimating phase and decoding phase-amplitude coded symbols leads to an efficient two step decoding procedure for decoding phase-symbol sequences. Simulation results for binary, 8-ARY PM, and 16-QASK symbol sets transmitted over random walk and sinusoidal jitter channels are presented, and compared with results one may obtain with a decision-directed algorithm, or with the binary Viterbi algorithm introduced by Ungerboeck. When phase fluctuations are severe, and the symbol set is rich (as in 16-QASK), MAP phase-symbol sequence decoding on circles is superior to Ungerboeck's technique, which in turn is superior to decision-directed techniques.



I. Introduction

Phase fluctuations can significantly increase the error probability for coded or uncoded symbols transmitted over a channel that may or may not have been equalized. This is especially true for PSK and QASK symboling in which case accurate phase discrimination is essential for symbol decoding[1]. Even when the receiver contains a decision-directed phase-locked loop (DDPLL), performance loss in SNR with respect to a coherent decoding system can be in the range 5-10dB. This fact is established in [1] for practical symbol sets and typical values of the phase variance parameter and symbol error probability.

On telephone lines linear distortion and phase jitter dictate the use of a channel equalizer and some kind of phase estimator to achieve high rate, low error probability, data transmission. A common approach to phase estimation and data decoding is to use a decision-directed algorithm in which a phase estimate is updated on the basis of old phase estimates and old symbol decisions. The DDPLL of [5] is a first-order digital phase-locked loop (PLL) in which the phase estimate is updated on the basis of a new measured phase and an old symbol decision. In the jitter equalizer (JE) of [3] and [4] a complex gain is updated according to a simple decision directed stochastic approximation algorithm. The complex gain is used to scale and rotate the received signal, thereby correcting phase jitter and normalizing rapid fading variations. Although there is no explicit interest in phase estimation itself in the JE, it is possible to interpret the structure as an adaptive gain-phase correcting equalizer.

---

[1] The modifiers PSK and QASK stand for "Phase Shift Keyed" and "Quadrature Amplitude Shift Keyed," respectively; SNR will mean signal-to-noise-ratio.

Both the DDPLL and the JE are very simple to implement, but apparently neither achieves optimality with respect to any statistical criterion for symbol (or data) decoding. Furthermore, neither the DDPLL nor the JE is optimum for estimating and/or correcting phase. Therefore an important question to be answered is whether or not symbol decoding can be improved using a better phase estimator. The answer, based on the results of [1] and this paper, is that significant improvements can be realized when the phase fluctuations are severe if one is willing to pay the price of an increased computational burden. In practice, cases of severe phase fluctuation can occur in high data rate PSK and QASK systems in which the angular distance between symbols is small.

In [1] Ungerboeck recognized the potential of maximum a posteriori (MAP) sequence estimation for jointly estimating phase and decoding data symbols. A path metric was derived and its rôle in a forward dynamic programming algorithm for obtaining MAP phase symbol sequences was indicated. Because of the way phase was modelled in [1], the dynamic programming algorithm could not be solved directly. Using two approximations, Ungerboeck derived an implementable algorithm and obtained performance results that were on the order of 3dB superior in SNR to the DDPLL in a 16-QASK system,at interesting values of the phase variance parameter. We call the algorithm of [1] a discrete binary Viterbi algorithm (DBVA).

The reader is referred also to [5] and [6] for discussions of other sub-optimum, but computationally tractable, algorithms for simultaneously estimating phase and decoding data symbols.

In this paper we observe that baseband data is invariant to modulo-2π transformations on the phase sequence. This motivates us to wrap the

phase around the circle, so to speak, and obtain folded probability models for transition probabilities on the circle. When the phase process is normal random walk on the circle, then the transition probabilities are described by a folded normal model. This model has also been used in [7] and [8]. It is then straight-forward to pose a MAP sequence estimation problem for simultaneous phase and symbol sequence decoding as described in [8] and [9]. The basic idea is to discretize the phase space $[-\pi,\pi)$ to a finite dimensional grid and to use a dynamic programming algorithm (Viterbi algorithm) to keep track of surviving phase-symbol sequences that can ultimately approximate the desired MAP phase-symbol sequence. The MAP phase-symbol sequence, itself, is the entire sequence of past phases and symbols that is most likely, given an entire sequence of recorded observations. Details of the algorithm are given in [8] and [9]. For PSK and QASK symbol sets an interesting principle of optimality leads to an efficient two-step decoding procedure. With this procedure computational complexity is reduced by a factor near to the square of the number of admissible phase values per amplitude level. This amounts to a factor of 16 for the 16-point QASK diagram that has been recommended by CCITT for data transmission on telephone lines at 9600 b/s. Finally, in order to make the computation and storage requirements tractable in the Viterbi algorithm, we use it in a fixed delay mode, as do other authors. By appealing to known results for fixed-lag smoothing of linearly-observed data, we are able to intelligently choose the fixed delay. Without significant performance loss we decode phase-symbol pairs at a depth constant of $k_0 = 10$. This obviates the need for huge storage requirements for long sequences. With these modifications the Viterbi algorithm becomes a feasible, albeit sophisticated, decoding procedure.

Simulation results for the proposed Viterbi algorithm (VA) are presented for several symbol sets consisting of 2, 8 or 16 symbols. Several types of phase jitter are investigated such as Gaussian and non-Gaussian random walk, and sinusoidal phase jitter. The resulting error probabilities are compared with those of the simpler decision-directed algorithms (JE and DDPLL), and those of the DBVA. As expected, performance of the VA is always superior to that of the other systems. On the other hand the increase in computational burden is substantial and the improvement in performance is not always great enough to warrant the use of the VA. In our concluding remarks we discuss situations in which one might reasonably use the VA rather than a simpler decision-directed algorithm such as the JE or the DDPLL.

Remarks on Notation:

Throughout this paper $\perp\!\!\!\perp$ denotes statistical independence. The notation $\{\phi_k\}_1^K$ will mean the set $\{\phi_k, k=1,2,\ldots,K\}$. When the indexes 1 and K are missing (e.g., $\{\phi_k\}$), it is understood that K is infinite. The symbol N^+ denotes the positive integers. The notation $x:N_x(\mu,\sigma^2)$ means the random variable x is normally distributed with mean μ and variance σ^2; $N_x(\mu,\sigma^2)$ will also be used to denote the function $(2\pi\sigma^2)^{-\frac{1}{2}} \exp\{-(x-\mu)^2/2\sigma^2\}$. When x is complex, $x:N_x(\mu,\sigma^2)$ means x is complex with density $N_x(\mu,\sigma^2) = (2\pi\sigma^2)^{-1}\exp\{-|x-\mu|^2/2\sigma^2\}$. By f(x/y) we mean the conditional probability density of the random variable x, given the random variable y. Thus f(x/y) is generally a different function than f(w/z), even though we use no explicit subscripting such as $f_{w/z}(\cdot/\cdot)$ to indicate so. We make no notational distinction between a random variable and its realizations, relying instead on context to make the meaning clear. A density function for a random variable, evaluated at a particular realization of the random variable is termed a likelihood function. "Hatted" variables such as $\hat{\phi}_k$ refer always to MAP estimates that maximize an a posteriori density.

Finally it is convenient to define the function

$$g_M(x) = M^{-1} \sum_{m=1}^{M} \sum_{\ell=-\infty}^{\infty} h[x-\ell 2\pi-(m-1)\ 2\pi/M] \qquad (1)$$

where $h(\cdot)$ is a probability density. The function $g_M(\cdot)$ plays an important role in our discussion of phase-symbol decoding on QASK symbol sets.

II. Signal and Phase Models

Assume complex data symbols $\{a_k\}$ are phase or phase-amplitude modulated onto a carrier and transmitted over a channel with linear distortion and phase jitter. The received signal, call it y(t), is typically processed as illustrated in Figure 1. The signal y(t) is passed through a bandpass noise filter and demodulated with two quadrature waveforms. The resulting complex baseband signal $x_1(t) + jx_2(t)$ is equalized with a complex adaptive equalizer in order to reduce the intersymbol interference due to linear distortion in the channel. The equalized signal is a sequence of samples at symbol rate $1/\Delta$ (Δ is the interval between successive data symbols). The output of the equalizer is a complex sequence $x_k = x_k^{(1)} + jx_k^{(2)}$ which is a noisy, phase-distorted, version of the original transmitted sequence. Thus we write

$$x_k = a_k e^{j\phi_k} + n_k, \quad k\varepsilon N^+ \qquad . \qquad (2)$$

Here $\{a_k\}$ is the complex symbol sequence, typically encoded according to one of the diagrams illustrated in Figure 2. The sequence $\{\phi_k\}$ represents phase fluctuations (jitter and frequency drift) in the channel. The two real components $n_k^{(1)}$ and $n_k^{(2)}$ of the complex noise sequence $n_k = n_k^{(1)} + jn_k^{(2)}$ are the noise variables in the respective baseband quadrature equalized channels. The variables $n_k^{(1)}$ and $n_k^{(2)}$ can be shown to be independent when the carrier frequency is in the middle of the input noise filter bandwidth and the additive channel noise is white. If the equalizer is perfect, then n_k is the usual Gaussian, additive noise with zero-mean. If the equalizer is not perfect, then n_k contains a residual of the intersymbol interferences, and is not Gaussian; nor are successive variables $n_k^{(1)}$, $n_{k+1}^{(1)}$,..., independent. However, for a

reasonably good equalizer, we may assume that $\{n_k\}$ is a sequence of independent identically distributed (i.i.d.) complex Gaussian variables. Strictly speaking this assumption is valid only at the input to the equalizer when the baseband equivalent of the input noise filter and low-pass demodulator is the so-called sampled, whitened-matched filter of [10]. In practice the assumption of Gaussianity is more realistic than the assumption of independence for the sequence $\{n_k\}$. Assuming that the equalizer of Figure 1 is perfect we model the noise sequence $\{n_k\}$ as follows:

$$\begin{aligned} & n_k = n_k^{(1)} + jn_k^{(2)}, \quad k\epsilon N^+ \\ & n_k^{(1)} \perp\!\!\!\perp n_\ell^{(2)} \ \forall(k,\ell) \\ & n_k^{(1)} \perp\!\!\!\perp n_\ell^{(1)}, \ k \neq \ell; \ n_k^{(2)} \perp\!\!\!\perp n_\ell^{(2)}, \ k \neq \ell \\ & n_k^{(1)}: \ N(0,\sigma_n^2); \ n_k^{(2)}: \ N(0,\sigma_n^2) \end{aligned} \tag{3}$$

Here $2\sigma_n^2$ is the variance of the complex noise variable n_k and σ_n^2 is the variance of each real component.

Consider now the phase distortion $\{\phi_k\}$. The term generally reflects two effects, one long-term and the other short-term. In modern high speed data modems no carrier or pilot tone is transmitted for locking the local oscillator at the receiver. Thus long-term, large-range linear phase variations result from frequency drift in the channel which cannot be eliminated. In addition, nonlinear intermodulation with local power supplies gives rise to short-term, small-range phase variations. The variations exhibit energetic harmonic content at the harmonics of the fundamental power supply frequency. Hence a realistic model for $\{\phi_k\}$ is

$$\phi_k = (\phi_0 + 2\pi Bk) + \sum_{\ell=1}^{p} A_\ell \sin(2\pi\nu_\ell k\Delta + \rho_\ell),\ k\epsilon N^+ \tag{4}$$

where $\nu_\ell = \ell \cdot 50$ Hz or $\nu_\ell = \ell \cdot 60$ Hz, depending on the place of use. A typical phase process is depicted in Figure 3. The first term in parentheses in (4) is the so-called frequency drift term and the summation term is the phase jitter. In practice the constants ϕ_0, B, $\{A_\ell, \nu_\ell, \rho_\ell\}_{\ell=1}^{p}$ vary with time $k\Delta$, but at an extremely slow rate.

The spectrum of the phase jitter, i.e. the behavior of A_ℓ vs. ν_ℓ has been investigated experimentally in [14]. The spectrum is roughly fitted by a $1/\nu^2$ curve. A phenomenological model for phase having a $1/\nu^2$ spectrum (like that of phase jitter at high frequencies) is the Wiener-Levy continuous time process,

$$\frac{d\phi(t)}{dt} = w(t),\ t \geq 0, \tag{5}$$

where $\{w(t)\}$ is a white noise process. The discrete time analog is the independent increments sequence

$$\phi_k = \phi_{k-1} + w_k,\ k\epsilon N^+ \tag{6}$$

where $\{w_k\}$ is a sequence of i.i.d. random variables with even probability density h(w).[2] When $w_k : N_w(0,\sigma_w^2)$, then $\{\phi_k\}$ is the so-called normal random walk.

