Noncausal Telemetry Data Recovery Techniques

H. Tsou, R. Lee, A. Mileant, and S. Hinedi
Communications Systems and Research Section

Cost efficiency is becoming a major driver in future space missions. Because of the constraints on total cost, including design, implementation, and operation, future spacecraft are limited in terms of their size, power, and complexity. Consequently, it is expected that future missions will operate on marginal space-to-ground communication links that, in turn, can pose an additional risk on the successful scientific data return of these missions. For low data-rate and low downlink-margin missions, the buffering of the telemetry signal for further signal processing to improve data return is a possible strategy; it has been adopted for the Galileo S-band mission. This article describes techniques used for postprocessing of buffered telemetry signal segments (called gaps) to recover data lost during acquisition and resynchronization. Two methods, one for a closed-loop and the other one for an open-loop configuration, are discussed in this article. Both of them can be used in either forward or backward processing of signal segments, depending on where a gap is specifically situated in a pass.

I. Introduction

With all the budget cuts under way, cost efficiency is becoming a major driver in future space missions. Because of the constraints on total cost, including design, implementation, and operation, future spacecraft are limited in terms of their size, power, and complexity. Consequently, it is expected that future missions will operate on marginal space-to-ground communication links. As a result, data compression techniques aboard the spacecraft and other advanced signal-processing techniques on the ground are important for providing alternatives in order to increase the scientific data return of a mission by improving the communication link margin. One adopted concept is buffering the telemetry signal for further signal processing to improve data return for low data-rate and low downlink-margin missions. This has been implemented first in the buffered telemetry demodulator (BTD) to support NASA's Galileo Mission [1]. In this mission, because of a malfunctioning high-gain antenna, the Galileo spacecraft has to rely on its low-gain antenna to transmit data from Jupiter back to Earth. The mission operation can only support very low symbol signal-to-noise ratios (SNRs) (ranging from -10 to -6 dB) at low symbol rates (up to 640 symbols per second). For such a low SNR range, it is estimated that the signal acquisition (including carrier, subcarrier, and symbol) can take up to 10 minutes when a single ground antenna is used. Furthermore, in order to maximize the data return, several data rate changes intending to take advantage of all the available antenna apertures on the ground can occur in a pass, and each may require reacquisition of the signal, depending upon the availability of exact knowledge of phase and timing at the rate change. The combination of the above two conditions can result in significant data loss if the signal is not buffered.

With the data buffered, various nonreal-time (noncausal) signal-processing techniques can be performed to reduce the data loss due to acquisition, resynchronization, and loss of lock. For example, the

C-2

use of different loop bandwidths and/or different quadrature windowing may realize better acquisition performance [2]. In this article, we will focus on techniques used for reprocessing segments of buffered telemetry signal (called gaps), by using checkpoint information obtained from the real-time processing of signal segments next to the gap (called pads), to recover the lost data. These techniques, being able to greatly reduce telemetry data loss under low-SNR scenarios, present a unique opportunity to employ non-causal signal-processing techniques for the purpose of signal demodulation and detection. In Section II, a brief description of the BTD and the nature of gaps is given. Two methods, one for a closed-loop and the other one for an open-loop configuration, are discussed in Sections III and IV, respectively. Both of them can be used for either forward or backward gap processing, depending on where a gap is specifically situated in a pass. The overall gap-processing strategy is then discussed in Section V. Finally, the loop initialization technique associated with closed-loop gap closure processing can be found in the Appendix.

II. Buffered Telemetry Demodulator and Gaps

The BTD is a software receiver implemented on a general-purpose, multiple central-processing unit (CPU) workstation. It performs acquisition and tracking functions for the carrier, subcarrier, and symbol as well as providing miscellaneous monitoring functions, such as lock indicators, symbol SNR estimators, etc. It is designed to take advantage of multiple CPUs in simultaneously doing several processes on different segments of digitally sampled and recorded signal. For example, it can process real-time samples forward (in terms of time) and reprocess any other segment of samples in the past at the same time. The necessity to reprocess a segment of samples arises from the following two scenarios: (1) the out-of-lock indication is detected in any of the loops in the BTD and (2) the succeeding decoder fails to extract valid information from the BTD output symbols. Both situations indicate that demodulation was not successfully performed for that segment of samples, which is called a gap. Typically, gaps are caused by either acquisition/reacquisition or cycle slip in one of the three loops. Gaps due to acquisition/reacquisition can be found in the beginning of each pass and at the instants of data rate change. However, gaps due to cycle slips are usually accompanied by the drop of loop SNR and can occur randomly in a pass. In rare cases, a gap may happen as demodulated symbols are mishandled in the data flow between the BTD and the succeeding decoder. The processing of a gap to extract any valid information not available when that segment of signal samples was first processed is referred to as gap closure processing (GCP).

