Hindawi Publishing Corporation
International Journal of Navigation and Observation
Volume 2012, Article ID 135401, 14 pages
doi:10.1155/2012/135401

Research Article

Two-Step Galileo E1 CBOC Tracking Algorithm: When Reliability and Robustness Are Keys!

Aleksandar Jovanovic,[1] Cécile Mongrédien,[2] Youssef Tawk,[1] Cyril Botteron,[1] and Pierre-André Farine[1]

[1] *Electronics and Signal Processing Laboratory (ESPLAB), École Polytechnique Fédérale de Lausanne (EPFL), Rue A.-L. Breguet 2, 2000 Neuchâtel, Switzerland*
[2] *Fraunhofer Institute for Integrated Circuits IIS, Nordostpark 93, 90411 Nuernberg, Germany*

Correspondence should be addressed to Aleksandar Jovanovic, aleksandar.jovanovic@epfl.ch

Received 30 December 2011; Revised 4 April 2012; Accepted 22 April 2012

Academic Editor: Heidi Kuusniemi

The majority of 3G mobile phones have an integrated GPS chip enabling them to calculate a navigation solution. But to deliver continuous and accurate location information, the satellite tracking process has to be stable and reliable. This is still challenging, for example, in heavy multipath and non-line of sight (NLOS) environments. New families of Galileo and GPS navigation signals, such as Alternate Binary Offset Carrier (AltBOC), Composite Binary Offset Carrier (CBOC), and Time-Multiplex Binary Offset Carrier (TMBOC), will bring potential improvements in the pseudorange calculation, including more signal power, better multipath mitigation capabilities, and overall more robust navigation. However, GNSS signal tracking strategies have to be more advanced in order to profit from the enhanced properties of the new signals.In this paper, a tracking algorithm designed for Galileo E1 CBOC signal that consists of two steps, coarse and fine, with different tracking parameters in each step, is presented and analyzed with respect to tracking accuracy, sensitivity and robustness. The aim of this paper is therefore to provide a full theoretical analysis of the proposed two-step tracking algorithm for Galileo E1 CBOC signals, as well as to confirm the results through simulations as well as using real Galileo satellite data.

1. Introduction

New GPS and Galileo signals use new modulations, such as AltBOC, CBOC, and TMBOC that have the potential to improve navigation through advanced signal properties, such as more signal power, better multipath mitigation capabilities, and overall improved signal cross-correlation properties. Certainly, a major innovation brought by the new modulation schemes consists of the presence of two distinct components, namely, the data and pilot channels that carry two different pieces of information. The data channel contains the navigation message, whereas the pilot channel is dataless, allowing long coherent signal integration that, in turn, allows more precise determination of the ranging information. For carrier tracking, the presence of a pilot channel enables the combined use of pure PLL (Phase Lock Loop) discriminators and longer coherent integration time. Code tracking can be organized as data/pilot collaborative tracking [1, 2], where two channels (data and pilot) are used in the estimation of the code error, decreasing the thermal noise error and improving overall tracking.

Several tracking algorithms proposed for Galileo E1 CBOC signals were derived from tracking schemes developed for BPSK (Binary Shift Keying) and BOC(1,1) signals. BOC(1,1) tracking faces the problem of biased tracking as already explained in [3]. Until now, the main algorithms that were proposed to specifically address the problem of biased tracking for BOC(1,1) and can be applied to CBOC tracking too are Single Side Lobe, bump and jump algorithm [4], ASPeCT (Autocorrelation Side-Peak Cancellation Technique) [3], and Double Estimator [5, 6]. The Single Side Lobe technique provides a robust solution to resolving BOC ambiguity, but it is suitable only for low-precision receivers. Note that the ASPeCT algorithm modifies the shape of the autocorrelation function and eliminates side peaks that can be points of false locking. However, it requires a different and more complex correlator architecture. Double Estimator assumes an additional loop, the Subcarrier Lock Loop (SLL), that tracks subcarrier delay. More correlators are needed, as well as more complex loop implementation. In addition, two tracking techniques have been proposed exclusively for Galileo E1 CBOC tracking: TM61 and Dual Correlator. The TM61 technique generates a one-bit local replica, either BOC(1,1) or BOC(6,1), simplifying the architecture, but also degrading tracking by at least 3 dB [7]. The Dual Correlator technique is based on the investigation of two parallel correlations: one between the incoming MBOC and a BOC(1,1) replica, and one between the incoming MBOC and a BOC(6,1) replica. Each correlation is weighted, and two outputs are linearly added, such that by changing the values of the weights the tracking can be easily modified [8]. These two E1 CBOC tracking techniques assume separate correlations, which degrades the tracking and brings additional complexity. In contrast, the two-step tracking algorithm considered in this paper is based on full four-level bit local replica generation and consists of two steps, coarse and fine, with different tracking parameters. The main requirements that were taken into consideration when defining this algorithm were (1) a relatively low implementation complexity suitable for a mass-market solution, (2) the shape of the E1 CBOC autocorrelation function, and (3) the correlator's structure since it conditions the tracking algorithm's properties. Regarding the second point, since the CBOC autocorrelation function has secondary peaks that can be potential false lock points, one of the main objectives was to design a tracking algorithm that can avoid such false locks or, at least, minimize their occurrence. Therefore, five complex correlators are used with several correlator spacing options (very early (VE) and a very late (VL) correlators have been added to the three conventional Early (E), Prompt (P), and Late (L) correlators). Regarding the third point, we wanted to design a tracking architecture that can minimize the tracking error caused by thermal noise and multipath. Moreover, DP (Dot-Product) and HRC (High Resolution Correlator) discriminators are used equally in the fine tracking step, depending on the tracking conditions.

This paper extends our previous results by fully analyzing the theoretical performance of the previously proposed two-step tracking schemes with respect to tracking accuracy, sensitivity, and robustness, as well as testing the algorithm using real satellite data. The paper is organized as follows. In Section 2, the Galileo E1 CBOC signal is described. Section 3 provides an overview of the two-step tracking algorithm with a thorough analysis of the discriminator curve outputs and discussion of its stability and linearity regions. Section 4 provides theoretical analysis of the tracking loops' performance, especially derivation and calculation of code tracking error and tracking thresholds for different discriminators. A multipath mitigation analysis is provided in Section 5, followed by simulation-based and realistic results provided in Section 6. Finally, conclusions and outlook are provided in Section 7.

2. Galileo E1 CBOC Signal Properties

New generations of Global Navigation Satellite Systems (Galileo, Glonass, Compass, etc.) are rapidly evolving, and old GPS system served localization purposes to users for more than twenty years in positioning that is currently undergoing a modernization process, and new European Galileo system will be available within a few years. MBOC modulation was chosen to be implemented on the Galileo E1 and GPS L1C signals [9]. The power density function of the MBOC modulation is a sum of the 10/11 normalized BOC(1,1) and the 1/11 normalized BOC(6,1) signal spectrum. MBOC is defined in the frequency domain, and two different implementations have been specified for Galileo and GPS: CBOC and TMBOC, respectively. Although having different time domain implementations, the power spectral densities in both configurations of the MBOC(6,1,1/11) are the same.

The GPS L1C signal has a pure BOC(1,1) data channel carrying 25% of the total signal power, while the pilot signal uses a TMBOC modulation with 75% of the total signal power. The Galileo E1 signal on the other hand shares its power equally between data and pilot channels, with both channels using a CBOC modulation based on a four-level subcarrier formed by the weighed sum of BOC(1,1) and BOC(6,1). To acquire 100% of the signal energy, a bandwidth of 16 MHz is needed. The only difference between data and pilot channels besides having different PRN (Pseudo-Random Noise) codes is in the sign of the weighted sum. The data channel is "in phase" (CBOC(6,1,"+")), and the pilot channel is in "antiphase" (CBOC(6,1,"−")). "Antiphase" configuration exhibits better tracking performance: autocorrelation peak is narrower. The analytical expression for the E1 CBOC signal can be represented as

$$\begin{aligned}
s_{\text{Gal}}(t) = {} & \sqrt{\frac{1}{2}}\Big[d_{\text{Gal}}(t)c_d(t)\text{CBOC}(6,1,p,\text{“}+\text{”})(t) \\
& \qquad + c_p(t)\text{CBOC}(6,1,p,\text{“}-\text{”})(t)\Big], \\
\text{CBOC}(6,1,p,\text{“}+\text{”})(t) = {} & \Big(\sqrt{(1-p)}\,s_{\text{BOC}(1,1)}(t) + \sqrt{p}\,s_{\text{BOC}(6,1)}\Big), \\
\text{CBOC}(6,1,p,\text{“}-\text{”})(t) = {} & \Big(\sqrt{(1-p)}\,s_{\text{BOC}(1,1)}(t) - \sqrt{p}\,s_{\text{BOC}(6,1)}\Big),
\end{aligned} \tag{1}$$

and $p = 1/11$ for OS (Open Service), $c_d(t), c_p(t)$ are the data and pilot spreading codes, and $\text{CBOC}(6,1,p,$ "+") and $\text{CBOC}(6,1,p,$"–") are the pilot and data subcarriers, respectively.

