Carrier Phase Discrimination for a Common Correlation Interval GNSS Receiver Architecture

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*—Basic measurements of global navigation satellite system receivers are obtained after the correlation of the incoming signal with locally generated replicas. Usually, correlation intervals are chosen synchronously with the data-bits sent with each satellite signal to avoid bit transitions. As a consequence, the ensuing code and carrier phase estimation signal processing operates at its own time and the navigation task must extrapolate loop measurements to a common instant. We have proposed an alternative receiver architecture using a common correlation interval for all satellite signals. Under this scheme, the correlations made for each satellite in-view have a common interval, chosen in synchronism with the navigation process rather than with the data bits. Naturally, now the bit transitions within a correlation interval require special treatment. The advantages of avoiding measurement extrapolation are shown with a scalar phase lock loop structure intended for high dynamics real-time receivers. The operation of this loop in a common correlation interval receiver needs a carrier phase discriminator structure able to produce outputs for the correlation with bit transitions inside the interval. Three possible carrier discriminator schemes are analyzed in this work. It is shown that the loops operating data-bit asynchronously with any of these schemes have similar tracking threshold and phase estimation quality than those working bit-synchronously. The proposed architecture naturally generates a vector of simultaneous measurements and then it is particularly suited for the implementation of real-time vector tracking loops.**

Keywords- Phase Discrimination; GNSS; Real-Time Receivers; Digital Phase Locked Loops; Vector Tracking Loops.

I. INTRODUCTION

Measuring the propagation delay of broadcast signals is the key to 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 [2]. 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 [3]. 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 to 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 as indoor positioning, the use of some long-correlation techniques is unavoidable [4]. 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 [2], [3]. 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 [2]. 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 [5]. Vector tracking loops have been mainly applied in software-based receivers [6], [7]. Recently, a real-time implementation using field-programmable-gate-

arrays (FPGA) with a fast microprocessor has been reported in [8]. This implementation operates with asynchronous 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 [9].

In a previous work, we proposed 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 [1]. 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. The suggested approach is to compute partial correlations before and after the bit edge and calculate a discriminated error based on them. In this work we analyze particularly three different carrier phase discriminators based on this partial correlations. In high signal-to-noise ratio the three schemes show similar performance, which is also similar to the bit synchronous tracking loop results. The main difference between them is their non-linear characteristic, which is evidenced in low signal-to-noise and/or high dynamic situations. This aspect of the discriminators is quantified by means of the resulting tracking threshold, i.e., the lowest signal-to-noise ratio the tracking loop can operate at, computed for different dynamic scenarios.

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 discriminator, our unambiguous frequency-aided phase-locked loop (UFA-PLL) scheme will be briefly explained [10]. The advantages of avoiding measurements extrapolation are shown with this tracking loop structure intended for high dynamics real-time receivers. The three phase error discriminators schemes under consideration for the correlation periods with possible bit-transition are presented in Section 3. The computational load of each scheme and their extension for code delay discrimination is also shortly discussed. The considered phase discriminators are applied to UFA-PLL structure in Section 4, and their effects in the phase measurements quality and tracking threshold are determined by means of Monte Carlo simulations. One of the schemes show better performance for low dynamics situations but its performance is severely degraded in high dynamics. In the other two cases, the degradation in the tracking threshold is less than 0.5dB compared with a loop that works with known data-bits, for all the dynamic scenarios considered. Curves of pull-out probability of the different schemes are also provided. The small degradation in tracking threshold caused by the data-bits asynchronous operation is completely insignificant compared to the improvement in the quality of phase measurements due to the absence of extrapolation and the common correlation interval architecture. Finally, the obtained conclusions and future work lines are given in Section 5.

II. Digital Measurements Model

In a GNSS receiver, the incoming signal must be correlated with the locally generated replicas for each visible satellite. 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 [2]

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

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 $\text{sinc}(x) = \sin(\pi x)/(\pi x)$. 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 [2], code and frequency estimation errors are sufficiently small so that the functions $\text{sinc}(\cdot)$ and $R(\cdot)$ can be approximated by one. 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 $\phi_i = \pi f_i T + \theta_i$ and $\hat{\phi}_i = \pi\hat{f}_i T + \hat{\theta}_i$. 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. 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 [11], [10]. The UFA phase discriminator

