

Analysis of Galileo E1 OS unbiased BOC/CBOC tracking techniques for mass market applications

Olivier Julien, Christophe Macabiau, Emmanuel Bertrand

► **To cite this version:**

Olivier Julien, Christophe Macabiau, Emmanuel Bertrand. Analysis of Galileo E1 OS unbiased BOC/CBOC tracking techniques for mass market applications. NAVITEC 2010, 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals, Dec 2010, Noordwijk, Netherlands. pp 1-8, 2010, <10.1109/NAVITEC.2010.5708070>. <hal-01022203>

HAL Id: hal-01022203

<https://hal-enac.archives-ouvertes.fr/hal-01022203>

Submitted on 29 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis of Galileo E1 OS unbiased BOC/CBOC tracking Techniques for Mass Market Applications

Olivier Julien, Christophe Macabiau
 Laboratoire de Traitement du Signal et des
 Télécommunications
 Ecole Nationale de l'Aviation Civile (ENAC)
 Toulouse France

Emmanuel Bertrand
 Université de Toulouse / ENSEEIHT
 Toulouse, France

Abstract —BOC modulation is used for Galileo E1 OS signals. The Offset Carrier (OC) modulation is interesting to transmit the Galileo signal because it offers a high degree of spectral separation with signals centered in the used band and a better robustness against noise, multipath and interference. However, BOC modulation introduces several peaks to the autocorrelation function, potentially biasing the pseudorange measurements. Several techniques were proposed in the last ten years to overcome this problem. The aim of this paper is to present an analysis of the suitability of unbiased BOC/CBOC tracking techniques of the Galileo E1 OS signal implemented in mass market receivers by reviewing their models and by testing them.

Keywords-component; Galileo, Tracking, ASPeCT, DET

I. INTRODUCTION

While with the GPS L1 C/A signals the BPSK modulation is used, BOC modulation is applied to Galileo E1 OS signals. The Offset Carrier (OC) modulation is interesting to transmit the Galileo signal because it provides a simple and effective way of moving signal energy away from the band centre and consequently allows the use of a same frequency band several GNSS. Moreover it offers a high degree of spectral separation with signals centered in the used band and a better robustness against noise, multipath and interference. However, BOC modulation introduces several peaks to the autocorrelation function, so we must make sure the measurement is made on the main peak in order not to bias the pseudorange measurements.

Several techniques were proposed in the last ten years to overcome this problem. In the frame of the FP7 HIMALAYA project, we are analyzing the potential of some of these techniques to fight the biased measurements for mass market applications with severe implementation constraints.

The aim of this paper is to present an analysis of the suitability of unbiased BOC/CBOC tracking techniques of the Galileo E1 OS signal implemented in mass market receivers by reviewing their models and by testing them.

The paper focuses on two techniques that both resist to the bias due to the BOC signal : ASPeCT and DET.

Autocorrelation Side-Peak Cancellation Technique (ASPeCT)'s principle is to remove the side-peaks of the sine-BOC(n,n) autocorrelation function in order to be able to reliably obtain unambiguous measurements. The fundamental concept of the double estimation technique (DET) is to track separately and independently the subcarrier and code delays using different delays. It is based on the fact that the time delay of the received signal is the same in both the code and the sub-carrier. Both methods do not rely on peak discrimination to determine valid tracking states.

The second part of the article reviews the performance of these techniques for CBOC tracking in presence of acquisition bias, in presence of noise and multipath.

II. GALILEO E1 OS SIGNAL MODEL

A. Galileo E1 Signal Model

The received Galileo E1 OS signal can be modelled at the Intermediate Frequency (IF) as [1]:

$$s(t) = A \begin{pmatrix} c_D(t - \tau) d_D(t - \tau) CBOC_-(t - \tau) \\ -c_P(t - \tau) s_c(t - \tau) CBOC_+(t - \tau) \end{pmatrix} \times \cos(2\pi f_{IF} t + \theta) \quad (1)$$

with

$$\begin{aligned} CBOC_-(t) &= \sqrt{10/11} BOC_1(t) - \sqrt{1/11} BOC_6(t) \\ CBOC_+(t) &= \sqrt{10/11} BOC_1(t) + \sqrt{1/11} BOC_6(t) \\ BOC_x(t) &= \text{sign}(\sin(x \cdot 2\pi \cdot 1.023e^6 \cdot t)) \end{aligned}$$

where

- A is the amplitude of the incoming signal at the correlator input,
- c_P and c_D are the spreading sequences carried by the pilot and data components. They have a chipping rate equal to 1.023 Mega-chips per second (Mcps) and a length of 4092 chips.
- d_D represents the symbols of the I/NAV navigation message [1] modulating the data component. It has a rate of 250 symbols per second (sps).

- s_c represents the secondary code present on the pilot component that has a length of 25 chips and a rate of 250 Hz.
- τ is the delay of the spreading sequence,
- f_{IF} is the intermediate frequency, and
- θ is the phase offset of the carrier.