2.1. Galileo E1 CBOC Autocorrelation Function. The study of the autocorrelation function is important when trying to evaluate the tracking performance of the signal. The E1 CBOC autocorrelation function for the data and pilot channels, as well as for the combination of the two channels (for the combined tracking case), is shown in Figure 1 assuming an 18 MHz bandwidth. The CBOC autocorrelation function exhibits a very narrow main peak and two side-correlation peaks located at approximately 0.5 chips around

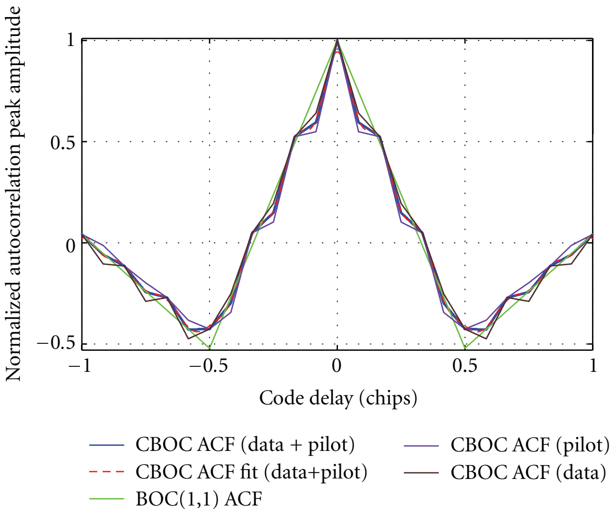


Figure 1: CBOC autocorrelation function for pilot, data, and averaged data/pilot channels and its fit by a sum of sinusoidal functions for bandwidth of 18 MHz.

the main peak. Dangerously, these secondary peaks can be potential false locking points, as explained earlier.

The CBOC autocorrelation function has a narrower main peak when compared to BOC(1,1), but it is not fully linear. The expression for the autocorrelation function can be approximated more simply, as it is done with the expression for the BOC(1,1) autocorrelation function inside the one chip width of the main autocorrelation peak [3]:

$$R_{\mathrm{BOC}(1,1)}(\tau) = 1 - \alpha|\tau|, \tag{2}$$

where α corresponds to the absolute value of the slope of the spreading sequence autocorrelation function's main peak. The value of α is equal to 3 for the BOC(1,1) signal, and it is larger than 3 for the CBOC data/pilot autocorrelation function for the narrow E-L spacing around the main correlation peak. Due to the existence of ripples in CBOC autocorrelation function, the value for the α is not constant and it depends on correlator spacing.

In the following, the first derivative of the autocorrelation function (α) is used as a parameter for the evaluation of the code tracking error and tracking threshold. α varies with the E-L correlator spacing, but for fixed E-L distances it can be considered constant. Analytical expressions for autocorrelation function should be able to provide both the properties of the CBOC modulation and the effects of front-end filtering on the autocorrelation peak. In order to compute α, the analytical expression for the E1 CBOC autocorrelation function is determined by fitting the autocorrelation curve with a sum of sinusoidal functions. By doing so, it is possible to compute α of the filtered autocorrelation function for different E-L correlator spacing. The resulting analytical expression for the CBOC autocorrelation function obtained by fitting the autocorrelation curve and α can be written as [10]:

$$\widetilde{R}_F(t) = \sum_{i=1}^{8} a_i \cdot \sin(b_i \cdot t + c_i),$$
$$\alpha = \left(\frac{d\widetilde{R}_F(x)}{dx}\right)\Bigg|_{x=-d/2}, \tag{3}$$

where t is the distance from the main correlation peak (half of E-L spacing), and a_i, b_i, and c_i, where $i = 1, \dots, 8$, are fitted constants that are different for the data and pilot channels. Since we assume later both data and pilot channels are used for tracking, and thus the averaged CBOC(+) and CBOC(−) autocorrelation is used for fitting. In this case, the infinite bandwidth is used so that 99.99% of the CBOC signal is received. Therefore, three sets of a_i, b_i, and c_i parameters were used. As can be observed from Figure 1, the analytical fit closely follows the averaged CBOC autocorrelation function over the considered range. α for different E-L spacings and three different combinations is provided in Table 1. If the correlator spacing decreases below $d = 1/12$ chips, the slope of the autocorrelation peak decreases as well, because of the rounding effect of the front-end filtering on the autocorrelation peak. Additionally, a smaller E-L spacing reduces the linear tracking region which, at some point, can make the tracking unreliable. Therefore, $d = 1/12$ was chosen as a good tradeoff between accuracy and reliability for the considered bandwidth of 18 MHz.

The existence of secondary code on the pilot channel on top of the primary PRN code of duration 100 ms or 25 chips additionally complicates the tracking and requires techniques for wiping off the secondary code before further process the signal. The main advantages of secondary code are increased resistance to narrow-band interference due to the additional spectrum line, better cross-correlation properties (which mostly help during acquisition to avoid the near-far effects), and more robust data bit synchronization than with the histogram method, although that does not apply to Galileo E1-B/C, since the data bit and PRN code have the same duration.

The algorithm used for secondary code extraction is based on a combination of serial and parallel searches. It is well described in [11, 12]. The search for the primary code phase is performed serially within one primary code length, and the secondary code phase is searched in parallel over the entire length of the secondary code. The algorithm takes into consideration the residual frequency offset, and it is shown to reduce the acquisition time compared to other methods.

3. Two-Step Tracking Scheme and Discriminator Curves Analysis

Achieving the favorable properties of the new signals is possible by increasing the complexity of the tracking loops design. There are many techniques proposed to provide unambiguous tracking, as already explained in the introduction. Most of them are listed with the corresponding tracking architectures in [13, 14] as well as the code tracking

Table 1: Slope of the autocorrelation function for different early-late spacing for data, pilot channels, and averaged data/pilot channel for infinite bandwidth.

	Autocorrelation slope α		
Correlator spacing	Data/pilot averaged	Pilot channel	Data channel
1/20	3.98	4.50	3.39
1/12	5.35	5.96	4.63
1/10	5.51	6.06	4.82
1/5	1.49	0.09	1.89

error comparison. Most of them need complex tracking architecture or modify the correlation function. In this section, we present the analysis about our proposed two-step tracking scheme, simple and with less complex tracking architecture. Carrier and code delay tracking are analyzed separately, putting more weight on code delay tracking.

3.1. Galileo E1 CBOC Carrier Tracking (PLL). The right choice of the PLL discriminator is the first step towards obtaining accurate phase error estimation. The choice of the PLL discriminator is dependent upon the parameters of the E1 CBOC signal structure. The pilot channel alone can be used for the phase error estimation, since there is no data bit on it, and, theoretically, coherent integration can be used for as long as needed. As a consequence, a discriminator that is insensitive to phase jumps can be used. Using pure PLL tracking on the pilot channel as well as longer coherent integration improves carrier tracking sensitivity. It also enhances the carrier tracking loop resistance to receiver dynamics. Although 3 dB are lost by ignoring the data channel for the carrier phase estimation, the noise sensitivity is improved by 3 dB. Therefore, the use of an extended arctangent discriminator has been selected (atan2), providing the widest linear tracking region. The discriminator output can be analytically expressed as

$$D_{\text{atan2}} = \arctan 2\left(\frac{Q}{I}\right). \quad (4)$$

I and Q are in-phase and quadrature-phase correlator outputs for the pilot channel. The extended *arctangent* (four-quadrant) discriminator has an operational range twice as large as the traditional *arctangent* discriminator ($[-\pi, \pi]$). It has good noise resistance performance for high C/N_0 (carrier-to-noise ratio). It can also track the phase modulo 2π, without a half cycle of ambiguity. Although pilot-only tracking is noisier than data/pilot combined tracking, the chance of losing lock is smaller, and the tracking is more stable and less complex.

3.2. Galileo E1 CBOC Code Delay Tracking (DLL). Code delay tracking is more stable than carrier tracking, since it provides the user with more robust measurements and initial estimates of the receiver position through pseudoranges. The main idea behind the considered code tracking algorithm that we call the "two-step tracking technique" [1, 15] is to benefit from the narrowness of the CBOC autocorrelation peak while minimizing the risk of locking onto one of the secondary peaks of the autocorrelation function. This algorithm can also be applied to BOC(1,1) as well as CBOC(6,1,1/11) tracking and consists of two steps. The first step is a coarse tracking step that is used to ensure proper convergence towards the true lock point, since it relies on an unambiguous discriminator. This is achieved using an original unambiguous combination of all five correlators (E, L, P, VE, and VL), where VE and VL are always positioned on the secondary peaks. This ensures that the prompt correlator is indeed located on the main correlation peak.

However, once the code tracking has converged to the main peak, the VEMLP (Very Early Minus Late Power) discriminator that uses five correlators' outputs offers sub-optimal code tracking performance since it does not benefit from the narrow CBOC correlation peak. At this point, the tracking process switches to a fine tracking step, where a DP (or HRC) discriminator is used with a narrow correlator spacing. The description and parameters of the two-step tracking technique are provided in Table 2 [1]. Power in the name VEMLP comes from the fact that the discriminator function contains squared correlator outputs that relate to the power of the signal.

As shown in Table 2, the fine tracking step fully exploits the narrow autocorrelation peak by using (1) a DP (HRC) discriminator with narrow E-L spacing and (2) a reduced loop bandwidth (as enabled by the use of carrier aiding). We proposed an HRC discriminator [16] for the fine tracking step instead of a DP discriminator in order to improve multipath mitigation [1]. More specifically, a combination of traditional tracking (DP) and tracking with an HRC is used. In the case of strong multipath, an HRC correlator is used, since it shows good multipath mitigation, otherwise traditional tracking (DP) is assumed, since it has a wider stability tracking region. Details are provided in Section 5.2. Normalization of the discriminator outputs was also performed in order to provide an unbiased estimate of the small code errors. The use of a bump-jump algorithm [4] minimizes the false lock risk. It measures and compares the received power in the VE, P, and VL correlator outputs and jumps left or right if the VE or VL correlator output power is found to be consistently higher than that of the P. Note that the E-L correlator spacing should remain flexible, depending on the front-end bandwidth of the receiver. We used bandwidth of 18 MHz throughout the paper and generated the results based on this assumption. However, the parameters of the two-step tracking scheme, such as early-late spacing, width of the stability and linearity tracking regions, slope of the autocorrelation functions (as it is provided in Table 1), and tracking thresholds change depending on the available bandwidth. The scheme is designed to be flexible, for example, when decreasing the front-end bandwidth, early-late correlator spacing increases, and so forth.