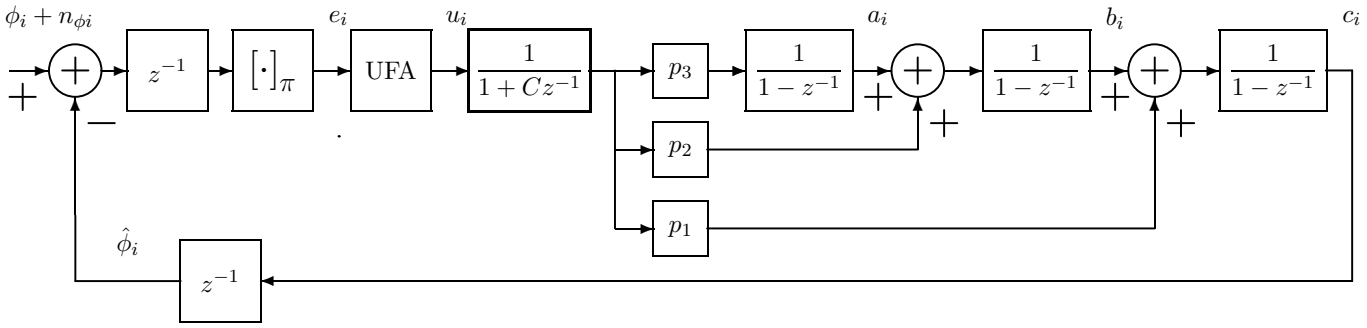

Fig. 1. Block diagram of the UFA-PLL model.

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

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

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 of 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, than an equivalent FLL [12].

B. Carrier Phase Tracking Loop

For the purpose of comparing different correlation intervals approaches, throughout the rest of this work we consider 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 [11]. According to the analysis made in [13] 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. More details and properties of this loop can be found in [10].

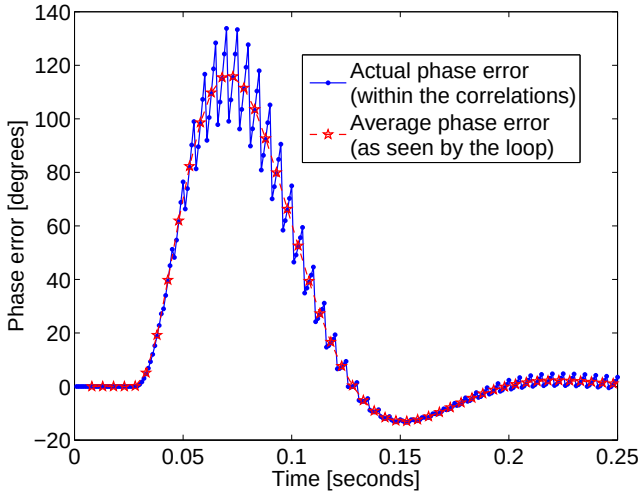

Fig. 2. Phase error during a step of 20g. Loop transient response.

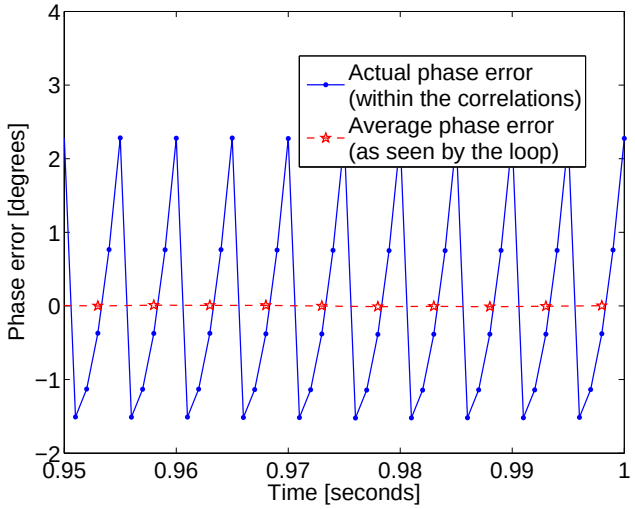

Fig. 3. Phase error during a step of 20g. Loop stationary state response.

This loop is simulated at a higher sampling rate than

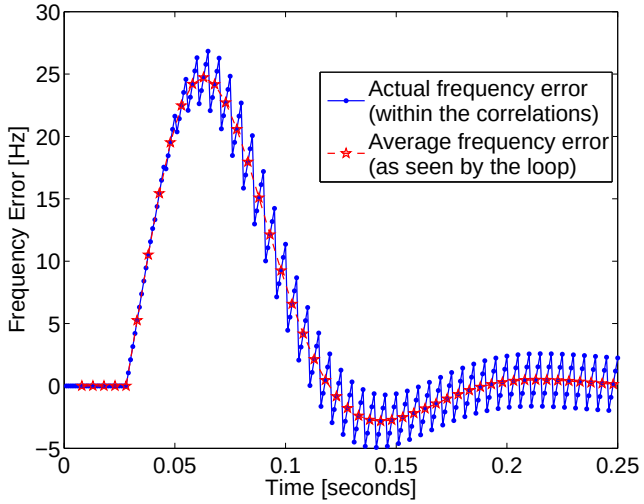

Fig. 4. Frequency estimation error during a step of 20g. Loop transient response.