B. Correlator Output Model

Four different correlators are considered: I_D , Q_D , I_P and Q_P . The correlator output model for these correlators is [10]:

$$I_D(k) = \sqrt{\frac{P}{4}} (d(k) \tilde{R}_{CBOC(+),L_D}(\varepsilon_\tau) - s_{\text{sec}}(k) \tilde{R}_{CBOC(-),L_D}(\varepsilon_\tau)) \text{sinc}(\pi \varepsilon_f T_I) \cos(\varepsilon_\phi) + n_{I_D}(k)$$

$$Q_D(k) = \sqrt{\frac{P}{4}} (d(k) \tilde{R}_{CBOC(+),L_D}(\varepsilon_\tau) - s_{\text{sec}}(k) \tilde{R}_{CBOC(-),L_D}(\varepsilon_\tau)) \text{sinc}(\pi \varepsilon_f T_I) \sin(\varepsilon_\phi) + n_{Q_D}(k)$$

$$I_P(k) = \sqrt{\frac{P}{4}} (d(k) \tilde{R}_{CBOC(+),L_P}(\varepsilon_\tau) - s_{\text{sec}}(k) \tilde{R}_{CBOC(-),L_P}(\varepsilon_\tau)) \text{sinc}(\pi \varepsilon_f T_I) \cos(\varepsilon_\phi) + n_{I_P}(k)$$

$$Q_P(k) = \sqrt{\frac{P}{4}} (d(k) \tilde{R}_{CBOC(+),L_P}(\varepsilon_\tau) - s_{\text{sec}}(k) \tilde{R}_{CBOC(-),L_P}(\varepsilon_\tau)) \text{sinc}(\pi \varepsilon_f T_I) \sin(\varepsilon_\phi) + n_{Q_P}(k)$$

with

$$\tilde{R}_{X,Y}(x) = \int_{-\infty}^{+\infty} H(f) F_X(f) F_Y^*(f) e^{-2j\pi f x} df.$$

where

- P represents the total power of the incoming signal at the correlator input (and thus includes a certain number of losses due to the front-end),
- the subscripts D and P are for the incoming "data component" and "pilot component" respectively,
- the subscripts L is for the local spreading sequence,
- \tilde{h} and H are the equivalent front-end filter impulse response and transfer function respectively,
- F_X is the Fourier transform of the signal X ,
- $\tilde{R}_{X,Y}$ is the correlation function between the incoming spreading sequence X and the locally generated spreading sequence Y taking into account the effect of the front-end filter H ,
- ε_τ is the code delay error between the incoming code delay and the one estimated by the receiver,
- ε_ϕ is the phase error (in the middle of the correlation interval) between the incoming carrier and the locally generated carrier,
- ε_f is the frequency error between the incoming carrier and the locally generated carrier,
- T_I is the correlation time,
- n_{I_D}, n_{Q_D} are the noise components of the two discriminator outputs associated with the data component respectively; n_{I_P} and n_{Q_P} are the noise components of the 2 discriminator outputs associated with the pilot component respectively. All these noise components are all uncorrelated, and: $\text{Var}[n_{I_D}] = \text{Var}[n_{Q_D}] = \text{Var}[n_{I_P}] = \text{Var}[n_{Q_P}] = \frac{N_0}{4T_I} \beta_D = \frac{N_0}{4T_I} \beta_P$
- $N_0/2$ is the equivalent double-sided thermal noise PSD at the correlator input. In the context of this study, it also includes the interference components that can be

equivalently represented by a white noise component at the correlator input.

- $\beta = \int_{-\infty}^{+\infty} |H(f)|^2 |F_L(f)|^2 df$ is the loss of noise power due to the front-end filter.

As a first remark, it is important to understand that the variance of the noise on the data and pilot correlator output might be different due to the different CBOC subcarriers used on each component ($\beta_P \neq \beta_D$ since $L_P = CBOC(6,1,1/11, '-')$ \neq $L_D = CBOC(6,1,1/11, '+')$). This is however not the case if a BOC receiver is used since in this case the same replica is used on the data and pilot components (BOC(1,1)). This is shown in Figure 1 for different values of front-end filter bandwidth.

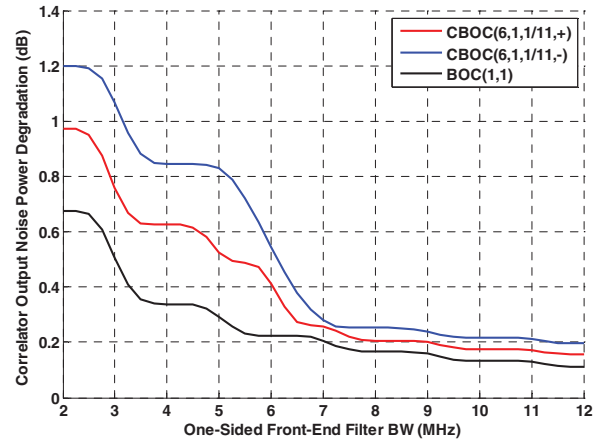


Figure 1 – Correlator Output Noise Power Degradation According to the Local Replica.

III. OPPORTUNITY OF CBOC FOR MASS MARKET RECEIVERS

Galileo E1 OS signal will use the CBOC(6,1,1/11) modulation. It is interesting to highlight the differences in performance between a BOC receiver (narrow-band) and a CBOC receiver (wide band).

It is known that a CBOC receiver requires more complexity due to the need for a higher sampling rate, and due to the need for a more complex 4-level local replica.

Regarding acquisition performances, a BOC receiver requires a wider code delay acquisition bin and thus has a faster acquisition time. To be able to have the same kind of performance, the CBOC receiver would need a narrower code delay acquisition bin, and a corresponding increased number of correlators to compensate for this. This could be acceptable, however, pending the fact that the information on the signal characteristics (Doppler, time) is fairly accurate.

Regarding code tracking, a CBOC receiver exhibits performances that are significantly higher than those of a BOC receiver. This is due to the fact that the CBOC autocorrelation function is very sharp, and to the fact that a narrower early-late spacing can be used (due to the wider front-end filter bandwidth). This is quite significant regarding the capability

of the receiver to provide proper measurements (with limited impact from the thermal noise and multipath).