In order to improve code tracking accuracy, both pilot and data channels are used for the code tracking delay estimation, making it more robust. Common data/pilot code

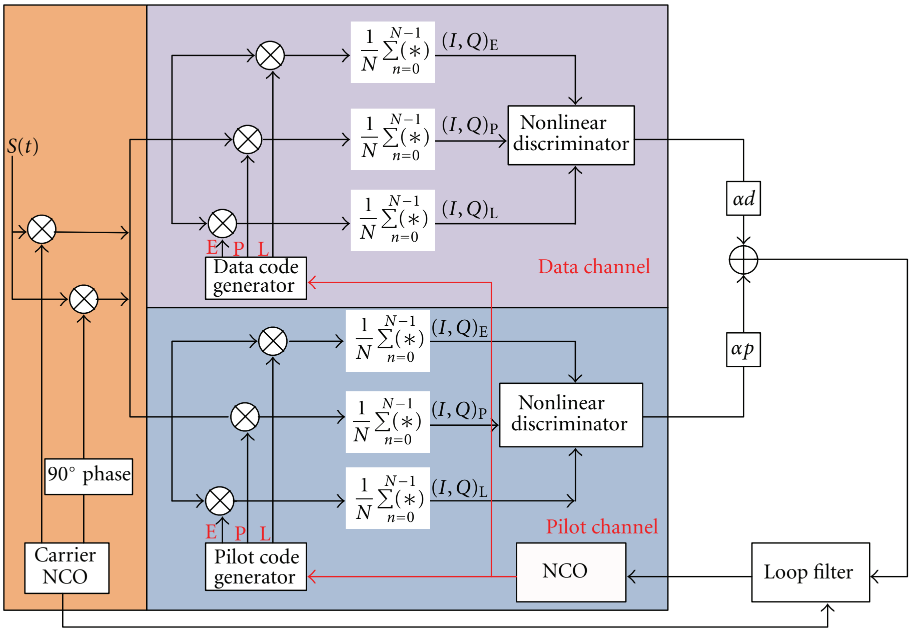


FIGURE 2: Noncoherent collaborative tracking architecture for the fine tracking step of two-step tracking algorithm.

TABLE 2: Two-step tracking scheme description for the front-end bandwidth of 18 MHz.

	Two-step tracking technique		
Parameters	Coarse tracking	Fine tracking (pilot channel)	Fine tracking (data channel)
E-L spacing	[−1/2; −1/4; 0; 1/4; 1/2]	[−1/2; −1/24; 0; 1/24; 1/2]	[−1/2; −1/24; 0; 1/24; 1/2]
Discriminator type	VEMLP	DP/HRC	DP
Loop filter BW	2 Hz	1 Hz	1 Hz
False lock detector	Embedded	Bump-jump	Bump-jump
Carrier aiding	No	Yes	Yes
Advantage	Unambiguous	Accurate multipath gain	Accurate
Disadvantage	Noise and MP increased	Ambiguous	Ambiguous

tracking is performed by linearly combining them noncoherently. The incoming signal is separately combined with data and pilot PRN codes. The outputs are then combined in a noncoherent way, as shown in Figure 2. Coherent combining outperforms noncoherent combining from an accuracy standpoint, but it also requires data bit sign recovery for integration times greater than 4 ms, which increases the loop complexity and becomes unreliable at low C/N_0. Therefore, data and pilot discriminator outputs are averaged using a noncoherent combining method and fed into a unique loop filter that updates both pilot and data NCO (Numerically Controlled Oscillator). The discriminator output can then be written as

$$D_{\text{out}} = \alpha_p D_d + \alpha_d D_p, \tag{5}$$

where α_p and α_d are weights for the pilot and data channels, respectively, applied to the discriminator. The weights have to fulfill the following requirements:

$$\begin{aligned} \alpha_p + \alpha_d &= 1, \\ \alpha_p &= \frac{\sigma_p^2}{\sigma_p^2 + \sigma_d^2}, \\ \alpha_d &= \frac{\sigma_d^2}{\sigma_p^2 + \sigma_d^2}, \end{aligned} \tag{6}$$

where σ_p and σ_d are the variances of the discriminator on the pilot and data channel, respectively. If the same modulation and code tracking loop update rate are used for the data and pilot channels, the same variance will be observed on both channels, reducing the channel combining to a simple averaging [2, 15]. However, since the data and pilot autocorrelation functions are not rigorously identical, the pilot channel having a slightly sharper main lobe peak that could provide slightly better tracking accuracy, it is therefore possible to further refine the weighting scheme so that the tracking relies more heavily on the pilot channel. Theoretical values for the weights for the pilot and data channels are shown to be approximately 0.6 and 0.4, respectively. They were obtained empirically, by computing the standard variation of the data and pilot channels and using a setup that consists of a Spirent GSS8000 simulator [17], connected with a wideband front-end [18], and the data was postprocessed in software.

The resulting collaborative tracking architecture for the fine tracking step of the two-step tracking algorithm is shown in Figure 2, highlighting the DLL part divided in separate tracking architectures for data and pilot channels. After wiping off the carrier using multiplication of the incoming signal with a local replica that is aligned using the information from the carrier NCO, the resulting in-phase and quadrature-phase components are multiplied with five code replicas: E, P, and L as well as VE and VL (not shown in Figure 2). Ideally, if the carrier phase is aligned with the carrier phase of the incoming signal, all the energy will be in the in-phase component. Estimation of the code delay error is performed in the DLL loop, using DLL discriminators. The output is then filtered by the DLL loop filter, and updates are provided for the code delay rate for the NCO. In the next section, analysis of the three different types of discriminators used in the tracking scheme described above is provided.

3.3. DLL Discriminators. As shown in Table 2, three types of discriminators are used in the two-step tracking scheme: VEMLP and DP, for the coarse and fine tracking steps, respectively, and HRC as an option when high multipath error is present in the fine tracking step. The VEMLP discriminator is noncoherent and needs two more correlators (VE and VL) with a fixed distance between them, to mark points that can be potential sources of false locks. It has a higher computational load than DP, which is quasicoherent. After converting a signal to *IF* frequency, *I* and *Q* baseband correlation outputs for the E, L, P, VE, and VL pilot channel can be written as

$$
\begin{aligned}
I_X &= \sqrt{\frac{\mathrm{P}}{2}}\widetilde{R}(\varepsilon_\tau+\beta)\frac{\sin\left(\pi\varepsilon_f T_i\right)}{\pi\varepsilon_f T_i}\cos\left(\varepsilon_{\phi_p}\right)+n_{IX},\\
Q_X &= \sqrt{\frac{\mathrm{P}}{2}}\widetilde{R}(\varepsilon_\tau+\beta)\frac{\sin\left(\pi\varepsilon_f T_i\right)}{\pi\varepsilon_f T_i}\sin\left(\varepsilon_{\phi_p}\right)+n_{QX},
\end{aligned}
\tag{7}
$$

where $\widetilde{R}$ is the correlation of the local spreading code with the filtered incoming spreading code, ε_f is the frequency error, ε_τ is the code group delay error, and ε_ϕ is the carrier phase delay error. The noise components n_{IX}, n_{QX} are independent Gaussian noise components. X relates the correlator type (E, P, L, VE, VL), and β represents the spacing between correlators (for P $\beta = 0$, E $\beta = \delta/2$, for L $\beta = -\delta/2$, VE $\beta = \delta$, and VL $\beta = -\delta$). δ represents the E-L correlator spacing and for the coarse tracking step ($\beta = 2\delta$). Therefore, the output for the types of the discriminators can be written analytically as

$$
D_{\mathrm{VEMLP}} = \sqrt{I_{\mathrm{VE}}^2+Q_{\mathrm{VE}}^2+I_{\mathrm{E}}^2+Q_{\mathrm{E}}^2}-\sqrt{I_{\mathrm{VL}}^2+Q_{\mathrm{VL}}^2+I_{\mathrm{L}}^2+Q_{\mathrm{L}}^2},
\tag{8}
$$

$$
D_{\mathrm{DP}} = (I_{\mathrm{E}}-I_{\mathrm{L}})I_{\mathrm{P}}+(Q_{\mathrm{E}}-Q_{\mathrm{L}})Q_{\mathrm{P}},
\tag{9}
$$

$$
D_{\mathrm{HRC}} = (I_{\mathrm{E}}-I_{\mathrm{L}})+\frac{(I_{\mathrm{VE}}-I_{\mathrm{VL}})}{2}.
\tag{10}
$$

Analytical expressions for the VEMLP, DP, and HRC discriminators functions can be written using filtered autocorrelation function ($\widetilde{R}$) with the approximation that in tracking stage $\sin(\pi\varepsilon_f T_i)/\pi\varepsilon_f T_i \simeq 1$ and $\cos(\varepsilon_{\phi_d}) \simeq 1, \sin(\varepsilon_{\phi_d}) \simeq 0$ as

$$
\begin{aligned}
D_{\mathrm{VEMLP}} = \sqrt{\frac{\mathrm{P}}{2}}\Bigg(&\sqrt{\widetilde{R}^2\left(\tau+\frac{\delta}{2}\right)+\widetilde{R}^2(\tau+\delta)}\\
&-\sqrt{\widetilde{R}^2\left(\tau-\frac{\delta}{2}\right)+\widetilde{R}^2(\tau-\delta)}\Bigg),
\end{aligned}
\tag{11}
$$

$$
D_{\mathrm{DP}} = \sqrt{\frac{\mathrm{P}}{2}}\left(\widetilde{R}\left(\tau+\frac{\delta}{2}\right)-\widetilde{R}\left(\tau-\frac{\delta}{2}\right)\right)\widetilde{R}(\tau),
\tag{12}
$$

$$
D_{\mathrm{HRC}} = \sqrt{\frac{\mathrm{P}}{2}}\left(\widetilde{R}\left(\tau+\frac{\delta}{2}\right)-\widetilde{R}\left(\tau-\frac{\delta}{2}\right)-\frac{\widetilde{R}(\tau+\delta)-\widetilde{R}(\tau-\delta)}{2}\right).
\tag{13}
$$

The performance of the tracking loops in the presence of multipath and code tracking errors can be assessed by studying the discriminators curves. These curves present the discriminators' outputs as a function of code delays (inputs). Tracking parameters of interest that can be derived from the curves are the linear tracking region, defined as the region in which the discriminator responds without any bias, and the stability region (pull-in region), defined as the region where the discriminator reacts in the right direction [3].