In detail the model of (6) falls well short of a reputable probabilistic model for phase, because at low frequencies the spectrum is unbounded. Furthermore the spectrum is not integrable, corresponding to the unbounded growth of the variance in the diffusion model of (6). However, in gross terms, i.e. for short-term fluctuations, the model captures, with appropriate selection of h(w), the correlated evolution of phase. The main virtue of the independent increments model is that

[2] That is, h(w) = h(-w).

it forms a convenient basis from which to derive estimator structures which may then be evaluated against more realistic phase sequences.

As the measurement model of (2) is invariant to modulo-2π translates of ϕ_k we may represent phase as if it were a random sequence on the unit circle C or equivalently on the interval $[-\pi,\pi)$. Call $\bar{\phi}_k$ this representation of ϕ_k. Note $\bar{\phi}_{k+1}$ may be written

$$\bar{\phi}_{k+1} = \bar{\phi}_k + \bar{w}_k \tag{7}$$

where the + denotes modulo-2π addition of real variables or equivalently rotation with positive (counter-clockwise) sense on C. The variable $\bar{w}_k$ is a modulo-2π version of w_k.

The conditional density of $\tilde{\phi}_{k+1} \triangleq \bar{\phi}_k + w_k$, given $\bar{\phi}_k$, is $h(\tilde{\phi}_{k+1}-\bar{\phi}_k)$. Since $\bar{\phi}_{k+1}$ is a modulo-2π version of $\tilde{\phi}_{k+1}$, we may reflect all of the conditional probability mass into C to obtain the transition (or conditional) probability density

$$f(\bar{\phi}_{k+1}/\bar{\phi}_k) = \sum_{\ell=-\infty}^{\infty} h(\bar{\phi}_{k+1}-\bar{\phi}_k-\ell 2\pi) = g_1(\bar{\phi}_{k+1}-\bar{\phi}_k) \tag{8}$$

where g_1 is the function defined in (1). Hereafter $g_1(\cdot)$ is called the folded density of the phase increments. Usually the phase increment is small and its distribution $h(\cdot)$ is very narrow with respect to 2π. Therefore, in the sum of (8) only one term is relevant and $f(\bar{\phi}_{k+1}/\bar{\phi}_k) \doteq h(\bar{\phi}_{k+1}-\bar{\phi}_k)$. In the normal case this imples $\sigma_w << 2\pi$, where σ_w^2 is the variance of w_k. As it is cumbersome to carry around the overbar notation $\bar{\phi}_{k+1}-\bar{\phi}_k$, we drop it with the caution that from here on ϕ_k is defined on C unless otherwise stated.

In the normal case [7], [8], the density $g_1(\phi_{k+1}-\phi_k)$ may be written

$$g_1(\phi_{k+1}-\phi_k) = \sum_{\ell=-\infty}^{\infty} N_{\phi_{k+1}}(\phi_k+\ell 2\pi,\sigma_w^2) \tag{9}$$

This case and the Cauchy case (in which the distribution tails are much heavier than the normal tails) are studied in Appendix A. It is shown that $g_1(x)$ achieves its maximum at $x = 0$ and that it is monotone decreasing on $0 \leq x \leq \pi$.

The sequence $\{\phi_k\}_1^K$ is Markov. Therefore, we may write for the joint density of the K phases $\{\phi_k\}_1^K$

$$f(\{\phi_k\}_1^K) = \prod_{k=1}^{K} f(\phi_{k+1}/\phi_k) \tag{10}$$

$f(\{\phi_1/\phi_0) \triangleq f(\phi_1)$: the marginal density of ϕ_1 .

Usually ϕ_1 is uniformly distributed on C, because phase acquisition starts at k = 1 with no prior information about its value. By the independence of the n_k in (2) it follows that the conditional density of the measurement sequence $\{x_k\}_1^K$, given the phase and data sequences $\{\phi_k\}_1^K$, $\{a_k\}_1^K$, is

$$f(\{x_k\}_1^K/\{\phi_k\}_1^K, \{a_k\}_1^K) = \prod_{k=1}^{K} N_{x_k}(a_k e^{j\phi_k}, \sigma_n^2) \quad . \tag{11}$$

Equations (8)-(11) form the basis for the derivation of a MAP sequence estimator. The key element is that $\{\phi_k\}$ is a Markov sequence with a bounded range space $[-\pi,\pi)$. Discretization of this bounded interval leads to a finite-state model from which a finite dimensional dynamic programming algorithm can be derived.

III. Decision-Directed Algorithms

The usual way of dealing with phase fluctuations is to design a phase estimator and use the estimated phase, call it $\hat{\phi}_k$, to rotate the received signal as follows:

$$y_k = x_k e^{-j\hat{\phi}_k}, \quad k\epsilon N^+ \qquad (12)$$

The phase corrected signal y_k is then fed to a decision device which, in turn, delivers the symbol estimate $\hat{a}_k$. Typically the phase estimate $\hat{\phi}_k$ is functionally dependent on the old measurements $\{..., x_{k-2}, x_{k-1}\}$ and the past symbol estimates $\{..., \hat{a}_{k-2}, \hat{a}_{k-1}\}$. If a carrier or pilot tone is transmitted as in single sideband (SSB) systems, then $\hat{\phi}_k$ is obtained from a simple phase-locked loop (PLL). In suppressed carrier systems such as PSK or QASK systems the PLL is "decision-directed". That is, $\hat{\phi}_k$ is updated on the basis of $\hat{a}_{k-1}$. For instance in [5]

$$\begin{aligned}\hat{\phi}_{k+1} &= \hat{\phi}_k + \mu \mathrm{Im}[x_k \hat{a}_k^* e^{-j\hat{\phi}_k}] \\ &\doteq \hat{\phi}_k + \mu_k \sin(\arg x_k - \arg \hat{a}_k - \hat{\phi}_k) \quad , \quad \mu_k = \mu |x_k|^2\end{aligned} \qquad (13)$$

where * denotes complex conjugate and μ is a constant that depends on the signal-noise ratio. The estimator of (13) is called a DDPLL.

In the jitter equalizer (JE) of [3] and [4], x_k is rotated and scaled as follows:

$$\begin{aligned}y_k &= x_k G_k \\ G_k &= G_{k-1} + \mu(\hat{a}_{k-1} - y_{k-1}) x_{k-1}^*\end{aligned} \qquad (14)$$

The complex gain G_k is the single complex coefficient of a one-coefficient rapidly-adaptive equalizer. We may think of $G_k/|G_k|$ as the phase correction $e^{-j\hat{\phi}_k}$, and $|G_k|$ as a gain correction $\hat{c}_k$. Thus, although there is no explicit formulation of a phase-gain estimation problem in [3] and [4], the net effect of the JE is to correct phase

and normalize rapid fading variations. As explained in [4], when phase fluctuations are large, the JE performance may be improved by setting a constraint on G_k that keeps its value inside a given domain including the complex point (0,1).

Geometrical Comments:

The combined effects of random phase fluctuations and additive noise may be illustrated as in Figure 4(a). The transmitted symbol $a_k = a^{(0)}$ (say) is rotated by the random phase angle ϕ_k to give $a_k e^{j\phi_k}$. To this is added the complex noise sample n_k to give the measurement x_k defined in (2). For the case illustrated, the resultant measurement is closer to symbol $a^{(1)}$ than to $a^{(0)}$ and consequently, with no phase or phase-gain correction, a decoding error would be made. To emphasize the combined effects of phase fluctuation and additive noise, we have illustrated a case for which either phase jitter or additive noise alone would cause no error. See [11] for a probabilistic discussion of this issue. Figure 4(b) is an illustration of how a DDPLL works. The angle ψ_k is the noisy measured phase (arg x_k) minus the sum of the phase of the decoded symbol and the previously estimated phase (arg $\hat{a}_k+\hat{\phi}_k$). A given amount μ_k of this angle is added to $\hat{\phi}_k$ as a correction to get the new phase estimate $\hat{\phi}_{k+1} = \hat{\phi}_k + \mu_k\psi_k$. Note that only phase is corrected. In the JE both phase and gain are corrected, offering potential for improved performance. This potential is particularly important in QASK symbol sets where amplitude errors in x_k can result in decoding errors.

IV. Map Phase and Symbol Sequence Decoding with the Viterbi Algorithm

The basic idea behind MAP sequence decoding is to find a sequence of phase-symbol pairs $\{\phi_k, a_k\}_1^K$ that, based on the observation sequence $\{x_k\}_1^K$, appears most likely. The application of this idea to data communication was first proposed in [1] and refined in [9]. The most likely sequence, call it $\{\hat{\phi}_k, \hat{a}_k\}$, is the sequence that maximizes the natural logarithm (or any other monotone function) of the a posteriori density of $\{\phi_k, a_k\}_1^K$, given the sequence of observations $\{x_k\}_1^K$. Thus we pose the maximization problem:

$$\max_{\{\phi_k\}_1^K,\ \{a_k\}_1^K} \ln f(\{\phi_k\}_1^K,\ \{a_k\}_1^K / \{x_k\}_1^K) \quad . \tag{15}$$

This is equivalent to maximizing the natural logarithm of the likelihood function $f(\{x_k\}_1^K,\ \{\phi_k\}_1^K,\ \{a_k\}_1^K)$, obtained by evaluating the joint density function for $\{x_k\}_1^K$, $\{\phi_k\}_1^K$, and $\{a_k\}_1^K$, at the observed values of $\{x_k\}_1^K$. Using the results of (10) and (11) we may write:

$$f(\{x_k\}_1^K,\ \{\phi_k\}_1^K,\ \{a_k\}_1^K) = \prod_{k=1}^{K} N_{x_k}(a_k e^{j\phi_k}, \sigma_n^2)\ f(\phi_k/\phi_{k-1}) f(\{a_k\}_1^K). \tag{16}$$

Assuming the $\{a_k\}_1^K$ to be a sequence of independent, equally likely symbols, using (8), and neglecting uninteresting constants, we may write the maximization problem as

$$\max_{\{\phi_k\}_1^K,\ \{a_k\}_1^K} \Gamma_K$$

$$\Gamma_K = -\frac{1}{2\sigma^2} \sum_{k=1}^{K} |x_k - a_k e^{j\phi_k}|^2 + \sum_{k=2}^{K} \ln g_1(\phi_k - \phi_{k-1}) + \ln f(\phi_1) \quad . \tag{17}$$

Note that Γ_k satisfies the recursion

$$\Gamma_k = \Gamma_{k-1} + p_k \qquad k = 2,3,\ldots$$

$$p_k = -\frac{1}{2\sigma_n^2}\left|x_k - a_k e^{j\phi_k}\right|^2 + \ln\, g_1(\phi_k - \phi_{k-1}),\ k = 2,3,\ldots \tag{18}$$

$$\Gamma_1 = -\frac{1}{2\sigma_n^2}\left|x_1 - a_1 e^{j\phi_1}\right|^2 + \ln\, f(\phi_1)$$

where p_k is the so-called path-metric. For convenience, let us make explicit in Γ_K the last phase and symbol: $\Gamma_K(\phi_K, a_K)$. The other arguments $\{\phi_k\}_1^{K-1}$, $\{a_k\}_1^{K-1}$, remain implicit. Then, from (18)

$$\Gamma_K(\phi_K, a_K) = \Gamma_{K-1}(\phi_{K-1}, a_{K-1}) + p_K(x_K, a_K, \phi_K, \phi_{K-1}) \tag{19}$$

Thus, the maximizing sequence, call it $(\{\hat{\phi}_k\}_1^K, \{\hat{a}_k\}_1^K)$, passing through $(\hat{\phi}_{K-1}, \hat{a}_{K-1})$ on its way to $(\hat{\phi}_K, \hat{a}_K)$, must arrive at $(\hat{\phi}_{K-1}, \hat{a}_{K-1})$ along a route $(\{\hat{\phi}_k\}_1^{K-2}, \{\hat{a}_k\}_1^{K-2})$ that maximizes $\Gamma_{K-1}(\hat{\phi}_{K-1}, \hat{a}_{K-1})$. It is this observation which forms the basis of forward dynamic programming. In the actual implementation of a dynamic programming algorithm, one must discretize the phase space C to a finite dimensional grid of phase values $\underline{\underline{\xi}} = \{\xi_n\}_{n=1}^m$. The function $\ln\, g_1(\phi_k - \phi_{k-1})$ is then defined on the two-dimensional grid $\underline{\underline{\xi}} \times \underline{\underline{\xi}}$. However, as discussed in [8] and [9] the resulting m x m matrix of conditional probabilities has Toeplitz symmetry which means only an m vector of conditional probabilities must be computed and stored.