To lower the complexity of the CBOC receiver, two techniques using only 1-bit local replicas can be used. The two methods exhibit interesting performances. The simplest one, TM61[11], degrades the resistance to thermal noise, but improves the resistance to multipath. Its resistance to false lock still needs some work. The second method [12], which requires twice as many correlators exhibits excellent performance due to its adaptive possibility. It can be adapted to best fight thermal noise, or multipath, or interference.

Both BOC and CBOC receivers will have to include a specific tracking module to fight against biased measurements inherent to a BOC modulation. However, this threat is the same for both the BOC and CBOC receivers (the CBOC receiver does not seem to have more false lock points).

Regarding phase tracking, the CBOC receiver will have better performance than the BOC receiver since the correlation losses are lower for the former.

As a conclusion, if the main requirement for the receiver is the complexity, then it seems that the choice of a narrow band BOC receiver would be indicated. This is very important also for acquisition performance with a simple receiver (acquisition time and detection threshold) that is critical for mass market receiver.

That being said, it appears that a CBOC receiver would significantly improve the quality of the measurements. This would however have to be confirmed with tests in a real environment.

IV. CONSIDERED UNAMBIGUOUS BOC TRACKING TECHNIQUES

A variety of new modulation schemes including the BOC has been introduced. BOC signals have several advantages over traditional BPSK signals such as an increased robustness against multipath and improved tracking performance. However, they are characterized by autocorrelation functions with multiple peaks which have secondary peaks which are not much smaller than the main (central) peak and that may lead to false code lock. In the case of the Galileo E1 OS signal, this can lead to a bias around 150 m and has thus to be taken into account in the design of the tracking loops. This has led to the design of various BOC tracking algorithms. In the frame of this study 2 widely spread techniques were investigated:

- Autocorrelation Side-Peak Cancellation Technique (ASPeCT)
- Double Estimator (DE)

The purpose of these methods is to solve the problem of incorrectly locking onto secondary peaks.

V. AUTOCORRELATION SIDE-PEAK CANCELLATION TECHNIQUE

ASPeCT's principle is to remove the side-peaks of the BOC(1,1) autocorrelation function in order to be able to reliably obtain unambiguous measurements [4]. To do so, it is

interesting to calculate the correlation of the sine-BOC(n,n) modulated spreading code (PRN code \times square-wave sub-carrier) with the PRN code only. Without considering front-end filtering, it is equal to [5]:

$$R_{B/P}(\tau) = \frac{1}{2} \text{tri} \left(\frac{\tau - \frac{1}{2}}{\frac{1}{2}} \right) - \text{tri} \left(\frac{\tau + \frac{1}{2}}{\frac{1}{2}} \right)$$

It can be seen that $R_{B/P}$ consists of two triangles perfectly located on the side-peaks of the BOC(1,1) autocorrelation function, and with exactly the same magnitude. As a consequence, the idea on which ASPeCT is based is to form a synthesized correlation function by subtracting $R_{B/P}^2$ from R_B^2 to remove the undesired side-peaks as shown in Figure 2.

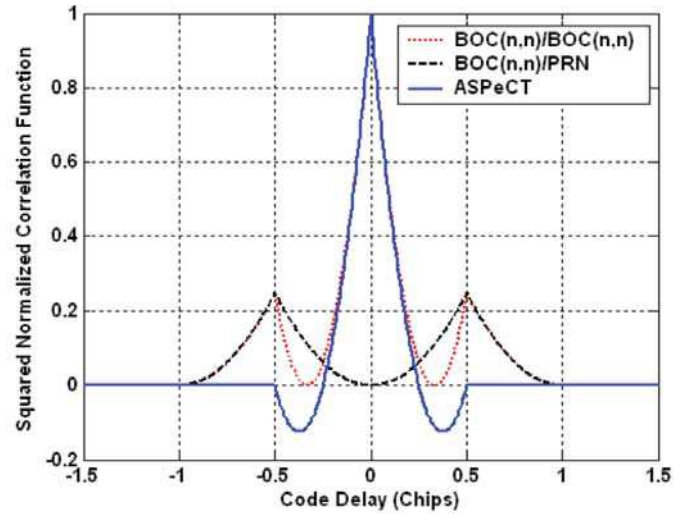


Figure 2 –BOC(1,1)/ BOC(1,1), BOC(1,1)/PRN Squared Normalized Correlation Functions and ASPeCT Modified Correlation Function Without Front-End Filter

However, in reality, one has to take into account the impact of the front-end filter on each correlation function, and as a consequence, on the alignment of the peaks and their respective magnitudes. Figure 2.22 shows the impact of a 6 MHz front-end filter bandwidth (double-sided) on the BOC(1,1) signal. It underlines that a narrow front-end filter could misalign both correlation functions' peaks, inducing the appearance of very small peaks around ± 0.6 chips. This can translate into false lock points for high C/N0 (only peaks of the squared correlation function pointing upward can lead to stable false lock points). This problem has to be taken into account in the design of the code delay discriminator. It is thus recommended to use a coefficient β in the combination of the two squared correlation functions in order to eliminate any small remaining peak. This can be modeled as:

$$R_{ASPeCT}(\tau) = \bar{R}_B^2(\tau) - \beta \bar{R}_{B/P}^2(\tau)$$

where the bar represents the effect of the front-end filter on the correlation functions

Figure 3 shows the ASPeCT modified correlation function using an experimental β value of 1.4 for a BOC(1,1) signal using a 6 MHz front-end filter bandwidth (double-sided). There is no side-peak remaining in the modified correlation function close to the BOC(1,1) autocorrelation side-peaks.