A wider stability tracking region means more robust tracking. A wider linear tracking region corresponds to correct and unbiased discriminator response. As already shown in [3], the discriminator output does not directly estimate the input error. In order to obtain unbiased code delay error, the discriminator output has to be normalized by its gain, therefore achieving perfect normalization. Normalization removes amplitude sensitivity, improving performance under rapidly changing SNR (Signal-to-Noise Ratio) conditions and providing unbiased code delay error estimation. As normalization depends on the power of the signal, discriminator outputs should be first normalized by the estimated power of the signal. Therefore, using this approach, the normalization factors for three discriminator types can be written as

$$N_{\text{VEMLP}} = \sqrt{I_{\text{VE}}^2 + Q_{\text{VE}}^2 + I_{\text{E}}^2 + Q_{\text{E}}^2} + \sqrt{I_{\text{VL}}^2 + Q_{\text{VL}}^2 + I_{\text{L}}^2 + Q_{\text{L}}^2}, \tag{14}$$

$$N_{\text{DP}} = (I_{\text{E}} + I_{\text{L}})I_{\text{P}} + (Q_{\text{E}} + Q_{\text{L}})Q_{\text{P}}, \tag{15}$$

$$N_{\text{HRC}} = (I_{\text{E}} + I_{\text{L}}) + \frac{(I_{\text{VE}} + I_{\text{VL}})}{2}. \tag{16}$$

Expressions for the discriminator outputs can be simplified using the assumptions that the estimated code delay error is smaller than half of the correlator spacing, and the E and L correlator values correspond to the points on the autocorrelation function that are on the slope forming its main peak. Normalization provides unbiased estimation of the code delay error within ±$\delta/2$ chips. Using the equations for the correlator outputs provided above and using the assumption that the autocorrelation function can be approximated by small linear parts with different slopes, the discriminator outputs can be written as

$$N_{\text{VEMLP}} = \sqrt{2\text{P}}\sqrt{2 - 2\alpha\varepsilon_\tau(\alpha\varepsilon_\tau - 2\delta) - \alpha\delta\left(\delta - 1 - \alpha - \alpha\frac{\delta}{4}\right)}, \tag{17}$$

$$N_{\text{DP}} = \sqrt{\frac{\text{P}}{2}}(2 - \alpha\delta)(1 - \alpha|\varepsilon_\tau|), \tag{18}$$

$$N_{\text{HRC}} = \sqrt{\frac{\text{P}}{2}}(2 - \alpha\delta)(1 - \alpha|\varepsilon_\tau|) + \sqrt{\frac{\text{P}}{2}}(1 - \alpha\delta). \tag{19}$$

Therefore, using the normalization expressions provided above, as well as the normalization factors that relate to discriminator gain, the expressions for the normalized VEMLP, HRC, and DP discriminators can be written as

$$D_{\text{VEMLP}_n} = \frac{2 - 3\alpha\delta + (5/4)\alpha^2\delta^2}{3\alpha\delta - 4}\frac{D_{\text{VEMLP}}}{N_{\text{VEMLP}}}, \tag{20}$$

$$D_{\text{DP}_n} = \frac{(2 - \alpha\delta)}{2\alpha}\frac{D_{\text{DP}}}{N_{\text{DP}}}, \tag{21}$$

$$D_{\text{HRC}_n} = \frac{2\alpha\delta - 3}{3\alpha}\frac{D_{\text{HRC}}}{N_{\text{HRC}}}. \tag{22}$$

These expressions are obtained from (18), (19), and (20) using the analytical expression for the autocorrelation function and obtaining the slope of the autocorrelation peak at $\alpha = -\delta/2$. Normalization factors were obtained analytically and tested empirically. They include normalization by the discriminator gain obtained by derivation of the discriminator output. Whereas normalization is straightforward for DP discriminator, for VEMLP and HRC discriminators, the following approximations should be used: $\alpha^2\varepsilon_\tau^2 \sim 0$ and $\alpha\varepsilon_\tau \sim 0$. These are valid for small code tracking errors, while the tracking is maintained.

These factors are constant for BOC(1,1) and BPSK signals, but for CBOC they vary depending on the E-L correlator distance. In both tracking steps, the correlators are located on the linear part of the discriminator curve, and the factors can be considered constant for fixed E-L spacing, as shown in Table 3. This table contains the discriminator normalization factors for the combined data/pilot tracking and the slope of the autocorrelation curve as provided in Table 1.

Table 3: Discriminator normalization factor for the two-step tracking scheme (E-L = 1/12, VE-VL = 1/2).

	Discriminator normalization factors	
Discriminator type	Pilot channel	Data channel
DP	0.32	0.28
VEMLP	0.6	0.55
HRC	0.25	0.35

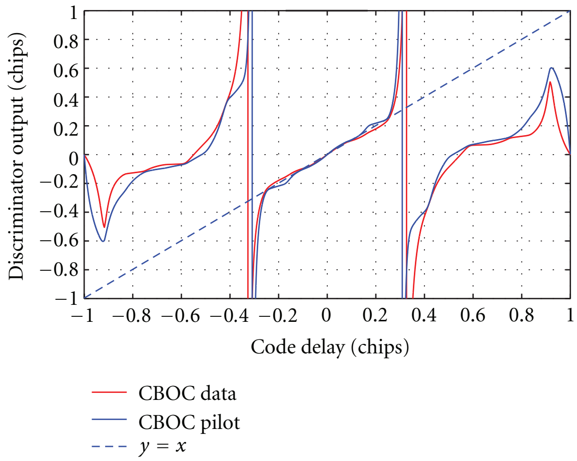


Figure 3: Normalized DP discriminator output for CBOC data and pilot channels and an E-L spacing of 1/12 chips.

The discriminator outputs for DP and VEMLP discriminators are shown in Figures 3 and 4. The most important point that can be observed from these figures is that, by applying normalization, the discriminator curve's linear tracking region increases. Also, the discriminator curves for the pilot channel alone and the VEMLP discriminator (Figure 4) contain additional ripples. These ripples are due to high-frequency components introduced by BOC(6,1) modulation. As expected, the discriminators provide an unbiased response for input code errors below roughly 0.25 chips. For input code errors greater than that, the discriminators' output tends to flatten. This, in turn, should provide a smooth convergence phase. Since VEMLP offers only limited noise and multipath mitigation capabilities, it is therefore used directly after acquisition to ensure convergence of the code tracking loop to the main peak of the autocorrelation function. Upon convergence of both the code and carrier tracking loops, the tracking software switches to the fine tracking step.

As was the case for the VEMLP discriminator, the overall behavior of the DP discriminator is invariant across the two

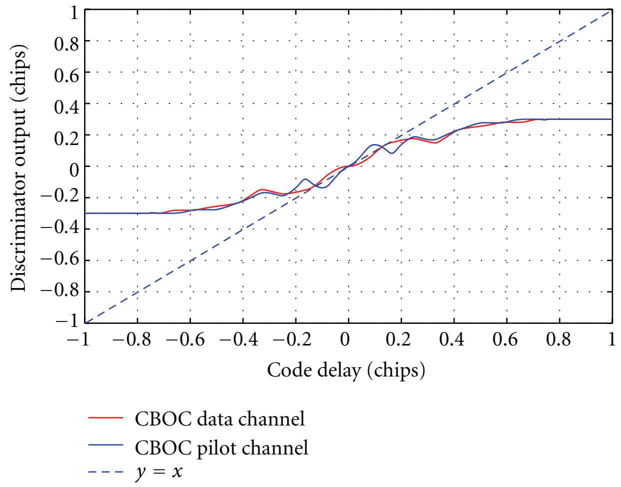


Figure 4: Normalized VEMLP discriminator output for CBOC data and pilot channels and an E-L spacing of 1/2 chips.

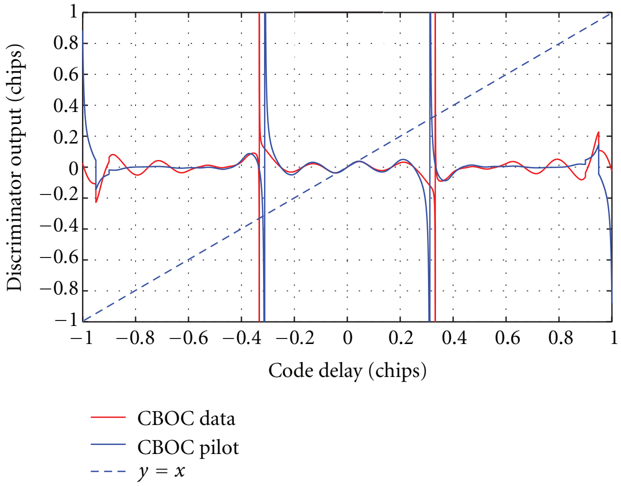


Figure 5: Normalized HRC discriminator output for CBOC data and pilot channels and an E-L spacing of 1/12 chips.

MBOC modulations apart from the additional ripples that can be observed on the two CBOC discriminators. Figure 3 also highlights the false-lock point issue inherent to all BOC and CBOC modulations (e.g., [3]). Indeed, it is obvious from this figure that the zero-crossing observed at approximately 0.5 chips would provide a stable lock point which would lead to a ranging bias of 150 m. The HRC normalized discriminator output is shown in Figure 5. It can be seen that it has a much narrower linear tracking region in comparison to DP and VEMLP.

