An Estimator–Predictor Approach to PLL Loop Filter Design

J. I. Statman and W. J. Hurd
Communications Systems Research Section

A new approach to the design of digital phase locked loops (DPLLs), using estimation theory concepts in the selection of a loop filter, is presented. The key concept is that the DPLL closed-loop transfer function is decomposed into an estimator and a predictor. The estimator provides recursive estimates of phase, frequency, and higher order derivatives, while the predictor compensates for the transport lag inherent in the loop. This decomposition results in a straightforward loop filter design procedure, enabling use of techniques from optimal and sub-optimal estimation theory. A design example for a particular choice of estimator is presented, followed by analysis of the associated bandwidth, gain margin, and steady state errors caused by unmodeled dynamics. This approach is under consideration for the design of the DSN Advanced Receiver Carrier DPLL.

I. Introduction

A phase locked loop (PLL) is a feedback system that tracks the phase of a received signal. A typical PLL structure is shown in Fig. 1: the difference between the received phase and a voltage controlled oscillator (VCO) phase is filtered and used to steer the VCO, so that the VCO phase tracks the received phase. Designers of PLLs select a loop filter to meet performance requirements, usually specified in terms of bandwidth, gain margin, and dynamic errors. Design methods for analog PLLs are well documented in the literature (Refs. 1 and 2).

It is assumed here that the DPLL is characterized by constant loop update rate. Discussion of DPLLs with variable loop update rate is provided in Refs. 3-6. DPLLs with fixed update rate can be designed either by analogy to continuous domain PLLs or based on an optimality criterion.

Design by analogy to continuous time PLLs, i.e., applying s-plane design rules by translation to the z-plane, suffers from two major disadvantages. The design is based on iteratively placing open-loop poles and zeroes at "well chosen" locations until satisfactory performance is achieved. Such locations are selected based on design experience rather than on a well established set of rules. Also, the design does not account for the transport delay in the digital loop, caused by hardware and loop filter computations. This delay is either "tolerated," i.e., the degradation due to the delay is analyzed and found to be acceptable (Ref. 7), or "compensated" by the equivalent of a lead-lag network (Ref. 8).

Design based on an optimality criterion attempts to optimize a concise performance measure, usually a weighted sum of transient response, gain margin, and noise bandwidth. For some cases (Ref. 9), analytical expressions for the performance measure can be obtained, and closed-form expressions for loop filters defined.

The estimator-predictor approach described in this article offers an alternate method for DPLL design. The DPLL is shown to be equivalent to an estimator followed by a predictor, compensating for the transport lag. The estimator derives an estimate of the state vector, consisting of phase, frequency, and perhaps frequency rate, based on measurements of phase alone. Selection of the specific estimator can then be based on the vast resources available in linear optimal estimation theory. In the next sections, a generic form of the design procedure is presented, followed by a design example using specific estimator, predictor, and for specified transport lags. Then, performance of the designed loops is analyzed. A future study will compare performance of loops designed using this approach to those designed according to alternative criteria.

II. Design Approach

This section discusses the rationale for the new design approach and then presents the resulting design procedure. Time dependent variables, e.g., phase, are represented by their z-plane transforms, and difference equations are replaced by corresponding z-plane transfer functions. Even though the notation does not explicitly account for time variation of difference equation coefficients, such variations can be accommodated by the design procedure.

A. Rationale

The basic purpose of a DPLL is to generate a signal, with phase $\overline{\theta}(z)$, that approximates $\theta(z)$, the phase of a received signal, as shown in Fig. 1. From a linear estimation theory viewpoint, this is a pure and simple estimation problem, and the solution is straightforward. First, a linear state model for $\theta(z)$ is defined, with a corresponding measurement model. Then, statistical models for state (or process) noise and measurement noise are developed. Finally, an appropriate estimator is selected (a fixed-coefficient, recursive least squares estimator is a prime candidate).

There is a slight complication since a non-zero transport delay is associated with DPLL mechanization. The effect of this delay can be compensated by a predictor that extrapolates the state estimate by an appropriate time interval. Thus, the closed-loop DPLL transfer function has the form:

$$H(z) = D(z)\,C(z)\,z^{-N} \tag{1}$$

where

$H(z)$ = closed-loop DPLL transfer function

$D(z)$ = predictor transfer function

$C(z)$ = estimator transfer function

N = DPLL transport lag, in units of loop update time

Equation (1) assumes that the DPLL is updated at uniformly spaced time points, with update interval of T, hence the transport lag is an integer multiple of T. Figure 2 shows a DPLL structure that addresses Eq. (1). The three functional blocks on the right are the elements of $H(z)$: estimator, predictor, and transport lag. In addition, the figure incorporates a phase detector, with gain A, and another summing junction. Note that, when phase detector gain has a nominal value of $A = 1$, the feedback is cancelled, and closed-loop transfer function is $H(z)$, as in Eq. (1). The closed-loop transfer function, for general phase detector gain, is denoted $P(z)$.