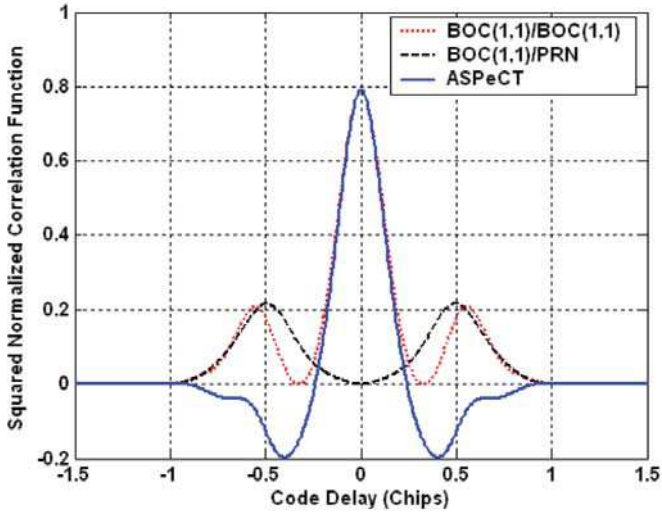


Figure 3 - BOC(1,1)/ BOC(1,1), BOC(1,1)/PRN Squared Normalized Correlation Functions and ASPeCT Modified Correlation Function With a 6 MHz Front-End Filter (Double Sided) and

In order to focus on the tracking performance, a modified version of the normalized DP discriminator based on ASPeCT-modified correlation function is proposed:

$$D_{DP}^{ASPeCT}(\epsilon_r) = \frac{((I_{E_B} - I_{L_B})I_{P_B} + (Q_{E_B} - Q_{L_B})Q_{P_B}) - \beta((I_{E_{B/P}} - I_{L_{B/P}})I_{P_{B/P}} + (Q_{E_{B/P}} - Q_{L_{B/P}})Q_{P_{B/P}})}{(6 + \beta d)(I_{P_B}^2 + Q_{P_B}^2)}$$

where

- I_{X_B} and Q_{X_B} are the in-phase and quadra-phase correlator output associated to the BOC local replica (X=E, P or L), and
- $I_{X_{B/P}}$ and $Q_{X_{B/P}}$ are the in-phase and quadra-phase correlator output associated to the BPSK local replica (X=E, P or L)

It is interesting to see that this discriminator requires the use of 4 complex correlators (1 for the E-L with BOC replica, 1 for the P with BOC replica, 1 for E-L with BPSK replica, and 1 with P with BPSK replica)

Figure 4 shows ASPeCT DP discriminator output for a BOC(1,1) signal using a 6 MHz (double-sided) front-end filter for $\beta=1$ and $\beta=1.4$ and $d=0.2$ chip. It also shows the traditional BOC DP discriminator output using the same front-end filter. The choice of $\beta=1.4$ clearly removes any potential false lock point around ± 0.56 chips. Moreover, it is interesting to note that the discriminator stability domain slightly increases with β : from $[-0.33; 0.33]$ chips for the traditional tracking technique to $[-0.36; 0.36]$ chips for ASPeCT with

$\beta=1$, to $[-0.39; 0.39]$ chips for ASPeCT with $\beta=1.4$. This is very important when considering the DLL stability for low C/N0. However, ASPeCT stability domain remains significantly lower than for the BPSK(1) signal tracking, which constitutes the main advantage of the SSL technique.

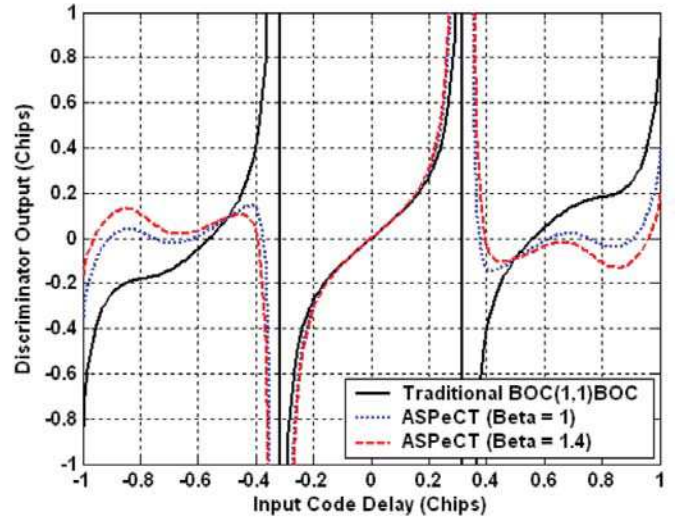


Figure 4 – ASPeCT Dot Product Proposed Discriminator Output with and for an Early-Late Spacing of 0.2 Chips and a 6 MHz Front End Bandwidth (Double Sided)

Finally, $\beta=1.4$ appears to create a false lock point around ± 0.95 chips. This is, however, not a real threat as there is no energy at that location on the ASPeCT-modified correlation function ($R_{ASPeCT}(0.95) \approx 0$) as already seen in Figure 2.22. As a consequence, no false lock will occur even for high C/N0.

It is important to note that the optimal choice of β depends upon the front-end filter and the early-late correlator chip spacing.

The DLL architecture adapted to ASPeCT is shown in Figure 5.