The Viterbi algorithm for simultaneous phase and symbol decoding consists simply of an algorithm which determines survivor phase-symbol sequences terminating at each possible phase-symbol pair. One of these surviving sequences is ultimately decoded as the approximate MAP phase-symbol sequence. The complexity c of the algorithm lies mainly in the

evaluation of the mM possible values of $|x_k - a_k e^{j\phi_k}|^2$, for each new measurement x_k. Here M is the symbolling alphabet size and m is the number of discrete phase values. For each calculation of $|x_k - a_k e^{j\phi_k}|^2$ there are 6 real multiplies. Compared to this multiplication load of 6mM per sample, the determination and addition of the m possible values of $\ell n\, g_1(\phi_k - \phi_{k-1})$ that appear in (18) is negligible. The determination of $|x_k - a_k e^{j\phi_k}|^2$ would likely be computed in a pipe-lined parallel architecture, while the terms $\ell n\, g_1(\cdot)$ would be read by appropriately addressing ROM. When there are many symbols and short-term phase fluctuations have small amplitude (σ_w small), so that m must be large for accurate phase tracking, then the complexity is great. For example with M=8 and m=48, c α (384), indicating on the order of 2×10^3 computations at each k-step.

As we show in the next section the complexity of the Viterbi algorithm can be dramatically reduced by making a change of variable and tracking a total phase variable that is the sum of ϕ_k and the symbol phase, arg a_k. And, of course, for PSK symbol sets M may be set to unity because only one symbol amplitude is admissible and admissable symbol phases may be chosen to fall on one of the discrete phase values. Thus for PSK symbol sets the complexity is simply m and the number of path metric computations is on the order of 300 for m=48. Even this figure may be reduced by using one of a variety of so-called M-algorithms in which all survivor states are saved but only a handful of candidate originator states are considered for each survivor.

V. A Principle of Optimality for Phase-Amplitude Coded Symbols and an Efficient Two-Step Decoding Procedure

In order to simplify matters and to illustrate the key ideas, let us consider PSK symbols of the form

$$a_k = e^{j\theta_k} \tag{20}$$

with $\{\theta_k\}$ drawn independently from an M-ary equi-probable alphabet $\Theta = \{(\ell=1)2\pi/M\}_{\ell=1}^{M}$. Write the measurement model of (2) as

$$x_k = e^{j\psi_k} + n_k \tag{21}$$

where the total phase ψ_k is represented as follows:

$$\psi_k = \phi_k + \theta_k$$

$$\theta_k = \sum_{\ell=1}^{k} \Delta\theta_\ell, \quad \Delta\theta_k = \theta_k - \theta_{k-1}, \quad \Delta\theta_1 = \theta_1 \quad . \tag{22}$$

It is clear that $\hat{\theta}_k = \sum_{\ell=1}^{k} \Delta\hat{\theta}_\ell$ and $\hat{\phi}_k = \hat{\psi}_k - \hat{\theta}_k$. Thus we may replace the MAP sequence estimation problem posed in (15) by the problem

$$\max_{\{\psi_k\}_1^K,\ \{\Delta\theta_k\}_1^K} f(\{x_k\}_1^K,\ \{\psi_k\}_1^K,\ \{\Delta\theta_k\}_1^K) \tag{23}$$

The joint density $f^K \triangleq f(.,.,.)$ in (23) may be written

$$f^K = \prod_{k=1}^{K} N_{x_k}(e^{j\psi_k}, \sigma_n^2) f(\psi_k, \Delta\theta_k / \{\psi_j\}_1^{k-1}, \{\Delta\theta_j\}_1^{k-1}) \tag{24}$$

where for k=1, $f(\psi_1, \Delta\theta_1/.,.)$ is simply the marginal density $f(\psi_1, \Delta\theta_1)$. The conditional density on the right hand side of (24) is easily evaluated with Bayes' rule:

$$f(\psi_k, \Delta\theta_k / \{\psi_j\}_1^{k-1}, \{\Delta\theta_j\}_1^{k-1}) = f(\psi_k / \{\psi_j\}_1^{k-1}, \{\Delta\theta_j\}_1^{k}) \cdot$$

$$f(\Delta\theta_k / \{\psi_j\}_1^{k-1}, \{\Delta\theta_j\}_1^{k-1}) \quad . \tag{25}$$

Now $\Delta\theta_k$ is independent of the previous data, additive noise and phase fluctuations. Thus

$$f(\Delta\theta_k/\{\psi_j\}_1^{k-1},\ \{\Delta\theta_j\}_1^{k-1}) = \frac{1}{M} \quad . \tag{26}$$

Moreover if we rewrite ψ_k as

$$\begin{aligned}\psi_k &= \phi_{k-1} + w_k + \theta_{k-1} + \theta_k - \theta_{k-1} \\ &= \psi_{k-1} + \Delta\theta_k + w_k \quad ,\end{aligned} \tag{27}$$

we see immediately that

$$f(\psi_k/\{\psi_j\}_1^{k-1},\ \{\Delta\theta_j\}_1^{k}) = g_1(\psi_k - \psi_{k-1} - \Delta\theta_k) \quad . \tag{28}$$

Recall ψ_k is defined on the circle C. Therefore, for clarity we might think of ψ_k as a random variable $\psi_{k-1} + \Delta\theta_k + w_k$, whose density is folded in $[-\pi,\pi)$. Putting (24)-(28) together, we have for the joint density f^K

$$\begin{aligned}f^K &= \prod_{k=1}^{K} N_{x_k}(e^{j\psi_k}, \sigma_n^2)\ \frac{1}{M}\ g_1(\psi_k - \psi_{k-1} - \Delta\theta_k) \\ \Delta\theta_1 &\triangleq \theta_1,\ \psi_0 \triangleq 0 \quad .\end{aligned} \tag{29}$$

Principle of Optimality: Call $\{\psi_k\}_1^K$, $\{\Delta\theta_k\}_1^K$ the MAP sequences that maximize f^K; $\{\Delta\theta_k\}_1^K$ enters only in the $g_1(\cdot)$ term on the right hand side of (29). Now let us suppose (as is usual) that $g_1(w)$, which is even, is also unimodal with a peak at $w = 0$. This single-mode assumption for $g_1(\cdot)$ is valid in particular when the phase increment w_k in the Markov-process (6) has a Gaussian or Cauchy distribution $h(w)$. See Appendix A. It follows that f^K is maximized by choosing

$$\Delta\hat{\theta}_k = [\hat{\psi}_k - \hat{\psi}_{k-1}] \tag{30}$$

where $[x]$ denotes the closest value of $(\ell-1)2\pi/M$ to x. By substitution of the constraint (30) into (29) and defining the "rest" function $R(x)$ on the circle C by

$$R(x) = x - [x] \tag{31}$$

we find that

$$\hat{f}^K = \prod_{k=1}^{K} N_{x_k}(e^{j\psi_k}, \sigma_n^2)\, \frac{1}{M}\, g_1(R(\psi_k - \psi_{k-1})) \tag{32}$$

The maximization of $\hat{f}^K$ with respect of $\{\psi_k\}_1^K$ is formally equivalent to maximizing the joint density $f(\{x_k\}_1^K, \{\psi_k\}_1^K)$ when the total phase ψ_k follows a Markov-model similar to (6):

$$\psi_k = \psi_{k-1} + u_k \tag{33}$$

Here the independent increments u_k have "probability density", folded on the circle C,

$$f(u) = \frac{1}{M}\, g_1(R(u)) \quad . \tag{34}$$

This interpretation is purely formal since f(u) is not generally a probability density. However when

$$g_1(u) = 0, \quad |u| \geq \frac{\pi}{M} \tag{35}$$

then f(u) is a probability density because in that case

$$\frac{1}{M}\, g_1(R(u)) = g_M(u) \quad . \tag{36}$$

Thus (34) can be interpreted as an approximate density when the peak of g(u) is narrower than the minimum phase distance between the symbols. This condition is always satisfied in communications applications. Otherwise phase distortion is so large that data transmission is not possible. Thus we have a pure phase-tracking problem as in [8] and [9] and we may proceed accordingly. Taking the natural logarithm of $\hat{f}^K$ we have the maximization problem:

$$\max_{\{\psi_k\}_1^K} \Gamma_K' ,$$

$$\Gamma_k' = \Gamma_{k-1}' + p_k' \; ; \; \Gamma_1' = -\frac{1}{2\sigma_n^2} |x_1 - e^{j\psi_1}|^2 + \ell n \; g_1(R(\psi_1)) \tag{37}$$

$$p_k' = -\frac{1}{2\sigma_n^2} |x_k - e^{j\psi_k}|^2 + \ell n \; g_1[R(\psi_k - \psi_{k-1})]$$

which is solved by the dynamic programming algorithm discussed in Section IV. The complexity c′ of this algorithm lies essentially in the evaluation of the m possible values of $|x_k - e^{j\psi_k}|^2$ for each new data value x_k. The m different values of $\ell n \; g_1[R(\cdot)]$ will be pre-computed and stored in ROM. For each computation of $|x_k - e^{j\psi_k}|^2$ there are 6 multiplies, so complexity is simply proportional to m.

This represents a reduction in complexity proportional to M for M-ary PSK.

Usually, the phase is differentially modulated rather than directly modulated and therefore the relevant symbol is $\Delta\hat{\theta}_k$ itself (see (30)). For the purpose of data transmission, there is no need to reconstruct the absolute data phase $\hat{\theta}_k = \sum_{\ell=1}^{k} \Delta\hat{\theta}_\ell$. This reconstruction has, however, been carried out in the simulations in order to recover the estimates $\hat{\phi}_k = \hat{\psi}_k - \hat{\theta}_k$ of the phase fluctuations, and to get the approximate variance of the phase estimates

$$\hat{\sigma}_\phi^2 = \frac{1}{K} \sum_{k=1}^{K} |\phi_k - \hat{\phi}_k|^2 \quad . \tag{39}$$

Geometrical Comments and Densities Galore: The entire development of this section has a nice geometric interpretation which we illustrate in Figure 5. In Figure 5(a) the basic phase noise density h(x) is illustrated on $(-\infty,\infty)$. Figure 5(b) is the folded version $g_1(x)$ of h(x) to account for the wrapping on the unit circle C. Figure 5(c) is the function $g_1[R(x)]$ that arises in our discussion of the principle of

optimality, sketched in the case of 4-ary phase modulation. Figure 5(d) shows $g_1[R(x)]$ wrapped around the circle C. Since $g_1(x)$ is very narrow, $g_1[R(x)]$ is approximately the repeated copy of $g_1(x)$ at all possible values of data phase. With $x = \hat{\psi}_k - \hat{\psi}_{k-1}$, Figure 5(d) illustrates the choice of $\Delta\theta_k$ nearest $\hat{\psi}_k - \hat{\psi}_{k-1}$ ($\Delta\theta_k = \pi/2$ is the best choice here), and the resulting value of $g_1[R(\hat{\psi}_k - \hat{\psi}_{k-1})]$ is shown by the heavy segment on the axis $\hat{\psi}_k$, terminated by the heavy dot.