The transformation of Fig. 2 into an implementation-oriented structure is presented in Figs. 3 and 4. In Fig. 3, the estimator-predictor-delay combination is replaced by actual components: a digital filter, $S(z)$, and a numerically controlled oscillator, $Q(z)$, where $S(z)$ satisfies:

$$S(z) = \frac{D(z)\,C(z)}{Q(z)}\,z^{-N} \tag{2}$$

Finally, Fig. 4 illustrates a proposed DPLL implementation. Phase detection and "hardware numerically controlled oscillator (NCO)" functions are performed by custom circuitry, while summing junction, $S(z)$, and "software NCO" functions are executed in a computer. Both "hardware NCO" and "software NCO" have identical mathematical representations, as the latter simulates the operation of the former. When comparing Figs. 1 and 4, it is apparent that the function of the loop filter is performed, in Fig. 4, by the inner loop, implemented in the computer-resident portion of the DPLL.

B. Design Procedure

The design can be divided into five distinct steps. The first step is to select a model for the received phase process. This model can include the process dynamics and statistics on process noises and measurement noise. The second step is to model the hardware: NCO, transport lag, and phase detector. As shown later, the transport lag usually can be incorporated as part of the transfer function of the NCO, $Q(z)$.

The next step is selection of an estimator, based on phase model and process noise and measurement noise covariance matrices. Possible realizations of the estimator are some varia-

tions of Kalman filters, or other least squares estimators. The fourth step is determination of a predictor to compensate for transport lag. The last, but definitely not the least, step is to assure that the loop is stable.

III. Design Example

In this section, the design approach presented above is implemented for a particular case.

A. Model of Received Phase

Two separate models are used for the received phase. The first model has two state variables: phase and frequency, while the second model uses three state variables: phase, frequency and frequency rate. Later, these models will be addressed as second- and third-order models, respectively.

B. Model of Phase Detector and NCO

The phase detector inputs are actually time continuous signals. The phase detector measures the average phase difference between the phase of the input process and the NCO phase. This is equivalent to the difference between the phase of the input process and the NCO phase. Thus, $\theta(z)$ represents the input phase, averaged over the measurement interval, and $\overline{\theta}(z)$ represents the NCO phase, averaged over the same interval.

Hurd and Aguirre (Ref. 7) and Simon and Mileant (Ref. 8) discuss a general form for a DPLL NCO. The NCO is updated gT seconds after the phase measurement is made, where T is the loop update interval and $0 \leqslant g \leqslant 1$. The case $g=0$ corresponds to NCO update immediately following the measurement, while $g=1$ is caused by a T-second delay before the NCO update. Such a delay occurs in practical systems while the computer determines the next NCO input. $Q(z)$ is the transfer function between the NCO input and the average NCO phase at the next measurement, including all delays. This transfer function is:

$$Q(z) = T\,\frac{(1-g)^2 z^2 + (1+2g-2g^2)z + g^2}{2z^2(z-1)} \tag{3}$$

In particular, for the cases $g=0$ and $g=1$, $Q(z)$ has the simple forms:

$$g=0: \qquad Q(z) = \frac{T}{2}\,\frac{z+1}{z-1}\;z^{-1} \tag{4}$$

$$g=1: \qquad Q(z) = \frac{T}{2}\,\frac{z+1}{z-1}\;z^{-2} \tag{5}$$

We observe that in Eqs. (4) and (5), the NCO function can be separated into two parts: integer number of delays, and a kernal, $Q_1(z)$. $Q_1(z)$ is characterized by a numerator and a denominator of the same degree. The cases of $g=0$ and $g=1$ are then examples of $N=1$ and $N=2$ computational delays, respectively. The general expressions for a delay of N update times is:

$$Q(z) = \frac{T}{2}\,\frac{z+1}{z-1}\;z^{-N} = Q_1(z)\,z^{-N} \tag{6}$$

In particular, cases with $N = 3,4\ldots$ correspond to pipelined feedback computations.

C. Selection of an Estimator

For this example, we select a least squares, fading memory estimator, with fixed gains (Ref. 10). The estimator computes a state estimate based on a set of measurements, applying an exponentially decaying weight to past data. This "aging function" effectively discards measurements that are older than three or four estimator time constants. The estimator transfer functions for fixed or steady state gain are presented in Table 1, and depend on a single parameter, α, a decay factor:

$$\alpha = e^{-\frac{T}{\tau}} \tag{7}$$

where τ is the estimator time constant.

D. Selection of a Predictor

The predictor generates a phase predict by extrapolating the current phase estimate using either an estimate of frequency, for a second-order model, or estimates of frequency and frequency rate, for a third-order model. The extrapolation time, NT, is the number of integer delays in the loop multiplied by the update interval. Predictor equations are given in Table 2.

E. Loop Transfer Functions

Recall from Eq. (1) that the transfer function $H(z)$ equals $D(z)\,C(z)\,z^{-N}$. Thus, in this example, $H(z)$ depends only on the order of the phase model, the number of delays, and the estimator decay factor, α. The open loop, $G(z)$, closed loop, $P(z)$, and error, $E(z)$, transfer functions for the DPLL are easily expressed in terms of the function $E_1(z) = 1 - H(z)$:

$$G(z) = \frac{\overline{\theta}(z)}{\phi(z)} = \frac{AH(z)}{1-H(z)} = \frac{A(1-E_1(z))}{E_1(z)} \tag{8}$$

$$P(z) = \frac{\bar{\theta}(z)}{\theta(z)} = \frac{G(z)}{1+G(z)} = \frac{AH(z)}{1+(A-1)H(z)}$$

$$= \frac{A(1-E_1(z))}{1+(A-1)(1-E_1(z))} \quad (9)$$

$$E(z) = \frac{\phi(z)}{\theta(z)} = 1 - P(z) = \frac{E_1(z)}{1+(A-1)(1-E_1(z))} \quad (10)$$

where $\phi(z)$ is the z-transform of the phase out of the phase detector, not including phase detector gain, i.e., $\phi(z) = \theta(z) - \bar{\theta}(z)$. Expressions for $E_1(z)$ are presented in Table 3. A special case of interest occurs when the phase detector gain, A, has its nominal value, $A = 1$. Then:

$$G(z) = \frac{H(z)}{1-H(z)} \; ; \quad P(z) = H(z) \; ; \quad E(z) = E_1(z) \quad (11)$$