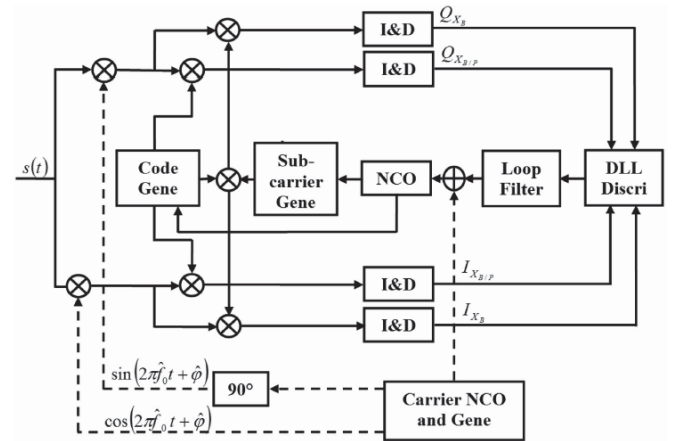


Figure 5 – DLL Architecture adapted to ASPeCT

VI. DUAL ESTIMATION TECHNIQUE

The fundamental concept of the Dual Estimation Technique (DET) is to track separately and independently the subcarrier and code delays using different delay estimates. It is based on the fact that the time delay of the received signal is the same in both the code and the sub-carrier. The following presentation is based on the tracking of a BOC signal. However, [6] claims that the method can be extended to CBOC tracking.

The DET uses three tracking loops (architecture shown in [6])

- a delay-locked loop (DLL) tracks the code phase of the received signal based on the generation of a Prompt version of the sub-carrier multiplied by an Early-Minus-Late version of the spreading code. Thus, this can be linked with a BPSK tracking: not very accurate but robust to tracking ambiguity.
- a sub-carrier locked loop (SLL) tracks the sub-carrier phase of the received signal based on the generation of an Early-Minus-Late version of the sub-carrier multiplied by a Prompt version of the spreading code. Thus this can be linked with a sub-carrier tracking: accurate but not robust to tracking ambiguity
- a third loop, either a frequency locked loop (FLL) or a phase locked loop (PLL) tracks the carrier frequency and / or phase of the received signal.

Assuming that the tracking of the code is done perfectly by the DLL, the SLL will rely on the sub-carrier correlation function shown on Figure 2.29 (case of BOC(1,1) tracking). It can be seen that the subcarrier correlation function is ambiguous with peaks every 0.5 chips (including down-pointing peaks). It can also be seen that the slope of the correlation function is steeper than that of the BOC(1,1). It can thus be thought that SLL tracking will be more accurate than the BOC(1,1) tracking. This however, might be compensated by the noise on the DLL.

Assuming that the SLL tracks perfectly the subcarrier, the DLL will be based on the code correlation function shown on Figure 6 (case of BOC(1,1) tracking). It can be seen that the code correlation function is unambiguous, but with a slope flatter than that of the BOC(1,1). Consequently, the DLL should provide an unambiguous tracking.

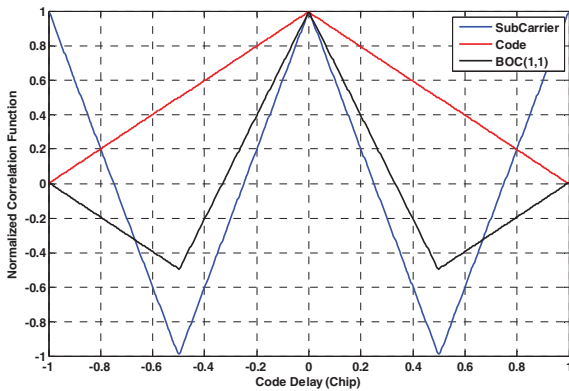


Figure 6 – Correlation Functions Used in DET

The ambiguity of the SLL estimate can thus be solved through the noisier but unambiguous DLL estimate. An example is shown in [7]. The SLL settles to integer subchip offsets from the correct, valid delay estimate. The delay estimate derived from the DLL allows correction of the SLL delay estimate wherever the SLL has settled. Therefore, the corrected delay estimate of the DET, is instantaneously corrected for tracking slips of integer subchips and the integrity of the receiver not compromised. One of the advantages of the DET is that it can recover from a large initial error since it relies on an unambiguous DLL that is comparable to a DLL tracking a BPSK.

[7] states that there are no false-lock or invalid tracking states when operating the DET. The method is said to function with the cos-BOC(15,2.5), which is a very difficult BOC to track due to the high number of side-peaks compared to the BOC(1,1) or CBOC signals.

This method is said to maintain reliable and robust tracking in high noise conditions (24 dB-Hz) and with strong multipath interference (half-amplitude) [7], although it is not demonstrated. The authors mention that the tracking jitter of the corrected DET delay estimate is equivalent to that achieved by the BJ algorithm (standard early-minus-late discriminators), preserving the precision of the BOC modulated signal. This seems in line with the correlation function analysis made earlier. In addition, it appears that multipath mitigation techniques such as narrow correlators and double-delta correlation can be easily implemented into the DET correlator structure, in particular into the SLL.

No theoretical derivation or simulations providing the code tracking standard deviation (in thermal noise) using DET as a function of the C/N0 was found in the literature. The multipath error envelope using DET was also not found in the literature. It is difficult to assess the effect of the DLL tracking noise on the SLL. In particular, this might be quite important at low C/N0.

[6,7] proposes a model of the correlator outputs based on the assumption that the correlation function of a BOC incoming signal with a replica composed of the code delayed by ϵ_1 multiplied by a subcarrier delayed by ϵ_2 is approximated by:

$$R(\epsilon_1, \epsilon_2) = R_{BPSK}(\epsilon_1) \times \Lambda(\epsilon_2)$$

Where

- R_{BPSK} is the autocorrelation of a BPSK(1) signal (as shown in Figure 6), and
- Λ is a periodical triangular function with magnitude 1 and period the sub-carrier period (as shown in Figure 6).