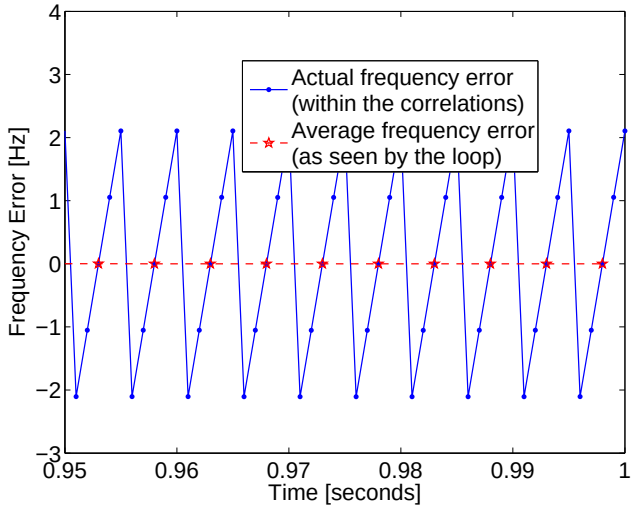

Fig. 5. Frequency estimation error during a step of 20g. Loop stationary state response.

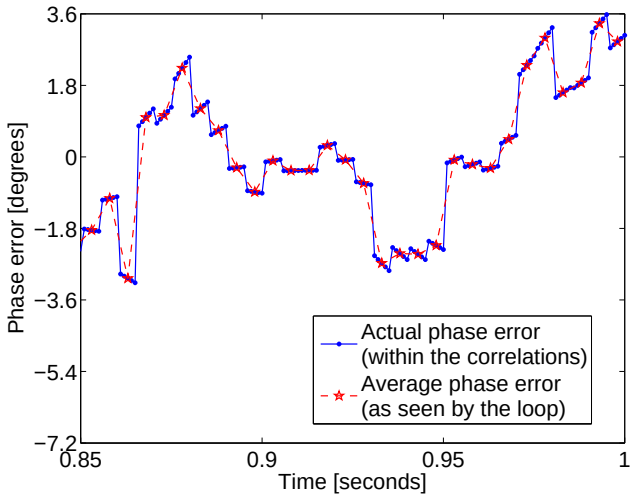

Fig. 6. Phase estimation error with C/N_0 =48 dB/Hz.

the loop iteration in order 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. The phase measurement extrapolation made by the navigation process is done based on the same information that the loop has about the phase dynamic. Typically, if the measurement instant is not the middle of the correlation interval, the loop frequency estimate for this interval is used to linearly extrapolate this value. The phase measurement obtained in this way, is the same that the instant phase error inside the correlation interval because the carrier local replica is generated with the same logic. In order to quantify the effects of navigation measurements extrapolation without increasing the simulation time excessively, a time step of 1ms was selected. This implies a quantization of the measurement times to 5 possible values within a correlation interval of $T = 5$ms. 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.

As an example, consider the phase estimation error produced by the loop for an acceleration step of 20 g without noise. The result is plotted in Figures 2 and 3 to appreciate the estimation error inside the correlation interval. 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. Even when the digital loop has zero stationary state response to acceleration steps, the error in the middle of the correlation interval is not zero because the loop works with the average phase error of each interval. The same situation is found for the frequency estimation, plotted in Figures 4 and 5. Here, the residual error is a linear ramp inside each interval, and then average error equal to zero produce zero error in the middle of the 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 with $C/N_0 = 48$dB/Hz is plotted in Figure 6. In this case, the resulting 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

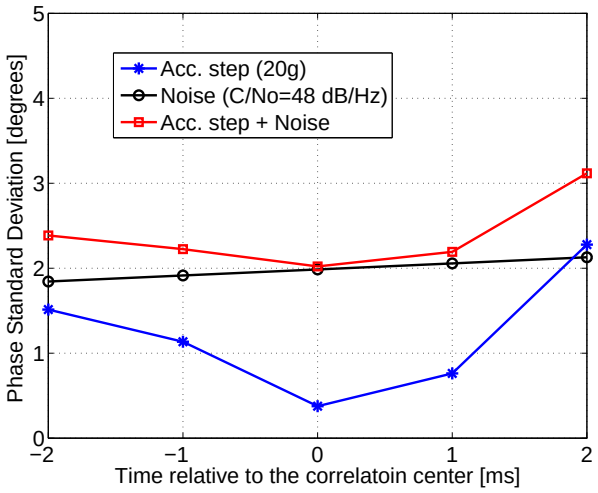

Fig. 7. Phase estimation error within the correlation interval.