Again, normalization brings improvement by increasing the linearity region of the discriminator, making tracking more robust. However, the stability region remains similar to the unnormalized case. Normalization thus reveals the true code error present at the output of the discriminator. To conclude this section, the main advantage of using normalization is that the discriminator will perfectly react to a wide range of code delays. Both DP and VEMLP discriminators have similar widths of the linearity and stability regions before performing normalization. After normalization, the DP discriminator shows better stability and linearity (approximately 5% and more). Also, the normalized DP discriminator has vertical asymptotes. This means that the loop will overreact for large errors, and this can cause sudden tracking jumps that might lead to a false lock or degraded tracking.

So far, tracking loop and code discriminators have been described assuming no external disturbances. Since the DLL tracking loop can be affected by different sources of disturbances, such as thermal noise, multipath and signal dynamics, our next step is to investigate the influence of these error sources on the overall behavior of the tracking loop.

4. Code Tracking Error and Tracking Threshold

Code tracking error besides multipath and dynamic stress error is the dominant source of range errors in a GNSS receiver's DLL [19, 20]. Using carrier aiding, the dynamic stress error is negligible; therefore, only the error induced by thermal noise will be considered. The code tracking threshold can be derived using an analytical expression for the thermal noise-induced CBOC code tracking error variance. For the DP discriminator used in the fine tracking step, it is given in [3, 21], whereas, for the VEMLP discriminator, it is derived below. Analytical expressions for both discriminators can be written as

$$\sigma^2_{(\mathrm{DP,VEMLP,HRC})} = \frac{2B_{\mathrm{L}}(1-0.5B_{\mathrm{L}}T_I)S_N(0)}{K_{(\mathrm{DP,VEMLP,HRC})}}. \tag{23}$$

$S_N(0)$ is the noise power spectral density, and $K_{(\mathrm{DP,VEMLP,HRC})}$ represents the discriminator loop gain given by

$$K_{(\mathrm{DP,VEMLP,HRC})} = \left.\frac{dD_{(\mathrm{DP,VEMLP,HRC})}}{d\varepsilon_\tau}\right|_{\varepsilon_\tau=0}. \tag{24}$$

Combining this equation with the expressions for the correlator outputs provided in Section 3.3 and arranging them using the following equality for the filtered version of the autocorrelation function, we derived theoretical expressions for the code tracking errors for both DP and VEMLP discriminators for infinite front-end bandwidth:

$$\sigma^2_{\mathrm{VEMLP}} = \frac{B_{\mathrm{L}}(1-0.5B_{\mathrm{L}}T_I)\left(\left(\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\right)\widetilde{R}_F(\delta/2)\left(\widetilde{R}_F(\delta/2)+\widetilde{R}_F(\delta)\right)+\left(\widetilde{R}_F(0)-\widetilde{R}_F(2\delta)\right)\widetilde{R}_F^2(\delta)\right)}{2(\mathrm{P}/N_0)\left(\widetilde{R}_F(\delta)\left(d\widetilde{R}_F(x)/dx\right)\Big|_{x=\delta}+\widetilde{R}_F(\delta/2)\left(d\widetilde{R}_F(x)/dx\right)\Big|_{x=\delta/2}\right)^2}\cdot\theta,$$

$$\sigma^2_{\mathrm{DP}} = \frac{B_{\mathrm{L}}(1-0.5B_{\mathrm{L}}T_I)\left(\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\right)}{2(C/N_0)\left\{d\widetilde{R}_F(x)/dx\right\}^2\Big|_{x=d/2}}\cdot\chi. \tag{25}$$

Expressions $\alpha_1 = (d\widetilde{R}_F(x)/dx)|_{x=\delta}$ and $\alpha_2 = (d\widetilde{R}_F(x)/dx)|_{x=\delta/2}$ represent the slopes of the autocorrelation function at E-L spacings δ and $\delta/2$. Expressions for θ and χ can be approximated as

$$\theta\left(\frac{\mathrm{P}}{N_0},T_I\right) = 1+\frac{3\left(\widetilde{R}_F^2(0)-\widetilde{R}_F^2(\delta)\right)+\widetilde{R}_F^2(0)-\widetilde{R}_F^2(2\delta)}{(2CT_I/N_0)\left(\left(\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\right)\widetilde{R}_F(\delta/2)\left(\widetilde{R}_F(\delta/2)+\widetilde{R}_F(\delta)\right)+\left(\widetilde{R}_F(0)-\widetilde{R}_F(2\delta)\right)\widetilde{R}_F^2(\delta)\right)}, \tag{26}$$

$$\chi\left(\frac{\mathrm{P}}{N_0},T_I\right) = 1+\frac{1}{(CT_I/N_0)\widetilde{R}_F(0)}.$$

B_{L} is DLL loop bandwidth, T_I is coherent integration time, δ is the early-late spacing, C is the incoming power of the carrier, N_0 is the thermal noise, and $\widetilde{R}_F$ is the filtered correlation function of the incoming signal and $\theta,\chi = f(C/N_0,T_I)$ are factors that are a function of the integration time and carrier-to-noise ratio. The expression for the VEMLP discriminator can be simplified, using the following equality:

$$\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\simeq\widetilde{R}_F\left(\frac{\delta}{2}\right)-\widetilde{R}_F\left(\frac{3\delta}{2}\right). \tag{27}$$

This simplification is valid only for wide E-L correlator spacings (E-L = 1/2) that are used in the coarse tracking step. Using the values for the slope of the autocorrelation function α from Table 1 for different correlator spacings, the code tracking error for the CBOC fine and coarse tracking steps can be computed. From our analysis, we observed that VEMLP discriminator provides very high code tracking noise error (33 m for C/N_0 of 25 dB-Hz) for the basic configuration (E-L = 1/2). High code tracking noise is the price one pays in the coarse tracking step. Decreasing the inner correlator spacing (E-L) while keeping the outer spacing (VE-VL) to one chip decreases the code tracking noise as well.

When increasing the correlator spacing, the DP discriminator code tracking error increases, but less dramatically pace than is observed for the VEMLP discriminator. The DP code tracking noise is very similar for different spacings since the E-L spacing is small (around 1.7 m for C/N_0 of 25 dB-Hz). The code tracking error for an E-L distance of 1/20 is higher than for E-L distance of 1/12. This happens due to front-end filtering, which rounds the autocorrelation peak, increasing the noise. DP outperforms VEMLP for large correlator spacings. Since the first stage of the two-step tracking scheme is short, the large code tracking noise does not greatly influence the overall tracking performance.

As was previously shown in [2], the code tracking error is approximately $\sqrt{2}$ lower for combined pilot and data channels tracking than for single channel tracking. Therefore, the tracking threshold decreases using combined tracking schemes. An analysis of the tracking threshold is provided below, starting from the code tracking noise derivation. Following the analysis provided in [19], the resulting analytical expression for the tracking threshold is obtained, and it is equal to

$$\left(\frac{C}{N_0}\right)_{\mathrm{ThDP}} = \frac{18\cdot W_D\left(1+\sqrt{1+\delta^2/9W_DT_IR_{\mathrm{CBOC}}}\right)}{\delta^2}. \tag{28}$$

In the same way, the resulting tracking threshold for the VEMLP discriminator can be written as

$$\left(\frac{C}{N_0}\right)_{\mathrm{ThVEMLP}} = \frac{18\cdot W_T\left(1+\sqrt{1+\delta^2/9W_TT_IY}\right)}{\delta^2}. \tag{29}$$

Parameters W_T and Y depend on the slope of the autocorrelation function, integration time, and DLL parameters and can be written as

$$W_T = \frac{B_{\mathrm{L}}(1-0.5B_{\mathrm{L}}T_I)\left(\left(\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\right)\widetilde{R}_F(\delta/2)\left(\widetilde{R}_F(\delta/2)+\widetilde{R}_F(\delta)\right)+\left(\widetilde{R}_F(0)-\widetilde{R}_F(2\delta)\right)\widetilde{R}_F^2(\delta)\right)}{2\left(\widetilde{R}_F(\delta)\left(d\widetilde{R}_F(x)/dx\right)\Big|_{x=\delta}+\widetilde{R}_F(\delta/2)\left(d\widetilde{R}_F(x)/dx\right)\Big|_{x=\delta/2}\right)^2}, \tag{30}$$

$$Y = \frac{2\left(\left(\widetilde{R}_F(0)-\widetilde{R}_F(\delta)\right)\widetilde{R}_F(\delta/2)\left(\widetilde{R}_F(\delta/2)+\widetilde{R}_F(\delta)\right)+\left(\widetilde{R}_F(0)-\widetilde{R}_F(2\delta)\right)\widetilde{R}_F^2(\delta)\right)}{3\left(\widetilde{R}_F^2(0)-\widetilde{R}_F^2(\delta)\right)+\widetilde{R}_F^2(0)-\widetilde{R}_F^2(2\delta)}.$$

Following our analysis that cannot be fully presented here, the tracking threshold decreases as the integration time increases and DLL bandwidth decreases. For very long integration durations, extremely low C/N_0 values can be tracked, such as 10 dB-Hz. What was observed is that the results for tracking threshold for two types of discriminators DP and VEMLP are very similar. For VEMLP, as the inner correlator spacing decreases, the tracking threshold decreases as well whereas, for coarse tracking step, does not drop below 20 dB-Hz. For DP, the trend is the same except the fact that for E-L spacing of 1/20 the threshold is increased, due to the rounding of autocorrelation peak.

5. Multipath Mitigation Analysis

Multipath represents a phenomenon that disturbs tracking causing phase offsets in code and carrier and should thus be properly mitigated. For automotive applications in urban environments, the multipath conditions will change continuously, making multipath one of the most disturbing problems of GNSS-based navigation systems in urban environments. As already analyzed and shown in [22], carrier multipath is not critical, causing maximum absolute multipath error of only 0.015 m for E1 CBOC. Code multipath represents a much more important issue and will be discussed here.