This model is valid when the DLL and SLL tracking errors are very small. It is however quite clear that it is not valid for

large errors as represented in Figure 6 in the case of an infinite front-end filter bandwidth. In particular, it seems that even small errors in the DLL tracking can significantly impact the SLL (with an uneven tracking).

Still, the combination of the DLL and SLL will be absorbed by the neighboring higher peak, which means that the tracking should unavoidably go towards an unambiguous tracking for the DLL and a potentially ambiguous tracking for the SLL. However, it can be anticipated that strong tracking errors in either the DLL or SLL will have a strong impact on the overall tracking accuracy. This is true also for tracking in low C/N0 conditions.

Figure 7 shows the true DET correlation function taking into account the use of a 6 MHz (Double-Sided) front-end filter. It can be seen that in this case the model from [6] cannot be used.

It seems that the only way to effectively test this method in terms of resistance to thermal noise and interference is through an independent testing.

The DET method, as presented in [6,7], uses a coherent discriminator for both the SLL and the DLL. However, it seems reasonable to think that an adaptation of the DET tracking algorithm using a non-coherent discriminator would be more adapted for a mass-market receiver. With this architecture, the DET method requires 3 complex correlators, thus 1 less than ASPECT and BJ.

It is important to note that this method seems conceptually very similar to the patent proposed in [8].

This method is patented by SSTL for BOC and CBOC use [6].

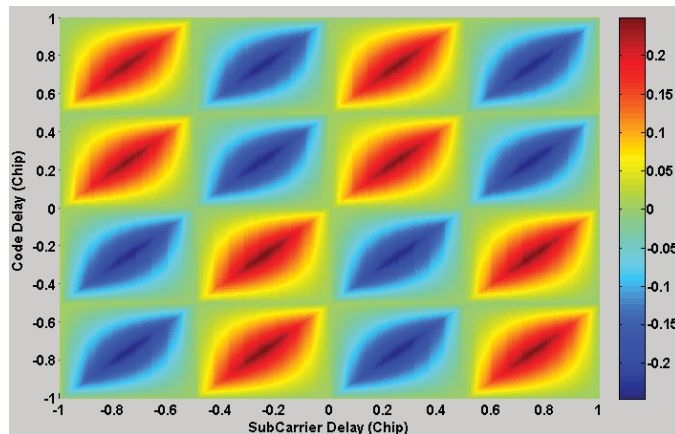
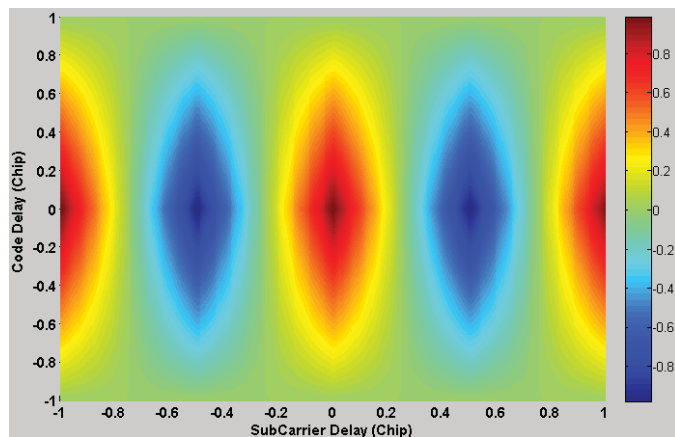
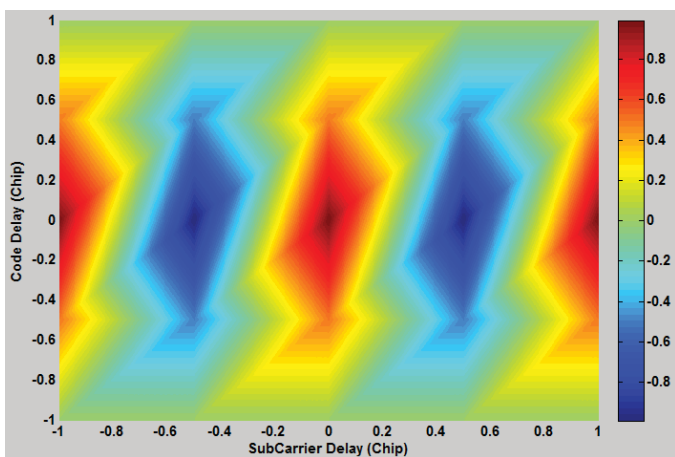


Figure 7 – True Normalized DET Correlation Function (Top), DET Correlation Function as Modeled by [6] (Middle), and Difference between the two (Bottom)

VII. SIMULATION RESULTS

The simulations are based on the reception of BOC signal, which is an approximation of a BOC receiver receiving a CBOC signal, as seen earlier. In the frame of the simulations, it is assumed that only the pilot signal is tracked. Also, it is assumed that the secondary code has been acquired. As a consequence, code tracking will be based upon a 100-ms correlation time while the phase/frequency tracking will be based on a 20-ms correlation time (to take into account the oscillator drift and signal dynamics although they are not simulated herein).

The receiver is assumed to have a 9-MHz (double-sided) front-end filter. The correlator spacing is assumed to be close to the inverse of the one-sided front-end filter bandwidth, or approximately 0.14 chips. For ASPECT's DLL and for DET's DLL and SLL, this same correlator spacing will be used. Regarding ASPECT, the value of β is chosen equal to 1.3. This value is a comfortable value that allows an unambiguous DP discriminator.