We now extend this principle of optimality to phase-amplitude encoded symbols. Assume the independent, equally probable data symbols are complex symbols of the form

$$a_k = A_k e^{j\theta_k} \tag{40}$$

with the A_k positive real numbers drawn independently from the alphabet $A = (\alpha_1, \alpha_2, \ldots, \alpha_L)$. Denote by $p(A_k)$ the probability mass function for the random variable A_k. Assume the θ_k are drawn from the alphabet $B = (\beta_1, \beta_2, \ldots, \beta_M)$. Denote the conditional probability mass function of θ_k, given by A_k, by $p(\theta_k/A_k)$. For the (4,4) diagram of Figure 2,

$A = (\sqrt{2}\alpha_1, 3\alpha_1, 3\sqrt{2}\alpha_1, 5\alpha_1)$; $B = \{b_i\}_{i=1}^{8}$, $b_i = (i-1)\frac{\pi}{4}$. The probabilistic description of the source is

$$p(A_k) = 1/4 \text{ for all } A_k$$

$$p(\theta_k/A_k = \alpha_1) = \begin{cases} 1/4, & \theta_k = \beta_2, \beta_4, \beta_6, \beta_8 \\ 0, & \text{otherwise} \end{cases}$$

$$p(\theta_k/A_k = \alpha_3) = p(\theta_k/A_k = \alpha_1) \tag{41}$$

$$p(\theta_k/A_k = \alpha_2) = \begin{cases} 1/4, & \theta_k = \beta_1, \beta_3, \beta_5, \beta_7 \\ 0, & \text{otherwise} \end{cases}$$

$$p(\theta_k/A_k = \alpha_4) = p(\theta_k/A_k = \alpha_2)$$

In place of the maximization problem posed in (23) we write

$$\max_{\{\psi_k\}_1^K,\ \{\Delta\theta_k\}_1^K,\ \{A_k\}_1^K} f(\{x_k\}_1^K,\ \{\psi_k\}_1^K,\ \{\Delta\theta_k\}_1^K,\ \{A_k\}_1^K) \tag{42}$$

with ψ_k and $\Delta\theta_k$ defined as in (22). The density $f^K(.,.,.,.)$ appearing in (42) may be written

$$f^K = \sum_{k=1}^{K} N_{x_k}\left(A_k e^{j\psi_k}, \sigma_n^2\right) f(\psi_k, \Delta\theta_k, A_k / \{\psi_j\}_1^{k-1},\ \{\Delta\theta_j\}_1^{k-1},\ \{A_j\}_1^{k-1}) \ . \tag{43}$$

The conditional density on the right-hand side of (43) is simply

$$f(\psi_k, \Delta\theta_k, A_k/.,.,.) = g_1(\psi_k - \psi_{k-1} - \Delta\theta_k)\ p(\Delta\theta_k / A_k, A_{k-1})\ p(A_k) \tag{44}$$

where $p(\Delta\theta_k / A_k, A_{k-1})$ is the conditional probability mass function for $\Delta\theta_k$, given A_k and A_{k-1}. Putting (43) and (44) together we have as the joint density function to be maximized

$$f^K = \prod_{k=1}^{K} N_{x_k}\left(A_k e^{j\psi_k}, \sigma_n^2\right)\ g_1(\psi_k - \psi_{k-1} - \Delta\theta_k)\ p(\Delta\theta_k / A_k, A_{k-1})\ p(A_k) \tag{45}$$

It is important to note in this expression that the $N_{x_k}(.,.)$ term is dependent only on the measurement model; $g_1(\cdot)$ is dependent only on the random phase model, and $p(\Delta\theta_k/.,.)\ p(A_k)$ is dependent only upon the symbolling constellation (or encoding scheme). Thus (45) is a useful canonical decomposition that is generally applicable to communications problems involving additive independent noise and independent increments phase processes.

For the (4,4) diagram of Figure 2 we may compute $p(\Delta\theta_k / A_k, A_{k-1})$ as follows:

$$p(\Delta\theta_k / A_k = \alpha_i, A_{k-1} = \alpha_j) = \begin{cases} 1/4, \ \Delta\theta_k = \beta_1, \beta_3, \beta_5, \beta_7, \ i,j \text{ even-even or odd-odd} \\ 1/4, \ \Delta\theta_k = \beta_2, \beta_4, \beta_6, \beta_8, \ i,j \text{ even-odd or odd-even} \end{cases} \tag{46}$$

It is a straight-forward matter to substitute these results into (45) and derive a path-metric as in (37).

VI. Linear Performance Results and the Selection of a Fixed Lag

There is one more simplification to be made: namely the selection of a depth constant k_0 such that phase-symbol pairs may be decoded at a fixed-lag k_0, thereby obviating the need to store long survivor sequences. The idea is the following. Call $\{\hat{\psi}_{k/K}\}_1^K$ the MAP phase sequence based on measurements $\{x_k\}_1^K$. The subscript k/K indicates that $\hat{\psi}_{k/K}$ depends on all measurements up to time K. In general the MAP sequence $\{\hat{\psi}_{k/K+1}\}_1^{K+1}$ based on measurements to time (K+1) may differ from $\{\hat{\psi}_{k/K}\}_1^K$ at all values of $1 \leq k \leq K$. However, one expects that for large K and for $k \leq K - k_0$, the sequences $\{\hat{\psi}_{\ell/K}\}_1^k$ and $\{\hat{\psi}_{\ell/K+1}\}_1^k$ will not be very different for a well chosen depth k_0. In other words, long survivor sequences have one common trunk up to $K-k_0$, at which point they may diverge as illustrated in Figure 6. Thus we may use $\hat{\psi}_{K-k_0/K}$ as a final estimate of $\hat{\psi}_{K-k_0}$ since $\hat{\psi}_{K-k_0/K+\ell} \doteq \hat{\psi}_{K-k_0/K}$ for all ℓ. Thus, as a practical matter, one may choose a depth constant k_0 such that the sequence of fixed-lag estimates $\hat{\psi}_{k-k_0/k}$, $k = k_0 + 1, k_0 + 2, \ldots$, gives an approximate MAP sequence. Here $\hat{\psi}_{k-k_0/k}$ is simply the phase value, k_0 samples back, in the MAP sequence based on measurements up to time k. In this way phase values are estimated with delay k_0 and only survivor sequences of length k_0 must be stored.

How to choose k_0? This is a difficult question to answer precisely because there exist no analytical results for the performance of nonlinear phase trackers of the Viterbi-type. We can, however, study the filtering behavior of a related linear problem and find how performance varies with fixed-lag k_0. To this end, we consider the problem of tracking phase when there is no data symboling. Assume $\{\psi_k\}$ is a normal random walk of the form (6) with $w_k : N_{w_k}(0, \sigma_w^2)$. Let $x_k = e^{j\psi_k} + n_k$,

$\{n_k\}$ a sequence of complex i.i.d. $N_{n_k}(0,\sigma_n^2)$ random variables. A PLL with gain K_1 for estimating $\{\psi_k\}$ is the following:

$$\hat{\psi}_k = \hat{\psi}_{k-1} + K_1|x_k| \sin(\arg x_k - \hat{\psi}_{k-1}) \tag{47}$$

Note that this is similar to (13) when there is no data.

For $\sigma_n^2 << 1$ we approximate (47) with

$$\hat{\psi}_k + \hat{\psi}_{k-1} + K_1(\arg x_k - \hat{\psi}_{k-1}) \quad . \tag{48}$$

When K_1 is selected to be

$$K_1 = (\sigma_w^2/\sigma_n^2)[-0.5 + 0.5\ (1+4\sigma_n^2/\sigma_w^2)^{\frac{1}{2}}] \tag{49}$$

then (48) is the Kalman filter for the "linear observation model"

$$\arg x_k = \psi_k + n_k \rightarrow x_k = \exp[j(\psi_k+n_k)] \tag{50}$$

The steady-state filtering error P_0 for this linear problem is related to K_1 as follows:

$$K_1 = \frac{\sigma_w^2}{\sigma_n^2} \cdot \frac{P_0}{\sigma_w^2} \quad . \tag{51}$$

A general result due to Hedelin [12] for fixed-lag smoothing may be adapted to random walk smoothing from observations of the form (50). The steady-state fixed-lag smoothing variance P_{k_0} at delay k_0 is

$$P_{k_0}/\sigma_w^2 = P_0/\sigma_w^2 - \sum_{\ell=1}^{k_0} G^{2\ell}$$

$$= P_0/\sigma_w^2 - G^2(1-G^{2k_0})/(1-G^2) \tag{52}$$

$$G = 1 - K_1 \quad .$$

The infinite-lag smoothing variance is

$$P_\infty/\sigma_w^2 = P_0/\sigma_w^2 - G^2/(1-G^2) \tag{53}$$

In Figure 7 several error expressions and asymptotic forms are plotted versus σ_w^2/σ_n^2, which is a kind of SNR. For large σ_w^2/σ_n^2, the

error variances P_o/σ_w^2, P_{10}/σ_w^2 and P_∞/σ_w^2 go as $(\sigma_w^2/\sigma_n^2)^{-1}$. For small σ_w^2/σ_n^2, they go as $(\sigma_w^2/\sigma_n^2)^{-\frac{1}{2}}$ although infinite-lag smoothing offers 6dB improvement in σ_w^2/σ_n^2 over zero-lag smoothing for a fixed smoothing variance. Over the range of values $0.01 \leq \sigma_w^2/\sigma_n^2 \leq 10$, a delay of $k_0 = 10$ offers all but 1 to 2dB of the theoretically achievable gain from infinite delay. In communication problems for which random phase is a significant effect, the ratio σ_w^2/σ_n^2 is typically in this range. Only at very small values of σ_w^2/σ_n^2 can very large delays k_0 provide large performance gains. But in this case there is no real phase fluctuation problem for the purpose of data decoding, and the gain is not worth the large delay. Shown also in Figure 7 is the Kalman gain K versus σ_w^2/σ_n^2.

The problem considered in Section IV is admittedly different from the linear problem considered here. However, the numerical results given in Figure 8 for the Viterbi phase tracker illustrate that the performance gain to be achieved with a fixed-lag of $k_0 = 10$ is much as predicted by the linear theory. Furthermore, over the range of values $0.1 \leq \sigma_w^2/\sigma_n^2 \leq 2$, the phase estimator variance for the Viterbi phase tracker operating with delay $k_0 = 10$ is essentially equivalent to the filtering variance of a Kalman filter that has access to linear observations and provides estimates without delay. Performance is not measurably degraded by the presence of data which is concurrently decoded. For the results of Figure 8, the phase space was discretized to $m = 48$ values. Data transmission was 8-ary PSK.

VII. Simulation Results: Gaussian Increments

For all simulation results discussed in this section the phase space $[-\pi,\pi)$ has been discretized to 48 equally-spaced phase values and a Viterbi algorithm has been programmed to solve the MAP sequence estimation problem. The principle of optimality established in Section V has been used to derive the appropriate path metric and thereby reduce computational complexity. The choice of a fixed-lag decoding (or depth) constant is $k_0=10$. Source symbols have been generated independently. The random phase sequence has been governed by the independent increments model of (2) with $w_k:N(0,\sigma_w^2)$ and initial phase uniformly-distributed on $[-\pi,\pi)$. Initial phase acquisition has been achieved by transmitting a preamble according to one of the following schemes.

a) During a pre-transmission period of length N, the sequence of transmitted data is known to the receiver. Thus, in the DBVA and VA systems, based upon MAP estimation, the Viterbi algorithm works as a pure phase estimator during this period. At the end of the preamble, the Viterbi algorithm is turned into a joint phase-data MAP estimator. In the DDPLL and JE systems, based upon decision-directed algorithms, the algorithm is directed by the true data during the preamble period.

b) During the preamble period, identical (but unknown) data are emitted. This keeps the phase away from severe sudden fluctuations, and makes the joint phase-data estimator able to adequately acquire the initial phase.

In our simulations the VA has achieved the same data-error probability for both methods; i.e. its performance has not depended upon which learning procedure was used. On the other hand, the DBVA has proved to be sensitive to the learning procedure.

For example, at SNR = 20dB, with phase variance $\sigma_w^2 = 4\ \sigma_n^2$, for a learning period of N=60 data, the number of errors during an emitting period of 490 data values has jumped from 7 for procedure a) - known data - to 59 for procedure b) - constant but unknown data. Moreover the DBVA typically requires a longer learning period than does the VA (roughly two times longer). A value of N=50 is sufficient for the VA, while the DBVA needs N=100 learning iterations in our simulations. The decision-directed systems (DDPLL and JE) work as the VA in these respects. That is, a preamble period of 50 data values is sufficient. These data may be unknown to the receiver, provided they are kept constant (procedure b). No degradation with respect to procedure a) results.

Binary Symboling: Shown in Figure 9 are binary symboling results for the VA when $\sigma_w^2 = 0.01\ \text{rad}^2$ (σ_w=5.7°) and SNR ranges from 4 to 10dB. (Recall SNR = $10\ \log_{10}\ 1/2\sigma_n^2$). For comparison the performance curves for coherent binary orthogonal and coherent binary antipodal systems are also shown. The simulation results for binary orthogonal symboling are of no inherent interest in their own right because even fully coherent binary orthogonal symboling provides only marginal gains over incoherent binary symboling at SNRs of practical interest. This point is made with curves 1 and 2 of Figure 9. However, the simulation results for binary orthogonal symboling serve to validate the simulation. The simulation results for binary antipodal symboling are interesting because incoherent reception is not possible with antipodal symboling.

The results indicate that performance with the VA is essentially equivalent to that of a fully coherent receiver - even for a relatively large value of σ_w.

8-PSK: Shown in Figure 10 are simulation results for 8-PSK when SNR ranges from 16-19dB and $(\sigma_w^2\sigma_n^2)^{\frac{1}{2}}$ remains fixed at 4.4×10^{-3} rad^2. The solid circles correspond to the VA and the solid triangles correspond to the markedly simpler JE. Also shown on Figure 1 are performance bounds for fully coherent 8-PSK and 16-PSK symboling. The values of σ_w^2 under investigation range from 1.6° to 2.2° and the ratio σ_w^2/σ_n^2 is very small, ranging from 0.03 to 0.12. In this case neither the VA nor the DBVA provides significant improvement over the JE or DDPLL. The latter two receivers are simpler than the DBVA which, in turn, is simpler than the VA. Therefore for such cases of weak phase noise, and a simple symbol constellation, neither the VA nor the DBVA would be favored over the JE or the DDPLL.