Along its demodulation process, the BTD keeps a record of its internal states, including the lock indicator states, the state variables inside the loop filter, and the numerically controlled oscillator (NCO) for all three loops. These state variables are recorded at fixed intervals and are stored as checkpoints. The checkpoint information will serve as a reference when a gap closure process is necessary. For a gap situated inside a pass, it is possible to perform the GCP by using the available checkpoint information either prior to or after the gap. The former is referred to as forward gap closure processing (FGCP) because it loads in the checkpoint information prior to the gap and starts to process the gap from the beginning toward the end; the latter is referred to as backward gap closure processing (BGCP) since the gap is processed in the order of reversed time. Obviously, only the FGCP can be performed when the gap is at the end of a pass, and only the BGCP can be performed when the gap is at the beginning of a pass. There is no essential difference between FGCP and BGCP except that the order of signal samples processed is reversed. The exclusion of either one depends solely on the availability of sufficient checkpoint information. Since every state variable of each checkpoint is a random variable, it is necessary to take more than one checkpoint to make a good estimation of states at the start of GCP. A pad is referred to a segment of signal samples from which all the checkpoints used for a GCP are derived. Since reliable checkpoint information is essential to a successful GCP, it is always true that the receiver remains in lock throughout a pad while performing real-time signal processing. It will be clear that the pad-to-gap size ratio is an important factor in determining the strategy of GCP.

The concept of FGCP and BGCP is simply illustrated in Fig. 1, where two gaps, two pads, and associated GCPs are given. The gap #1 is caused by initial acquisition at the start of a pass. It can be
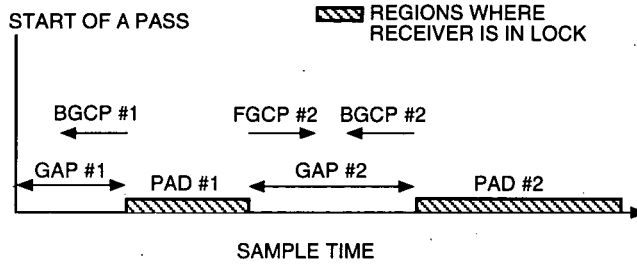
Fig. 1. Conceptual illustration of gap closure processing.

recovered by the BGCP #1 using checkpoint information obtained from pad #1. The same checkpoint information can also support the FGCP #2 in reprocessing gap #2, where it is inside a pass and probably caused by a data rate change or an unexpected drop of loop SNR. Of course, BGCP #2, using checkpoint information obtained from pad #2, is an alternative for the GCP of gap #2.

III. Closed-Loop Gap Closure Configuration

The most straightforward way to reprocess a gap is to run the same receiver over that segment of sampled signal with some new a priori information pertinent to the starting point. The a priori information is typically obtained by estimates of the state prior to the gap (in the case of FGCP) or after the gap (in the case of BGCP) in segments when the receiver is in lock. Then, a loop-filter coefficients initialization method [3] that was first proposed to reduce the transient response of a digital phase-locked loop (DPLL) can be used to set up the loop using the estimated state information. As shown in the Appendix, a baseband equivalent phase-locked loop can be augmented by including additional loop parameters for the purpose of phase, frequency, and frequency rate initialization. For example, a typical third-order DPLL for deep-space applications, given in Fig. 2, can track a signal with third-order Doppler effect, whose phase can be expresssed as

$$\theta(t) = \theta_0 + \theta_1 t + \frac{\theta_2}{2!}t^2 + \frac{\theta_3}{3!}t^3 \tag{1}$$

The loop transient state can be eliminated by setting the parameters C_0, C_1, and C_2 as follows:

$$C_0 = \theta_0 - \frac{\theta_3 T_u^2}{G_3} \tag{2a}$$

$$C_1 = \theta_1 + \theta_2 T_u - \frac{G_1}{G_3}\theta_3 T_u^2 + \frac{1}{2}\theta_3 T_u^2 \tag{2b}$$

$$C_2 = \frac{T_u}{G_3}\left[\theta_2 + \frac{1}{2}\theta_3 T_u - \frac{G_2}{G_3}\theta_3 T_u\right] \tag{2c}$$

where $G_i, i \in \{1, 2, 3\}$ are internal gain factors of the loop filter and T_u is the loop filter update interval. With good a priori knowledge on θ_0, θ_1, θ_2, and θ_3 as checkpoint information available right before a gap, Eqs. (2a), (2b), and (2c) can be used for FGCP.