A 3rd order PLL aided by a 2nd order FLL will be used. The PLL is a true PLL using the Q discriminator [9] and uses an equivalent loop bandwidth equal to 15 Hz. The FLL will use

the differential 4-quadrant arctangent discriminator with a loop bandwidth equal to 18 Hz.

The FLL-aided PLL will aid the 1st order DLL. The DLL will use a DP discriminator and will have a 1 Hz equivalent loop bandwidth.

A. Bias Test

It has been shown in [9] that ASPeCT was unambiguous by nature. Thus, this section investigates the unambiguous property of DET. Figure 8 shows the response of DET to an initial error of 0.5 chips for two C/No values. It can be seen that DET is able to converge towards the true code delay. It can be noticed that this convergence is fairly slow (more than 5 seconds), which is due to fact that the correlation functions used by the DLL and SLL are distorted by large tracking errors, as already highlighted in Figure 6. This might be a problem if the dynamic of the signal is large during the acquisition phase.

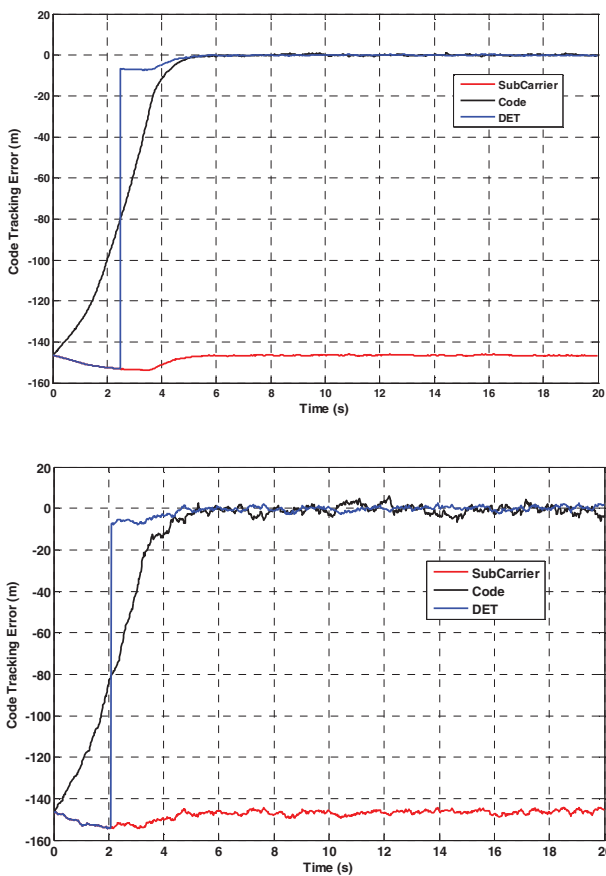


Figure 8 – DET Code Delay Tracking Error for an Initial Error of 0.5 Chip and a C/No of 45 dB-Hz (Top) and 30 dB-Hz

B. C/No Test

The theoretical formulation of code delay tracking can be found in the literature for traditional BOC tracking and ASPeCT. However, such expression cannot be found in the literature for DET. The main goal of this section is thus to find this expression and test it through simulations using the receiver parameters found above.

As already explained, assuming that the DLL tracks perfectly the code, the SLL can be considered as functioning based upon the correlation function shown in Figure 6 (assuming an infinite front-end filter). This correlation function can be used to adapt the generalized code tracking error variance. It means that the DET tracking error standard deviation can be roughly (assuming no front-end filter and a perfect DLL) estimated by:

$$\sigma_{DP,DET}^2 = \frac{B_L d}{8 \frac{C}{N_0}} \times \left(1 + \frac{1}{T_I \frac{C}{N_0}} \right)$$

By comparison, traditional BOC(1,1) tracking leads to the following expression:

$$\sigma_{DP,BOC(1,1)}^2 = \frac{B_L d}{6 \frac{C}{N_0}} \times \left(1 + \frac{1}{T_I \frac{C}{N_0}} \right)$$

This means that under our hypothesis, DET should improve lower the BOC(1,1) tracking standard deviation by $\sqrt{\frac{4}{3}} \approx 1.15$, which is also interesting. Because ASPeCT has performance very similar to traditional BOC(1,1) tracking, this applies also to ASPeCT.

Figure 9 shows the results of the simulations done to test the code tracking error standard deviation of ASPeCT and DET. It can be seen that DET follows very well the theoretical formula displayed above. This means that slight errors of the DLL do not significantly degrade the SLL behavior, at least regarding the tested configuration (indeed, for a wider early-late spacing, the DLL error would be larger and would degrade more significantly the SLL). It can also be seen that DET slightly outperforms ASPeCT against thermal noise.

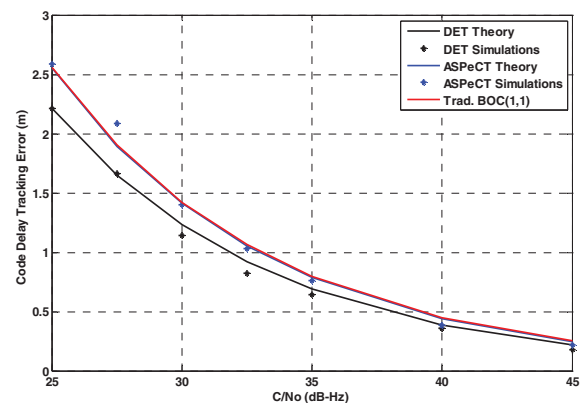


Figure 9 – ASPeCT and DET Tracking Code Tracking Standard Deviation as a Function of the C/No

C. Multipath Test

This test is meant at assessing the multipath envelope for both DET and ASPeCT. This is done through simulations by injecting a multipath with a slowly growing delay, which will also modify its relative phase compared to the direct path. For this simulations, to ensure that the tracking loops reach their stability point for each multipath delay value, the equivalent loop bandwidth of the DLL (and SLL for DET) were set to 40 Hz, and no thermal noise was generated.