16-QASK: Shown in Figures 11, 12 and 13 are simulation results for 16-QASK symbols encoded according to the (4,4) CCITT rule. The decoding procedures are JE, DDPLL, DBVA and VA, for three distinct values of the ratio σ_w^2/σ_n^2. Figure 11 is concerned with a weak phase noise ($\sigma_w^2/\sigma_n^2 = 0.25$). Figure 12 is concerned with an average phase noise ($\sigma_w^2/\sigma_n^2 = 1$), and Figure 13 is concerned with a large phase noise ($\sigma_w^2/\sigma_n^2 = 4$). We recall [1] that the DBVA performs some kind of phase estimation along a path that satisfies

$$\hat{\psi}_n = \hat{\psi}_{n-1} \pm \sigma_w \quad , \qquad (32)$$

using a Viterbi algorithm. The DBVA that we have simulated is somewhat different from Ungerboeck's DBVA, in which the number of possible phase states at each iteration is limited to 6 or 8. In our simulation the number of phase states is not limited, thus avoiding one possible cause of errors and improving the error rate, but also increasing the computational complexity with respect to [1].

Behavior of DDPLL and JE on CCITT (4,4) Constellation: The decision-directed algorithms (DDPLL and JE) have essentially the same performance as shown in Figures 11-13. The DDPLL is superior to the JE by only 0.5dB. The slight inferiority of the JE is largely conpensated by the fact that the complex gain of the JE can also correct rapid gain fluctuations in the channel. We emphasize that the curves of the DDPLL and JE are biased and cannot be trusted just as they are, because of the occurrences of very large bursts of errors at relatively high error probabilities. When such bursts have occurred in the simulation runs, they have been withdrawn from the error rate computation. For instance, with $\sigma_w^2 = 0.25\sigma_n^2$ and SNR = 17 dB, at an error probability on the order of 10^{-2}, between one fourth and one third of the simulation runs (with length 500 data values) have exhibited bursts of about a hundred errors. In the simulations, the bursts began to occur at SNR = 18dB, 21.5dB, and 26dB for $\sigma_w^2/\sigma_n^2 = 0.5$, 1 and 4 respectively. This corresponds to a value of σ_w such that $4\sigma_w$ ranges between 11.5° and 20°. The phenomenom of error bursts can be explained as follows: because the phase increment is Gaussian it will occasionally reach the value $4\sigma_w$. If, at the same time the noise is relatively large, the angle between the observed data and the transmitted symbol will exceed the value 22.5° that corresponds to the angular threshold for an error in the 16-point CCITT diagram (see Figs. 2d and 4a). No type of decision-directed phase estimator can correct such an error. Therefore the phase estimate will become incorrect (by a shift of ±45°), causing a burst of errors. Moreover, as expected, the error probability at which bursts of errors occur decreases as the phase fluctuations increase, making the receiver less and less reliable. For instance, at the error probability of about 10^{-3} with $\sigma_w^2 = 0.25\ \sigma_n^2$ no

bursts occurred; on the other hand, at the error probability of about 10^{-4} with $\sigma_w^2 = 4\sigma_n^2$, we have obtained one 500 sample run out of 40 such runs that was a burst. For the other 19500 samples of the other 39 runs, no error was observed.

With respect to burst phenomena, the DDPLL and JE behave similarly.

Behavior of DBVA and VA on CCITT (4,4) Constellation: The performance of the VA is superior to that of the DBVA. The gain achieved by the VA over the simpler DBVA is monotone increasing in the ratio of phase fluctuation variance σ_w^2 to additive noise variance σ_n^2. While there is no gain when $\sigma_w^2/\sigma_n^2 = 0.25$, the gain is 1dB for $\sigma_w^2/\sigma_n^2 = 1$ and 2dB for $\sigma_w^2/\sigma_n^2 = 4$. Both systems perform better than the DDPLL or JE, the improvement again being a monotone increasing function of σ_w^2/σ_n^2.

A very important point is that the use of either of the two MAP phase estimators precludes the occurrence of error bursts. The errors seem to be grouped by two or three and no error multiplicastion occurs since the phase estimator is not decision-directed. Thus such MAP sequence estimators can be used even at high error probabilities on the order of 10^{-2} or 10^{-1}.

Comparison between MAP and Decision-Directed Phase Estimators: The improvement that can be gained by using any type of MAP estimator for phase rather than a simple decision-directed algorithm is again an increasing function of σ_w^2/σ_n^2. Figure 11 shows that 1dB only is gained by the DBVA and the VA over the DDPLL if $\sigma_w^2 = 0.25\ \sigma_n^2$. This gain is realized at a high computational price. For phase fluctuations and additive noise of the same importance ($\sigma_w^2/\sigma_n^2=1$), the VA outperforms the DDPLL by 3 dB (see Fig. 12), but the gain is reduced to 2dB for the simpler DBVA. For large phase fluctuations, the gain is important. For instance

Figure 13 shows that the VA outperforms the DDPLL by 5dB when $\sigma_w^2/\sigma_n^2 = 4$. In addition the VA brings the insurance that no burst of errors can occur, even for very poor SNR and large phase fluctuations.

Sensitivity to imperfect knowledge of σ_w^2/σ_n^2: It is easily seen in (18) or (37) that the only parameter required in order to proceed with the VA algorithm is the ratio of phase variance to additive noisepower. The same holds for the DDPLL whose optimal gain K_1 depends on this ratio (see (49)), and for the JE whose step-size μ (see (14)) is to be kept close to K_1, but smaller, provided the data diagram has unit power. As for the DBVA it requires only the knowledge of σ_w^2 in order to determine the number m of discretized phase levels. Thus an important feature of each system is its sensitivity to an imperfect knowledge of σ_w^2/σ_n^2 (or σ_w^2) because firstly σ_w^2 can vary with time and secondly the actual phase can fluctuate according to a statistical model that is different from the one expected. The less sensitive the system is to the knowledge of σ_w^2/σ_n^2 (or σ_w^2), the more robust it is.

a) Sensitivity of the decision-directed systems. Let us denote σ_w^2/σ_n^2 by α. The function $K_1(\alpha)$ that gives the optimum loop-gain of the DDPLL is sketched in Figure 14. It is quite flat except for α very close to zero (e.g. $\alpha < 0.2$).

Now the case $\alpha << 1$ is of no real interest for the purpose of this paper. Indeed it has been seen previously that, in this case, no MAP phase estimator is worth being worked out. Moreoever any type of (reasonable) phase estimator will perform satisfactorily. When α is not negligible, $K_1(\alpha)$ is slowly varying. For example $K_1(1)/K_1(0.25) = 1.59$, and $K_1(4)/K_1(1) = 1.34$. Thus the value $K_1(1) = 0.62$ for the

DDPLL gain is correct for a large range of values of α. This fact is largely confirmed by the simulations. However, due to the risk of error multiplication that increases very rapidly with K_1, it should rather be set to the lower bound $K_1(\alpha_{min})$ corresponding to the smallest α that can be expected, rather than to an average value $K_1(\alpha_{ave})$, which will sometimes be too large and bring error bursts. Thanks to this precaution, the DDPLL is insensitive to α. It is a robust system.

The robustness of the JE is also excellent. This fact was checked on numerous computer simulations: as a function of the step-size μ the error probability P(E;u) exhibits a minimum which is very flat, as sketched in Figure 15. The range where the minimum is reached does not depend critically upon α. A value such as μ=0.4 corresponds to the minimum of error probability for α in the range [0.25-1], and for a unit energy data diagram.

b) Sensitivity of the MAP phase estimators. The VA sensitivity to imperfect knowledge of α has been tested in our computer simulations. It appears that the VA performance is not appreciably degraded by an error of ±6dB for α. Hence the VA robustness is at least as good as that of the decision-directed algorithms.

On the other hand the DBVA robustness has turned out to be poor. For instance, with SNR=21dB and α=4, the DBVA is supposed to work with $m = \frac{2\pi}{\sigma_w} = 50$ phase levels. If only 45 levels are used, corresponding to a 0.9dB error for α, then the error probability is increased by a factor of 2. In fact, as a function of m, P(E;m) exhibits a minimum, but it is a sharp minimum. This poor robustness can be understood by noting that in the DBVA, the path metric is not a function of $\alpha = \sigma_w^2/\sigma_n^2$, but only of σ_w^2. This may be one of the main drawbacks of the DBVA. This

observation suggests that one should implement the DBVA with a conservatively large number of phase levels to provide robustness. This increases complexity.

VIII. Simulation Results: Bounded-Increments Phase Jitter

For all simulation results of this section the phase space $[-\pi,\pi)$ has been discretized to 32 equally-spaced phase values and a VA has been programmed to solve (17). The assumed increment density h(w) is the uniform density

$$h(w) = \begin{cases} \frac{1}{2a} , & -a \leq w < a \ ; \ 2a = 2\pi/16 \\ 0 , & \text{otherwise} \end{cases} \tag{33}$$

The corresponding discrete transition density for use in the path metric is

$$f(\phi_k/\phi_{k-1}) = \begin{cases} 1/3 , & \phi_k - \phi_{k-1} = -\pi/16 \ , \ 0 \ , \ \pi/16 \\ 0 , & \text{otherwise} \end{cases} \tag{34}$$

The resulting VA is related to the class of so-called M-algorithms in which all survivors are saved, but only M (in this case 3) candidate originator states are allowed. This significantly reduces calculations and results in an algorithm similar in spirit to the DBVA of [1]. Still, however, phase is tracked only on $[-\pi,\pi)$ rather than on $(-\infty,\infty)$.

Source symbols have been generated independently from a 4-PSK alphabet and used to differentially-encode phase according to a Gray code. The random phase sequence has been generated in ways to be discussed below.

Markov Phase with Non-Gaussian Increments: Here the phase is generated according to (6) with h(w) given by (33). Thus the algorithm is matched to the actual phase sequence. Shown in Fig. 16 are performance results for the VA and for the JE. The VA outperforms the JE by 1.5dB over the

range 10dB $\leq$ SNR < 15dB. The probability of error is "probability of bit error."

Sinusoidal Phase Jitter: Here the phase jitter is sinusoidal (see (4)) with uniformly-distributed initial phase and frequency ν. The frequency is chosen such that $\nu\Delta = 1/24$, corresponding to a transmission rate of 4800 b/s with baud rate $1/\Delta$ = 2400Hz and jitter frequency ν = 100Hz. The runs are 2000 to 10,000 steps long, corresponding to 4000 to 20,000 transmitted bits. The peak-to-peak phase deviation is 20° or 60°. For these experiments the VA outperforms the JE by 1.5-1.7dB. This gain is, of course, achieved at a high price in complexity.

Comparison of the JE and VA: In the simulations reported above, the ratio $\alpha = \sigma_w^2/\sigma_n^2$ ranges from 0.02 to 0.81, that is from small to average values. No burst of errors has ever been observed for the JE. This is due to the fact that the phase increment is always bounded as appears in (4) and also (33). The bound is much smaller than the angular distance between adjacent data. Thus there is no risk of a ±90° slip (corresponding to the 4-PSK diagram) in the JE phase estimation. Hence the errors will be scattered rather than grouped, and no error multiplication phenomenon can happen.

Owing to this consideration, to the fact that the VA outperforms the JE by only 1.5dB, and to the complexity of the VA, a practical system will implement the JE (or DDPLL) rather than the VA (or DBVA), in the case of bounded increment phase jitter and a simple symbol constellation such as the 4-PSK.

IX. Conclusions

We have derived a principle of optimality for phase-amplitude encoded symboling that allows one to simultaneously track random phase and decode data symbols using the VA derived in [8] and [9]. The VA is designed for a random walk phase process, a very severe type of phase process. In such a process there exists the possibility of large phase jumps. The VA gives excellent performances.

In order to reach conclusions about the type of phase estimation that should be used for given type of phase fluctuations, performance comparison of the VA with two simple decision-directed phase estimators, namely the JE of [3] and the DDPLL of [5], and with the DBVA, have been thoroughly investigated on computer simulations, with various data diagrams. They indicate that the choice among the four systems is to be made according to four parameters:

(i) the error probability P(E) at which the system is to be used;

(ii) The relative importance $\alpha = \sigma_w^2/\sigma_n^2$ of phase fluctuations with respect to additive noise;

(iii) The complexity C that is technologically feasible and acceptable;

(iv) the maximum phase increment $\Delta\Phi_{max}$ that is to be expected, as compared to the angular distance between points of the data diagram.

Suggestions for this choice are sketched in the Tables 1 and 2 where Table 2 is concerned with cases 2 and 3 of Table 1.