Besides the required checkpoint information obtained right after the end of a gap, the BGCP needs another slight modification in these coefficient settings by changing the sign in front of the terms containing
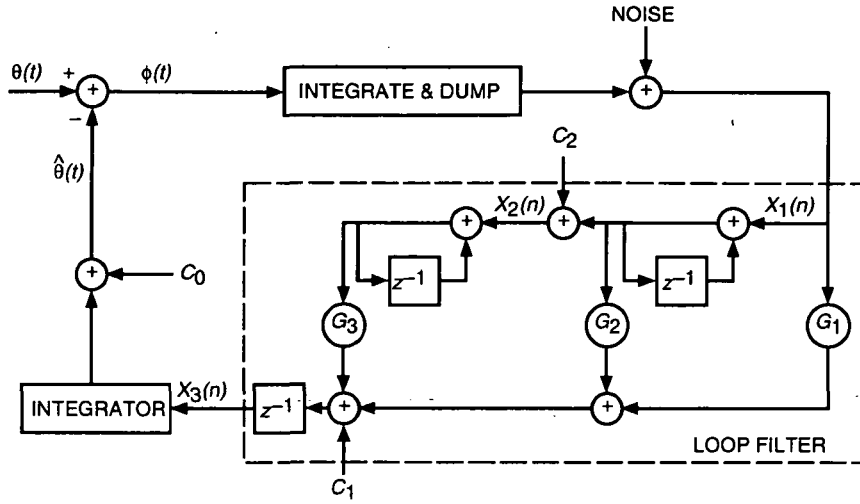
Fig. 2. Augmented third-order digital phase-locked loop.

θ_i, where i is a positive even integer. In the above example, C_0 remains unchanged but C_1 and C_2 need to be changed so that

$$C_0 = \theta_0 - \frac{\theta_3 T_u^2}{G_3} \tag{3a}$$

$$C_1 = \theta_1 - \theta_2 T_u - \frac{G_1}{G_3}\theta_3 T_u^2 + \frac{1}{2}\theta_3 T_u^2 \tag{3b}$$

$$C_2 = \frac{T_u}{G_3}\left[-\theta_2 + \frac{1}{2}\theta_3 T_u - \frac{G_2}{G_3}\theta_3 T_u\right] \tag{3c}$$

The reason for this modification is that the odd-order derivatives of a phase process do not need a sign change as the time is reversed; however, the even-order derivatives do need one. This can be explained by a simple example, as follows: The frequency for a periodic movement at a rate of $2\pi/s$ is always 1 Hz, whether its associated phasor is described as rotating clockwise or counterclockwise. On the other hand, a frequency ramp increasing (positive slope) with time will be decreasing (negative slope) when the time is reversed.

Similar results for a second-order DPLL can be readily given, by setting $G_3 = \theta_3 = 0$ in the derivation of third-order DPLL results shown before, as follows for FGCP:

$$C_0 = \theta_0 - \frac{\theta_2 T_u}{G_2} \tag{4a}$$

$$C_1 = \theta_1 + \theta_2 T_u - \left(\frac{G_1}{G_2}\right)\theta_2 T_u \tag{4b}$$

and for BGCP:

$$C_0 = \theta_0 + \frac{\theta_2 T_u}{G_2} \tag{5a}$$

87

$$C_1 = \theta_1 - \theta_2 T_u + \left(\frac{G_1}{G_2}\right)\theta_2 T_u \tag{5b}$$

IV. Open-Loop Gap Closure Configuration

A gap can be reprocessed without the use of a PLL to adaptively estimate the signal phase, provided that the signal phase is reasonably stable or slowly varying in that gap. The open-loop gap closure configuration uses an estimated phase profile to serve as the reference phase. The estimated phase profile can be obtained from the phase observation over an adjoint region where phase has been successfully tracked.

It is assumed that the measurements at the output of the NCO can be modeled as

$$y(k) = \theta(k) + \nu(k) = \theta_0 + \theta_1(kT) + \frac{\theta_2(kT)^2}{2} + \nu(k) \tag{6}$$

where $\theta(k)$ is the input phase process sampled at interval T and $\nu(k)$ is a zero-mean Gaussian random variable with variance σ_y^2. Given m noisy correlated samples of $y(k)$, the least-squared estimate of the input phase parameters, namely, θ_0, θ_1, and θ_2, is easily shown as [4]

$$\hat{\underline{\theta}} = \left[H^T H\right]^{-1} H^T \underline{y} \tag{7}$$

where $\hat{\underline{\theta}}$ is the estimation vector and $\underline{y}$ is the measurement vector, given as

$$\hat{\underline{\theta}} = \begin{bmatrix} \hat{\theta}_0 \\ \hat{\theta}_1 \\ \hat{\theta}_2 \end{bmatrix} \quad \text{and} \quad \underline{y} = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(m) \end{bmatrix}$$

and the transpose of the observation matrix H is given as

$$H^T = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ T & 2T & \cdots & mT \\ \dfrac{T^2}{2} & \dfrac{(2T)^2}{2} & \cdots & \dfrac{(mT)^2}{2} \end{bmatrix}$$

Therefore, the estimated phase at time t will be

$$\hat{\theta}(t) = \hat{\theta}_0 + \hat{\theta}_1 t + \frac{\hat{\theta}_2 t^2}{2} \tag{8}$$

which is an unbiased estimate with variance

$$\sigma_{\hat{\theta}}^2(t) = \sigma_y^2 \left[p_{11} + p_{22}t^2 + \frac{p_{33}t^4}{4} + 2\left(p_{12}t + \frac{p_{13}t^2}{2} + \frac{p_{23}t^3}{2}\right)\right] \tag{9}$$

where p_{ij} is the (i,j)th component of the covariance matrix

$$P = \left[H^T H\right]^{-1} H^T R H \left[H^T H\right]^{-1}$$

with $R = E\{\underline{\nu}\underline{\nu}^T\}$ being the covariance matrix of the NCO output, determined by sampling the correlation function of the phase process given in [5].

It is interesting to note that the normalized variance of the open-loop, least-squared estimated phase, defined as the ratio of $\sigma_{\hat{\theta}}^2(t)/\sigma_y^2$, serves as an important indicator showing whether the open-loop configuration outperforms the closed-loop configuration or not. A ratio smaller than 1 actually implies that the open-loop estimated phase has smaller variance than the NCO output phase derived from the closed-loop configuration. Figure 3 gives a typical example of the performance of the least-squares smoother/predictor versus the sample number, in which the observation window consists of 2000 samples starting from the sample one. Note that the region where normalized variance is less than unity extends both sides symmetrically beyond the observation window. The ratio of the single-sided extension size to the observation window size, denoted as $\eta(m, B_L T)$, is listed in Table 1 for various m, the number of observed samples, and $B_L T$, the product of the loop bandwidth of the counterpart closed-loop configuration and the sampling interval. For example, as listed in Table 1 and also shown in Fig. 3, the open-loop configuration should outperform the closed-loop configuration over any gap with a size up to 133 percent of the available observation size $m = 2000$ and $B_L T = 1.0$.
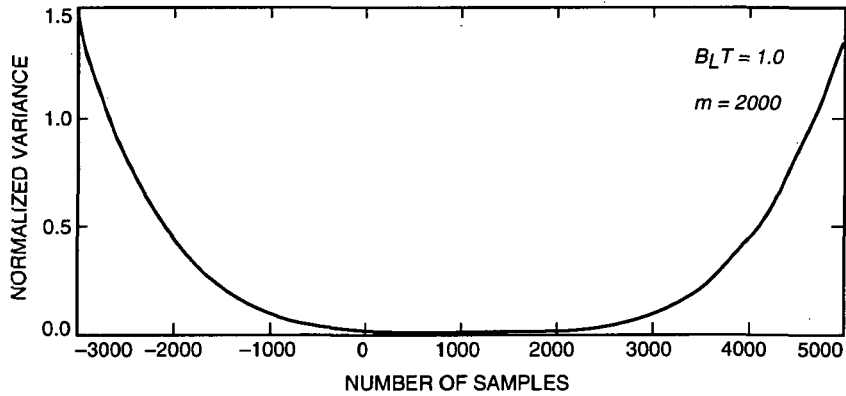

Fig. 3. Performance of the least-squares smoother/predictor.

Table 1. The $\eta(m, B_L T)$ values.

m	$B_L T$				
	1.0, %	0.2, %	0.1, %	0.05, %	0.01, %
100	38.00	14.00	5.00	0.00	0.00
500	80.80	43.60	29.60	18.20	0.00
1000	105.00	60.40	43.80	29.80	6.20
2000	133.85	80.55	60.65	43.90	15.05
5000	180.62	113.32	88.12	66.84	29.94
10,000	223.91	143.71	113.64	88.17	43.92
15,000	252.95	164.12	130.79	102.57	53.41
20,000	275.43	179.92	144.07	113.72	60.76

V. Overall Gap Closure Strategy

It has been shown that open-loop configuration can simply outperform the closed-loop configuration for a certain length of time into a gap, and then its performance deteriorates rapidly when further into the gap. Based upon this phenomenon, the overall gap closure strategy that optimizes the GCP performance is depicted as the flow chart of Fig. 4.
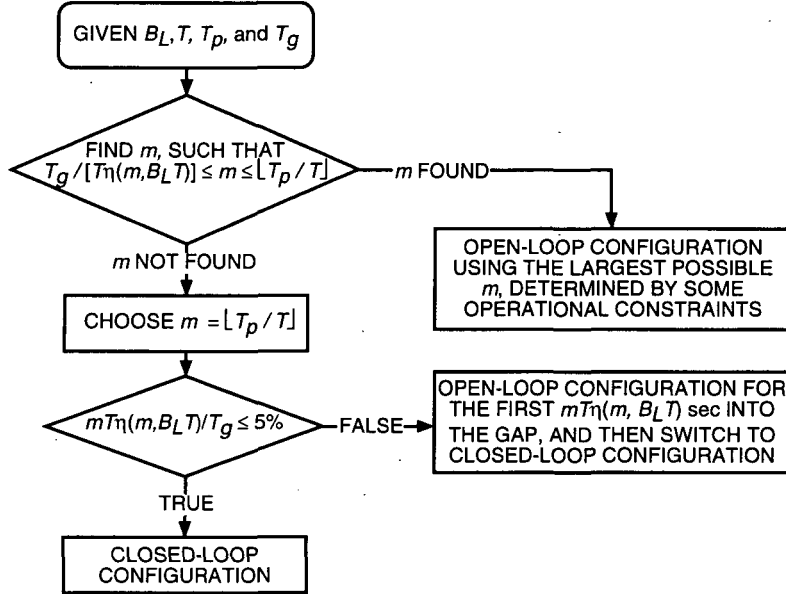

Fig. 4. The overall gap closure strategy.

For a given loop bandwidth B_L, checkpoint interval T, gap size T_g, and available pad size T_p, the open-loop configuration is able to process the complete gap with better performance, only if there exists at least an integer number of checkpoints, denoted as m, which satisfies

$$\frac{T_g}{T \cdot \eta(m, B_L T)} \leq m \leq \left\lfloor \frac{T_p}{T} \right\rfloor \tag{10}$$

The upper limit of Eq. (10) is the number of available checkpoints from the given pad, and the lower limit is the number of required checkpoints to open loop cover the whole gap. When such m exists, the largest possible m, determined by some operational constraints, is chosen to ensure the best open-loop gap closure performance. When no such m can be found to satisfy Eq. (10), it is still possible to use open-loop configuration up to the point where the open-loop performance is going to be worse than the closed-loop performance, namely the first $mT \cdot \eta(m, B_L T)$ seconds into the gap, and then close the loop for the rest of the gap. A decision needs to be made here whether the benefit justifies including an additional switching mechanism from open-loop to closed-loop configuration. Typically, the break point is arbitrarily set as

$$\frac{mT \cdot \eta(m, B_L T)}{T_g} = 5\%$$

Figure 5 shows the result from gap closure processing of a gap in the real Galileo telemetry data received at DSS 14. The received signal has a predicted P_T/N_0 ratio around 16 dB-Hz with its carrier
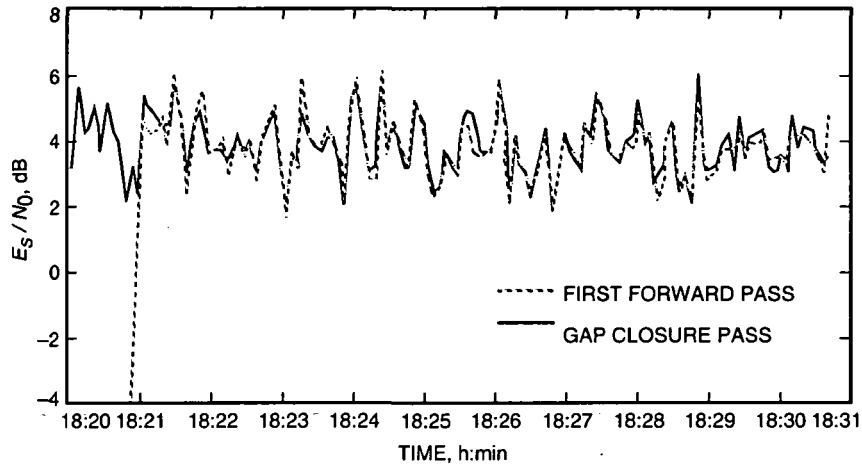
Fig. 5. The estimated symbol SNR versus time.

component being fully suppressed. The symbol rate is 20 symbols per second, which is translated to the predicted symbol SNR, denoted as E_s/N_0, at the 3-dB level. The gap, characterized by the below-average estimated SNR of demodulated symbols from the BTD in Fig. 5, happens at the beginning of the first forward pass and lasts about 1 minute (from 18:20 to 18:21). After this initial acquisition effort, the BTD maintains its lock on the signal, and the demodulated symbol SNR is stabilized around its predicted level. During BGCP, the telemetry data in this gap are successfully recovered as the estimated symbol SNR remains at its normal level until the beginning of the data pass is reached.

VI. Conclusion

Techniques used to recover data loss from buffered telemetry signal during acquisition/reacquisition, data rate changes, and cycle slips are presented in this article. Two different configurations—one a simple least-squares phase estimator used for the open-loop gap closure processing and the other involving loop filter initialization for the closed-loop gap closure processing—can be chosen to recover lost data, depending upon the size of a gap and the size of its surrounding pad(s). Both methods have the flexibility to be used in either forward or backward processing, which helps to cover gaps at various positions, especially at the beginning of each pass where initial acquisition is always required. The overall strategy that optimizes the gap closure-processing performance is also presented. With the ease of implementation on general-purpose workstations and the flexibility to work on the reversed time order, these techniques should be crucial to the future low-cost space missions, since the expected very low downlink communication margin is a threat to the scientific data return.

Besides the techniques presented in this article, other techniques, including Kalman filters and the use of both the past and future information jointly to perform the gap closure processing, have also been studied. The performance comparison will be addressed later.

Acknowledgments

References

[1] H. Tsou, B. Shah, and S. Hinedi, "A Functional Description of the Buffered Telemetry Demodulator," *SUPERCOM/ICC'94*, New Orleans, Louisiana, May 1994.

[2] M. Aung, W. J. Hurd, C. M. Buu, and J. B. Berner, "The Block V Receiver Fast Acquisition Algorithm for the Galileo S-Band Mission," *The Telecommunications and Data Acquisition Progress Report 42-118, April–June 1994*, Jet Propulsion Laboratory, Pasadena, California, pp. 83–114, August 15, 1994.

[3] S. A. Stephens and J. B. Thomas, "Controlled-Root Formulation for Digital Phase-Locked Loops," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 31, no. 1, pp. 78–95, January 1995.

[4] H. W. Sorenson, *Parameter Estimation*, New York: M. Dekker, 1985.

[5] A. Mileant and S. Hinedi, "Lock Detection in Costas Loops," *IEEE Trans. on Communications*, vol. 40, no. 3, pp. 480–483, March 1992.

Appendix

Loop Initialization of the Augmented Digital Phase-Locked Loop

Figure 2 shows an augmented baseband-equivalent DPLL used for deep-space applications, in which the loop filter coefficients can be initialized to reduce the transient response. It is assumed that the phase detector and the NCO are updated at a much higher rate—usually the sampling rate—as compared to the loop update rate with which the rest of the loop is operating. Therefore, in this baseband-equivalent model, the NCO is treated as an integrator in the continuous-time domain, and the loop filter remains in the discrete-time domain. The loop filter z-domain transfer function is given as

$$F(z) = G_1 + \frac{G_2}{1 - z^{-1}} + \frac{G_3}{\left(1 - z^{-1}\right)^2} \tag{A-1}$$

where

$$G_1 = \frac{rd}{T_u} \tag{A-2a}$$

$$G_2 = \frac{rd^2}{T_u} \tag{A-2b}$$

$$G_3 = \frac{krd^3}{T_u} \tag{A-2c}$$

and

$$d = \frac{4B_L T_u(r-k)}{r(r-k+1)} \tag{A-2d}$$

with T_u as the loop update interval, B_L as the designed loop bandwidth, r being equal to 4ξ where ξ is the damping ratio, and k as a third-order loop gain parameter ($k = 0$ for a second-order loop). This augmented baseband-equivalent model includes three constant parameters, C_0, C_1, and C_2, which can be used to initialize the loop, as described later.

The estimated phase for this loop can be derived as follows:

$$\hat{\theta}(t) = C_0 + \int_0^t X_3(\tau)d\tau \tag{A-3}$$

where

$$X_3(k) \triangleq \left. \frac{d\hat{\theta}(t)}{dt} \right|_{t=kT_u} = C_1 + G_1 X_1(k-1) + G_2 \sum_{i=1}^{k-1} X_1(i) + G_3 \left[C_2(k-1) + \sum_{i=1}^{k-1} \sum_{j=1}^{i} X_1(j) \right] \tag{A-4}$$

is the loop filter output representing the discrete-time estimated frequency process updated at the loop update rate. The first-order difference of $X_3(k)$ can be found as

$$\Delta X_3(k) \triangleq X_3(k) - X_3(k-1)$$

$$= G_1\left[X_1(k-1) - X_1(k-2)\right] + G_2 X_1(k-1) + G_3\left[C_2 + \sum_{j=1}^{k-1} X_1(j)\right] \tag{A-5}$$

and the second-order difference of $X_3(k)$ can be found as

$$\Delta^2 X_3(k) \triangleq \Delta X_3(k) - \Delta X_3(k-1)$$

$$= G_1\left[X_1(k-1) - 2X_1(k-2) + X_1(k-3)\right] + G_2\left[X_1(k-1) - X_1(k-2)\right] + G_3 X_1(k-1) \tag{A-6}$$

The phase process of the incoming signal can be approximated by the following Taylor series expansion:

$$\theta(t) = \theta_0 + \theta_1 t + \frac{\theta_2}{2!}t^2 + \cdots + \frac{\theta_m}{m!}t^m \tag{A-7}$$

A third-order phase-locked loop can track $\theta(t)$ with $m \le 3$, while a second-order loop can only track $\theta(t)$ with $m \le 2$. In the following, we will describe the loop initialization for both second-order and third-order DPLLs, with the third-order DPLL discussed first and the second-order loop discussed as a simplified case from the previous one.

I. The Third-Order DPLL

Let us consider a third-order input phase process given as follows:

$$\theta(t) = \theta_0 + \theta_1 t + \frac{\theta_2}{2!}t^2 + \frac{\theta_3}{3!}t^3 \qquad \text{(A-8)}$$

We can find the sampled phase derivative process $X(n)$ as

$$X(n) \triangleq \left. \frac{d\theta(t)}{dt} \right|_{t=nT_u} = \theta_1 + \theta_2(nT_u) + \frac{\theta_3}{2}(nT_u)^2 \qquad \text{(A-9a)}$$

and its associated first and second-order differences as follows:

$$\Delta X(n) = \theta_2 T_u + \left(n - \frac{1}{2} \right) \theta_3 T_u^2 \qquad \text{(A-9b)}$$

$$\Delta^2 X(n) = \theta_3 T_u^2 \qquad \text{(A-9c)}$$

When the augmented third-order DPLL is used to track this input phase in a noiseless environment, the loop will eventually reach a steady state in which

$$\theta(t) - \hat{\theta}(t) = \phi_{ss} \qquad \text{(A-10)}$$

where ϕ_{ss} is called the steady-state phase error. Let us assume that the loop has reached its steady state at time instance $t = nT_u$, so that the loop filter input becomes the steady-state phase error

$$X_1(k) = \phi_{ss} \qquad \text{(A-11)}$$

for k being in the vicinity of n. In this case, Eqs. (A-4), (A-5), and (A-6) can be further simplified as

$$X_3(n) = C_1 + G_1\phi_{ss} + G_2 \sum_{i=1}^{n-1} X_1(i) + G_3 \left[C_2(n-1) + \sum_{i=1}^{n-1} \sum_{j=1}^{i} X_1(j) \right] \qquad \text{(A-12a)}$$

$$\Delta X_3(n) = G_2\phi_{ss} + G_3 \left[C_2 + \sum_{j=1}^{n-1} X_1(j) \right] \qquad \text{(A-12b)}$$

$$\Delta^2 X_3(n) = G_3\phi_{ss} \qquad \text{(A-12c)}$$

Note that the summations in the above equations involve the history of the $X_1(k)$ that should trace back to the beginning of loop operation, including any possible transient state.

Equation (A-10) suggests that, in the steady state, all the corresponding phase derivatives of the input phase process and the estimated phase process should be equal. By equating Eq. (A-9c) and Eq. (A-12c), we find the steady-state phase error is

$$\phi_{ss} = \frac{\theta_3 T_u^2}{G_3} \tag{A-13}$$

Using Eq. (A-13), we can solve for C_1 and C_2 by equating Eq. (A-9a) with Eq. (A-12a) and Eq. (A-9b) with Eq. (A-12b). It can be found that

$$C_1 = \theta_1 + \theta_2 T_u - \frac{G_1}{G_3}\theta_3 T_u^2 + (n-1)\frac{G_2}{G_3}\theta_3 T_u^2 - \frac{1}{2}(n^2 - 3n + 1)\theta_3 T_u^2$$

$$- G_2 \sum_{j=1}^{n-1} X_1(j) + (n-1)G_3 \sum_{j=1}^{n-1} X_1(j) - G_3 \sum_{i=1}^{n-1}\sum_{j=1}^{i} X_1(j) \tag{A-14}$$

$$C_2 = \frac{T_u}{G_3}\left[\theta_2 + \left(n - \frac{1}{2}\right)\theta_3 T_u - \frac{G_2}{G_3}\theta_3 T_u\right] - \sum_{j=1}^{n-1} X_1(j) \tag{A-15}$$

Also, from Eqs. (A-10) and (A-13), it is clear that

$$C_0 = \theta_0 - \frac{\theta_3 T_u^2}{G_3} \tag{A-16}$$

Ideally, the transient state can be eliminated by properly setting the parameters C_0, C_1, and C_2 so that, immediately as the loop begins its operation, the loop is in the steady state, namely,

$$X_1(k) = \phi_{ss} \quad \text{for every} \ \ k \geq 0 \tag{A-17}$$

Equation (A-17) suggests that the loop can be initialized so that it starts with its steady state by setting C_0 as given in Eq. (A-16) and, based on Eqs. (A-14) and (A-15),

$$C_1 = \theta_1 + \theta_2 T_u - \frac{G_1}{G_3}\theta_3 T_u^2 + \frac{1}{2}\theta_3 T_u^2 \tag{A-18}$$

$$C_2 = \frac{T_u}{G_3}\left[\theta_2 + \frac{1}{2}\theta_3 T_u - \frac{G_2}{G_3}\theta_3 T_u\right] \tag{A-19}$$

II. The Second-Order DPLL

As mentioned previously, the second-order DPLL is exactly the same as the third-order DPLL except that one of the loop filter branch gains, i.e., G_3, is set to be 0. Therefore, the augmented second-order DPLL has only two parameters, C_0 and C_1, to be set in the loop initialization. Accordingly, a second-order input phase process that can be tracked by a second-order DPLL has $\theta_3 = 0$.

Following the same argument about the steady state of the loop as given before, the second-order DPLL steady-state phase error can be found by equating Eq. (A-9b) with Eq. (A-12b) and setting both G_3 and θ_3 to be 0, as

$$\phi_{ss} = \frac{\theta_2 T_u}{G_2} \tag{A-20}$$

Thus, C_1 can be solved by equating Eq. (A-9a) with Eq. (A-12a) and using Eq. (A-20) as

$$C_1 = \theta_1 + n\theta_2 T_u - \frac{G_1}{G_2}\theta_2 T_u - G_2 \sum_{i=1}^{n-1} X_1(i) \tag{A-21}$$

and C_0 is found from Eqs. (A-10) and (A-20) as

$$C_0 = \theta_0 - \frac{\theta_2 T_u}{G_2} \tag{A-22}$$

This loop can be initialized so that it starts with its steady state by setting C_0 as given in Eq. (A-22) and

$$C_1 = \theta_1 + \theta_2 T_u - \frac{G_1}{G_2}\theta_2 T_u \tag{A-23}$$