prediction. Signal and noise results are summarized in Figure 7 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 10000 runs of 1 second with only Gaussian noise of variance $\frac{1}{2TC/N_0}$ as input, and discarding the first third of the record to avoid the main part of the 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,

$$\sigma^2_{\hat{\phi}_i} = \frac{1}{2TC/N_0} \times 2B_N T = \frac{B_N}{C/N_0} \tag{5}$$

The expression gives $\sigma_{\hat{\phi}_i} = 1.98^{\text{o}}$ when $C/N_0 = 48$dB/Hz, as seen in ordinates of Figure 7. 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 standard deviation is $3.1^{\text{o}}/2^{\text{o}} \approx 1.55$, i.e., 55%. This analysis clearly shows that even for a GNSS receiver with scalar tracking loops, the common correlation interval scheme can be considerably beneficial, if the effect of bit transitions can be properly handled. This is the aim of the next section.

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. Then, for a given correlation interval the partial correlation can be expressed as

$$C_1 = I_1 + jQ_1 = D_1\sqrt{T_1\frac{C}{N_0}}e^{j\Delta\phi_1} + n_1 \tag{6}$$

$$C_2 = I_2 + jQ_2 = D_2\sqrt{T_2\frac{C}{N_0}}e^{j\Delta\phi_2} + n_2 \tag{7}$$

where the subscripts i have been omitted to simplify the notation. Notice that $\Delta\phi_2 = \Delta\phi_1 + 2\pi f_i T$, if f_i is the frequency error, assumed constant during this whole correlation interval, and n_1 and n_2 are independent zero-mean and unit-variance complex gaussian noise. Based on these two partial correlations, different phase discriminators can be computed. In the following subsections we describe three of them. Their comparison in terms of tracking threshold is left for the next section.

A. Partial-Decision Directed (PDD) Discriminator

One possible approach to combine the phase information of both partial correlations is to estimate the corresponding data-bits based on the in-phase value. Then, this data-bit estimates, $\hat{d}_1$ and $\hat{d}_2$, can be used to correct the bit transition and add the partial correlations. Then, the phase of the resulting total correlation will be the sought for phase error value. The expression of this phase discriminator for the i-th correlation interval is

$$\begin{aligned} e_{PDD} &= \tan^{-1}\left(\frac{\hat{d}_1 Q_1 + \hat{d}_2 Q_2}{\hat{d}_1 I_1 + \hat{d}_2 I_2}\right) \\ &= \tan^{-1}\left(\frac{\text{sgn}(I_1)Q_1 + \text{sgn}(I_2)Q_2}{|I_1| + |I_2|}\right) \end{aligned} \tag{8}$$

where the notation $\text{sgn}(\cdot)$ indicate the sign function. Of course, the longer the correlation time, the better is the data-bit estimate. But, if one correlation is long necessarily the other will be short. However, it can be argued that when a short correlation is added, even in error, its effect will not be so harmful because the other partial correlation will dominate in the sum.

B. Partial-Phase Weighted Average (PWA) Discriminator

Another possible discriminator can be obtained from the phase errors of both partial correlations. This is the approach used in [1]. The phase errors, obtained as in (3), corresponding to C_1 and C_2 respectively are

$$e_1 = \tan^{-1}\left(\frac{Q_1}{I_1}\right) = [\Delta\phi_1 + n_{\phi_1}]_\pi \quad (9)$$

$$e_2 = \tan^{-1}\left(\frac{Q_2}{I_2}\right) = [\Delta\phi_2 + n_{\phi_2}]_\pi \quad (10)$$

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, adjusted to the duration of each correlation, should be

$$\begin{aligned} e_{PWA} &= \frac{T_1}{T}e_1 + \frac{T_2}{T}e_2 \\ &\approx \Delta\theta_i + \pi\Delta f_i T + n_{eq} = \Delta\phi_i + n_{eq} \end{aligned} \quad (11)$$

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 either 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 (11) the $[\cdot]_\pi$ function actually is applied before the average, and then the result 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 (9) must be bounded. Indeed, (9) can be written as

$$e_1 = \tan^{-1}\left(\frac{Q_1}{I_1}\right) = \Delta\phi_1 + n_{\phi_1} + k_1\pi \quad (12)$$

$$e_2 = \tan^{-1}\left(\frac{Q_2}{I_2}\right) = \Delta\phi_2 + n_{\phi_2} + k_2\pi \quad (13)$$

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 (11). This situation has to be detected, and a simple hypothesis test can be built. The decision variable is

$$e_1 - e_2 = \Delta\theta_1 - \Delta\theta_2 + n_d + k_d\pi = \pi\Delta f_i T + n_d + k_d\pi \quad (14)$$

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 (11) 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 this phase error discriminator is

$$e_{PWA} = \frac{T_1}{T}e_1 + \frac{T_2}{T}\left\{e_2 + I_\pi(e_1 - e_2)\right\}. \quad (15)$$

TABLE I
Standard deviation for different discrimination schemes.

Scheme	Noise (48dB/Hz)	Acc.(20g)	Acc. + Noise
KD	1.975	0.377	2.015
PDD	1.976	0.377	2.011
PWA	1.975	0.377	2.013
SIP	1.99	0.373	2.027

C. Squared-Input Phase (SIP) Discriminator

The bit transition caused by a BPSK modulation can be also avoided squaring the signal. This technique, usually called Squaring Loops has been used traditionally in analog BPSK synchronization schemes [14]. This idea was also used to extend the correlation interval non-coherently beyond the data-bit duration [15]. In our case, if the partial correlations are squared, then they can be added without the adverse effect of the data-bit. Clearly, the noise is also present and in low signal-to-noise ratio conditions this causes a squaring loss. The squaring operation also produces a duplication of the correlation phase and therefore a correction factor should be applied. The expression of this phase discriminator for the i-th correlation interval is

$$\begin{aligned} e_{SIP} &= \frac{1}{2}\angle\left\{(I_1 + jQ_1)^2 + (I_2 + jQ_2)^2\right\} \\ &= \frac{1}{2}\tan_2^{-1}\left\{\frac{2(I_1Q_1 + I_2Q_2)}{I_1^2 - Q_1^2 + I_2^2 - Q_2^2}\right\} \end{aligned} \quad (16)$$

where the subscript 2 indicates that actually a four-quadrant inverse tangent should be used.

The three discrimination schemes were simulated in the same conditions as the bit-synchronous loop. In this case, only the phase error in the middle of the correlation interval matters since under the common correlation interval architecture the receiver can choose this measurement instant and no extrapolations are needed. The results after 10000 runs of the loop with random data-bit transition location in each case are summarized in the Table I. In each case, the statistical significance of the result was checked, i.e., the empirical standard deviation of the estimated value was in the order of a thousandth of the empirical mean. For comparison purposes a loop with known data-bits (KD) was also included. This loop uses the same discriminator as in the bit-synchronous case because it can correct the partial correlations and add them coherently.

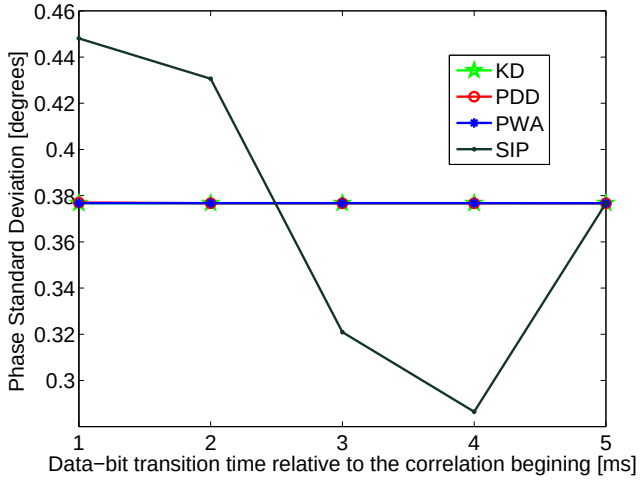

Fig. 8. Phase estimation error with different transition locations.

Except for the squaring scheme, the noise and signal response are the same as in the loop with known data-bits, and also in the data-bit synchronous loop. On the contrary, the last scheme (SIP) has a little more output noise and, in average, a little less noise in the signal estimation. However, this value is highly dependent on the data-bit transition location. The response to an acceleration step without noise is plotted in Fig. 8 for fixed transition location inside the correlation interval. While this value is almost constant for the other schemes, it changes considerably for the squaring one.

From the computational cost point of view, the first scheme is the best. One inverse tangent is used as the case of the usual phase discriminator. Only the sign corrections are needed before the complex sum. In the second case, one more inverse tangent calculation is needed each time a bit transition is 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. And in the last scheme, two complex square operations are required instead of and additional inverse tangent. Which one is more demanding will depend on the specific implementation of the receiver processor unit.

Naturally, the ideas of the partial data-bit estimation, the weighted average for combining partial discriminated errors, or the squaring operation can be extended to be applied for the case of code delay error estimation in the presence of a bit transition. 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 in the case of the weighted average.

IV. Tracking Threshold Estimation

In this section, the non-linear performance of the three proposed bit-asynchronous schemes 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.

Four UFA-PLLs were simulated for comparison. One operates with known data-bits and it is equivalent to a data-bit synchronous loop. The other three loops implement the proposed schemes for carrier phase discrimination during a bit transition according to (8), (15) and (16). Pull-out events were detected using the previously mentioned criteria, and the pull-out probability (POP) of each scheme was estimated as the ratio between the number of runs with a pull-out event and the number of total runs simulated. 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. 9, 10, 11, and 12, where POP level curves have been plotted for values of 0.1, 0.2 and so on. The tracking threshold of each scheme is defined usually when the POP reaches a level of 0.1. Then, the region of operation to the right of the 0.1 curve in each scheme. The first considered scheme shows practically the same tracking threshold as the loop with known data-bits in the low dynamic situation. However, its performance has an important degradation when the dynamics is increased. This effect can be explained by the fact that during high dynamic situations, the received phase changes appreciably during the correlation interval, and the bit detection based on partial correlations is poor. The other two proposed schemes exhibit only a small degradation in their tracking thresholds with respect to the loop with known data. Actually, comparing Figs. 10 and 11 with Fig. 9 it can be clearly seen that the difference between them is always less than 0.5 dB.

The pull-out probabilities estimated for the scenarios without acceleration, and with acceleration steps of 20g, 40g, and 60g are plotted in Figs. 13, 14, 15, and 16, respectively. These curves show in more detail that the scheme based on partial bit detection has a better performance when there

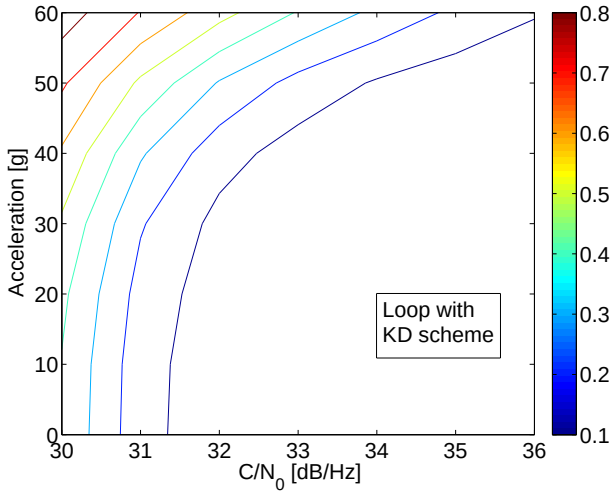

Fig. 9. POP of loop with known data and its tracking threshold.

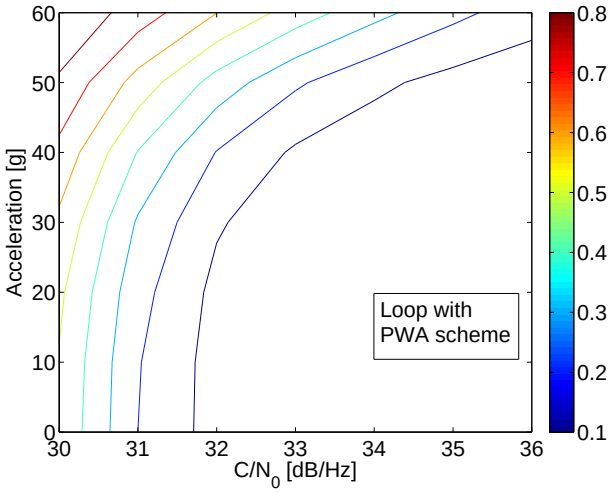

Fig. 11. POP of loop with PWA scheme and its tracking threshold.

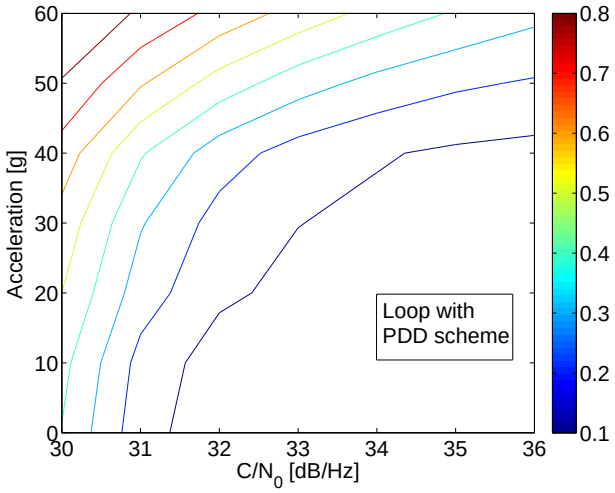

Fig. 10. POP of loop with PDD scheme and its tracking threshold.

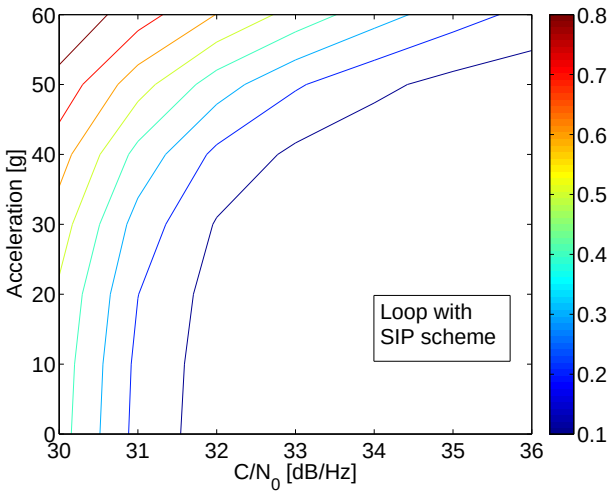

Fig. 12. POP loop with SIP scheme and its tracking threshold.

is no acceleration and how this performance change in the more demanding dynamic scenarios. It can also be seen that the partial-phase weighted average scheme has a worse performance in low and moderate dynamic scenarios than the squared-input one, but the situation changes when very high dynamics is considered. In particular, the weighted average scheme has the best performance for the 60g acceleration condition. Notice that in n Fig. 13 there are no results for $C/N_0 = 36$dB/Hz because the number of runs simulated do not allow a dependable estimate of them.

V. Conclusions and Future Work

The proposed GNSS receiver architecture is based on bit-asynchronous tracking loops and is intended for high performance real-time receivers. Under this architecture, the tracking loop operation is synchronous with the navigation measurement generation process, rather than with the data-bits as it is done traditionally. In this way, 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 standard deviation increase of up to 55% can be avoided by controlling the measurement instant. Three different carrier discrimination schemes designed for GNSS receivers operating with such a common correlation interval for all the tracked satellite signals have been presented.

The effect of a possible bit-transition inside the correlation interval is managed by the calculation of two partial correlations. The proposed carrier discrimination schemes work based on these partial correlations. The proposed schemes were applied to a UFA-PLL designed for high dynamic GNSS receivers. We found that the data-bit asynchronous operation with either of the proposed schemes produces negligible impact on tracking estimation phase noise for high signal-to-

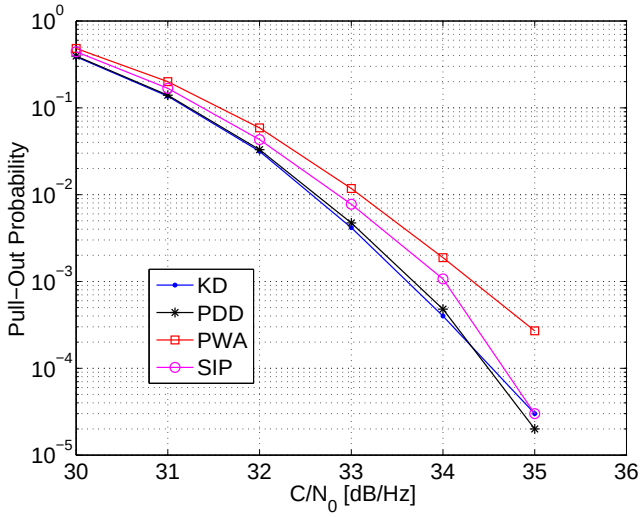

Fig. 13. POP of different schemes without acceleration.

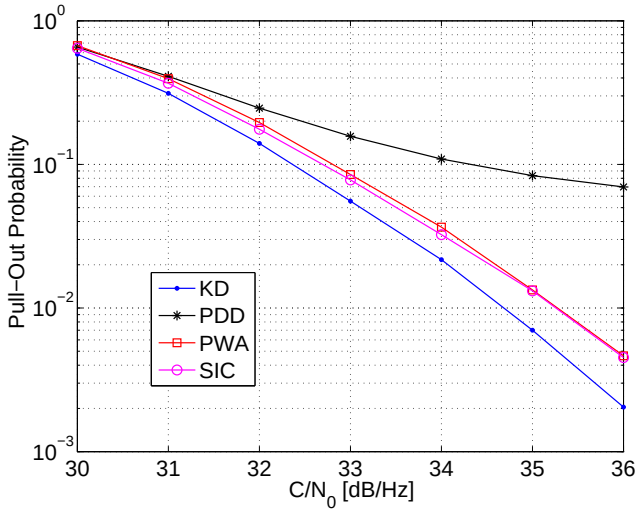

Fig. 15. POP loop of different schemes at 40g of acceleration.