5.1. Code Multipath Analysis. Multipath is the main factor that affects the pseudoranges, and it does so through parameters such as the number of multipath signals, geometric path delay, phase, and relative power. Pseudorange measurement is performed by the code tracking loop, which aligns the locally generated PRN code with the PRN code of the incoming signal. The impact of multipath on code tracking accuracy is often represented as an error envelope representing the maximum error resulting from a single multipath with a certain phase, delay, and amplitude. This is called multipath error envelope (MEE). MEEs are computed for each discriminator type in the two-step tracking scheme, as already shown in [1], and for both steps and analyzed the optimal solution in order to mitigate or severely minimize the multipath error.

The first step, a coarse tracking step, uses a noncoherent VEMLP discriminator with wide correlator spacings. The resulting MMEs for the CBOC and BOC(1,1) signals are high MMEs, going up to 25 m as shown in [1]. Large multipath error is the price paid for reliable tracking in the coarse tracking step. As already explained in [1], the E-L correlator spacing should be as low as possible since the multipath error rapidly increases for spacings beyond 1/20 chips. For an E-L spacing of 1/12 chips, the MEE is close to that obtained for 1/20 chips but the linear region is wider which implies more robust tracking in the fine tracking step. For wider E-L spacings between the performances of the CBOC and BOC(1,1) are almost equal. As it it shown in [1], it is clear that CBOC offers a better resistance to long delay multipath than BOC(1,1) tracking. However, for short delays (<8 m), it leads to the same multipath envelope. An additional way to mitigate multipath is to lower the loop bandwidth as much as possible (0.5 Hz). This way, the DLL will not be able to track the multipath-induced error, resulting in more accurate tracking.

5.2. Proposed Fine Tracking Step Implementation. The use of an HRC discriminator is proposed in the fine tracking step of two-step tracking algorithm. A potential issue with HRC is the limited stability tracking region, as shown in Section 3.3. This can be minimized through the implementation of carrier aiding which reduces the dynamics experienced by the discriminator. Since the HRC degrades the postcorrelation SNR and since the coherent integration time on the data channel is limited to 4 ms, it appears that implementing the HRC on the data channel would not provide a very robust solution. However, in the absence of unknown bit transition and with the help of carrier aiding, the coherent integration time can easily be extended to a couple hundred milliseconds on the pilot channel [3]. Consequently, implementation of the HRC on the pilot channel is the desired configuration.

Given the previous discussion, we proposed a CBOC tracking algorithm that combines a two-step tracking technique with the HRC on the pilot channel to lower the multipath error. Different correlator spacing configurations are used on the data and pilot channels. The data channel uses the configuration defined for the fine code tracking, that is, $[-1/2; -1/24; 0; 1/24; 1/2]$, and the pilot channel uses the HRC configuration defined as $[-1/12; -1/24; 0; 1/24; 1/12]$, where the spacings follow the pattern $[-2\delta_{\mathrm{HRC}}; -\delta_{\mathrm{HRC}}; 0; \delta_{\mathrm{HRC}}; 2\delta_{\mathrm{HRC}}]$ and δ_{HRC} is the E-L correlator spacing. The scheme then consists of using the data channel to perform false lock detection and, in case no false lock is detected, using the pilot channel to close the tracking loop, which may use HRC discriminator depending on the following two scenarios.

(1) *High Multipath Case.* Tracking using the HRC discriminator is desired. In this case, the use of the DP discriminator is not recommended as it would feed the combined discriminator with its multipath-induced error.

(2) *Reliably Critical Case.* Tracking using the DP discriminator is desired. In this case, the use of the HRC should be avoided so as to avoid integrating the potentially unreliable HRC discriminator output into the combined discriminator output.

The pilot channel should therefore implement an “either/or” discriminator combination that would alternately rely on the DP or the HRC discriminator. It was shown in [1] that the MEE does not exceed 0.01 chips for the considered E-L spacing. The discriminator output should not be greater than this if the tracking is correctly achieved. Therefore, to decide which discriminator to use, a test based on the DP correlator output is run. If the code tracking error is small (i.e., the DP discriminator output is in its linear tracking range), then the HRC is used. Otherwise, the tracking relies on the DP because of its wider stability range (0.3 chips in one direction). To summarize, the pilot discriminator output can be written as

$$\begin{aligned} D_{\text{pilot}} &= D_{\text{DP}}, \quad \text{if } D_{\text{DP}} \geqslant \delta_{\text{HRC}}, \\ D_{\text{pilot}} &= D_{\text{HRC}}, \quad \text{if } D_{\text{DP}} \leqslant \delta_{\text{HRC}}. \end{aligned} \tag{31}$$

The choice of threshold δ_{HRC} could be made to depend on C/N_0. This approach would bring robustness and will be investigated in the future. When higher C/N_0 is available, strong multipath mitigation could be achieved, while for lower C/N_0 the tracking would remain robust.

6. Results and Discussion

After providing the overview of the theoretical performance of the two-step tracking scheme, we present in this section further the simulation results in order analyze if they are consistent with the theoretically derived results. The data collection setup consists of a Spirent GSS8000 simulator [17] to emulate the Galileo E1 OS signal, a Fraunhofer triple-band front-end [18] with 18 MHz of bandwidth to filter, downconvert, and digitize the signal of interest (i.e., E1 CBOC signal), and a postprocessing architecture that contains a software receiver implemented in MATLAB, based on a modified version of the Kai Borre GPS L1 defined software radio [20].

6.1. Simulation-Based Tracking Results. To test the proposed tracking scheme, we created different scenarios. Using the Spirent simulator's build-in multipath simulation models, we tested two types of multipath models: first type of simple multipath models, that adds a multipath ray with 0.5 chips delay and 6 dB of attenuation to the direct signal, and in the second case, a multipath with a fixed delay of 100 m is added, since this represents the point at which multipath error is the highest. Third model adds three rays that are to the direct signal with delays of 0.5, 0.3, and 0.1 chips with the same attenuation of 6 dB. Fourth group of multipath models tested was based on Land Mobile Multipath (LMM) model, statistical model that emulates direct paths and reflected paths as Rician or Rayleigh, depending on the different type of environment: Urban, Suburban, and Urban Canyon scenarios were tested. Also, two different C/N_0 configurations were investigated: low (24 dB-Hz) and high (45 dB-Hz), as well as static and mobile receiver.

The outputs from the I correlator of the pilot channel for simple multipath model that assumes one reflected path with the delay of 100 m are shown in Figure 6 for all five correlators. The most important point to observe is that all outputs stabilize after approximately 300 code periods from the beginning of tracking. After this point, the tracking is stable. The transition from coarse to fine tracking happens around 1500 code periods, when it is assured that the main autocorrelation peak is being tracked. We could observe that late correlator has higher power at the beginning. The condition for transition from coarse to fine tracking step is related to the PLI (Phase Lock Indicator). If the PLI is less than 0.9 (0.7 in case of low C/N_0), the coarse tracking step proceeds. The transition from coarse to fine tracking step happens only when this value is exceeded. In the Figure 6, the threshold is achieved earlier, but the transition happens later in order to provide a better overview of the scheme.

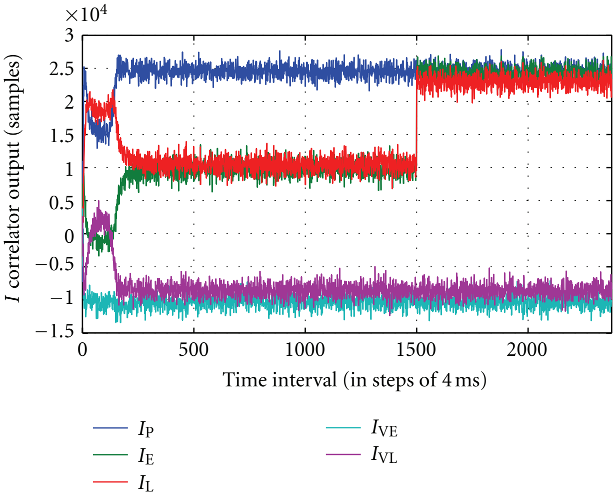


FIGURE 6: I correlation output for two-step tracking scheme with multipath model with fixed offset delay of 100 m.

The DLL outputs for the six different scenarios described above are shown in Figure 7. Note the different convergence times for some cases. Due to the deformation of the autocorrelation main peak, it takes more time for the DLL to converge when multipath is present, than when no multipath is present. Overall, approximately 500 periods are needed (or 2000 ms) to lock to the right correlation peak and then switch to the fine tracking step. The PLI indicators of the two multipath cases analyzed have multiple peaks lower than 0.5 until locked to the right correlation peak.

Once it is assured that the right correlation peak is being tracked, the fine tracking step turns on. Table 4 compares the fine and coarse tracking steps in terms of DLL output's standard deviation. It can be seen that the standard deviation drops by more than a factor of five when switching from coarse to fine tracking for all cases under investigation. It is also important to mention that the tracker succeeds to follow the right autocorrelation peak in all six cases. Peaks in the DLL output are present due to transition from coarse to fine step, as well as due to transition from HRC to DP in the fine tracking step. The PLL discriminator output is shown in Figure 8. It can be noted that variation of PLL error is similar for all cases considered, because carrier tracking is performed independently on the pilot channel only, and the frequency lock is well maintained. Only the secondary code was wiped off according to [12].

It has to be mentioned that in the fine tracking step a bump-jump algorithm was implemented, as previously proposed in [4]. Amplitude comparison was accomplished by a simple up/down counter mechanism. The absolute values of the VE, VL, and P correlators were compared and, if either VE or VL sample is the largest, the appropriate counter was increased and the other decreased. When the set

Table 4: Standard deviation of the DLL outputs for the two-step tracking scheme using six different scenarios.