$\alpha/P(E)$	small	large
small	case 1 JE or DDPLL	case 2 see Table 2
large	case 3 see Table 2	case 4 VA or DBVA

Table 1

$\Delta\Phi_{max}/C$	small	large
small	JE or DDPLL	JE or DDPLL
large	DBVA	VA or DBVA

Table 2

The choice between the two decision-directed phase estimators, JE or DDPLL, is irrelevant for the matters discussed in this paper. It appears in Tables 1 and 2 that the VA and DBVA are preferred when α, $P(E)$, and $\Delta\Phi_{max}$ are large. The comparison between these two MAP phase estimators shows that the VA is more robust, has a smaller learning period and outperforms the DBVA by 2dB or more when α is at least equal to 4.

Appendix: Monotonicity of Folded Normal and Cauchy Densities

There are many choices for the phase increment density h(x) that are physically interesting and mathematically tractable. Two of particular interest are the normal density and the Cauchy, the latter being useful in the modelling of "heavy-tailed" behavior. When folded around the unit circle according to (8) these densities yield transition densities which achieve their maximum at $\phi_k - \phi_{k-1} = 0$ and decrease monotonically on the interval $0 \leq \phi_k - \phi_{k-1} \leq \pi$.

Consider first the Cauchy case

$$g_1(x) = \sum_{k=-\infty}^{\infty} \frac{a/\pi}{a^2+(x+k2\pi)^2} \quad . \qquad \text{(A-1)}$$

According to Poisson's summation formula [13], this may be written

$$g_1(x) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} e^{-a|k|} e^{jkx}$$

$$= \frac{1}{2\pi} (1-e^{-2a})(1-2e^{-a}\cos x + e^{-2a})^{-1} \quad . \qquad \text{(A-2)}$$

This function achieves its maximum value at zero and decreases monotonically.

In the normal case

$$g_1(x) = \sum_{k=-\infty}^{\infty} (2\pi\sigma^2)^{-\frac{1}{2}} \exp\{-(x+2k\pi)^2/2\sigma^2\} \quad . \qquad \text{(A-3)}$$

Again, by Poisson's summation formula

$$g_1(x) = \sum_{k=-\infty}^{\infty} (2\pi)^{-1} \exp\{jkx-k^2\sigma^2/2\} \quad . \qquad \text{(A-4)}$$

This infinite sum goes by the name $J_3(x,q=e^{-\sigma^2/2})$ in the theory of Jacobian elliptic functions and theta functions [15]. The theta function $J_3(x,q)$ is known to be monotone decreasing on the interval $0 \leq x \leq \pi$.

Acknowledgments

The authors wish to thank Serge Kerbrat and Claude Pariente for their assistance with software development and simulations.

References

1. G. Ungerboeck, "New Application for the Viterbi Algorithm: Carrier Phase Tracking in Synchronous Data Transmission Systems," Nat. Telecomm. Conf., pp. 734-738 (1974).

2. H. Kobayaski, "Simultaneous Adaptive Estimation and Decision Algorithm for Carrier Modulated Data Transmission Systems," IEEE Trans. Comm. Tech., COM-19, pp. 268-280 (June 1971).

3. M. Levy and O. Macchi, "Auto-Adaptive Phase Jitter and Interference Intersymbol Suppression for Data Transmission Receivers," Nat. Telecomm. Conf., Dallas (November 1976).

4. M. Levy and O. Macchi, "Egaliseur de Gigue," 6ème Colloque GRETSI sur le Traitement du Signal et ses Applications," Nice (Avril 1977).

5. D. D. Falconer, "Analysis of a Gradient Algorithm for Simultaneous Passband Equalization and Carrier Phase Recovery," BSTJ, 55, pp. 409-428 (1976).

6. F. R. Magee, "Simultaneous Phase Tracking and Detection in Data Transmission over Noisy Dispersive Channels," IEEE Trans. on Comm., pp. 712-715 (July 1977).

7. A. S. Willsky, "Fourier series and Estimation on the Circle with Applications to Synchronous Communications - Part I: Analysis," IEEE Trans. Inform. Theory, IT-20, pp. 577-583 (September 1974).

8. L. L. Scharf, D. D. Cox, C. J. Masreliez, "Modulo-2π Phase Sequence Estimation," IEEE Trans. Inform. Theory (to appear).

9. L. L. Scharf, "A Viterbi Algorithm for Modulo-2π Phase Tracking in Coherent Data Communication Systems," ONR Technical Report #25, (February 1978).

10. G. D. Forney, "Maximum Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," IEEE Trans. Inform. Theory, IT-18, pp. 363-378 (May 1972).

11. G. J. Foschini, R. D. Gitlin and S. B. Weinstein, "On the Selection of a Two-Dimensional Signal Constellation in the Presence of Phase Jitter and Gaussian Noise," BSTJ, 52, pp. 927-965 (July-August 1973).

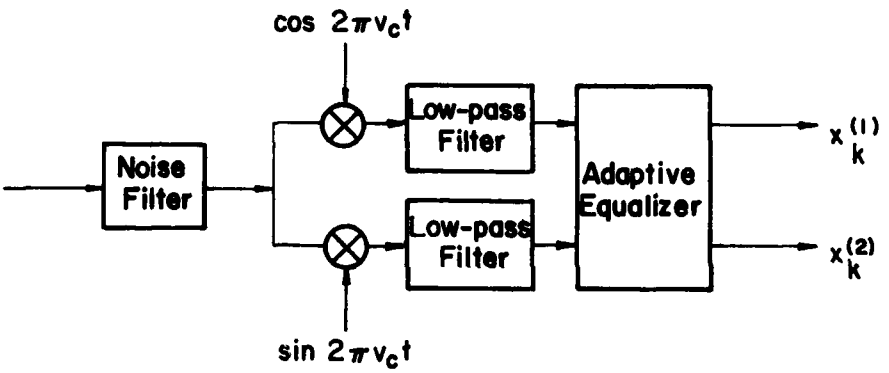
12. P. Hedelin, "Optimal Smoothing in Discrete Continuous Linear and Nonlinear Systems," Inform. Sci., 13, pp. 137-158 (1977).

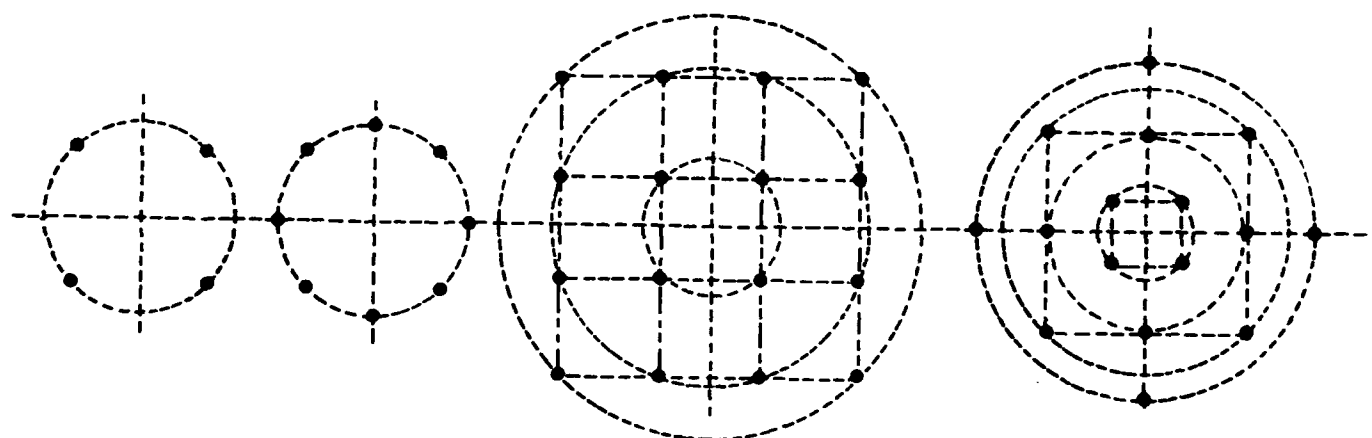
13. S. Bochner, "Harmonic Analysis and The Theory of Probability," Univ. of California Press, Berkeley, pp. 28-36 (1960).

14. "Data Communications Using the Switched Telecommunications Network" Bell System Technical Reference, pp. 15-16 (August 1970).

15. M. Abramowitz, I. Stegun, "Handbook of Mathematical Functions," National Bureau of Standards, U. S. Government Printing Office, pp. 576-578.

List of Figures


cos 2πv_c t
Noise Filter
Low-pass Filter
Low-pass Filter
Adaptive Equalizer
x_k^(1)
x_k^(2)
sin 2πv_c t

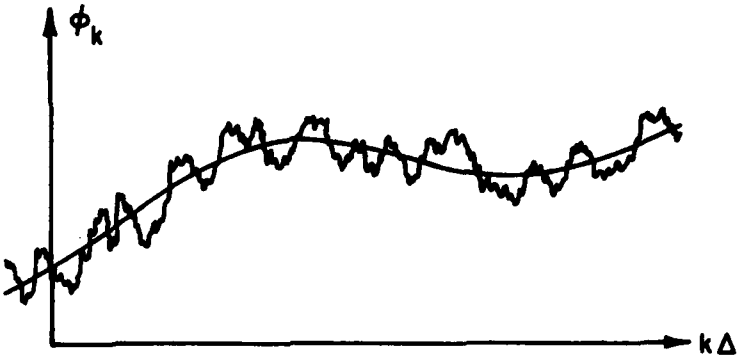


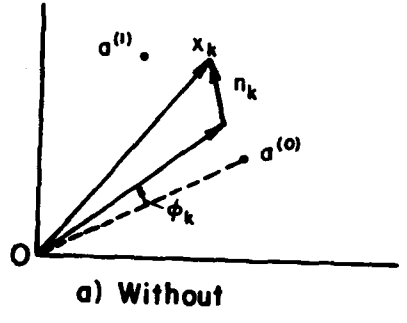
4-ARY-PSK

8-ARY-PSK

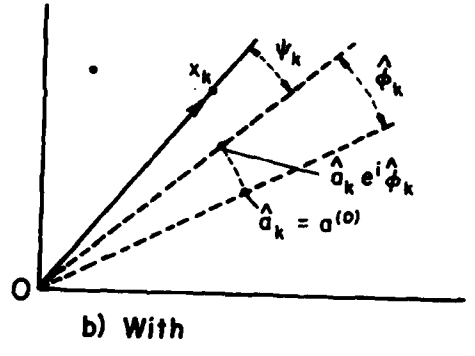
QASK

Four Amplitudes, Four Phases Modulation (4,4) (CCITT)