F. Realizability and Stability

The DPLL structure of Fig. 4 has three blocks that need to be realized: a "hardware NCO," a "software NCO" and a digital filter $S(z)$. The "hardware NCO" is realizable by definition. The "software NCO" is a mere simulation of the "hardware NCO" function and is thus realizable. This leaves the digital filter $S(z)$. A necessary and sufficient condition for the realizability of this filter is that the order of the numerator is less or equal to that of the denominator (Ref. 11). Recall, from Eqs. (2) and (6) that $S(z) = D(z)\,C(z)/Q_1(z)$. Since the above condition is satisfied for each component of $C(z)$ (Ref. 10), it is also true for $D(z)\,C(z)$ (the predictor merely computes a fixed weight sum of components of $C(z)$). The above condition is also satisfied for $1/Q_1(z)$ (see Eq. 6). Thus, $S(z)$ is realizable.

In this section, we restrict the stability discussion to the nominal operating point, $A = 1$, with analysis of the range of stable operating conditions deferred to Section IV. When $A = 1$, the closed loop function is:

$$P(z) = H(z) = 1 - E_1(z) \quad (12)$$

Equation (12), combined with Table 3, shows that $P(z)$ has poles at $z = \alpha$ and at $z = 0$. Since α is real and satisfies $0 < \alpha < 1$, all the poles of $P(z)$ are inside the unit circle and the closed-loop transfer function is stable. Gain margin is computed in the next section.

Even though $P(z)$ is stable, special care must be exercised in the implementation of $S(z)$. Note that $Q_1(z)$ has a zero at $z = -1$; thus, $S(z)$ has a pole at that point. This means that $S(z)$, were it not a part of the feedback loop, would be marginally stable. $H(z)$ is stable since the pole at $z = -1$, contributed by $S(z)$, is cancelled by the zero of $Q(z)$ at the same location. To maintain numerical stability, this pole of $S(z)$ may be moved slightly into the unit circle. In Section IV, we demonstrate that this shift in pole location has negligible effect on loop performance.

IV. Performance Analysis

In this section, performance of the loops is analyzed from three aspects. First, loop type and errors due to dynamics of the received phase are evaluated, then loop bandwidths are determined, and, finally, gain margins are computed. Discussions of dynamic errors and loop bandwidths is restricted to the nominal operating point, A = 1.

A. Loop Type and Dynamic Tracking Errors

Loop type is defined as the number of integrators in the open loop transfer function, i.e., the number of poles of $G(z)$ at $z = 1$. At the operating point, $G(z)$ is equal to $H(z)/E_1(z)$ forcing the poles of $G(z)$ to be either poles of $H(z)$ or zeroes of $E_1(z)$. However, $H(z)$ has no poles at $z = 1$, and $E_1(z)$ has as many zeros at $z = 1$ as the order of the estimator; thus, second- and third-order phase models correspond to type II and type III loops, respectively. We observe that while loop type is independent of the number of computational delays, order of polynomials in the transfer functions strongly depends on the number of delays. Specifically, the orders of the numerator and denominator of $E_1(z)$ are equal to the loop type plus $N - 1$, where N is the number of delays.

The steady state phase error for constant dynamics is computed using the final value theorem:

$$E_\infty = \lim_{z \to 1} \frac{z-1}{z} E(z)\,\theta(z) \quad (13)$$

It is well known that a type II loop has zero phase error for phase ramp, constant (non-zero) phase error for phase acceleration, and infinite phase error for phase jerk, or higher dynamics. Similarly, a type III loop has zero, constant, and infinite phase errors for phase acceleration, jerk, and higher dynamics, respectively. Thus, the dynamics of interest are phase acceleration for a type II loop, and phase jerk for type III loop. Table 4 lists the steady state phase errors for the types II and III loops, at the operating point $A = 1$. Figure 5 shows these phase errors as functions of the normalized value of the estimator time constant. The inputs are unit phase acceleration, e.g., rad/s^2, for a type II loop and unit phase jerk for a type III loop, and the errors are in consistent units. As the estimator time constant increases, the phase error increases. In fact, for $T \ll \tau$ and $N = 1$, the normalized phase

errors are τ^2 and τ^3 for type II and type III loops, respectively. Phase errors also increase slightly with increase in the number of delays. This is caused by the increased prediction error.

One must remember that, were a predictor not included, increased delays would have caused catastrophic performance degradation. As an exercise, the interested reader can derive expressions for $E_1(z)$, when no prediction is used, i.e.,

$$D(z) = (1, 0) \quad or \quad D(z) = (1, 0, 0) \tag{14}$$

and prove that both second- and third-order estimators result in type I loops.