Figure 10 shows the corresponding multipath envelope using a multipath with an amplitude that is half of the direct signal. It can be seen that here again, DET has a better resistance to multipath than ASPeCT for shorter delay multipath. On the other hand, ASPeCT behaves better than DET for long delay multipath.

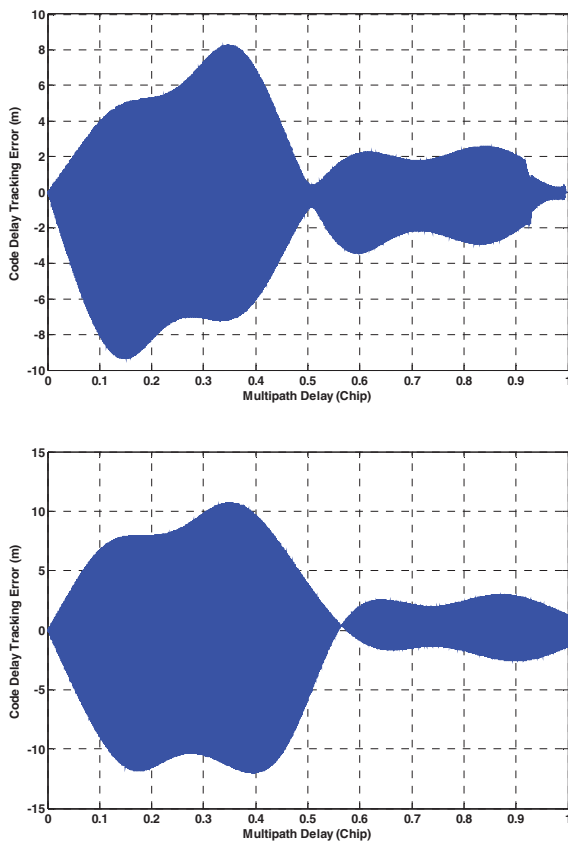


Figure 10 – DET and ASPeCT Code Delay Tracking Error in Presence of Multipath (half the Amplitude of the Direct Signal).

VIII. CONCLUSION

It appears that DET has an overall better performance than ASPeCT in the tests that we have conducted. This is true for tracking conditions, although this difference is fairly slim and might not be noticeable in operational conditions.

DET has the great advantage of having a very large pull-in region, and of recovering from initial code delay errors greater than 0.5 chips.

More tests in operational conditions should be performed to confirm this first assessment of the ASPeCT and DET techniques.

REFERENCES

- [1] [EC, 2010] European Commission (2010), European GNSS (Galileo) Open Service – Signal-In-Space Interface Control Document Issue 1, February.
- [2] RTCA SC-159 (2004), Assessment of Radio Frequency Interference Relevant to the GNSS L5/E5a Frequency Band, RTCA DO 292, July 2004.
- [3] Bastide, F., O. Julien, C. Macabiau, B. Roturier (2002), Analysis of L5/E5 Acquisition, Tracking and Data Demodulation Thresholds, Proceedings of the Institute of Navigation GPS (Portland, OR, USA, Sept. 24-27), pp. 2196-2207.
- [4] Julien O., C. Macabiau, M. E. Cannon, G. Lachapelle (2007), ASPeCT: an Unambiguous Sine-BOC(n,n) Acquisition/Tracking Technique for Navigation Applications, IEEE Transaction on Aerospace and Electronic Systems, vol. 43 n°1, January 2007, pp 150-162
- [5] Julien O., C. Macabiau, M.E. Cannon, and G. Lachapelle (2004), New Unambiguous BOC(n,n) Tracking Technique, NaviTec Conference, December 2004
- [6] Hodgart, M.S and Blunt Paul. (2008), A Receiver of Binary Offset Carrier (MBOC) Modulated Signals, WO 2008/017860, Apr 29, 2010
- [7] Hodgart, M.S. (2008), A Receiver of Multiplexed Binary Offset Carrier (MBOC) Modulated Signals, WO 2008/090323, June 3, 2010
- [8] Paul Blunt, Ruediger Weiler, Stephen Hodgart, Demonstration of BOC(15, 2.5) acquisition and tracking with a prototype hardware receiver, Surrey Space Centre Martin Unwin Surrey Satellite Technology Limited, 2007
- [9] P. Fenton, A. Van Dierendonck (2001), 'Split C/A Code Receiver', US Patent 6184822
- [10] Julien O. (2005), Design of Galileo L1F Receiver Tracking Loops, Ph.D. thesis, Department of Geomatics Engineering, University of Calgary, July 2005
- [11] O. Julien, C. Macabiau, J-L. Issler, L. Ries (2006), 1st CNES-ESA Workshop on GALILEO Signals and Signal Processing , Toulouse, 12-13 OCT 2006
- [12] O. Julien, C. Macabiau, J-L. Issler, L. Ries (2007), '1-bit Processing of Composite BOC (CBOC) Signals and Extension to Time-Multiplexed BOC (TMBOC) Signals', , ION NTM 2007, San Diego