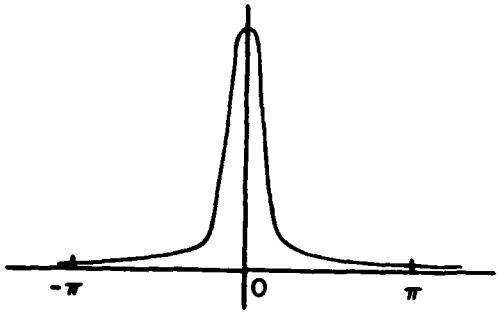

ϕ_k
$k\Delta$



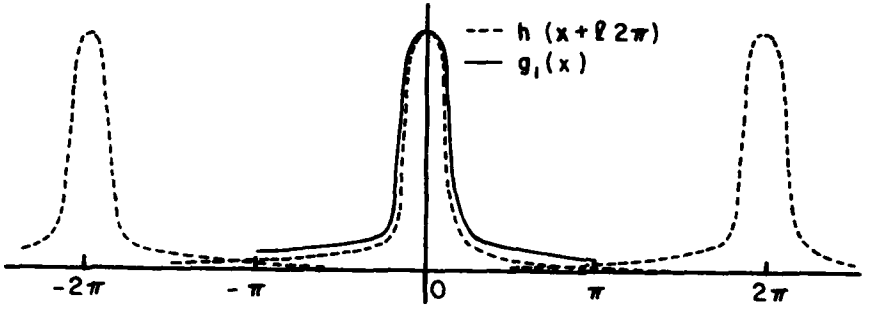
a) Without



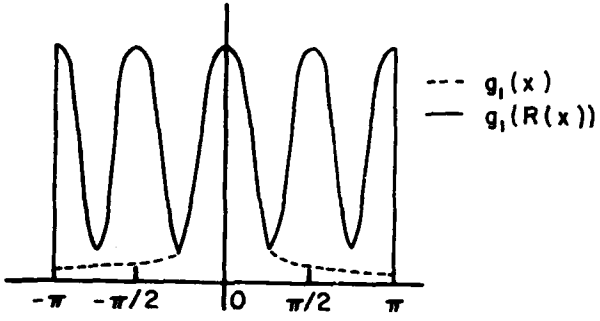
b) With



a) h(x) on $(-\infty, \infty)$

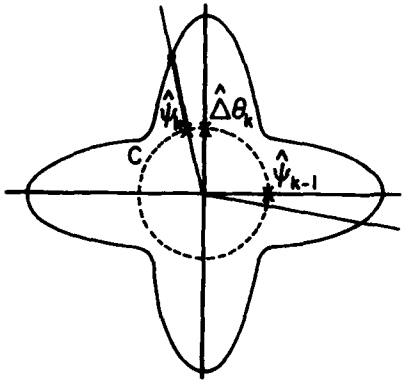


b) $g_1(x)$ on $[-\pi, \pi]$

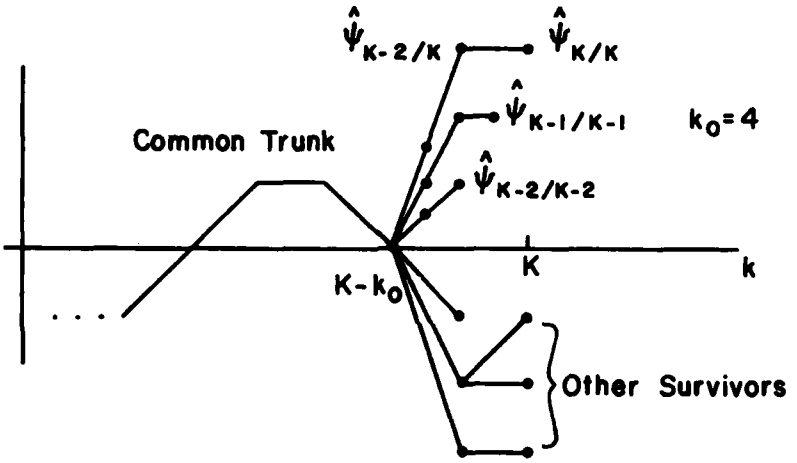


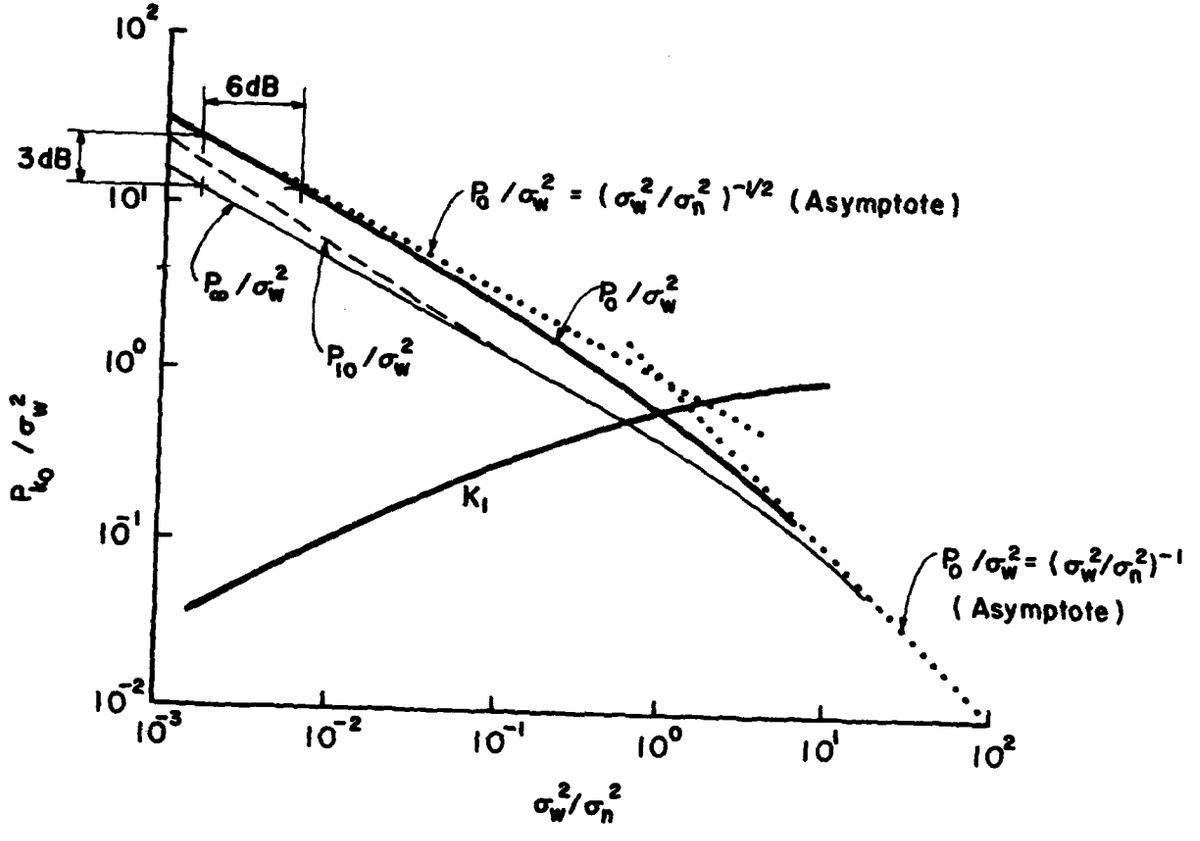
c) $g_1(R(x))$ on $[-\pi, \pi]$

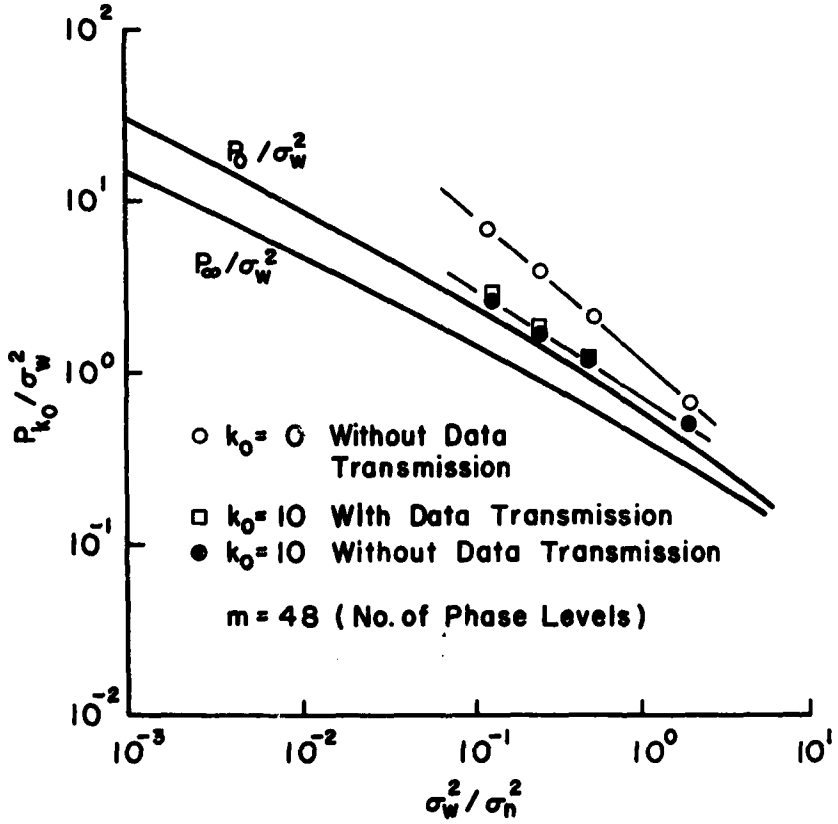
M=4

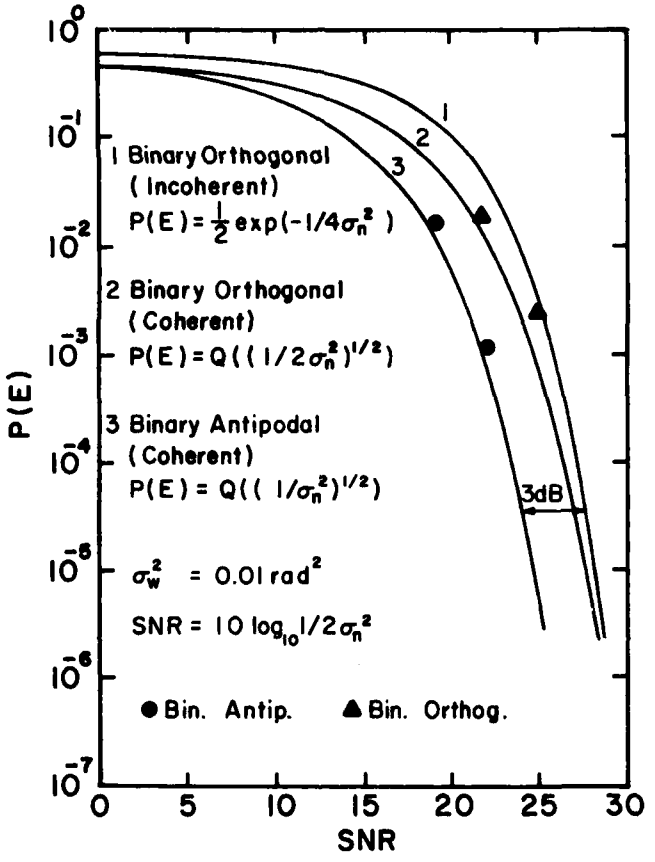


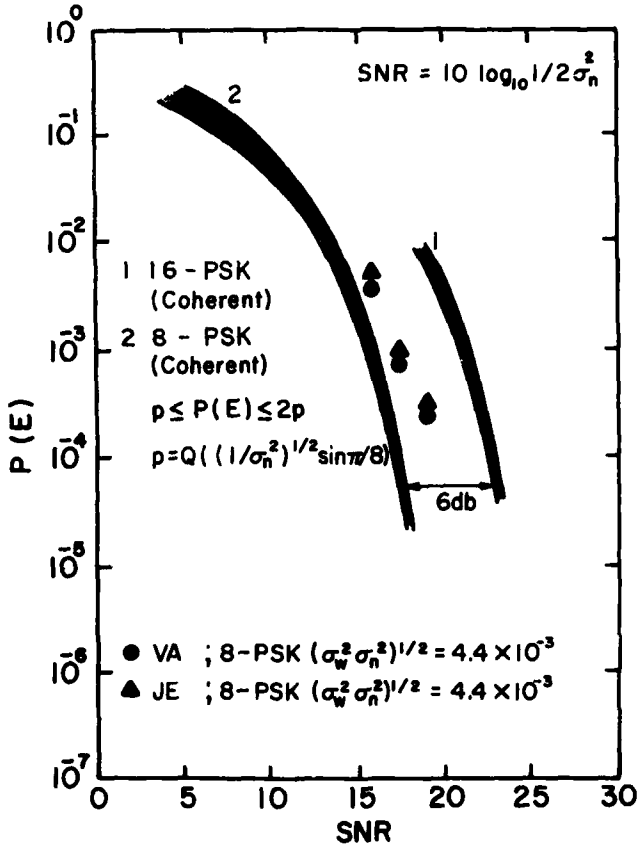
d) $g_1(R(x))$ Wrapped on the Circle C. M=4

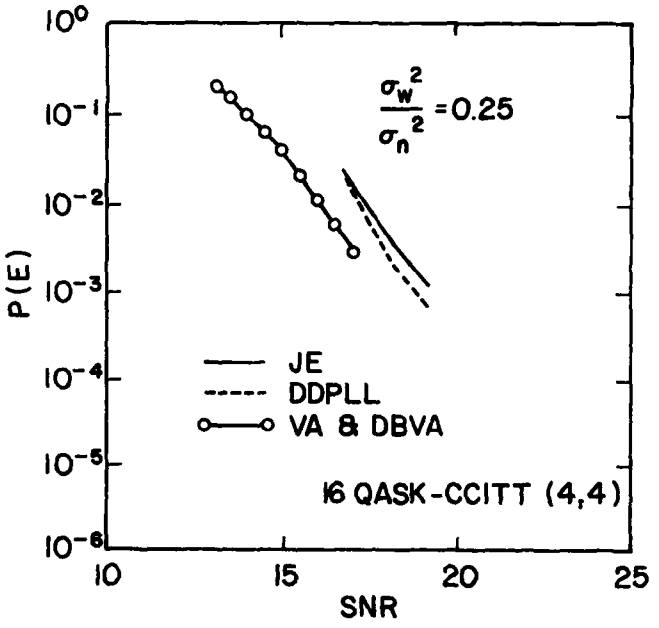

$\hat{\psi}_{K-2/K}$
$\hat{\psi}_{K/K}$
$\hat{\psi}_{K-1/K-1}$
$k_0=4$
Common Trunk
$\hat{\psi}_{K-2/K-2}$
K
k
$K-k_0$
Other Survivors