B. Bandwidth Results

One-sided bandwidth of the loop is defined as:

$$B_L = \frac{1}{2T} \frac{1}{|P(1)|^2} \frac{1}{2\pi j} \int_{\text{unit}} P(z) P(z^{-1}) \frac{dz}{z} \tag{15}$$

where $P(z)$ is the closed-loop transfer function, and the integral is evaluated along the unit circle. At the desired operating point, $A = 1, P(z) = H(z)$, and the integral can be written as:

$$B_L T = \frac{1}{2} \int_0^1 \left| H(e^{2\pi jx}) \right|^2 dx \tag{16}$$

The quantity $B_L T$ is also called normalized bandwidth. Equation (16) uses the fact that in our example, at $z = 1$, $H(z)$ is unity, as can be verified from the definitions in Section II. The integral was numerically evaluated for different estimators and delays. Figure 6 shows the bandwidth as a function of the estimator time constant and N for type II and type III loops. We observe that the introduction of extra delays in the loop increases the noise bandwidth for a fixed time constant, but the effect of the delays is less significant as the estimator time constant increases.

C. Stability and Gain Margin

Let us first discuss the closed-loop root loci for the designed DPLL. Recall the definition of the open loop transfer function in Eq. (8). The open-loop transfer function $G(z)$ has zeros at the zeros of $H(z)$ and poles at the zeros of $E_1(z)$; thus, with increase of the phase detector gain A, the poles of $P(z)$ move from the zeros of $E_1(z)$ to the zeros of $H(z)$.

Figure 7 shows root-locus plots for type II and III loops, for $N = 2$. The transfer function $E_1(z)$, for a type II loop, has three zeros, two at $z = 1$ and one at $z = -2(1 - \alpha)$. In Fig. 7(a) we observe that at low gain, the poles of $P(z)$ are near these points; then, as the gain increases, they move within the unit circle. Eventually, with higher gains, the poles of $P(z)$ go outside the unit circle and stability is lost. The root locus for a type III loop, illustrated in Fig. 7(b), has a similar form, except that at low gain roots may occur outside the unit circle, near $z = 1$.

As shown previously, the design can be marginally stable due to cancellation of a pole of $S(z)$ by a zero of the NCO. To assure stability, this $S(z)$ pole at $z = -1$ is slightly shifted to the inside of the unit circle. Effect of this shift in pole location is demonstrated in Figs. 7(c) and 7(d). These figures repeat the cases given in Figs. 7(a) and 7(b), with the pole shifted to $z = -0.98$. There is an extra root locus branch from $z = -1.0$ toward $z = -0.98$, with negligible effect on the rest of the root locus. In fact, since the whole root locus branch is inside the unit circle, an arbitrarily small shift in pole location is sufficient to guarantee loop stability. This small shift in pole location has a negligible effect on the three loop parameters of interest: dynamic phase error, gain margin, and noise bandwidth.

Locations of the poles of the closed-loop transfer function at the operating point are of special importance. These poles are commonly selected to be on the real axis to avoid instability and oscillations when the gain A changes slightly. At the operating point, $A = 1$, the closed loop transfer function is $H(z)$. For the estimator-predictor in our example, $H(z)$ has poles at $z = 0$ and two or three co-located poles at $z = \alpha$ for the second- and third-order estimators, respectively. This feature, real closed-loop transfer function poles at the operating point, assures stability.

Gain margin measures the effect of the phase detector gain, A, on the stability of the closed loop. It is defined as the ratio of the critical gain, i.e., gain that forces the poles of the transfer function outside the unit circle, to the nominal unity gain. Figure 8 shows the relationship between upper gain margin and normalized bandwidth. Here, as the loop becomes wider, the gain margin decreases. Similar results are presented in Fig. 9 for the lower gain margin of the type III loop. Note that the lower gain margin for the type II loop is infinity. These figures are useful for quick design evaluation, e.g., finding a favorable trade-off between any two variables.

D. Approximation Formulas

Figures 8 and 9 presented a trade-off between gain margin and normalized loop bandwidth, $B_L T$. Similar results can be obtained for a trade-off between dynamic errors and $B_L T$. Since dynamic phase error, E_∞, is approximately inversely

proportional to the second or third power of the normalized bandwidth, we define phase error coefficients, C_{ϕ_2} and C_{ϕ_3}, so that the steady state phase errors due to unit acceleration or jerk are:

Second-order estimator: $$E_\infty = \left(\frac{C_{\phi_2}}{B_L}\right)^2 \quad (17)$$

Third-order estimator: $$E_\infty = \left(\frac{C_{\phi_3}}{B_L}\right)^3 \quad (18)$$

These coefficients exhibit slower variation with $B_L T$ than do the dynamic phase errors. Figure 10 shows the phase coefficients as functions of normalized bandwidths. The phase coefficients, for narrow normalized bandwidths, are less than 1 for type II loops and slightly over 1 for type III loops, and they increase slowly with increased bandwidths.

Approximation formulas were also derived for dynamic phase error and gain margin as functions of the one-sided bandwidth, B_L. The resulting equations are valid for $B_L T < 0.1$ and $N = 2$, though the dependence of N is weak. Table 5 summarizes the equations.

V. Conclusions

Using an estimation theory approach, design of a DPLL can be accomplished in a systematic procedure rather than a trial-and-error approach. The examples presented here, second- or third-order fixed gain, fading memory estimators, are useful for many applications, and the resulting design curves may thus be directly applied. It is proposed that the design procedure be applied to the DSN Advanced Receiver Carrier DPLL.

References

1. Gardner, F. M., *Phaselock Techniques*, John Wiley and Sons, 1979.
2. Best, R. E., *Phase Locked Loops, Theory, Design and Applications*, Mcgraw Hill, 1984.
3. Gill, G. S., and Gupta, S. C., "First Order Discrete Phase Locked Loops with Applications to Demodulation of Angle Modulated Carrier," *IEEE Trans. on Communication Technology*, Vol. COM-20, pp. 454-462, June 1972.
4. Weinberg, A., and Liu, B., "Discrete Time Analysis of Nonuniform Sampling First- and Second-Order Digital Phase Locked Loops," *IEEE Trans. on Communication Technology*, Vol. COM-22, pp. 123-137, Feb. 1974.
5. Osborne, H. C., "Stability Analysis of Nth Power Digital Phase Locked Loop – Part I: First Order DPLLs," *IEEE Trans. on Communication Technology*, Vol. COM-28, pp. 1343-1354, Aug. 1980.
6. Osborne, H. C., "Stability Analysis of Nth Power Digital Phase Locked Loop – Part II: Second- and Third-Order DPLL's," *IEEE Trans. on Communication Technology*, Vol. COM-28, pp. 1355-1364, Aug. 1980.
7. Aguirre, S., and Hurd, W. J., "Design and Performance of Sampled Data Loops for Subcarrier and Carrier Tracking," *TDA Progress Report 42-79*, pp. 81-95, Jet Propulsion Laboratory, Pasadena, CA, Nov. 1984.
8. Simon, M. K., and Mileant, A., "Performance of the DSA's Subcarrier Demodulation Digital Loop," *TDA Progress Report 42-80,* pp. 180-194, Jet Propulsion Laboratory, Pasadena, CA, Feb. 1985.
9. Kumar, R., and Hurd, W. J., "A Class of Optimum Digital Phase Locked Loops for the DSN Advanced Receiver," *TDA Progress Report 42-83,* pp. 63-80, Jet Propulsion Laboratory, Pasadena, CA, Nov. 1985.
10. Statman, J., "A Recursive Solution for a Fading Memory Filter Derived From Kalman Filter Theory," TDA Progress Report 42-86 (this issue).
11. Oppenheim, A. V., and Schafer, R. W., *Digital Signal Processing*, Prentice-Hall, N.J., 1975.

Table 1. Closed form transfer functions from input phase to output parameter

Outside Parameter	Second-Order Estimator	Third-Order Estimator
Phase	$\frac{(1-\alpha^2)\, z\left(z-2\,\frac{\alpha}{1+\alpha}\right)}{(z-\alpha)^2}$	$\frac{(1-\alpha)\, z((\alpha^2+\alpha+1)\, z^2-3\alpha(1+\alpha)\, z+3\alpha^2)}{(z-\alpha)^3}$
Frequency	$\frac{(1-\alpha)^2\, z(z-1)}{T(z-\alpha)^2}$	$\frac{(1-\alpha)\, z(z-1)((3-3\alpha^2)\, z+5\alpha^2-4\alpha-1)}{2T(z-\alpha)^3}$
Frequency rate	–	$\frac{(1-\alpha)^3\, z(z-1)^2}{T^2(z-\alpha)^3}$

Table 2. Predictor formulas

Predictor	Second-Order Estimator	Third-Order Estimator
$D(z)$	$(1, NT)$	$\left(1, NT, \frac{(NT)^2}{2}\right)$

Table 3. Formulas for $E_1(z)$

N	Second-Order Estimator	Third-Order Estimator
1	$\frac{(z-1)^2}{(z-\alpha)^2}$	$\frac{(z-1)^3}{(z-\alpha)^3}$
2	$\frac{(z-1)^2}{(z-\alpha)^2}\ \frac{z+2(1-\alpha)}{z}$	$\frac{(z-1)^3}{(z-\alpha)^3}\ \frac{z+3(1-\alpha)}{z}$
3	$\frac{(z-1)^2}{(z-\alpha)^2}\ \frac{z^2+2(1-\alpha)\, z+(\alpha^2-4\alpha+3)}{z^2}$	$\frac{(z-1)^3}{(z-\alpha)^3}\ \frac{z^2+3(1-\alpha)\, z+(3\alpha^2-9\alpha+6)}{z^2}$

Table 4. Steady state phase errors due to dynamics (phase units)

N	Second-Order Estimator $\left(\theta(t) = \frac{1}{2} t^2\right)$	Third-Order Estimator $\left(\theta(t) = \frac{1}{6} t^3\right)$
1	$\frac{T^2}{(1-\alpha)^2}$	$\frac{T^3}{(1-\alpha)^3}$
2	$\frac{T^2}{(1-\alpha)^2}(3-2\alpha)$	$\frac{T^3}{(1-\alpha)^3}(4-3\alpha)$
3	$\frac{T^2}{(1-\alpha)^2}(\alpha^2-6\alpha+6)$	$\frac{T^3}{(1-\alpha)^3}(3\alpha^2-12\alpha+10)$

Table 5. Approximation formulas

DPLL Parameter	Second-Order Estimator	Third-Order Estimator
Gain margin (g_m)	$\frac{0.53}{(B_L T)^{0.9}}$	$\frac{0.89}{(B_L T)^{0.75}}$
Dynamic lag (E_∞)	$a\left(\frac{0.87}{B_L}\right)^2$	$j\left(\frac{1.17}{B_L}\right)^3$

E_∞ = units (e.g., rad)

a = acceleration, units/s^2 (e.g., rad/s^2)

j = jerk, units/s^3 (e.g., rad/s^3)

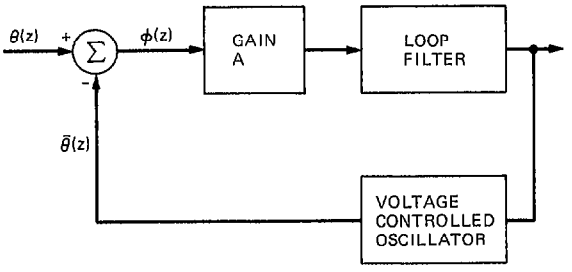

Fig. 1. Traditional phase-locked loop

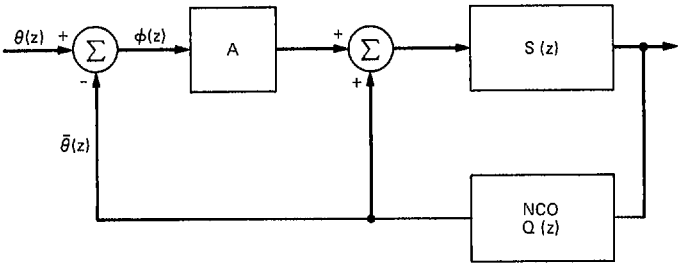

Fig. 3. DPLL physical block diagram

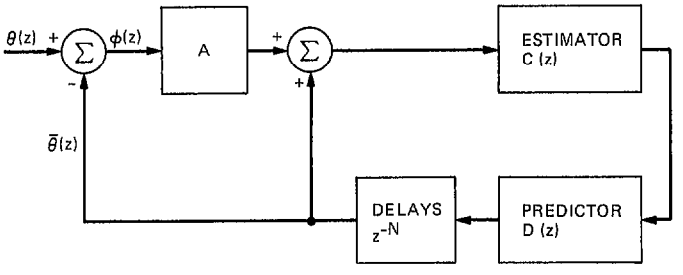

Fig. 2. DPLL functional block diagram

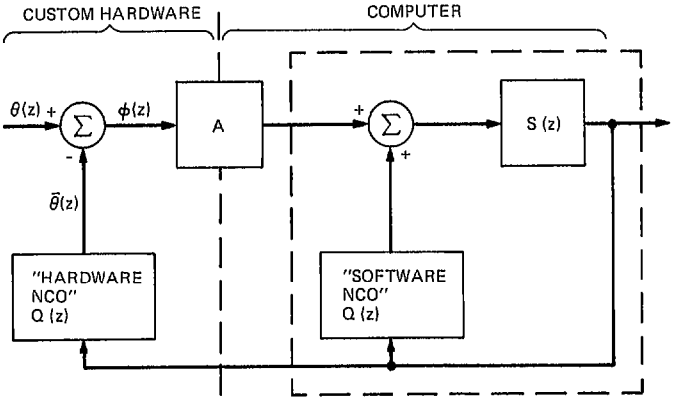

Fig. 4. Proposed DPLL implementation

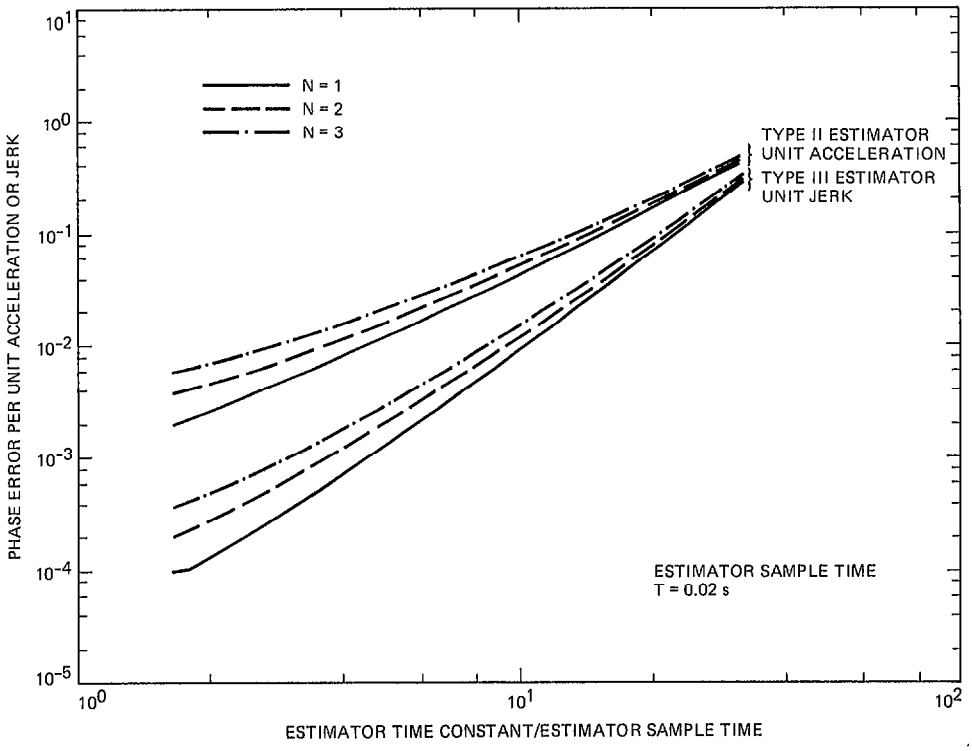

Fig. 5. Dynamic phase error versus estimator parameters

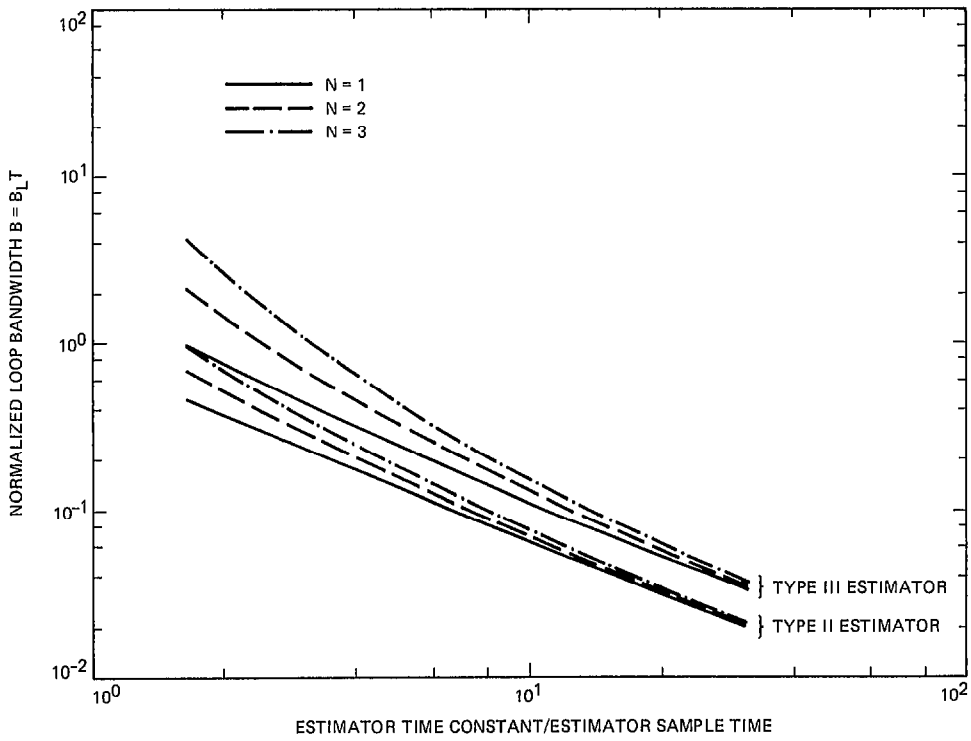

Fig. 6. Loop bandwidth versus estimator parameters

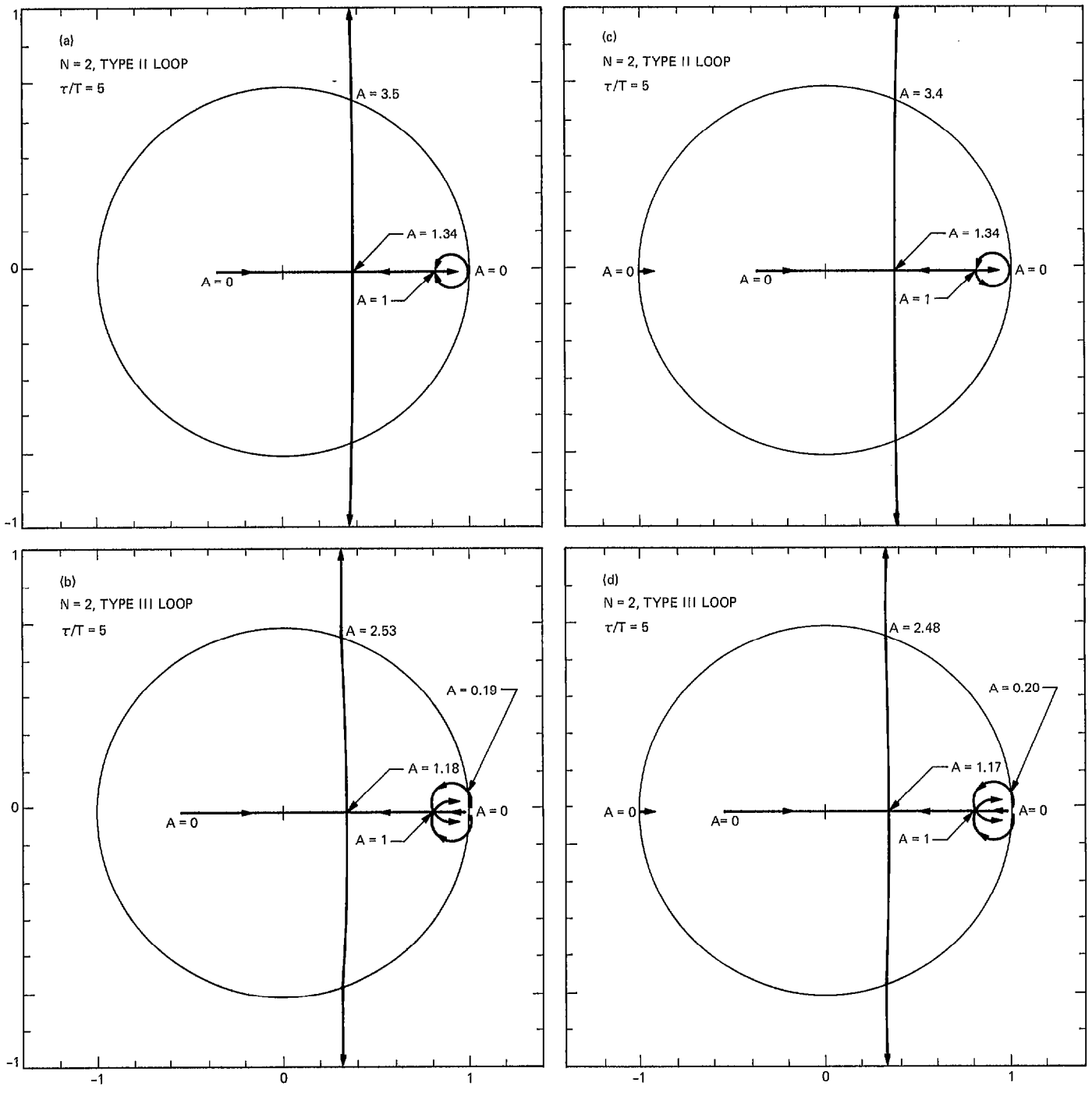

Fig. 7. Examples of root loci: (a) type II loop; (b) type III loop; (c) type II loop with pole at $z = -0.98$; (d) type III loop with pole at $z = -0.98$

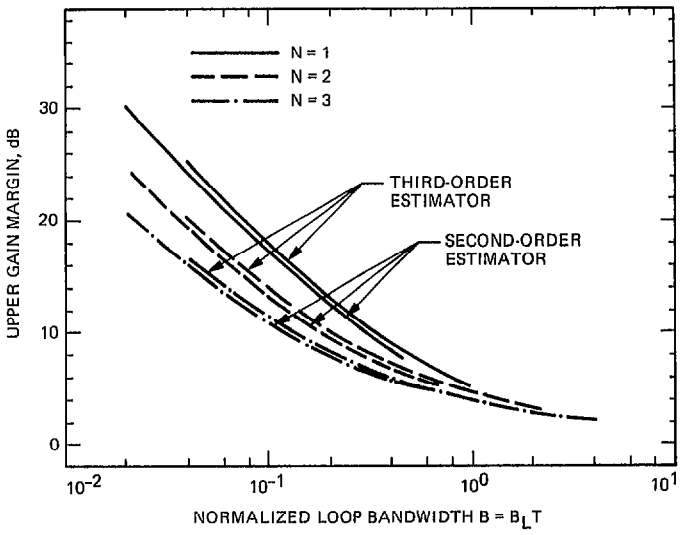

Fig. 8. Upper gain margin versus normalized loop bandwidth

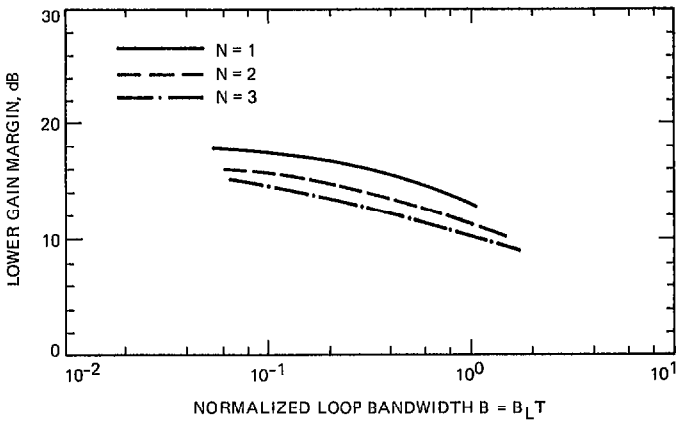

Fig. 9. Lower gain margin versus normalized loop bandwidth

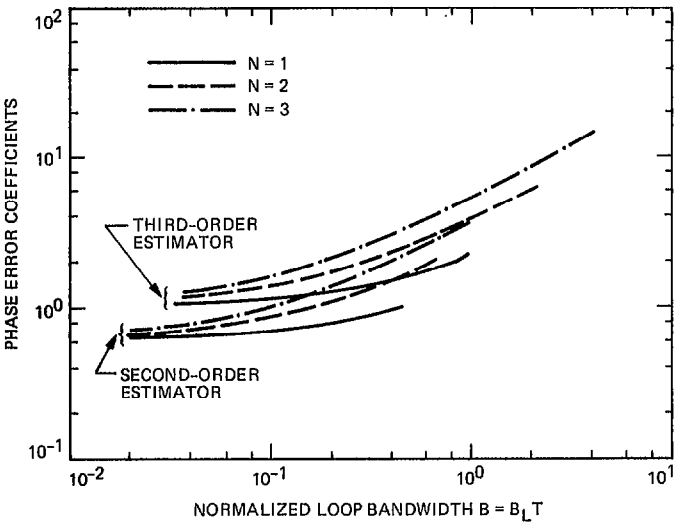

Fig. 10. Phase error coefficients versus normalized loop bandwidth