	DLL standard deviation	
Scenario	Coarse tracking step	Fine tracking step
Low C/N_0 simple multipath model	0.0585	0.0124
Mobile receiver no multipath	0.0192	0.0035
One multipath ray of 0.5 chips delay	0.0169	0.0045
One multipath ray of 100 m delay	0.0180	0.0037
Land multipath model Urban Canyon	0.0154	0.0033
High C/N_0 no multipath	0.0134	0.0022

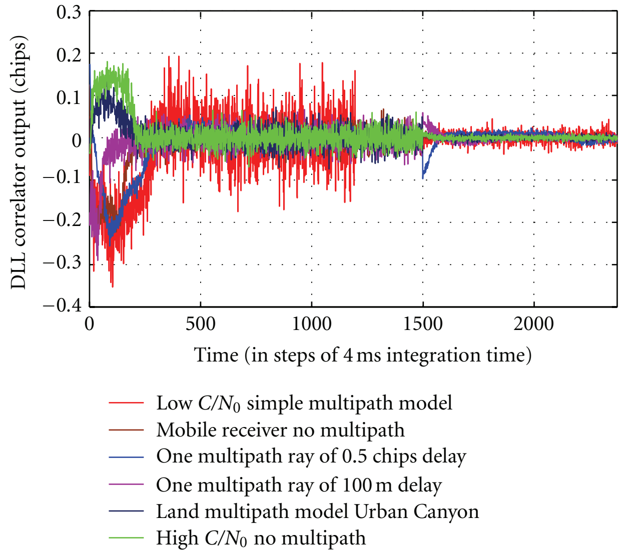


Figure 7: DLL loop output for the two-step tracking algorithm and six investigated scenarios.

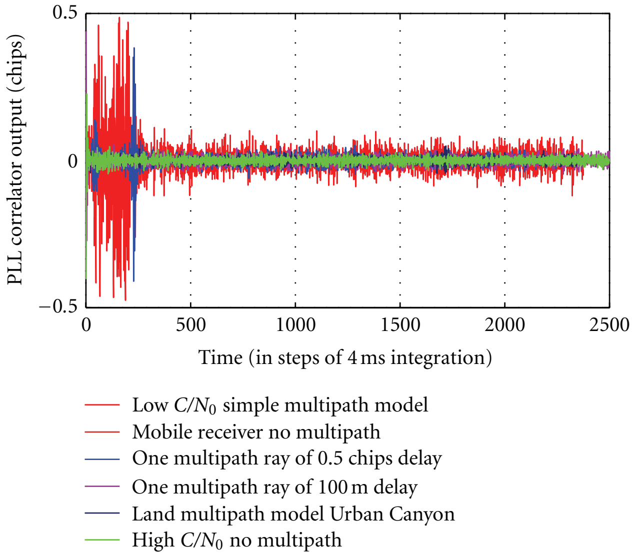


Figure 8: PLL loop output for the two-step tracking algorithm and different multipath scenarios.

threshold was exceeded, the jump to the new peak occurred. Here, we set the threshold to 50, which meant that if the VE or VL was larger than the P for 50 consecutive periods (or 200 ms), the tracking jumped to a new peak.

6.2. Galileo Real Data Tracking Results. On the 12th of December 2011, two Galileo in-orbit validation (IOV) satellites that were launched on 16th of October 2011, PFM and PM-FM2, started transmitting Galileo Open Service signals with pseudorandom code numbers SV11 and SV12 [9]. From the end of December 2011, the signals are available when the satellites are in the direct visibility domain of the receiver. Using the setup described above, we were able to receive signals from both satellites and successfully track them. We tested the two-step tracking scheme using real signals, and we show the first results here.

The setup differs from the simulation setup described above that differs only in the antenna. Instead of using the Spirent simulator output connected to the front-end receiver, we used a fixed, nondirectional rooftop wideband Antcomm antenna, connected via a two-stage amplifier to a front-end. The setup was located at our premises, in IMT Neuchatel, at latitude: 46.519617 degrees and longitude: 6.63221 degrees.

Digital *IF* (*IF* = 12.82 MHz) samples were stored in the memory and subsequently postprocessed, using the software receiver described earlier, on the evening of March 21st, 2012 when the satellites were in good view of the antenna. During recording, the elevation of the Galileo-PMF satellite (SV11) was 45.5 degrees and the elevation of the Galileo-FM2 satellite (SV12) 65.2 degrees.

The acquisition peak acquired in 3D space for satellite SV12 is shown in Figure 9. We can clearly observe the Galileo signal as the correlation peak rising above the noise level. After acquiring the signal, secondary code wipe-off was performed, and the values for the Doppler offset and code offset of the secondary code were passed to the tracking stage.

Correlator outputs for all five correlators are shown in Figure 10. The outputs are stable, and a clear transition from coarse to fine tracking is observed. The DLL discriminator output for both satellites is shown in Figure 11. The output is noisy in the coarse tracking step, but in the fine tracking step it is stable with a much smaller standard deviation and tracking is furthermore stable.

Once the receiver is in tracking mode, C/N_0 at the receiving antenna can be estimated. Approximately 20 code periods (around 100 ms) were used for the estimation of

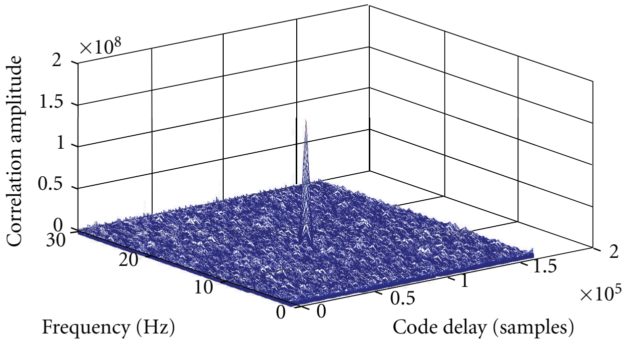


Figure 9: 3D acquisition peak acquired from the Galileo PFM satellite.

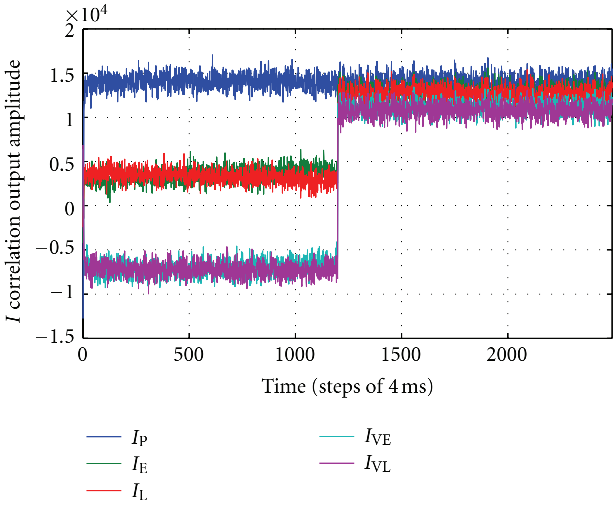


Figure 10: I correlator outputs for the two-step tracking algorithm using real data from Galileo FM2 and PFM satellites.

the C/N_0. The estimated ratio is shown in Figure 12 for both satellites and is very similar C/N_0 profiles for both satellites. Note that the displayed C/N_0 relates to both the data and pilot channels. The single channel C/N_0 is around 3 dB lower. Measured C/N_0 is relatively high due to the clear view of the satellites and good position of the satellites.

7. Conclusions

This paper provided both theoretical and experimental study of the previously proposed two-step tracking algorithm for Galileo E1 CBOC tracking. The proposed algorithm has two independent tracking steps that make the algorithm flexible enough to be easily adaptable to changing signal environments. It was shown that this approach provides robust, accurate, and reliable tracking since the coarse step ensures convergence to the main autocorrelation peak while the fine step minimizes noise and multipath errors.

The analytical expression for the CBOC autocorrelation function was analyzed, as it is more complex than the BOC(1,1) autocorrelation function, and cannot be considered to be fully linear. Depending on the E-L correlator spacing, the slope of the autocorrelation function varies in the range of 1–6, giving different code tracking noise behavior. Optimal correlator spacings were discussed (VE-VL = 1/2, E-L = 1/12), in order to minimize tracking errors. Still, scheme remains flexible, and, depending on the available front-end bandwidth, tracking parameters may vary.

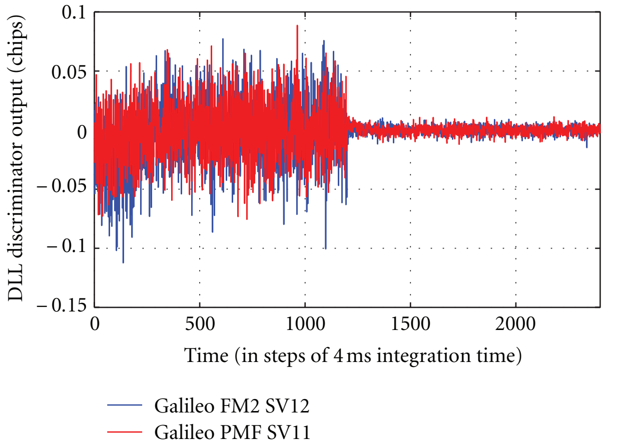


Figure 11: DLL loop output for the two-step tracking algorithm using real data from Galileo FM2 and PFM satellites.