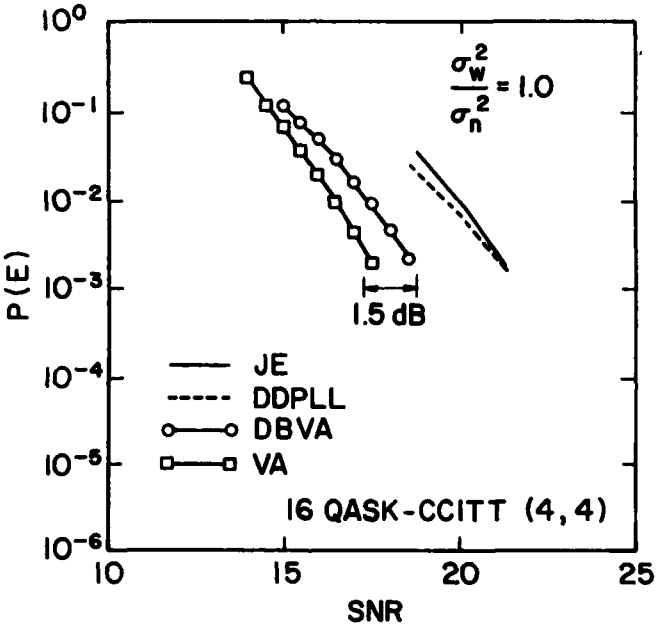

10^2
6dB
3dB
10^1
10^0
10^{-1}
10^{-2}
P_{k_0}/σ_w^2
$P_0/\sigma_w^2 = (\sigma_w^2/\sigma_n^2)^{-1/2}$ (Asymptote)
P_∞/σ_w^2
P_0/σ_w^2
P_{10}/σ_w^2
K_1
$P_0/\sigma_w^2 = (\sigma_w^2/\sigma_n^2)^{-1}$
(Asymptote)
10^{-3}
10^{-2}
10^{-1}
10^0
10^1
10^2
σ_w^2/σ_n^2

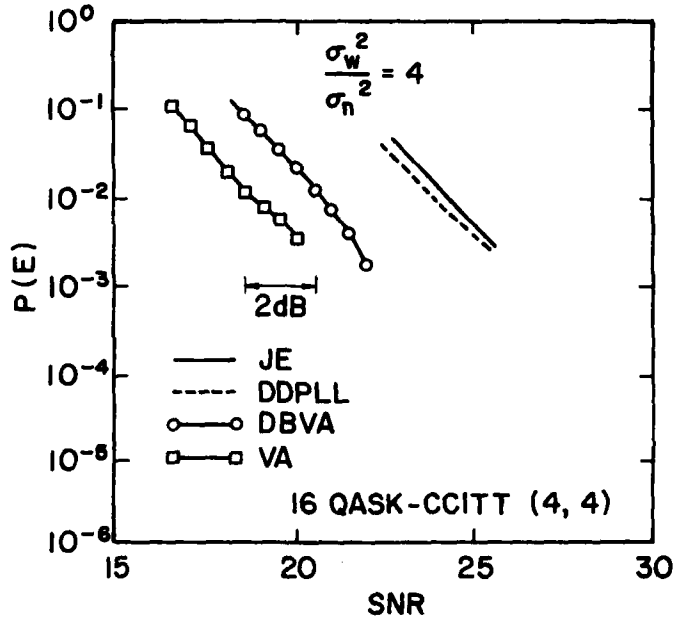

10^2
10^1
10^0
10^{-1}
10^{-2}
P_{k_0}/σ_w^2
P_0/σ_w^2
P_∞/σ_w^2
○ k_0= 0 Without Data Transmission
□ k_0= 10 With Data Transmission
● k_0= 10 Without Data Transmission
m = 48 (No. of Phase Levels)
10^{-3}
10^{-2}
10^{-1}
10^0
10^1
σ_w^2/σ_n^2

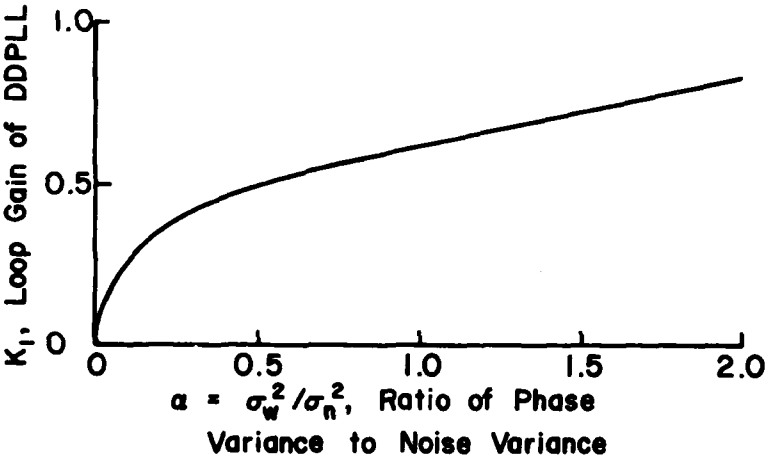

P(E)
SNR
1 Binary Orthogonal
(Incoherent)
P(E) = ½ exp(-1/4σn²)
2 Binary Orthogonal
(Coherent)
P(E) = Q((1/2σn²)^1/2)
3 Binary Antipodal
(Coherent)
P(E) = Q((1/σn²)^1/2)
σw² = 0.01 rad²
SNR = 10 log10 1/2σn²
● Bin. Antip.
▲ Bin. Orthog.
3dB
1
2
3

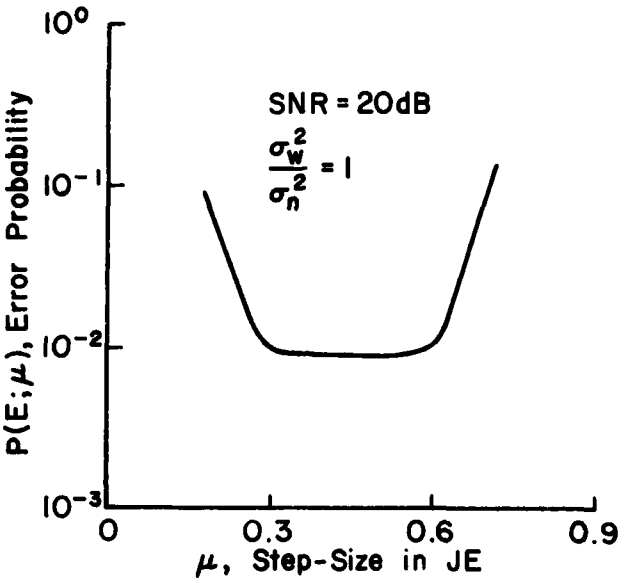

SNR = 10 log10 1/2σn²
2
1
1 16 - PSK
(Coherent)
2 8 - PSK
(Coherent)
p ≤ P(E) ≤ 2p
p=Q((1/σn²)^1/2 sinπ/8)
6db
● VA ; 8-PSK (σw²σn²)^1/2 = 4.4 X10^-3
▲ JE ; 8-PSK (σw²σn²)^1/2 = 4.4 X10^-3
P(E)
SNR
10^0
10^-1
10^-2
10^-3
10^-4
10^-5
10^-6
10^-7
0
5
10
15
20
25
30

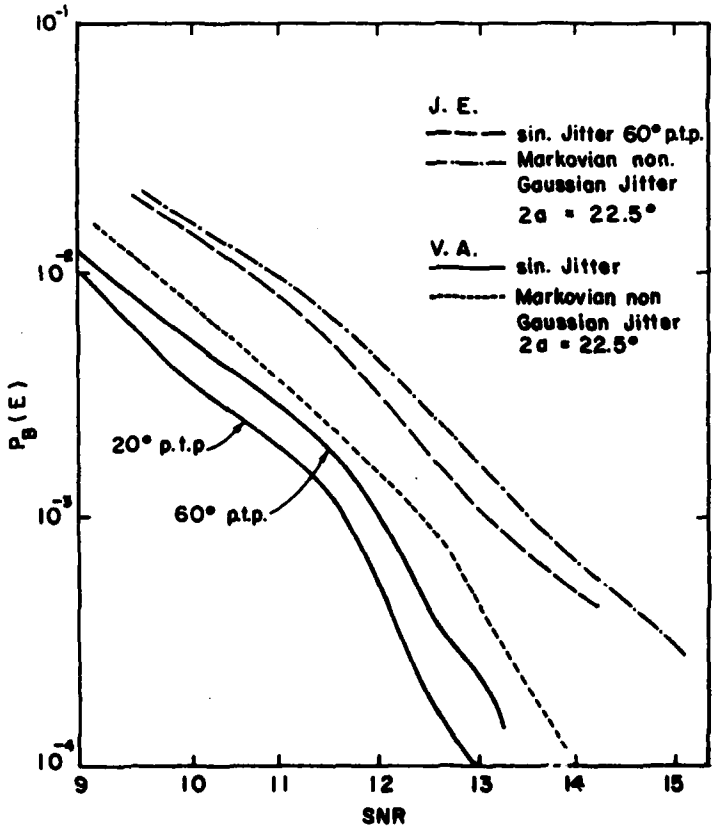

10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
P(E)
$\frac{\sigma_w^2}{\sigma_n^2}$ =0.25
JE
DDPLL
VA & DBVA
16 QASK-CCITT (4,4)
10
15
20
25
SNR


10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
P(E)
10
15
20
25
SNR
$\frac{\sigma_w^2}{\sigma_n^2} = 1.0$
1.5 dB
JE
DDPLL
DBVA
VA
16 QASK-CCITT (4, 4)


10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
P(E)
$\frac{\sigma_w^2}{\sigma_n^2} = 4$
2dB
JE
DDPLL
DBVA
VA
16 QASK-CCITT (4, 4)
15
20
25
30
SNR


1.0
0.5
0
K_1, Loop Gain of DDPLL
0
0.5
1.0
1.5
2.0
$\alpha = \sigma_w^2/\sigma_n^2$, Ratio of Phase Variance to Noise Variance


10^0
10^{-1}
10^{-2}
10^{-3}
SNR = 20dB
$\frac{\sigma_w^2}{\sigma_{n_2}^2} = 1$
$P(E;\mu)$, Error Probability
0
0.3
0.6
0.9
μ, Step-Size in JE


J. E.
sin. Jitter 60° p.t.p.
Markovian non. Gaussian Jitter
2α = 22.5°
V. A.
sin. Jitter
Markovian non Gaussian Jitter
2α = 22.5°
20° p.t.p.
60° p.t.p.
$P_B(E)$
10^{-1}
10^{-2}
10^{-3}
10^{-4}
9
10
11
12
13
14
15
SNR

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: Technical Report #DEC79-3	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER (9)
4. TITLE (and Subtitle): (6) A Dynamic Programming Algorithm for Phase Estimation and Data Decoding on Random Phase Channels	5. TYPE OF REPORT & PERIOD COVERED: Technical Report (12)
7. AUTHOR(s): (10) Odile Macchi, Louis L. Scharf	8. CONTRACT OR GRANT NUMBER(s): (15) N00014-75-C-0518
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Electrical Engineering, Colorado State University, Fort Collins, CO 80523	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS (11)	12. REPORT DATE: Dec 1979 / 13. NUMBER OF PAGES: 58
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Office of Naval Research, Statistics and Probability Branch, Arlington, VA (12) 62	15. SECURITY CLASS. (of this report): Unclassified / 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report): Approved for public release; distribution unlimited	
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report): Unlimited distribution (14) CSU-TR-DEC79-3	
18. SUPPLEMENTARY NOTES	
19. KEY WORDS (Continue on reverse side if necessary and identify by block number): Phase estimation, data decoding, dynamic programming, data transmission	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The problem of simultaneously estimating phase and decoding data symbols from baseband data is posed. The phase sequence is assumed to be a random sequence on the circle and the symbols are assumed to be equally-likely symbols transmitted over a perfectly equalized channel. A dynamic programming algorithm (Viterbi algorithm) is derived for decoding a maximum a posteriori (MAP) phase-symbol sequence on a finite dimensional phase-symbol trellis. A new and interesting principle of optimality for simultaneously

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

406434

Abstract (con't.)

estimating phase and decoding phase-amplitude coded symbols leads to an efficient two step decoding procedure for decoding phase-symbol sequences. Simulation results for binary, 8-ARY PM, and 16-QASK symbol sets transmitted over random walk and sinusoidal jitter channels are presented, and compared with results one may obtain with a decision-directed algorithm, or with the binary Viterbi algorithm introduced by Ungerboeck. When phase fluctuati are severe, and the symbol set is rich (as in 16-QASK), MAP phase-symbol sequence decoding on circles is superior to Ungerboeck's technique, which in turn is superior to decision-directed techniques.