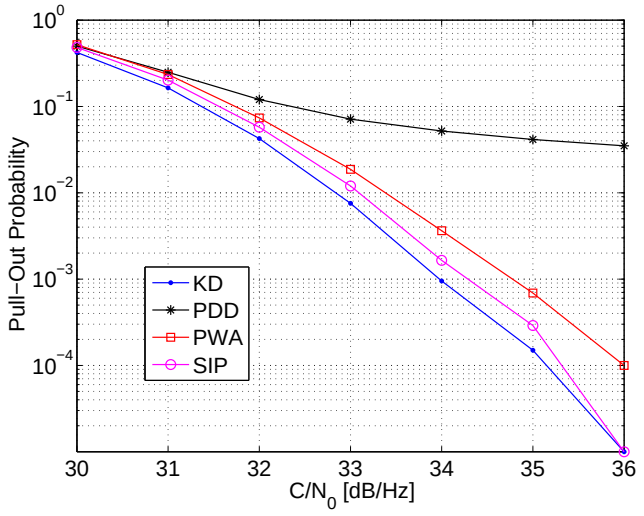

Fig. 14. POP of different schemes at 20g of acceleration.

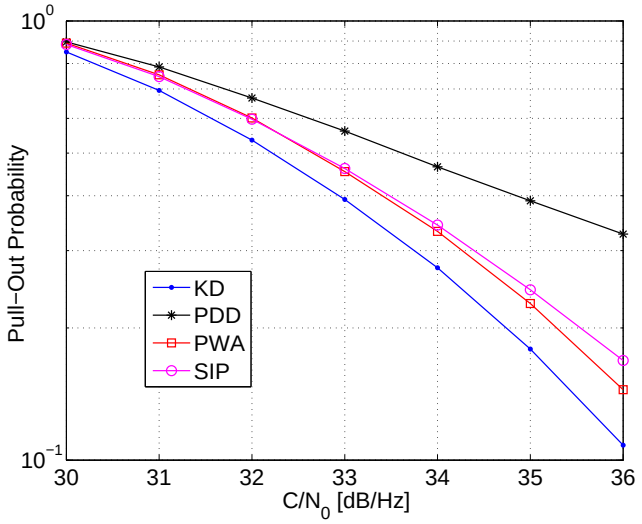

Fig. 16. POP loop of different schemes at 60g of acceleration.

noise ratio conditions. Their non-linear performance, in low signal-to-noise ratio conditions, was quantified through pull-out probabilities estimation based on extensive Monte Carlo simulations. It was found that each of the proposed schemes is more appropriate for different dynamic levels. The scheme based on partial-bit decision works better in low dynamic conditions, the squared-input one is preferable for moderate dynamics, while the partial-phase weighted average is better for very high dynamic situations. The extension of these schemes for code delay error estimation was briefly discussed.

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. Nevertheless, this new GNSS receiver architecture main potential 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 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. An implementation of such a correlator architecture in an FPGA has already been implemented and reported in [16]. The authors are now working on 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] P. A. Roncagliolo, J. García, and C. H. Muravchik, "Data-Bits Asynchronous Tracking Loop Scheme for High Performance Real-Time GNSS Receivers," in *Proceedings of The Fourth International Conference on Advances in Satellite and Space Communications, SPACOMM 2012*, Chamonix, Mont Blanc, France, April 2012, pp. 108 – 113.

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

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

[4] 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.

[5] 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.

[6] 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.

[7] S. Zhao and 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.

[8] 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.

[9] 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.

[10] P. A. Roncagliolo, J. G. García, and C. H. Muravchik, "Optimized Carrier Tracking Loop Design for Real-Time High-Dynamics GNSS Receivers," *International Journal of Navigation and Observation*, vol. 2012, p. 18.

[11] 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.

[12] P. A. Roncagliolo, J. G. 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.

[13] 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.

[14] J. G. Proakis, *Digital Communications, 3rd Edition*. New Jersey: McGraw Hill, 1995.

[15] D. Borio and G. Lachapelle, "A non-coherent architecture for GNSS digital tracking loops," *Annals of Telecommunications*, vol. 64, no. 9, 2009.

[16] J. G. Diaz, J. G. Garcia, and P. A. Roncagliolo, "An FPGA implementation of a data-bit asynchronous GPS/GLONASS correlator," in *Micro-Nanoelectronics, Technology and Applications (EAMTA), 2012 Argentine School of*, aug. 2012, pp. 27 –33.