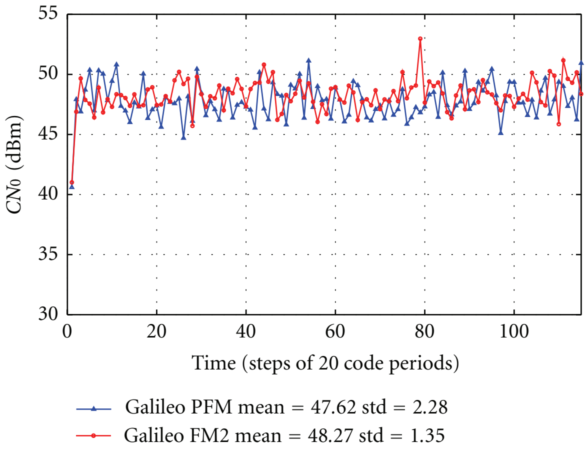


Figure 12: Estimated C/N_0 for the two-step tracking algorithm using real data from Galileo FM2 and PFM satellites.

The analytical expression for VEMLP code tracking error is derived, and this is one of the main contributions of this paper, since it was not provided before according to our best knowledge. Using this expression, it was shown that, although the code tracking error in the first step is very high (∼35 m), stable lock is provided in the first tracking step, and then it decreases exponentially (down to 1.8 m) in the fine tracking step.

We also showed experimentally using theoretical derivations that the two-step tracking algorithm has good tracking sensitivity in both tracking steps and can be used to track CBOC signals with low C/N_0 ratio without losing lock. The tracking stability was analyzed as well in form of discriminator curves analysis. It was shown that the proposed scheme does not lose lock easily since the risk of biased tracking is circumvented. Since the two-step tracking algorithm assumes the usage of three discriminator types (VEMLP, DP, and HRC), each of them were analyzed separately and the normalization parameters optimized in order to improve the two-step tracking algorithm.

The main source of errors, including multipath and thermal noise, were evaluated, and optimal algorithm parameters (shown in Table 2) were obtained through this evaluation, but still staying flexible to different front-end bandwidths, correlator spacing, and so forth. High multipath error (up to 20 m) in the first tracking step is compensated in the fine tracking step using a combination of DP and HRC discriminators, reducing multipath error down to less than a meter. It was shown by simulations that the proposed scheme is able to well mitigate multipath in different scenarios, including both statistical and theoretical models.

Finally, the parameters of the algorithm were further optimized and tested in different configurations through simulations using a Spirent GSS8000 simulator. The two-step tracking algorithm was also implemented in a software receiver, and performance assessments were conducted using real data recorded from the recently launched Galileo PFM and FM2 satellites.

Acknowledgments

The authors are grateful for the financial support received from the Swiss National Science Foundation (http://www.snf.ch/) under Grant 200020 134766/1, as well as from the European Community's Seventh Framework Program (FP7/2007–2013) under Grant Agreement no. 228339.

References

[1] A. Jovanovic, C. Mongrédien, C. Botteron, Y. Tawk, G. Rohmer, and P. A. Farine, "Requirements and analysis for a robust E1 Galileo tracking algorithm in the scope of the GAMMA-A project," in *Proceedings of the Institute of Navigation—International Technical Meeting 2010 (ION ITM '10)*, pp. 973–985, San Diego, Calif, USA, January 2010.

[2] D. Borio, C. Mongrédien, and G. Lachapelle, "Collaborative code tracking of composite GNSS signals," *IEEE Journal on Selected Topics in Signal Processing*, vol. 3, no. 4, pp. 613–626, 2009.

[3] O. Julien, *Design of Galileo L1F receiver tracking loops [Thesis]*, University of Calgary, 2005.

[4] P. Fine and W. Wilson, "Tracking algorithm for GPS offset carrier signals," in *Proceedings of the US Institute of Navigation NTM Conference*, San Diego, Calif, USA, January 1999.

[5] M. S. Hodgart, P. D. Blunt, and M. Unwin, "Double estimator—a new receiver principle for tracking BOC signals," Inside GNSS, pp. 26–36, 2008.

[6] M. S. Hodgart, R. M. Weiler, and M. Unwin, "A triple estimating receiver of multiplexed binary offset carrier (MBOC) modulated signals," in *Proceedings of the 21st International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '08)*, pp. 2295–2304, Savannah, Ga, USA, September 2008.

[7] O. Julien and C. Macabiau, "Two for one—tracking Galileo CBOC signal with TMBOC," Inside GNSS, 2007.

[8] O. Julien and C. Macabiau, "On potential CBOC/MBOC common receiver arcitectures," in *Proceedings of the International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS ITM '07)*, p. 1843, ENAC, 2007.

[9] Galileo Joint Undertaking, Galileo Open Service Signal in Space InterfaceControl Document, GAL OS SIS ICD, Draft 1, 2010.

[10] A. Jovanovic, C. Mongrédien, C. Botteron, Y. Tawk, G. Rohmer, and P. A. Farine, "Implementation and robustness analysis of the two-step CBOC tracking algorithm in the scope of the GAMMA-A project," in *Proceedings of the Proceedings of the ENC GNSS*, Braunscheig, Germany, October 2010.

[11] A. Jovanovic, C. Mongrédien, C. Botteron, Y. Tawk, G. Rohmer, and P. A. Farine, "Implementation and optimization of a Galileo E1 two-step tracking algorithm using data/pilot combining and extended integration time," in *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS '11)*, Portland, Ore, USA, September 2011.

[12] Y. Tawk, A. Jovanovic, J. Leclere, C. Botteron, and P. A. Farine, "A new FFT-based algorithm for secondary code acquisition for Galileo signals," in *Proceedings of the IEEE Vehicular Technology Conference (VTC '11)*, San Francisco, Calif, USA, September 2011.

[13] P. B. Anantharamu, D. Borio, and G. Lachapelle, "Pre-filtering, side-peak rejection and mapping: several solutions for unambiguous BOC tracking," in *Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation 2009 (ION GNSS '09)*, pp. 3182–3195, Savannah, Ga, USA, September 2009.

[14] J. C. Juang and T. L. Kao, "Generalized discriminator and its applications in GNSS signal tracking," in *Proceedings of the 23rd International Technical Meeting of the Satellite Division of the Institute of Navigation 2010 (ION GNSS '10)*, pp. 3251–3257, Portland, Ore, USA, September 2010.

[15] C. Mongrédien, M. Overbeck, and G. Rohmer, "Development and integration of a robust signal tracking module for the triple-frequency dual-constellation GAMMA-A receiver," in *Proceedings of the 23rd International Technical Meeting of the Satellite Division of the Institute of Navigation 2010 (ION GNSS '10)*, pp. 2808–2819, Portland, Ore, USA, September 2010.

[16] M. S. Braasch and G. A. Mcgraw, "GNSS multipath mitigation using gated and high resolution correlator concepts," in *Proceedings of the National Technical Meeting of the Satellite Division of the Institute of Navigation (ION NTM '99)*, pp. 333–342, San Diego, Calif, USA, January 1999.

[17] http://www.spirent.com/Solutions-Directory/GSS8000.aspx.

[18] http://www.iis.fraunhofer.de/.

[19] E. D. Kaplan and C. J. Hegarty, *Understanding GPS—Principles and Applications*, Artech House, 1996.

[20] N. Bertelsen and K. Borre, *A Software Defined GPS and Galileo Receiver*, Birkhauser, Boston, Mass, USA, 2007.

[21] J. W. Betz and K. R. Kolodziejski, "Generalized theory of code tracking with an early-late discriminator part II: noncoherent processing and numerical results," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 4, article 1551, 2009.

[22] A. Jovanovic, Y. Tawk, C. Botteron, and P. A. Farine, "Multipath mitigation techniques for CBOC, TMBOC and AltBOC signals using advanced correlators architectures," in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS '10)*, pp. 1127–1136, Palm Springs, Calif, USA, May 2010.

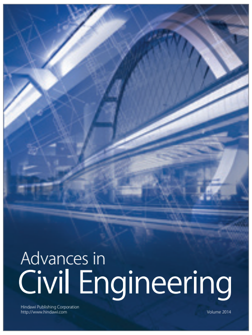
Journal of
Engineering

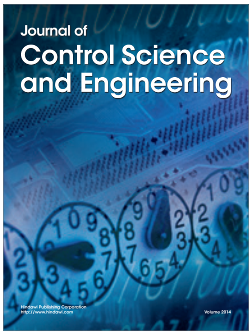
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

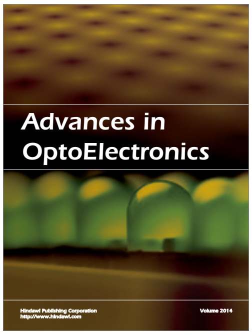
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

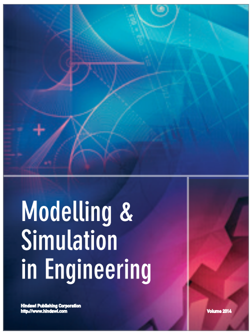
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

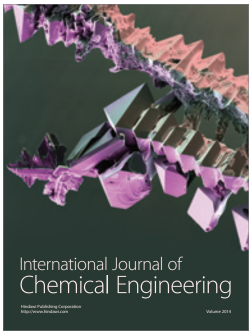

Advances in
OptoElectronics

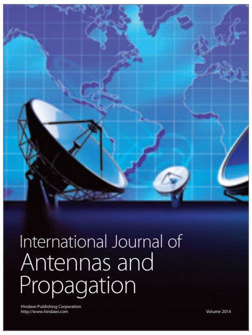
VLSI Design

International Journal of
Navigation and
Observation

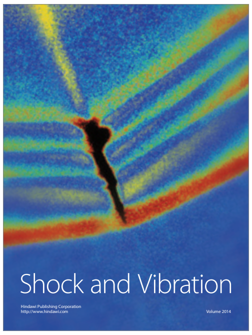

Modelling &
Simulation
in Engineering

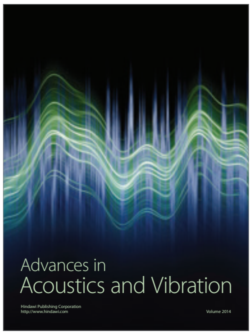
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration