Data-Bits Asynchronous Tracking Loop Scheme for High Performance Real-Time GNSS Receivers

Pedro A. Roncagliolo, Javier G. García and Carlos H. Muravchik
Laboratorio de Electrónica Industrial, Control e Instrumentación (LEICI),
Dto. Electrotecnia, Facultad Ingeniería, UNLP, La Plata, Argentina.
Emails: {agustinr, jgarcia, carlosm}@ing.unlp.edu.ar

***Abstract*—Tracking loops are very often chosen for code and carrier phase estimation in real-time Global Navigation Satellite System receivers due to their low computational complexity. The inputs to these loops are obtained from correlations of the received signal with locally generated replicas. Usually, correlation intervals are chosen synchronously with the data-bits sent with each satellite signal. As a consequence, each loop operates at its own time and the navigation task must extrapolate loop measurements to a common instant. We propose to change this philosophy using a common correlation interval for every satellite signal. In this way, the tracking loops work in synchronism with the navigation process, rather than with the data bits. We show how to account for the occurrence of bit transitions inside a correlation interval and how to derive a suitable discriminator for phase and code errors. The performance of this discriminator is very close to that obtained with the usual bit-synchronous correlations. The proposed scheme was applied to a scalar phase lock loop structure intended for high dynamics Global Navigation Satellite System receivers. The loop is shown to have almost the same tracking threshold and phase estimation quality than those working bit-synchronously. However, control of the measurement instant can produce significant improvements in phase estimation. Moreover, the main contribution of this scheme is for the implementation of real-time vector tracking loops, since it naturally generates a vector of simultaneous measurements in real-time.**

***Keywords*—GNSS; Real-Time Receivers; Phase Locked Loops; Vector Tracking Loops.**

I. Introduction

Measuring the propagation delay of the broadcasted signals is the key of the position calculations made in every modern Global Navigation Satellite System (GNSS) receiver. For this purpose, the receiver has to be synchronized with the visible satellite signals. Direct Sequence Spread Spectrum (DS-SS) signals are utilized due to their desired properties of high time-resolution and Code Division Multiple Access (CDMA) and therefore code and carrier synchronization are required [1]. A correlation stage is also needed at the receiver to de-spread the incoming signals so that the synchronization and navigation algorithms can operate with reasonable signal to noise ratios. The required economy of operations in real-time receivers makes impractical the use of complex estimation schemes and usually tracking loop schemes are adopted for synchronization purposes. Phase measurements are considerably less noisy than code delay and so, code loops are usually aided by carrier loops [2]. However, the signal phase is affected by the wavelength ambiguity and hence the basic measurement used for standard position determination is code delay. On the contrary, the techniques used in high precision positioning applications usually take advantage of the phase measurements. In general, code delay and carrier phase or frequency measurements used by the GNSS receiver for position and velocity determination are referred as navigation measurements or raw track data.

Typically, the GNSS signal has also a data structure to send useful information to the receivers, such as orbit parameters needed for satellite position calculations, clock corrections, ionospheric corrections, signal quality indexes, etc. The bits carrying this information are modulated usually in phase, and of course the receiver has to be able to demodulate them. The presence of these data-bits imposes restrictions to the receiver operation from the point of view of navigation measurements generation. Indeed, the correlation time is, in principle, limited to the bit duration time and the corresponding signal to noise ratio increase due to despreading gain is limited too. In some applications this is not a limitation at all, but in others, such us indoor positioning, the use of some long-correlation techniques is unavoidable [3]. Moreover, since different satellite signals experience different propagation delays, the edges of these bits are in general asynchronous. As a consequence, the correlation intervals used for each signal satellite are also asynchronous. In standard real-time receivers, this causes that the tracking loops for each satellite operate synchronously with the bit edges, but asynchronously among them. For the navigation process, this implies that the measurements do not correspond to the same time instant and the receiver has to extrapolate them [1], [2]. These lag differences make it difficult to take advantage of the correlation between the received signals, since each signal is tracked independently.

The convenience of joint tracking the signals by means of the so-called vector tracking loops, has been envisioned since the conception of the GNSS systems [1]. Nowadays, due to their potential advantages together with the growing computation capacity available in a GNSS receiver, many researchers and developers, are considering vector tracking loop schemes. These loops can obtain up to 6 dB of improvement in tracking threshold, in addition to high dynamic capacity, multipath immunity and robustness [4]. Vector tracking loops have been mainly applied in Software-based receivers [5], [6]. Recently, a real-time implementation using Field-Programmable-Gate-Arrays (FPGA) with a fast microprocessor has been reported in [7]. This implementation operates with asynchronous

 ISBN: 978-1-61208-194-6

correlations of the different signals, either extrapolating the navigation measurements or asynchronously incorporating the measurements to the main processing algorithm. Other off-line implementations use data bit removal in order to get simultaneous navigation measurements [8]. In this work, we make a different and novel approach, which is based on the use of synchronous correlations for the received satellite signals so that the navigation measurements are naturally simultaneous. As a consequence, the tracking loops operate asynchronously with respect to the bit edges of the signals and their inputs, i.e., the code and carrier phase errors, have to be calculated for signal intervals with a possible bit-transition inside. Our approach is simple: compute partial correlations before and after the bit edge and calculate a discriminated error based on them. By means of simulations made with scalar tracking loops, we show that this scheme offers some improvements in the measurement quality in high-dynamics conditions, and also that the degradation in the tracking threshold is less than 0.5dB compared with a bit-synchronous loop. This value is completely insignificant compared to the potential gain of using a vector loop in real-time.

The rest of paper is organized as follows. A digital model for the received GNSS signal is presented in Section 2. Since the emphasis on this work is on phase loops, our UFA-PLL scheme will be briefly explained. The proposed phase error discriminator for the correlation periods with possible bit-transition is presented in section 3. The extension of the idea to the code loops, which is straightforward, is also shortly discussed. The bit-asynchronous scheme is applied to a high-dynamic scalar carrier tracking loop and the effects in its performance are analyzed in terms of phase measurements quality and tracking threshold in section 4. Finally, the conclusions and future work lines are given in section 5.

II. Digital Measurements Model

As stated above, the received signal must be correlated with the locally generated replicas for each visible satellite in a GNSS receiver. The complex correlations of the signal from a given satellite with carrier power to noise power spectral density C/N_0 and for the i-th correlation interval of duration T can be expressed as [1]

$$C_i = D_i\sqrt{T\frac{C}{N_0}}\mathrm{sinc}(\Delta f_i)R(\Delta\tau_i)e^{j(\pi\Delta f_iT+\Delta\theta_i)} + n_i \quad (1)$$

where $\Delta\tau_i = \tau_i - \hat{\tau}_i$ is the code delay estimation error, $\Delta f_i = f_i - \hat{f}_i$ the frequency estimation error, both assumed constant during the integration time, and $\Delta\theta_i = \theta_i - \hat{\theta}_i$ the initial phase estimation error. The term n_i is a complex white Gaussian noise sequence with unit variance, $R(\cdot)$ is the code correlation function and $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$. This expression assumes that there are binary data bits $D_i = \pm 1$ and that correlations are computed within the same bit period. This type of modulation, i.e., Binary Phase Shift Keying (BPSK), is used in the GPS civil signal and in the data components of composite modernized GNSS signals.

After the acquisition process has been completed, i.e., in tracking conditions [1], code and frequency estimation errors are sufficiently small so that the functions $\mathrm{sinc}(\cdot)$ and $R(\cdot)$ can be approximated by 1. Hence, (1) becomes

$$C_i = I_i + jQ_i = D_i\sqrt{T\frac{C}{N_0}}e^{j\Delta\phi_i} + n_i \quad (2)$$

where we have defined $\Delta\phi_i = \phi_i - \hat{\phi}_i$, with $\phi_i = \pi f_iT + \theta_i$ and $\hat{\phi}_i = \pi\hat{f}_iT + \hat{\theta}_i$. With the help of these sequences the carrier tracking loop can be modeled as a digital single-input single-output (SISO) system. It is important to note that Δf_i and $\Delta\phi_i$ can be interpreted as the average frequency error and average phase error during the correlation interval respectively.

The phase estimation error is obtained from the angle of the complex correlation. In the case of BPSK modulation the phase error must be insensitive to the bit changes and a two quadrant discriminator should be utilized. Then,

$$e_i = \tan^{-1}\left(\frac{Q_i}{I_i}\right) = [\Delta\phi_i + n_{\phi_i}]_\pi \quad (3)$$

where the notation $[\cdot]_\pi$ indicates that its argument is kept within the interval $(-\frac{\pi}{2}, \frac{\pi}{2}]$ by adding or subtracting π as many times as needed. The noise term $n_{\phi i}$ has zero mean and a complicated probability distribution in general. However, in high C/N_0 conditions it can be approximated by a Gaussian distribution with zero mean and variance $1/(2TC/N_0)$.

A. UFA Phase Discriminator

The Unambiguous Frequency Aided (UFA) algorithm uses the frequency error information to correct the non-linearity of a Phase Locked Loop (PLL), instead of adding a Frequency Locked Loop (FLL) to cope with high dynamics. Thus, the advantages of a frequency loop are added to the PLL obtaining the same dynamic tolerance of an FLL but also avoiding cycle slips during tracking [9]. The UFA phase discriminator works correcting the ambiguous values of e_i by adding or subtracting an integer number of π. The correction is such that the difference between successive values of the corrected phase error u_i is less than a quarter of a cycle in magnitude. Then, the corrected phase error estimate, with starting value $u_0 = e_0$, is

$$u_i = e_i - I_\pi(e_i - u_{i-1}) \quad (4)$$

where $I_\pi(x) = x - [x]_\pi$ acts similarly to the integer part function, but with steps at the multiples of π. Created in this way, the sequence phase errors u_i has unambiguous values as long as the loop frequency error is lower than $1/(4T)$ in magnitude, i.e., half of the Nyquist rate from uniform sampling theory. Under this condition, the sequence u_i allows to measure the loop frequency error with a simple difference of successive phase errors, giving to the UFA-PLL the same extra-information that usually has an FLL but not a PLL. In previous works we have also shown that the UFA-PLL has the same noise resistance, and so the same tracking threshold, that an equivalent FLL [10].

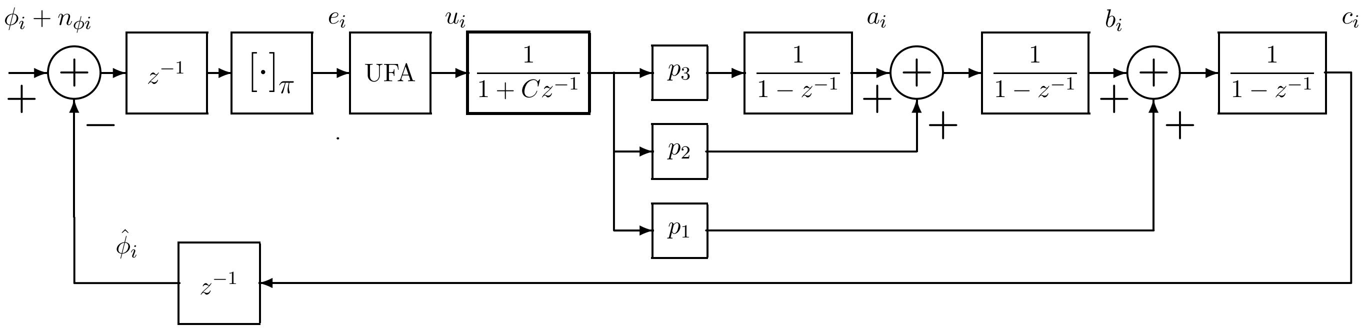

Figure 1. Block diagram of the UFA-PLL model.

III. Bit Asynchronous Phase Discrimination

Assume the receiver is tracking a given satellite and it knows when a data bit edge will occur during a correlation interval. This requires that a bit synchronization stage has been completed previously. This is not a limitation since the required signal strength for tracking at the high dynamics considered in this work must be high enough to detect bit transitions. For the same reason, multiple data-bits long correlation intervals will not be considered. However, notice that the receiver will not use bit transitions to synchronize the correlation intervals. In our scheme, the receiver uses them to compute the code and phase errors as described in the following and the correlation intervals are dictated by the navigation task. Specifically, assume for the i-th correlation interval of duration T the bit edge will occur T_1 seconds after the beginning and T_2 seconds before its end. Clearly, $T_1 + T_2 = T$. In that case, a coherent correlation of T seconds will not be effective since the possible change of phase will produce a signal cancelation. The worst case when there is a bit reversal is $T_1 = T_2 = T/2$, where a complete signal cancelation occurs. Therefore, the receiver should compute two partial correlations, namely C_1 and C_2. The corresponding phase errors, obtained as in (3), are

$$e_1 = \tan^{-1}\left(\frac{Q_1}{I_1}\right) = [\Delta\phi_1 + n_{\phi_1}]_\pi \tag{5}$$

$$e_2 = \tan^{-1}\left(\frac{Q_2}{I_2}\right) = [\Delta\phi_2 + n_{\phi_2}]_\pi \tag{6}$$

where $\Delta\phi_1 = \Delta\theta_i + \pi\Delta f_i T_1$ and $\Delta\phi_2 = \Delta\theta_i + 2\pi\Delta f_i T_1 + \pi\Delta f_i T_2$ according to the assumed linear evolution of the phase error. Leaving aside for a moment the nonlinearity of the $\tan^{-1}(\cdot)$ function, we can think that these two phase errors are partial averages and therefore they should be averaged to obtain the desired phase error for the i-th correlation interval. The weighted average of them according to the duration of each correlation should be

$$e_i = \frac{T_1}{T}e_1 + \frac{T_2}{T}e_2 \approx \Delta\theta_i + \pi\Delta f_i T + n_{eq} = \Delta\phi_i + n_{eq} \tag{7}$$

with n_{eq} equal to the weighted average of n_{ϕ_1} and n_{ϕ_2}. Under the Gaussian approximation for both noise terms, n_{eq} has a Gaussian distribution with zero mean and variance $1/(2TC/N_0)$. That is, the same variance as if the bit edge was not present. Of course, if T_1 or T_2 are not long enough the approximation is not valid, and we still have to deal with the nonlinearity of the $\tan^{-1}(\cdot)$ function.

The issue about the nonlinearity is caused by the ambiguity of the phase, indicated by the function $[\cdot]_\pi$. If this operation acts after the weighted average we would obtain a result equivalent to (3) for the correlation interval with a bit transition. However, in (7) the $[\cdot]_\pi$ function actually acted before the average, and then (7) is not correct. Fortunately, the same idea used to build the UFA algorithm can be applied here to test the result and correct it when needed. The hypothesis is that the frequency error is kept under $1/(4T)$ in magnitude. Hence, the signal part of a difference between the partial phase errors in (5) must be bounded. Indeed, (5) can be written as

$$e_1 = \tan^{-1}\left(\frac{Q_1}{I_1}\right) = \Delta\phi_1 + n_{\phi_1} + k_1\pi \tag{8}$$

$$e_2 = \tan^{-1}\left(\frac{Q_2}{I_2}\right) = \Delta\phi_2 + n_{\phi_2} + k_2\pi \tag{9}$$

with $k_1, k_2 \in \mathbb{Z}$. Then, different values of k_1 and k_2 will produce a wrong result at the average (7). This situation has to be detected, and a simple hypothesis test can be built. The decision variable is

$$e_1 - e_2 = \Delta\theta_1 - \Delta\theta_2 + n_d + k_d\pi = \pi\Delta f_i T + n_d + k_d\pi \tag{10}$$

where $n_d = n_{\phi_1} - n_{\phi_2}$ and $k_d = k_1 - k_2$. Since $|\Delta f_i T| < 1/4$ and n_d is a zero mean symmetrically distributed noise term, the optimum decision for the k_d value is $\hat{k}_d = I_\pi(e_1 - e_2)/\pi$. Notice that the possible values for k_d are only three: -1, 0 and 1. Then, if $\hat{k}_d = 0$ no correction is needed and (7) can be applied directly. If $\hat{k}_d \neq 0$ either e_1 or e_2 have to be corrected. Which one is not important since the π ambiguity of the e_i value will be solved later by the UFA algorithm. For simplicity, assume e_2 is corrected when $\hat{k}_d \neq 0$. Then, the final expression for the phase error is

$$e_i = \frac{T_1}{T}e_1 + \frac{T_2}{T}\left\{e_2 + I_\pi(e_1 - e_2)\right\}. \tag{11}$$

The computational cost of the new scheme is only one more phase error calculation each time a bit transition could be present, plus the weighted average. The logic needed for the last π ambiguity correction can be neglected compared with the cost of angle calculations and multiplications. Naturally, the idea of the weighted average for combining the discriminated errors from partial correlations in the presence of a

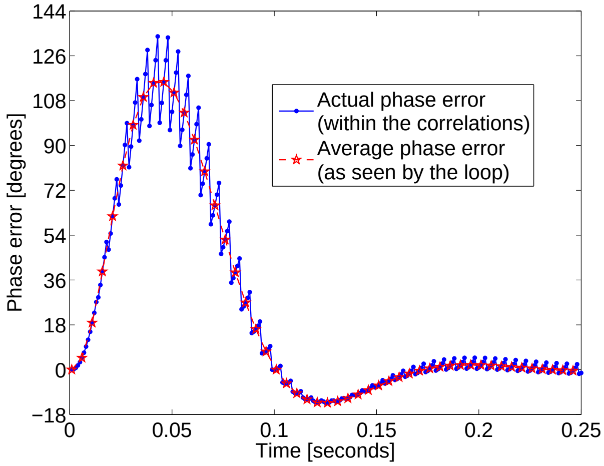

Figure 2. Phase estimation error during a step of 20g.

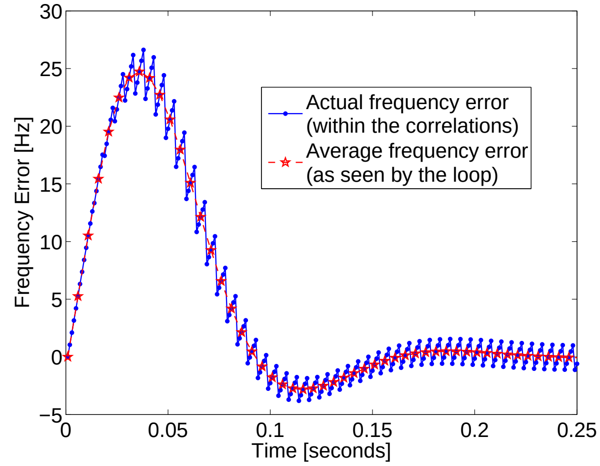

Figure 3. Frequency estimation error during a step of 20g.

bit transition can be also applied for the case of code delay. Significant changes of the code delay are not expected during a correlation interval, and in the case of high dynamics the code loop is aided by the carrier loop and then only the residual and low code dynamics is tracked. Thus, the code delay error will not evolve significantly during the two partial correlations, and the significant effect will be only to reduce the noise to the same level obtained with a correlation time T. In addition, the delay discriminators are not ambiguous and then no further corrections are needed to the average.

IV. Application to a Scalar Carrier Loop

In this section, the proposed bit-asynchronous scheme is applied to a specific carrier tracking loop. We chose a digital UFA-PLL as shown in Figure 1 whose filter coefficients are $C = 0.5$, $p_1 = C = 0.5$, $p_2 = 0.105$, and $p_3 = 0.0123$. For the selected correlation time, $T = 5$ms, the resulting PLL has an equivalent noise bandwidth $B_N = 75.6$Hz. Notice that two delays are included in the loop model. One of them is due to the time spent in computation of the correlation. The other delay appears because the estimated values used to compute the correlations have to be known before the calculations begin. That is, the value $\hat{\phi}_i$ is obtained with the loop filter output of the $(i-1)$-th correlation interval, which in turn is calculated with the estimation errors of $\hat{\phi}_{(i-2)}$. The loop filter is optimized for the tracking of acceleration steps, which produces a quadratic ramp of phase at the loop input. These demanding high dynamics scenarios can be found for example in sounding rockets, at engine turn-on and turn-off. This loop design has been implemented in experimental GPS receivers [9]. According to the analysis made in [11] this design is almost optimal for tracking steps of 20g, in the sense that for a given C/N_0 it approximately produces the smallest pull-out probability.

In order to consider the effects of asynchronous bit transitions without increasing the simulation time excessively, a time step of 1ms was selected. This implies a quantization of the transition times to 5 possible values within a correlation interval of $T = 5$ms. During each correlation interval, partial 1ms correlations are computed according to (1). Then, these values are added to form the two partial correlations if there is a bit transition, or a single correlation if not. The phase estimate used for each middle instant of this period is $\hat{\phi}_i = c_{(i-1)}$, and the frequency estimate for this whole 5 ms period is $\hat{f}_i = b_{(i-1)} + a_{(i-1)}/2$. This frequency estimate was chosen because, as can be seen in the following simulations, it has zero stationary error for acceleration steps. In the following, the influence of the synchronization of the tracking loop with the data-bits in the quality of the navigation measurements is analyzed first. This analysis shows that even for a GNSS receiver with scalar tracking loops, the proposed bit-asynchronous scheme is beneficial. The more important results are given in the second subsection, where the pull-out probabilities of the same loop, operating in a bit-synchronous and in a bit-asynchronous mode are presented.

A. Effects on Phase Estimation Quality

The simulation of the phase evolution with a higher sampling rate than the loop iteration allows us to quantify the quality of phase measurements obtained from the tracking loops. If the loop operates synchronously with the data bits, it cannot be synchronous with the navigation process in general. Therefore, the measurement instants can occur at any instant during a correlation interval, not necessarily in the middle. As an example, consider the phase estimation error produced by the loop for an acceleration step of 20 g without noise, plotted in Figure 2. Clearly, the phase error is not constant during each correlation period. In fact, since the estimated carrier has constant frequency for each period the loop fits the incoming phase with a piecewise linear approximation. Hence, a residual quadratic ramp of phase appears as an estimation error. Notice that the level of error is lower if the relative location within the correlation interval is close to the middle. The same situation is found for the frequency estimation, plotted in Figure 3. Here, the residual error is a linear ramp inside each interval.

As long as pull-out or cycle slips do not occur, the loop response is linear and the noise effects can be analyzed separately. As an example, the loop response to noise only

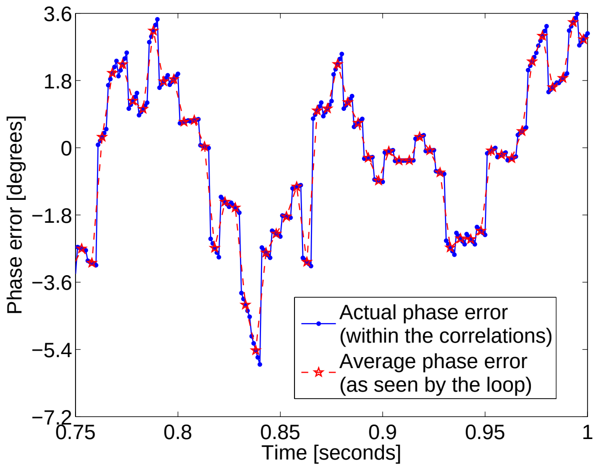

Figure 4. Phase estimation error with C/N_0 =48 dB/Hz.

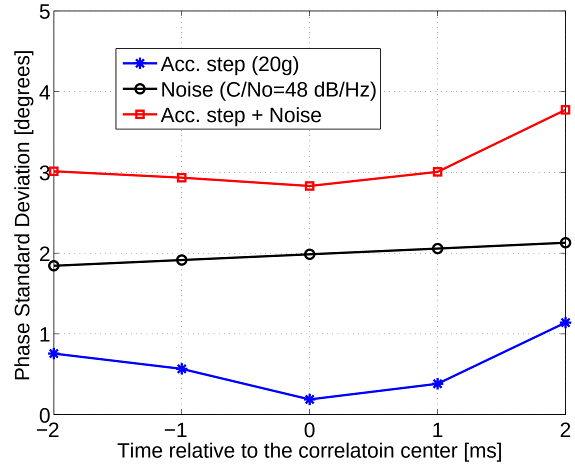

Figure 5. Phase estimation error within the correlation interval.

with $C/N_0 = 48$dB/Hz is plotted in Figure 4. In this case, the resultant phase error seems to be lower if the relative location within the correlation interval is close to the beginning. This effect can be understood if we notice that in fact, the loop calculates a carrier prediction for the following correlation interval based on the available measurements. And as the prediction time grows, so does the noise variance of this prediction. Signal and noise results are summarized in Figure 5 where each contribution to the phase standard deviation is plotted for the different relative location within the correlation period. The signal parts correspond to the time average of a 1-second run like that shown in Figure 2. The noise variance was estimated with an average of 1000 runs of 1 second with only Gaussian noise of variance $\frac{1}{2TC/N_0}$ as input, and discarding the first 0.15 seconds to avoid the main part of the initial loop transient. It is possible to verify that the standard deviation for the loop output, i.e., at the middle point, is the same as that obtained with the equivalent noise bandwidth of the loop and the input noise variance. Indeed,

$$\sigma^2_{\hat{\phi}_i} = \frac{1}{2TC/N_0} \times 2B_N T = \frac{B_N}{C/N_0} \qquad (12)$$

The expression gives $\sigma_{\hat{\phi}_i} = 1.98^{\text{o}}$ when $C/N_0 = 48$dB/Hz, as seen in ordinates of Figure 5. In summary, the phase estimation quality changes depending on the time when the navigation measurement is taken during the correlation interval. In the presented example better estimates are obtained if the measurements are taken at the middle, and this can only be done if the loop is synchronous with the navigation process and therefore it is bit-asynchronous. In a bit-synchronous loop, when the measurement instant is taken next to the end of the correlation interval the increase in noise variance is $(3.78^{\text{o}}/2.83^{\text{o}})^2 \approx 1.78$, i.e., 78%. Another 1000 runs of the same loop operating according to the bit-asynchronous scheme, with the bit transition location chosen randomly at each run, reveals that the noise output variance is almost not affected, since its increase is less than 0.5% and when plotted looks the same as in Figure 5.

B. Effects on Tracking Threshold

In this section the non-linear performance of the proposed bit-asynchronous scheme for different acceleration and signal levels is determined by means of simulation. Main consequences of this non-linear behavior are cycle slips and pull-out events, i.e., to lose lock with the tracked signal. If a cycle slip occurs, it will produce a loop transient that could end with a pull-out event or not. This temporary loss of phase lock can degrade the data bit demodulation, but as long as the frequency error is low enough useful navigation measurements can be generated. Actually, since the expression (1) was used for the calculation of 1ms correlations used in the simulations, a frequency error also produces a signal power reduction due to the sinc$(\cdot)$ function factor. Therefore, the adopted criterion to declare a pull-out was that the frequency error exceeds $1/T = 200$ Hz. In this situation, the signal power is completely attenuated and then it can be considered as a practically irrecoverable state. An error of less than 200 Hz is a critical situation but it could still be recoverable. For each value of acceleration and C/N_0, 100,000 runs of 1 second (200 samples) were computed. Each run has an acceleration step of the selected value at the beginning. Two UFA-PLLs were simulated for comparison. One operates synchronously with the data-bits whereas the other implements the proposed scheme according to (11). Using the criteria mentioned above, runs that presented a pull-out event were detected and the pull-out probability (POP) estimated. Since the POP is computed for 1 second of tracking it can be also interpreted as the inverse of the mean-time to lose lock (MTLL) in seconds. The results are presented in Figs. 6 and 7 where POP level curves have been plotted for values of 0.1, 0.2 and so on. Defining the tracking threshold when the POP reaches a level of 0.1, as is usually done, it is interesting to note that the proposed scheme exhibits practically the same threshold than the synchronous loop. Actually, comparing Figs. 7 and 6 it can be clearly seen that the difference between them is always less than 0.5 dB.

 ISBN: 978-1-61208-194-6

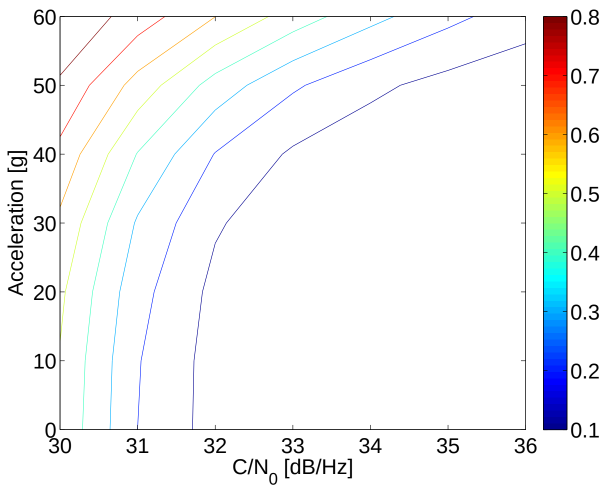

Figure 6. POP of bit-asynchronous loop and its tracking threshold.

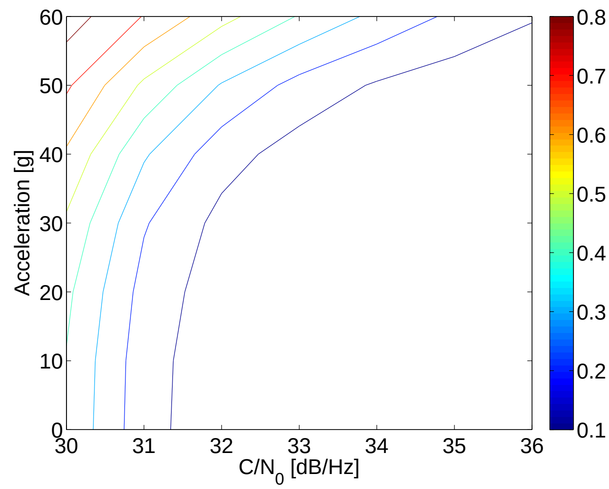

Figure 7. POP of bit-synchronous loop and its tracking threshold.

V. Conclusion and Future Work

An efficient bit-asynchronous tracking loop scheme for high performance real-time GNSS receivers has been presented. In the proposed scheme, the loop operation is synchronous with the navigation measurement generation process, rather than with the data-bits. The effect of a possible bit-transition inside the correlation interval is managed by the calculation of two partial correlations. We devised how to build a phase error discriminator based on these partial correlations and explained it in detail. The application of the same procedure for code delay error was briefly discussed. The proposed scheme was applied to a UFA-PLL intended for high dynamic GNSS receivers. We found that it produces an almost negligible impact on tracking threshold (< 0.5 dB) and estimation phase noise ($< 0.5\%$). However, the measurement instants of the tracked signals do not need to be extrapolated to a common instant and therefore a significant improvement can be obtained. An example was shown where a phase noise variance increase of up to 78% can be avoided controlling the measurement instant. Nevertheless, the new scheme main contribution is for the implementation of vector tracking loops in real time, since it will allow operating with a simultaneous vector of measurements from the received satellite signals obtained from correlations computed at a common time.

In terms of computational cost, there is some increase due to the calculation of the partial correlations and ensuing error discrimination. However, the operation of the different satellite tracking loops with the same timing can reduce the processor load depending on the adopted hardware/software architecture. In this case, a correlation stage capable of computing the two partial results for the same receiver estimates when the possible transition location is fed as an extra parameter could be very beneficial for a real-time implementation. The authors are now working on that correlator architecture to be implemented in FPGA, and in the vector tracking loop formulation in real-time with this philosophy.

VI. Acknowledgment

This work was funded by ANPCyT PICT 00535 and UNLP 11-I-127 and CIC-PBA.

References

[1] B. W. Parkinson and J. J. Spilker (eds.), *Global Positioning System: Theory and Applications*. Washington: American Institute of Aeronautics and Astronautics (AIAA), 1996.

[2] E. D. Kaplan, *Understanding GPS: Principles and Applications*. Boston: Artech House, 1996.

[3] van Diggelen, F., "Indoor GPS theory & implementation," in *Proceedings of The Position Location and Navigation Symposium (PLANS), 2002 IEEE/ION*, Palm Springs, CA , USA, August 2002, pp. 240 – 247.

[4] M. Lashley, D. M. Bevly, and J. Y. Hung, "A valid comparison of vector and scalar tracking loops," in *Proceedings of The Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, Indian Wells, CA, USA, May 2010, pp. 464 – 474.

[5] M. Lashley, D. M. Bevly, and J. Y. Hung, "Performance analysis of vector tracking algorithms for weak gps signals in high dynamics," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 4, pp. 661 – 673, August 2009.

[6] S. Zhao, D. Akos, "An Open Source GPS/GNSS Vector Tracking Loop - Implementation, Filter Tuning, and Results," in *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation, ION ITM 2011*, San Diego, CA, USA, January 2011, pp. 1293 – 1305.

[7] W. L. Edwards, M. Lashley, and D. M. Bevly, "FPGA Implementation of a Vector Tracking GPS Receiver using Model-Based Tools," in *Proceedings of the 22nd International Technical Meeting of The Satellite Division of the Institute of Navigation,ION GNSS 2009*, Savannah, GA, USA, September 2009, pp. 273 – 280.

[8] T. Lin, J. T. Curran, C. O´Driscoll, and G. Lachapelle, "Implementation of a Navigation Domain GNSS Signal Tracking Loop," in *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation, ION GNSS 2011*, Portland, Oregon, USA, September 2011, pp. 3644 – 3652.

[9] P. A. Roncagliolo and J. G. García, "High Dynamics and False Lock Resistant GNSS Carrier Tracking Loops," in *Proc. of The 20th Int. Technical Meeting of The Satellite Division of The Institute of Navigation, ION GNSS 2007*, Fort Worth, Texas, September 2007, pp. 2364 – 2375.

[10] P. A. Roncagliolo, J. García, and C. H. Muravchik, "Pull-out Probability and Tracking Threshold Analysis for High Dynamics GNSS Carrier Loops," in *Proceedings of The 21st International Technical Meeting of The Satellite Division of The Institute of Navigation, ION GNSS 2008*, Fort Worth, Texas, USA, September 2008, pp. 221 – 228.

[11] P. A. Roncagliolo, J. G. García, and C.H. Muravchik, "Pull-out Probability Considerations in High Dynamics GNSS Tracking Loops Design," in *Proceedings of The 10th International Symposium on Spread Spectrum Techniques and Applications, ISSSTA 2008*, Bologna, Italy, August 2008, pp. 53 – 57.