Semi-Analytic Techniques for Fast MATLAB Simulations

Daniele Borio and Eduardo Cano

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/46470

1. Introduction

Advances in electronics and telecommunications are leading to complex systems able to efficiently use the available resources. Fast electronics, complex modulation schemes and correction codes enable transmissions on channels with unfavorable characteristics, coexistence between different services in the same frequency bands and high transmission rates. However, the complexity of such communications systems often prevents analytical characterizations. For example, figures of merit such as the Bit Error Rate (BER) are difficult to determine analytically for transmission schemes involving correction codes and communication channels with Inter-Symbol Interference (ISI) and fading. In such cases, the system is characterized through Monte Carlo simulations [1, 2]. The Monte Carlo framework involves simulations of the whole system under analysis. For example, when considering a communications system, the whole transmission-reception chain is simulated. A large number of sequences are sent through the simulated system and the message recovered by the simulated receiver is compared to the original transmitted sequence. This comparison allows one to determine the average number of transmission errors and compute the BER.

Monte Carlo simulations can be applied to almost any system although their implementation and computation requirements can be significantly high. In addition to this, precision problems can arise when the quantity to be estimated is significantly low. For example, BERs of the order of $10^{-8} - 10^{-9}$ require at least $10^9 - 10^{10}$ simulation runs. Conversely, analytical models have a limited applicability and usually adopt approximations (i.e., model linearization) that can yield a poor description of the system under analysis.

In order to overcome the limitation of Monte Carlo and analytical techniques, semi-analytic approaches have been previously implemented [1, 2]. In a semi-analytic framework, the knowledge of the system under analysis is exploited to reduce the computational load and complexity that full Monte Carlo simulations would require. In this way, the strengths of both analytical and Monte Carlo methods are effectively combined. Semi-analytic techniques are a powerful tool for the analysis of complex systems.

INTECH
open science | open minds

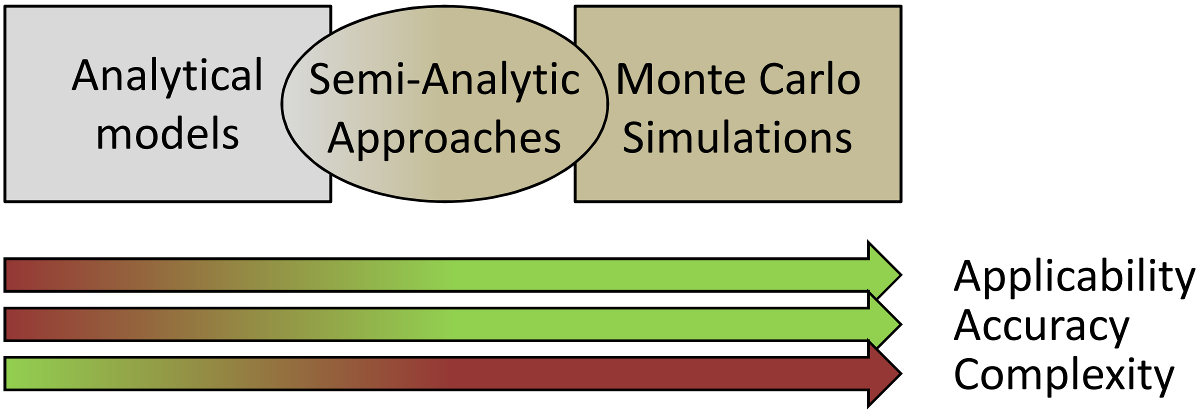

Figure 1. Different approaches available for the analysis of complex systems. Semi-analytic approaches represent a compromise in terms of applicability and complexity (computational and implementation) between analytical models and Monte Carlo simulations.

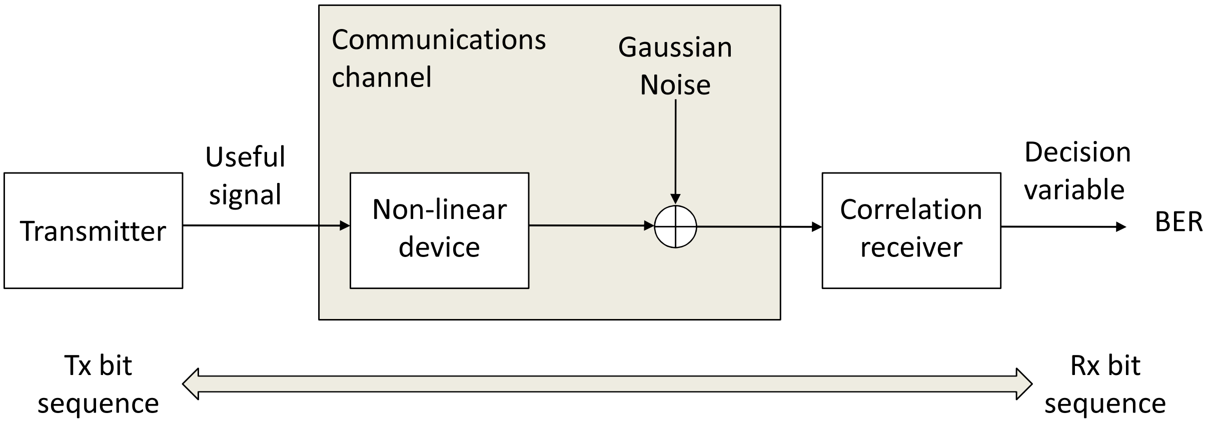

Figure 2. Model adopted for the evaluation of the BER using a semi-analytic approach. The communications channel is modeled as a non-linear device, which affects only the signal component, and the addition of a noise term supposed to be Gaussian.

The characteristics and relationships among the three aforementioned methods are shown in Figure 1: semi-analytic approaches represent a good compromise in terms of applicability and complexity, combining the strengths of Monte Carlo and analytical approaches.

The main goal of this chapter is to provide a general overview of semi-analytic techniques for the simulation of communications systems. Specific emphasis is given to their implementation in MATLAB and two examples from the communications context are analyzed in detail.

Despite their potential, semi-analytic techniques have received limited attention from the communications community. Reference books on simulation of communications systems such as [1], [2] and [3] dedicate only a few pages to this kind of techniques. The focus is usually on the computation of the BER, which represents one of the first applications of semi-analytic techniques in communication system analysis [4, 5]. In this case, the model depicted in Figure 2 is adopted. The communications channel is modeled as a non-linear device, which distorts the signal component alone, with the addition of a noise term that is assumed to be Gaussian. This model is quite general and can be used to represent several communications channels.

A classical example is the model of a transmission chain where a Traveling Wave Tube Amplifier (TWTA) is used to amplify the useful signal before transmission. The TWTA is highly non-linear and can lead to signal distortions. Since the signal is injected into the TWTA before transmission, the noise component entering the amplifier is negligible. In this case, the model depicted in Figure 2 is appropriate for describing the transmission chain including a non-linear amplifier.

The TWTA is a memory-less device and can be characterized using *AM-AM conversion* and *AM-PM conversion* (AM = Amplitude Modulation, PM = Phase Modulation) curves [3]. When a base-band signal model is used, the amplifier input and output are complex quantities; moreover, the response of the device usually depends only on the amplitude (instantaneous power) of the input signal. AM-AM and AM-PM conversion curves define the relationship between the input/output signal amplitudes and phases as a function of the input amplitude. Using these conversion curves, it is possible to simulate the behavior of the TWTA and other non-linear devices.

In the semi-analytic framework, the additivity of the noise component is exploited to compute the BER. More specifically, only the signal transmission chain is simulated and for each possible signal symbol, the Energy per bit (E_b) is computed. Since the noise properties are known, the BER for the ith symbol, s_i, is given by

$$BER_i = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_{b,i}}{N_0}}\right), \tag{1}$$

where $E_{b,i}$ has been obtained by simulating the transmission chain (including the non-linear device) in the absence of noise and transmitting the symbol s_i. The parameter N_0 is the noise power spectral density and it is a known value of the system. Finally, the BER of the system is obtained from

$$BER = \text{E}\left[BER_i\right] = \frac{1}{2}\sum_{i=0}^{N_s-1} p_i\text{erfc}\left(\sqrt{\frac{E_{b,i}}{N_0}}\right), \tag{2}$$

where p_i is the probability that the symbol s_i will be transmitted and N_s is the number of symbols of the signal constellation.

This simple example clearly illustrates the principles of semi-analytic techniques: the analytical knowledge of the system is exploited to reduce the computation load and complexity that full Monte Carlo simulations would require. In this case, only the transmission of the signal component is simulated.

In the literature, several generalizations of the aforementioned BER computation technique have been proposed. For example, [5] considered the case where the noise term at the input of the non-linear amplifier is not negligible. An equivalent model is proposed where the noise at the input is propagated after the non-linearity. [5] also considered the presence of a bandpass channel. More recently, [6] proposed a methodology for estimating the BER in the presence of ISI. All these examples show the potential and flexibility of the semi-analytic approach.

1.1. Building blocks

When considering the previous example it is possible to identify three building blocks that play different roles in the semi-analytic framework:

- **simulation block**
- **estimation block**
- **analytical model**.

The simulation block corresponds to that part of the system that is actually simulated. In the previous example, this block corresponds to the signal generation, the non-linear amplifier and the correlation receiver simulation. These blocks were used to determine the decision variable employed for recovering the transmitted symbol. The analytical model exploits the properties of the system to determine the quantities of interest. In the previous example, the fact that the noise introduced by the communication channel is Gaussian was exploited to determine the BER as a function of the E_b of each transmitted symbol. The estimation block is used as the interface between the simulated and analytical parts of the system. In BER computation case, the simulation part allows one to generate the different decision variables, whereas the analytical model is expressed as a function of energy per bit. The estimation block is used to determine E_b from the simulated decision variables.

The three functional blocks can be connected according to different configurations leading to different types of semi-analytic approaches. In the next section, two of these configurations are briefly discussed. Examples for each type of semi-analytic system are given in Section 2 and Section 3.

1.2. Main configurations

When considering the BER example, it is possible to note that the simulation, estimation and analytical blocks are connected in series. The simulation block is used at first to compute the different decision variables. The estimation block determines the E_b associated with each variable and finally the analytical model is used to compute the BER from the Energy per bit to Noise power spectral density ratio (E_b/N_0).

This type of configuration is defined here as *sequential* since there is no feedback between the different blocks and each element of the chain is run sequentially. The principle of this configuration is shown in the upper part of Figure 3.

A second type of configuration has been recently considered for the analysis of tracking loops in Direct Sequence Spread Spectrum (DSSS) and Global Navigation Satellite System (GNSS) receivers. The most computationally demanding operation in a DSSS/GNSS receiver is despreading, i.e, the correlation of the incoming samples with local replicas of the code and carrier. This operation is performed by the Integrate and Dump (I&D) blocks that rely on simple operations that can be analytically modeled. For this reason, semi-analytic models exploiting the knowledge of the I&D blocks and simulating only the non-linear parts of the system have been developed [7–10]. This resulted in efficient analysis tools, which require reduced processing time with the applicability of Monte Carlo simulations.

Different techniques for modeling the output of the I&D have been suggested. [7] modeled the correlator outputs evaluated by the I&D blocks as linear combinations of independent Gaussian random variables. Correlation among the different correlators was obtained by using, for the generation of different I&D outputs, a subset of the same random variables. This technique becomes complex as the number of correlators increases. Another attempt made in [8] assumed that the correlator outputs were independent which, in general, is not a

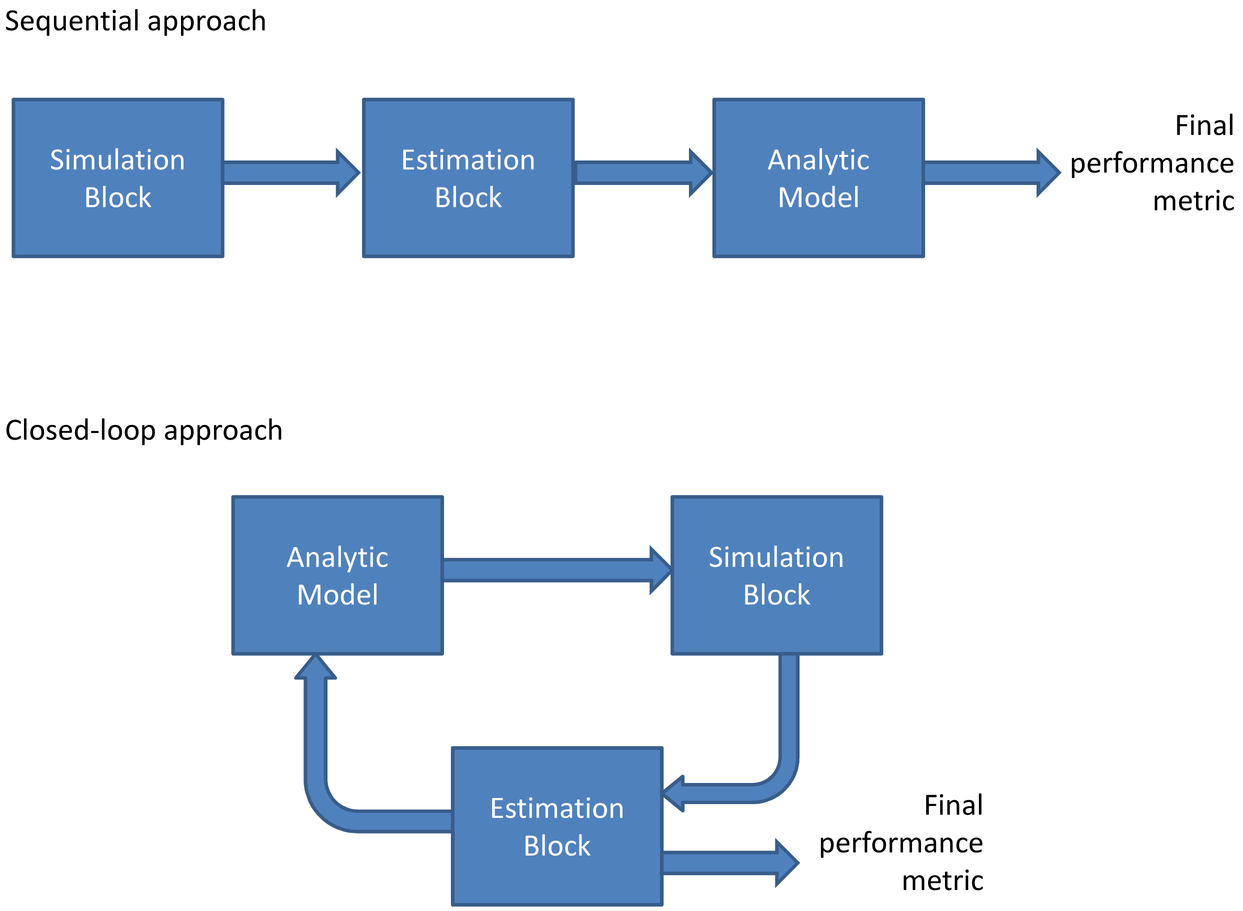

Figure 3. Basic blocks and main configurations adopted in semi-analytic approaches. An estimation block is used for determining key signal and system parameters and interfacing the simulation and analytic components of the system.

realistic condition. Finally, [9, 10] suggested the use of a technique based on the Cholesky decomposition detailed in [11]. This approach allows one to easily generate an arbitrary number of correlated Gaussian random variables. In this way, [9, 10] were able to simulate advanced tracking loops for new GNSS signals.

Regardless of the type of approach used for modeling the correlator outputs, the aforementioned semi-analytic configuration can be represented as in the bottom part of Figure 3. In this case, an analytical model is used to generate quantities that will be propagated by simulation. In the tracking loop case, an analytical model is used to generate the correlator outputs that are then processed through simulations. The non-linear parts of the system are fully simulated and quantities such as the loop discriminator and filter outputs are computed. Finally, an estimation block is used to interface the simulation and analytical components of the semi-analytic scheme. A new estimate of the signal parameters is obtained and used to generate new correlator outputs. The estimation block is also used to compute performance metrics such as tracking jitter or Mean Time to Lose Lock (MTLL) [12].

The three functional blocks described in Section 1.1 are connected in a loop and thus this type of configuration is named *closed-loop approach*. The technique developed by [9, 10] and the closed-loop approach will be detailed in Section 3.

2. Sequential approach: The inter-system interference case

As shown in Figure 3, the sequential approach, at first, requires an initial simulation block to generate the random processes and system functions that cannot be analytically described. Subsequently, the simulated processes are employed by the estimation unit to obtain key parameters required by the analytical model to compute metrics of interest. In the analytical model, the estimated parameters are plugged into mathematical expressions to obtain the desired final metrics. In the sequential architecture, the gain in computational load mainly depends on the simplifications allowed by the analytical model. The use of such a model allows one to simulate only a part of the system and eventually avoid computationally demanding error counting processes.

The computational complexity of Monte Carlo simulations increases significantly when two or more communication systems coexist within the same environment. In this case, it is necessary to account for the interaction of the different systems and determine potential inter-system interference. In addition, computational requirements of Monte Carlo simulations increase dramatically with the number of random elements included in each block of the communication chain. These requirements can be considerably reduced by adopting semi-analytic techniques in which the evaluated metrics are analytically expressed as a function of parameters estimated through simulations. This principle is the core idea behind the sequential semi-analytic approach.

In order to better illustrate the principles of the semi-analytic sequential approach, an inter-system interference scenario is considered in this section. The case of a satellite navigation receiver affected by interference generated by a communications system is considered. The primary system (i.e., the victim system) considered here is a Global Positioning System (GPS) L1 receiver affected by an interference signal caused by third order harmonics of a Digital Video Broadcasting - Terrestrial (DVB-T) signal. A comprehensive description of the sequential approach applied to an inter-system interference case is provided in the following.

The reception of GNSS signals is challenging due to low signal power, possible severe channel conditions and the presence of Radio-Frequency (RF) interference. The presence of RF interference can be particularly troublesome and the performance of a GNSS receiver can vary significantly depending on the type of interference. For this reason, significant research efforts have been devoted to the characterization of the receiver performance in the presence of different types of interference [13, 14]. Furthermore, the impact of interference originated by specific communication technologies, such as Ultra Wideband (UWB) transmissions [15], Distance Measuring Equipment (DME) signals [16] and DVB-T harmonics [17, 18], has been thoroughly investigated. It is noted that the interference impact strongly depends on the strategy adopted by a GNSS receiver for processing the useful signals. Moreover, different impacts are expected depending on the receiver operating mode. The first task of a GNSS receiver is to determine the presence of a specific GNSS signal. This task is accomplished by the acquisition block that implements a statistical test for the detection of useful signals. After acquisition, the useful GNSS signals are passed to the tracking stage that refines the estimates of different signal parameters. Since acquisition and tracking implement different processing strategies, RF interference will affect these two receiver blocks differently. In the following, the acquisition stage is considered. A semi-analytic approach for the analysis of GNSS tracking loops is discussed in Section 3.

The acquisition of a GNSS signal can be formulated as a classical detection problem [19], where the signal of interest is buried in noise. The outcome of the acquisition process is twofold. First, a decision relative to the signal presence is provided. If the signal is present, a rough estimate of signal parameters (defined in the following) is also obtained. The received useful GNSS signal, which is impaired by Additive White Gaussian Noise (AWGN) and interference, is processed by the acquisition block yielding a decision variable. If the decision variable is greater than a decision threshold the signal presence is declared. This decision variable is calculated by using the digital samples provided by the receiver front-end. The signal model and the acquisition process are briefly summarized in the following sections.

2.1. The GNSS signal

The signal at the input of a GNSS receiver, in a one-path AWGN channel and in the presence of RF interference, can be modeled as

$$r(t) = \sum_{l=0}^{L-1} y_l(t) + i(t) + \eta(t), \tag{3}$$

where $y_l(t)$ is the signal transmitted by the lth GNSS L1 satellite, L is the total number of satellites in view, $i(t)$ is the received interference signal and $\eta(t)$ is the noise term.

Each useful signal, $y_l(t)$, can be expressed as

$$y_l(t) = \sqrt{2C_l} d_l\left(t - \tau_{0,l}\right) c_l\left(t - \tau_{0,l}\right) \cos\left(2\pi(f_{RF} + f_{d0,l})t + \varphi_{0,l}\right), \tag{4}$$

where

- C_l is the power of the lth useful signal;
- $d_l(\cdot)$ is the navigation message;
- $c_l(\cdot)$ is the lth pseudo-random sequence extracted from a family of quasi-orthogonal codes and used for spreading the signal spectrum;
- $\tau_{0,l}$, $f_{d0,l}$ and $\varphi_{0,l}$ are the delay, Doppler frequency and phase introduced by the communication channel, and
- f_{RF} is the centre frequency of the GNSS signal.

It is noted that GNSS signals adopt a DSSS modulation. The pseudo-random sequences, $c_l(t)$, allow the simultaneous transmission of several signals at the same time and in the same band. Moreover, $c_l(t)$ sequences are characterized by sharp correlation functions that allow the precise measurement of the signal travel time. The travel time is then converted into distances that allows a GNSS receiver to determine its position.

The pseudo-random sequence, $c_l(t)$, is composed of several terms including a primary spreading sequence and a subcarrier:

$$c_l(t) = \sum_{i=-\infty}^{+\infty} c_{l,(i \bmod N_c)} s_b(t - iT_h). \tag{5}$$

The signal $s_b(t - iT_h)$ in (5) represents the subcarrier of duration T_h, which determines the spectral characteristics of the transmitted GNSS signal. The GPS L1 Coarse/Acquisition (C/A) component is Bi-Phase Shift Keying (BPSK) modulated, whereas the Galileo E1 signal adopts a Composite Binary-Offset Carrier (CBOC) scheme. The sequence $c_{l,i}$, of length N_c, defines the primary spreading code of the lth GNSS signal. In the following, only the BPSK case is considered. The results can be easily extended to different subcarriers.

A GNSS receiver is able to process the L useful signals independently since the spreading codes are quasi-orthogonal. Therefore, expression (3) can be simplified to

$$r(t) = y(t) + i(t) + \eta(t), \tag{6}$$

where the index l has been dropped for ease of notation.

The received signal in (6) is filtered and down-converted by the receiver front-end. Filtering is of particular importance since it determines which portion of the interfering signal, $i(t)$, will effectively enter the receiver. After down-conversion and filtering, the input signal is sampled and quantized. In this analysis, the impact of quantization and sampling is neglected. After these operations, (6) becomes:

$$r_{BB}[n] = y_{BB}(nT_s) + i_{BB}(nT_s) + \eta_{BB}(nT_s) = y_{BB}[n] + i_{BB}[n] + \eta_{BB}[n], \tag{7}$$

where the notation $x[n]$ is used to denote discrete time sequences sampled at the frequency $f_s = \frac{1}{T_s}$. In addition, the index "BB" is used to denote a filtered signal down-converted to baseband. Furthermore, the signal $y_{BB}[n]$ in (7) can be written as

$$y_{BB}[n] = \sqrt{C}d\left(nT_s - \tau_0\right) c\left(nT_s - \tau_0\right) \exp\left\{j2\pi f_0 nT_s + \varphi_0\right\}. \tag{8}$$

The noise term, $\eta_{BB}[n]$, is AWGN with variance σ^2_{BB}. This variance depends on the filtering, down-conversion and sampling strategy applied by the receiver front-end and can be expressed as $\sigma^2_{BB} = N_0 B_{RX}$, where B_{RX} is the front-end bandwidth and N_0 is the Power Spectral Density (PSD) of the input noise $\eta(t)$. The ratio between the carrier power, C, and the noise PSD, N_0, defines the Carrier-to-Noise density power ratio (C/N_0), one of the main signal quality indicators used in GNSS.

2.2. The DVB-T interfering signal

The interference term in (6), $i(t)$, originates from DVB-T emissions. The DVB-T system is the European standard for the broadcasting of digital terrestrial television signals and has been adopted in many countries, mainly in Europe, Asia and Australia. The standard employs an Orthogonal Frequency Division Multiplexing (OFDM)-based modulation scheme operating in the VHF III ($174 - 230$ MHz), UHF IV ($470 - 582$ MHz) and UHF V ($582 - 862$ MHz) bands [20]. It is noticeable that none of these bands fall within the bands allocated for GNSS signals. However, the second harmonics of UHF IV and third order harmonics of UHF V could coincide with the GPS L1 band, and, thus cause harmful interference. The case of the third order harmonics of a DVB-T signal is considered here. The DVB-T transmitted signal can be represented as

$$i_{DVB-T}(t) = \frac{1}{\sqrt{M}} \sum_{p=0}^{N_d-1} \sum_{h=0}^{M-1} I_{p,h} \exp\left(\frac{j2\pi ht}{M}\right), \tag{9}$$

where M is the modulation order, h is the subcarrier index, p is the symbol index, N_d represents the total number of transmitted symbols and $I_{p,h}$ models the hth constellation point of the pth symbol. Here, the term "subcarrier" should not be confused with the subcarrier used to modulate the GNSS signals in (5). In the OFDM context, several components are transmitted in parallel on different overlapping frequency bands. The term subcarrier denotes each individual transmitted component. In GNSS, the subcarrier is an additional component that modulates the transmitted signal and plays a role analogous to the carrier used for the signal up-conversion.

Third order harmonics are the consequence of the malfunctioning of the transmitter electronics. In particular, the presence of these harmonics are due to the non-linearities of an amplifier. The output of an amplifier can be modeled using a polynomial expansion of the amplifier input/output function:

$$p(t) = \sum_{n=1}^{\infty} a_n i^n_{DVB-T}(t), \tag{10}$$

where a_n are the polynomial coefficients of the Taylor series expansion of the amplifier input/output function. This type of model is an alternative to the AM-AM and AM-PM conversion functions discussed in Section 1 for the TWTA case.

The terms of order $n > 1$ in (10) model the amplifier non-linearities and the ratios a_n/a_1 are expected to be small for $n > 1$. Since only the third harmonics will fall into the GPS L1 band, the interference signal at the antenna of a GNSS receiver is given by

$$i(t) = a_3 i^3_{DVB-T}(t). \tag{11}$$

The signal $i(t)$ is filtered and down-converted by the receiver front-end and signal $i_F(t)$, the filtered version of $i(t)$, will affect receiver operations.

Finally, the interference term in (7) can be modeled as $i_{BB}[n] = i_F(nT_s)$.

2.3. The acquisition process

After signal conditioning, the sequence $r_{BB}[n]$ is correlated with local replicas of the useful signal code and carrier as shown in Figure 4. Since the code delay, τ_0, and the Doppler frequency, f_0, of the useful signal in (8) are unknown to the receiver, several delays and frequencies are tested by the acquisition block. In addition to this, several correlators, computed using subsequent portions of the input signal $r_{BB}[n]$, can be computed in order to produce a decision variable less affected by noise and interference. In this way, the output of the kth complex correlator can be expressed as

$$S_k = \frac{1}{N} \sum_{n=kN}^{(k+1)N-1} r_{BB}[n] c\left(nT_s - \tau\right) \exp\left\{-j2\pi f_d n T_s - j\varphi\right\}, \tag{12}$$

where τ, f_d and φ are the code delay, Doppler frequency and carrier phase tested by the receiver. The parameter N is the number of samples used for computing a single correlation output and $T_c = NT_s$ defines the coherent integration time. It is noted that the computation of correlation outputs is essential for the proper functioning of a GNSS receiver and they are both used in acquisition and tracking modes [21]. To further improve the acquisition performance,

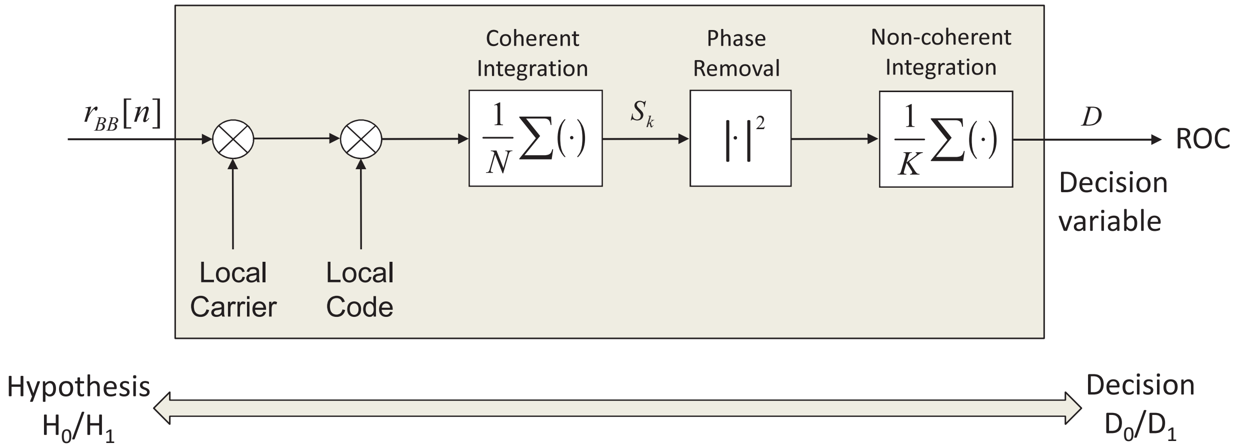

Figure 4. Schematic representation of the operations performed by the acquisition block of a GNSS receiver.

non-coherent integration can be implemented as illustrated in Figure 4. More specifically, the impact of the navigation message, $d(\cdot)$, is removed through squaring, $|S_k|^2$, and the final decision variable is computed as

$$D = \frac{1}{K}\sum_{k=0}^{K-1}|S_k|^2, \tag{13}$$

where K is the total number of correlation samples that are non-coherently integrated. It should be noted that for $K = 1$ only coherent integration is used. In order to determine the signal presence, the receiver compares D with a decision threshold, β. If D is greater than β then the useful signal is declared present.

It is noted that, as in any binary test, two hypotheses are possible:

- H_0: the signal is not present or it is not correctly aligned with the local code and carrier replica .
- H_1: the signal is present and the local code and carrier replica are aligned.

The null hypothesis, H_0 assumes that the correlator outputs, S_k, are made of noise alone. Since the pseudo-random sequences, $c(\cdot)$, are selected to have good autocorrelation properties, if the code delay and Doppler frequencies tested by the receiver do not match the parameters of the input signal, $y_{BB}[n]$, then the useful signal component is almost completely filtered out at the correlator output. Thus, also in this case, the H_0 hypothesis is verified.

Furthermore, H_1 is the alternative hypothesis and assumes that the signal is present and the local code and carrier replica are perfectly aligned. If H_1 is declared, then rough estimates of τ_0 and f_0 are also obtained.

Depending on the result of the test, $D > \beta$, two decisions can be taken by the receiver

- D_0: the signal is declared not present
- D_1: the signal is declared present

	D_0	D_1
H_0	Signal absence correctly declared	False alarm
H_1	Missed detection	Signal detection

Table 1. Confusion matrix describing the four events that can happen in the binary test performed by the acquisition process.

and the four events described in Table 1 can occur.

The off-diagonal events in Table 1 correspond to the different errors that the acquisition block can commit. The following probabilities are usually associated with the events in Table 1:

$$P_d(\beta) = \text{Prob}(D > \beta | H_1) \quad \text{Probability of detection} \tag{14}$$

and

$$P_{fa}(\beta) = \text{Prob}(D > \beta | H_0) \quad \text{Probability of false alarm.} \tag{15}$$

The probabilities of missed detection and correct signal absence decision are obtained as $1 - P_d(\beta)$ and $1 - P_{fa}(\beta)$, respectively. The performance of the acquisition process is characterized in terms of Receiver Operation Curves (ROC) [22], which plots the detection probability as a function of the false alarm rate. ROC curves capture the behavior of a detector as a function of the different decision thresholds.

2.4. The semi-analytic approach

The goal of the sequential semi-analytic approach considered in this section is the evaluation of the ROC in the presence of DVB-T interference. The full Monte Carlo approach would consist of simulating the full transmission/reception scheme shown in Figure 5 and generating several realizations of D both under H_0 and H_1. Probabilities of detection and false alarm would then be determined through error counting techniques. This approach is computationally demanding and does not exploit the analytical knowledge of the system. More specifically, under the hypothesis that the correlator outputs S_k are independent and identically distributed (i.i.d.) complex Gaussian random variables with independent real and imaginary parts, it is possible to show [23] that

$$P_{fa}(\beta) = \exp\left\{-\frac{\beta}{2\sigma_n^2}\right\} \sum_{i=0}^{K-1} \frac{1}{i!} \left(\frac{\beta}{2\sigma_n^2}\right)^i \tag{16}$$

and

$$P_{det}(\beta) = \text{Q}_K\left(\sqrt{K\lambda}; \sqrt{\beta\sigma_n^2}\right), \tag{17}$$

where $Q_K(a;b) = \int_b^{+\infty} x\left(\frac{x}{a}\right)^{K-1} \exp\left\{-\frac{x^2+a^2}{2}\right\} I_{K-1}(ax)dx$ is the generalized Marcum Q-function of order K. The function $I_K(\cdot)$ is the modified Bessel function of first kind and order K. In (16) and (17), σ_n^2 is the variance of the real and imaginary part of S_k. The parameter λ is given by

$$\lambda = \frac{|\text{E}[S_k]|^2}{\sigma_n^2} \tag{18}$$

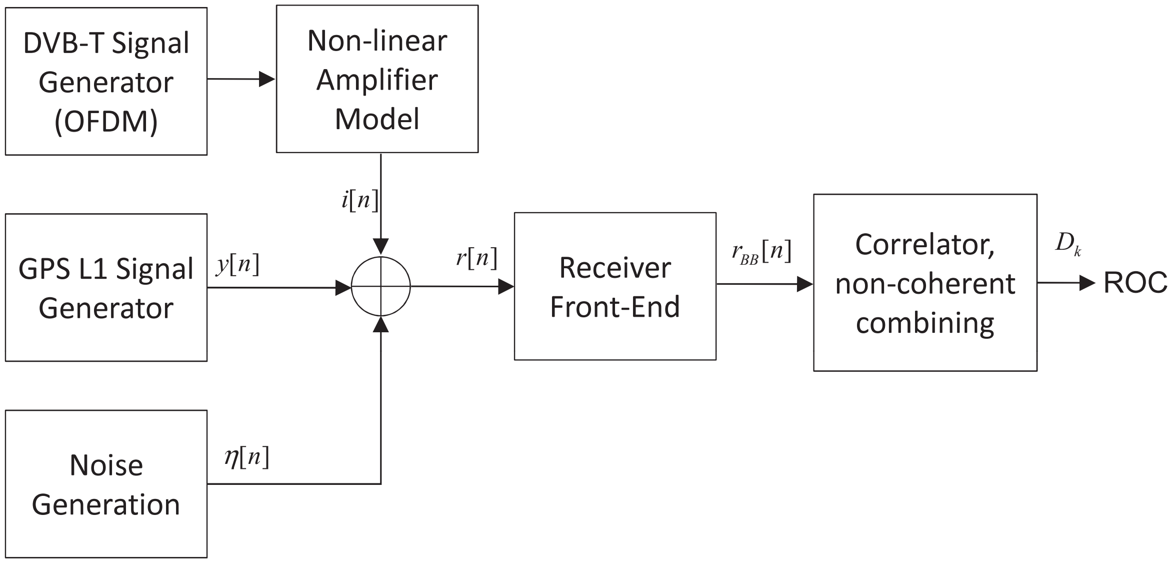

Figure 5. Schematic representation of the full Monte Carlo simulation system for the ROC evaluation of the acquisition of a GPS L1 signal in the presence of DVB-T third order harmonics.

and defines the Signal-to-Noise Ratio (SNR) at the correlator outputs.

The correlator output can be considered i.i.d. complex Gaussian random variables even in the presence of DVB-T interference. More specifically, the large number of terms in the sum performed in (12) allows one to invoke the central limit theorem and assume S_k is Gaussian. Independence derives from the down-sampling performed by the correlators. Since only one correlator is produced every N samples, the statistical correlation between subsequent correlators is significantly reduced. The lack of correlation translates into independence for Gaussian random variables. Thus, models (16) and (17) can be used and the only parameters that need to be estimated are σ_n^2 and λ.

The analytical knowledge of the system can be further exploited to simplify the evaluation of σ_n^2 and λ. In particular, since $i_{BB}[n]$ and $\eta_{BB}[n]$ are modeled as zero mean random processes, they only contribute to the variance of the correlator outputs. Thus, neglecting residual errors due to delay and frequency partial misalignments, yields

$$|\mathrm{E}\left[S_k\right]|^2 = C, \tag{19}$$

where C is the useful signal power and is one of the known parameters of the system. Thus, λ can be derived from C and σ_n^2.

Finally, exploiting the linearity of the correlation process, it is possible to express S_k as

$$S_k = S_{r,k} + S_{\eta,k} + S_{i,k}, \tag{20}$$

which is a linear combination of a useful signal term, derived from $y_{BB}[n]$, a noise term, derived from $\eta_{BB}[n]$, and an interference term derived from $i_{BB}[n]$. The variance σ_n^2 can be obtained as

$$\sigma_n^2 = \frac{1}{2}\mathrm{Var}\left\{S_k\right\} = \frac{1}{2}\mathrm{Var}\left\{S_{\eta,k}\right\} + \frac{1}{2}\mathrm{Var}\left\{S_{i,k}\right\}. \tag{21}$$

Using the results derived in [24], [13] and [23], it is possible to show

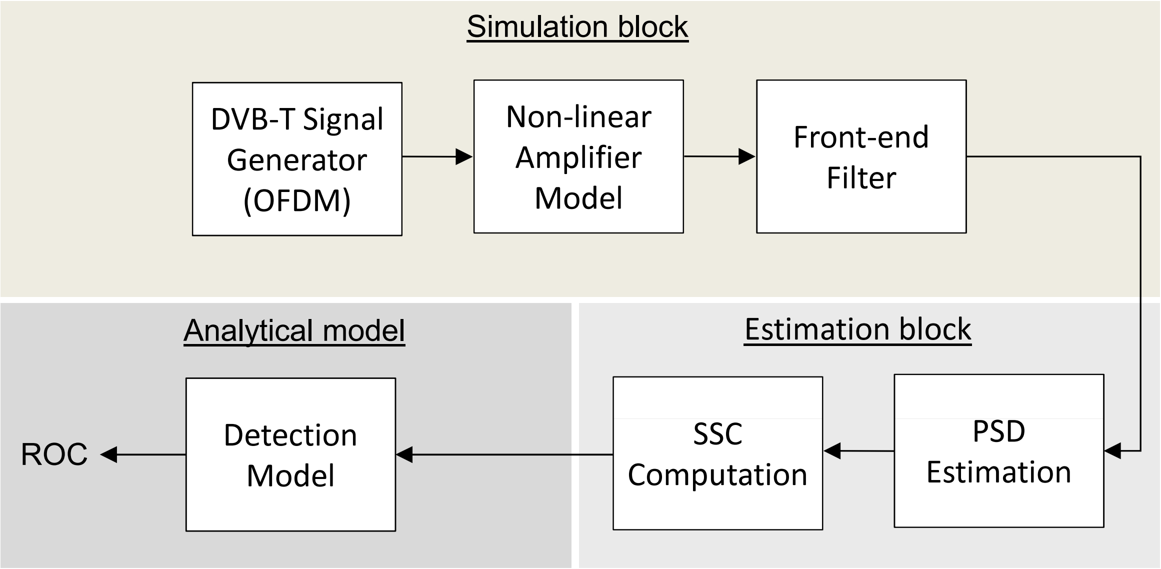

Figure 6. Schematic representation of the semi-analytic approach adopted for the evaluation of the ROC in the presence of DVB-T interference. The three functional elements of the semi-analytic approach are highlighted in different colors.

$$\frac{1}{2}\mathrm{Var}\left\{S_{\eta,k}\right\} = \frac{N_0}{2T_c} \tag{22}$$

and

$$\frac{1}{2}\mathrm{Var}\left\{S_{i,k}\right\} = \frac{C_i}{2T_c}k_a, \tag{23}$$

where C_i is the interference power and k_a is the Spectral Separation Coefficient (SSC) defined as [13, 24]

$$k_a = \int_{-B_{RX}/2}^{B_{RX}/2} G_i(f)G_c(f)df. \tag{24}$$

The function $G_i(f)$ in (24) is the normalized PSD of the DVB-T interference signal after front-end filtering. In addition, $G_i(f)$ is normalized such that

$$\int_{-B_{RX}/2}^{B_{RX}/2} G_i(f)df = 1. \tag{25}$$

The function $G_c(f)$ models the effect of the correlation on the interfering signal. Correlation can be modeled as an additional filtering stage and $G_c(f)$ can be shown to be well approximated by the PSD of the subcarrier used in the despreading process. Also, $G_c(f)$ is normalized to have a unit integral. It is noted that different subcarriers lead to different $G_c(f)$ and thus, $i_{BB}[n]$ will have different effects depending on the type of modulation considered.

The only unknown parameter in the previous equation is the SSC, which needs to be estimated using Monte Carlo simulations. Also, the interfering DVB-T signal is fully simulated. The resulting signal is filtered by the receiver front-end and the sequence $i_{BB}[n]$ is obtained. The samples of $i_{BB}[n]$ are used to estimate the normalized PSD, $G_i(f)$. This can be easily obtained using the MATLAB functions developed for spectral analysis. In this case, the `pwelch` function is used. The function $G_i(f)$ is used to compute the SSC, which is then used to determine the system ROC.

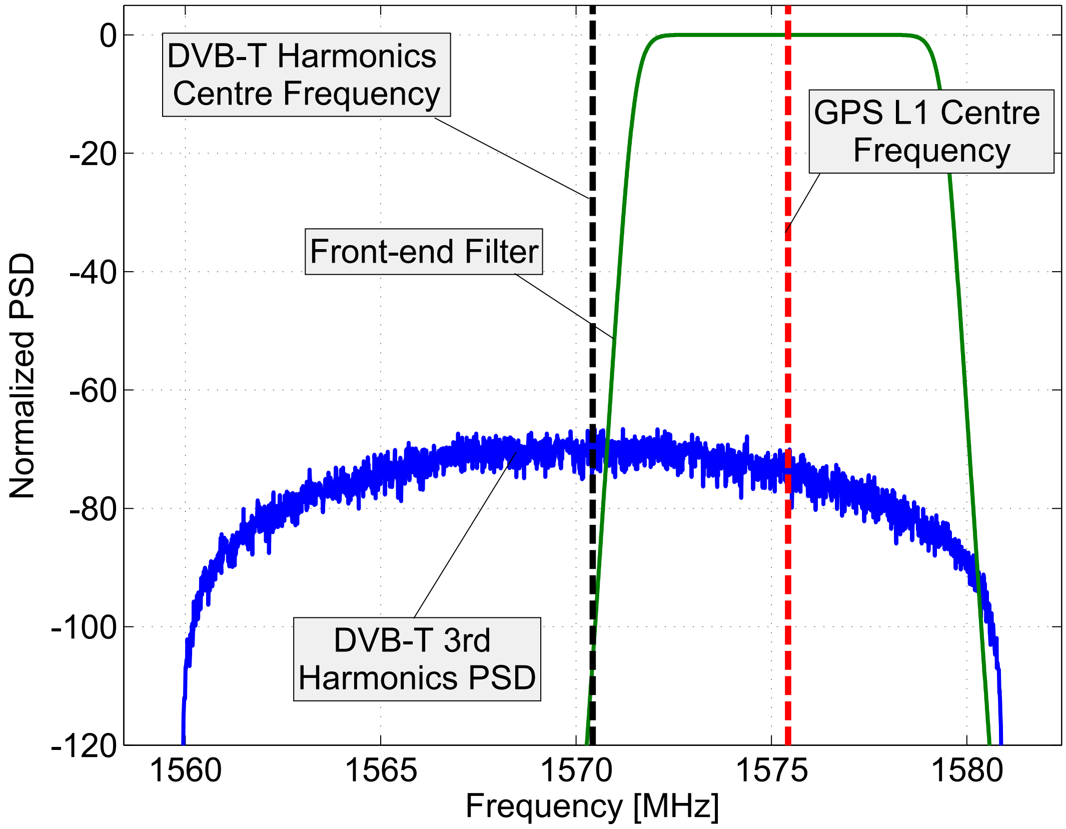

Figure 7. Representation of a normalized PSD realization of the third harmonic of the DVB-T signal and the frequency response of the GPS L1 front-end filter.

The developed semi-analytic approach is shown in Figure 6 where the simulation, estimation and analytic components are clearly highlighted.

2.5. Performance comparison

A comparison between a full Monte Carlo simulation and a semi-analytic technique, implemented for the evaluation of the acquisition performance of a GPS L1 receiver impaired by third order harmonics of a DVB-T signal, is presented in this section. Initially, the DVB-T interfering signal in time domain is programmed in MATLAB by following the DVB-T standard and the non-linear amplifier model, as illustrated in Figure 5. Note that the simulation of the interfering signal is required for both Monte Carlo and semi-analytic techniques. Subsequently, the estimated PSD of the interfering signal, needed for the estimation of the SSC in the semi-analytic method, is obtained by applying the `pwelch` function of MATLAB. A realization of the normalized PSD of the interfering signal is depicted in Figure 7. The centre frequency of the interfering signal is set to $f_I = f_{RF} + \Delta f$, where Δf is the frequency shift of the interference signal with respect to the centre frequency of the GPS L1 signal. The impact of selecting different values of Δf on the acquisition performance of a GPS L1 receiver is analyzed in [25]. Furthermore, the frequency response of the GPS L1 front-end filter is also plotted in Figure 7. In this case, the selected filter bandwidth is 8 MHz.

Sample results comparing ROC curves obtained using semi-analytic and Monte Carlo simulations are shown in Figure 8. The parameters used for the analysis are reported in

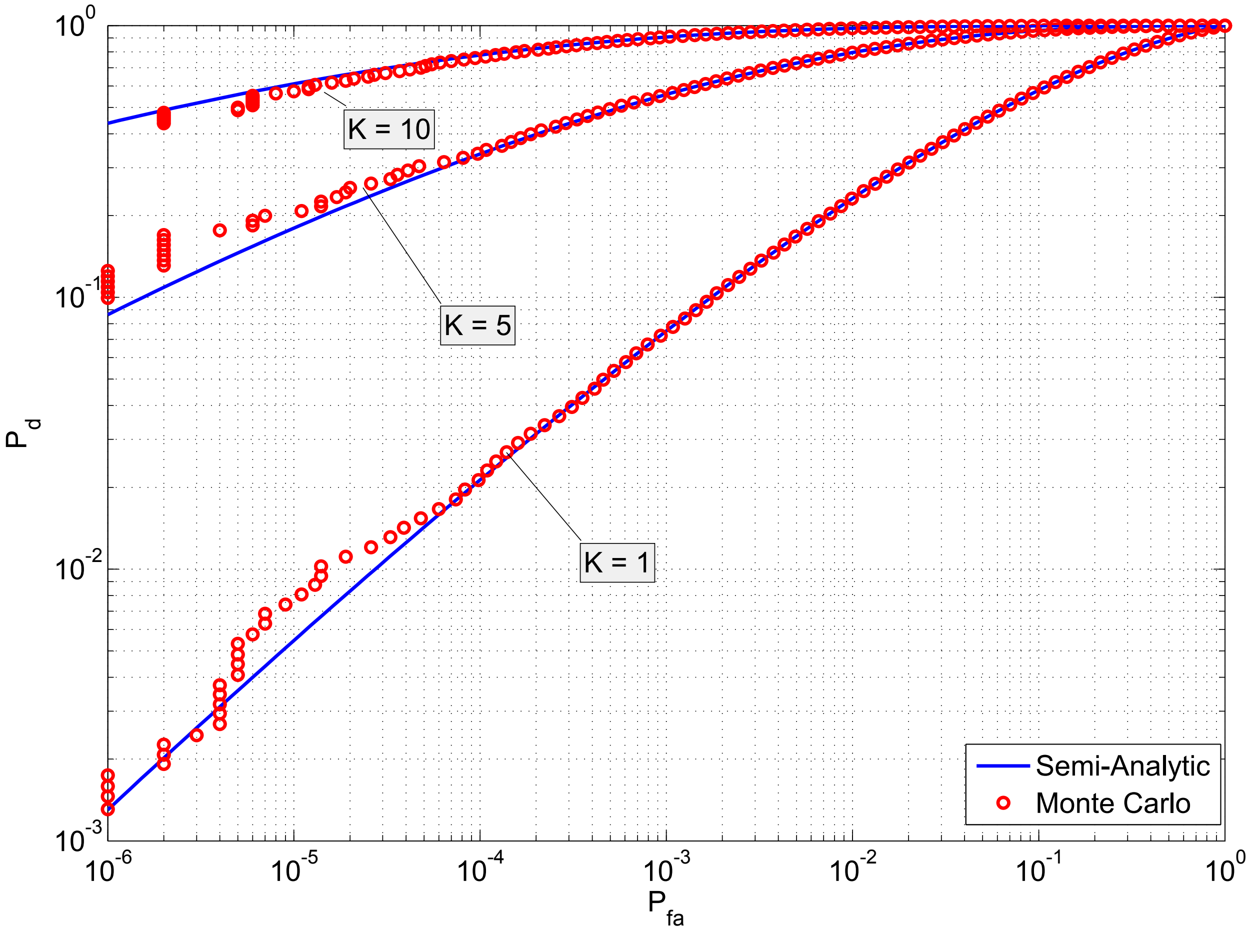

Figure 8. Comparison between ROC curves obtained using semi-analytic and Monte Carlo simulations. The semi-analytic framework considered provides increased precision and requires a lower computational complexity.

Parameter	Value
C/N_0	35 dB-Hz
Coherent integration time, T_c	1 ms
Interference to signal power ratio, $\frac{C_i}{C}$	30 dB
Centre frequency difference, Δf	0 Hz
Receiver bandwidth, B_{RX}	8 MHz
Number of Monte Carlo Simulation runs	10^6

Table 2. Parameters used for the evaluation of the ROC curves shown in Figure 8.

Table 2. From Figure 8, it can be observed that the Monte Carlo and semi-analytic approaches provide similar results and the curves obtained using the two methods overlap. However, the semi-analytic approach provides increased precision, particularly when small values need to be estimated, and a significant reduction in terms of computational complexity. Full Monte Carlo simulations require the implementation of the full transmission/reception chain and the evaluation of the ROC with a computational complexity significantly higher than that of the semi-analytic approach described above.

Additional results relative to the impact of DVB-T interference on GNSS can be found in [25].

3. Closed-loop approach: Digital tracking loops

As anticipated in Section 1, a second configuration, called *closed-loop approach*, has been recently proposed for the simulation of digital tracking loops in DSSS/GNSS receivers. The Semi-Analytic Tracking Loop Simulations (SATLSim) toolbox is a set of MATLAB functions implementing the semi-analytic closed-loop approach for the analysis of digital tracking loops. The SATLSim toolbox has been developed by [9, 10] and can be downloaded from the following websites:

- `http://www.ngs.noaa.gov/gps-toolbox/SATLSim.htm`
- `http://plan.geomatics.ucalgary.ca/publications.php`.

In the following, the closed-loop approach for the simulation of digital tracking loops is considered and the MATLAB code developed in the SATLSim toolbox is briefly analyzed.

A description of the correlator model used for reducing the computational complexity of the system is at first provided. The samples given by (7) at the input of a GNSS receiver are processed by the different functional blocks with different objectives. The acquisition process described in Section 2 is the first stage of a GNSS receiver and has the goal of determining the signal presence and provide a rough estimate of its parameters. These parameters include the code delay τ_0 and Doppler frequency f_0.

If the signal is successfully acquired then different tracking loops are used to refine the estimate of the signal parameters. A Delay Lock Loop (DLL) is usually used to provide accurate estimates of the code delay, τ_0, and track delay variations due to the relative motion between receiver and satellite. The Doppler frequency, f_0, is recovered using either a Frequency Lock Loop (FLL) or a Phase Lock Loop (PLL). If a PLL is used then the carrier phase, φ_0, is also estimated. The code delay and carrier phase allow the receiver to determine its position whereas the Doppler frequency can be used for computing the user velocity.

As indicated in Section 2, a subcarrier can be used for shaping the spectrum of the transmitted GNSS signal and improving its robustness against multipath. The presence of a subcarrier makes code tracking more complex since the correlation function of the transmitted signal may have multiple peaks. More specifically, fine delay estimation is obtained by maximizing the correlation between input signal and local code: the correlation function is maximized only when the delay of the locally generated code matches the delay of the input signal. The presence of several peaks in the correlation function may cause the DLL to converge to a local maximum causing biases in the delay estimation. For this reason, several solutions have been proposed to avoid lock on secondary correlation peaks [26, 27]. An effective solution is represented by the Subcarrier Lock Loop (SLL) proposed by [27]. In this case, the subcarrier is seen as a periodic waveform that further modulates the transmitted signal. The delays of code and subcarrier are decoupled and estimated separately. In this way, the ambiguous one-dimensional signal correlation is projected in an unambiguous bi-dimensional function. In the following, the joint simulation of DLL and SLL is considered.

In a GNSS tracking loop, the incoming signal is correlated with several locally generated code and carrier replicas and different correlator outputs are produced. This process is analogous to the correlation operations described in Section 2 and is performed by the I&D blocks.

Each correlator output is a function of the input signal and the parameters previously estimated by the tracking loop. The correlator outputs are passed to the non-linear discriminator that produces a first estimate of the tracking error that the loop is trying to minimize. The tracking error is filtered and passed to the Numerically Controlled Oscillator (NCO) that is used for generating new local signal replicas.

Efficient tracking loop simulations can be obtained by substituting the I&D blocks with their analytical model. More specifically, a correlator output can be modeled as:

$$\sqrt{C}\frac{\sin\left(\pi\Delta f_d T_c\right)}{\pi\Delta f_d T_c} R_l\left(\Delta\tau_d, \Delta\tau_s\right)\exp\left\{j\Delta\varphi\right\} + \eta_c, \tag{26}$$

where

- Δf_d and $\Delta\varphi$ are the residual frequency and phase errors;
- $\Delta\tau_d$ and $\Delta\tau_s$ are the code and subcarrier delay errors. The delay $\Delta\tau_s$ is present only when a SLL is used to correctly align the signal subcarrier [27];
- $T_c = NT_s$ is the coherent integration time where N is the number of samples used to compute a single correlator;
- $R_l\left(\Delta\tau_d, \Delta\tau_s\right)$ is the correlation function between incoming and locally generated code and is a function of both code and subcarrier delay errors. When the SLL is not used, $R_l\left(\Delta\tau_d, \Delta\tau_s\right)$ is replaced by the standard code correlation function;
- η_c is a zero-mean noise term whose variance depends on the input noise power, front-end filtering and the correlation process operated by the I&D blocks. More details on the properties of η_c can be found in [9].

From (26), it is possible to reconstruct the correlator outputs given the estimation errors generated by the tracking loops. Thus, the correlation process does not need to be simulated and only the estimation errors are determined using a Monte Carlo approach. Based on this principle, the simulation scheme shown in Figure 9 can be adopted for the fast simulation of digital tracking loops.

The functional elements in Figure 9 have been grouped to form the simulation block, the analytical model and the estimation part. The analytical model is used to convert the signal parameter errors, Δf_d, $\Delta\varphi$, $\Delta\tau_d$ and $\Delta\tau_s$, into the signal components of the correlator outputs. At the same time, the analytical model is used to determine the variance and correlation of the different noise terms used to simulate η_c. Since the noise components are simulated using parameters determined by the analytical model, the "noise generation" block is shared between the analytical and simulation parts. The remaining parts of the loop, including the non-linear discriminator, loop filter and NCO, are fully simulated. Finally, the estimation block determines the residual signal parameter errors by comparing true values (determined by the simulation scenario) and estimates produced by the NCO.

By modifying these functional blocks, it is possible to simulate different tracking loops. In the simulation scheme implemented in the SATLSim toolbox, a new estimate of the tracking parameters (Doppler frequency, carrier phase and code and subcarrier delays) is generated by

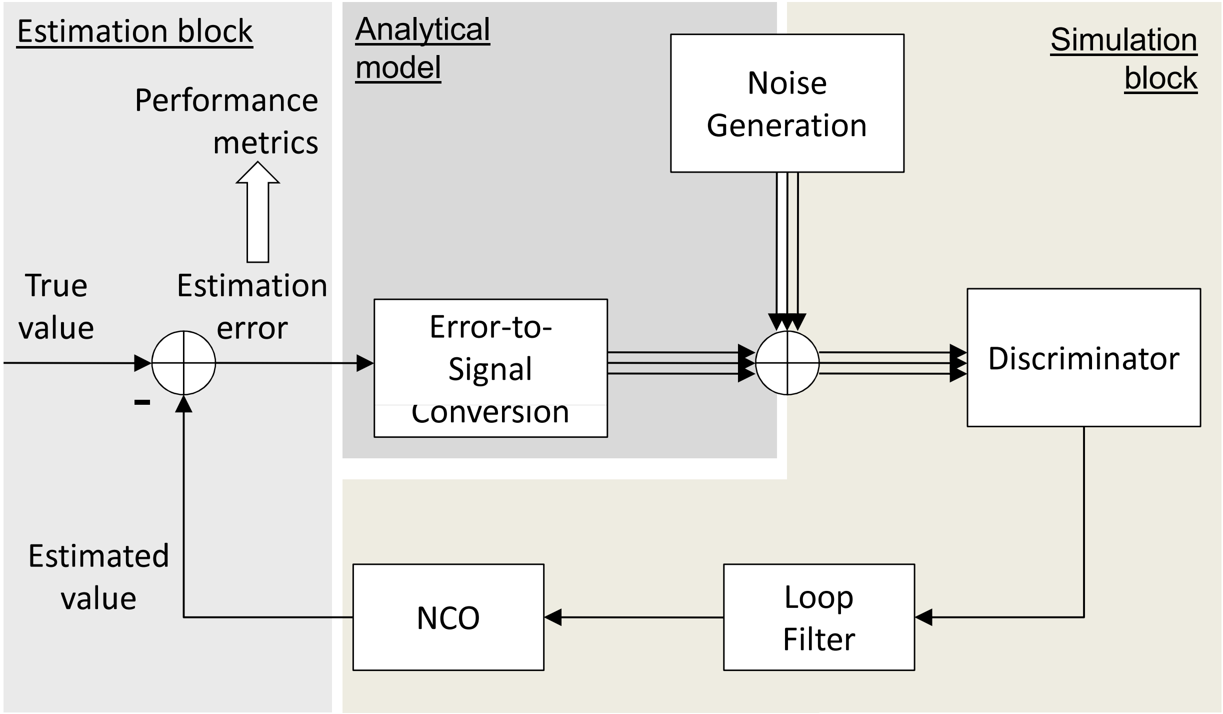

Figure 9. Semi-analytic scheme adopted for the simulation of GNSS tracking loops. Each element of the scheme proposed for the analysis of tracking loops has been implemented in a different function of the SATLSim toolbox.

an NCO model. This model accounts for the integration process performed by a real NCO and different update equations can be used [28]. A commonly used model is the rate-only feedback NCO [28], characterized by the following update equation:

$$\hat{\varphi}_k = \hat{\varphi}_{k-1} + \frac{T_c}{2}\left(\delta\hat{\varphi}_{k-1} + \delta\hat{\varphi}_{k-2}\right), \tag{27}$$

where $\hat{\varphi}_k$ denotes the k-th estimate of the tracking parameter under consideration and $\delta\hat{\varphi}_k$ is its estimated rate of change. The rate $\delta\hat{\varphi}_k$ is generally provided by the loop filter. It is noted that when several parameters are considered, equation (27) is used to update each term independently. The new parameter estimate is compared to the true value and a new estimation error is computed. This error is then used for the generation of the signal component at the output of the I&D block using equation (26). The noise term, generated separately, is then added to the signal component. When several correlators are required, the correlation among the different noise components has to be accounted for. This is simulated using the approach described in [9].

The operations required to convert the correlator outputs into a new estimate of the parameter rate, $\delta\varphi_k$, are fully simulated and correspond to the functional blocks that can be found in a real tracking loop. For instance, the correlator outputs are used to update the nonlinear discriminator and the loop filter. It is noted that a similar simulation scheme can be used for analyzing Kalman filter based tracking. In this case, the correlator outputs are fed to a Kalman filter that is used to produce new estimates of the tracking parameters.

```
Initialization:
    • Initial parameters: sampling frequency, integration time…
    • Loop filter design (FilterDesign.m)
    • Input (true) parameters generation
Main Simulation Loop
Loop on the Early-minus-Late spacing:
    Loop on the C/N_0 values:
        • Noise generation (GenerateNoiseVector.m)
        Loop on the simulation runs:
            1. NCO update (UpdateNCO.m)
            2. Evaluation of the estimation error
            3. Error-to-Signal conversion (GenerateSignalCorrelation.m)
            4. Signal and Noise combining
            5. Discriminator update (UpdateDiscriminator.m)
            6. Loop Filter update (UpdateFilter.m)
        End Loop on the simulation runs
        • Evaluate tracking jitter
    End Loop on the C/N_0 values
    • Plot tracking results
End Loop on the Early-minus-Late spacing
```

Figure 10. Structure of the SATLSim toolbox and list of the different MATLAB functions.

3.1. Code structure

The structure of the code developed in the SATLSim toolbox is provided in Figure 10. In this case, the code is used to estimate the tracking jitter of the loop as a function of different parameters, such as the Early-minus-Late spacing and the input C/N_0. In particular, the non-linear discriminator may use several correlators to compute the cost function that the loop is trying to minimize. A DLL usually requires at least two correlators, named Early and Late correlators, computed for the delays

$$\hat{\tau} \pm \frac{1}{2} d_s \tag{28}$$

where $\hat{\tau}$ is the best code delay estimate and d_s is the Early-minus-Late spacing. Early and Late correlators are computed symmetrically with respect to the best delay estimate and the non-linear discriminator computes a cost function proportional to the misalignment between these two correlators. Since the code correlation function is symmetric, the output of the discriminator is minimized when $\hat{\tau}$ corresponds to the delay of the input signal. The SLL works using similar principles. The performance of DLL and SLL depends on the Early-minus-Late spacing that is a simulation parameter. The C/N_0 is used to determine the correlator amplitude and the variance of the noise component, η_c.

The parameters required for initializing the simulation procedure are accessible through the function `InitSettings`. These parameters include the sampling frequency, the loop bandwidth and the coherent integration time that are used to design the loop filters through the function FilterDesign. In the code provided, standard formulae from [21] are used. However, FilterDesign can be modified in order to adopt a different approach, such as the controlled-root formulation proposed by [28]. During the initialization phase, the true input parameters are also generated. The simulation core consists of three nested loops, on the Early-minus-Late spacing, for different C/N_0 values and for the number of simulation runs. The loop on Early-minus-Late spacing can be absent if, for example, only a PLL is considered. For each Early-minus-Late spacing and for a fixed C/N_0, a noise vector containing the noise components of the correlator outputs is generated. The vector length is equal to the number of simulation runs and all the noise components are generated at once for efficiency reasons.

All intermediate results, such as the discriminator and loop filter outputs, are stored in auxiliary vectors and are used at the end of the loop on the simulation runs to evaluate quantities of interest such as the tracking jitter.

In the code provided, theoretical formulae for the computation of the tracking jitter are also implemented and used as a comparison term for the simulation results.

3.2. Standard PLL (PLL.m)

The simulation of a standard PLL requires the generation of the Prompt correlator alone (GenerateSignalCorrelation). The Prompt correlator is the output of the I&D block computed with respect to the best delay estimate provided by the loop [21]. For this reason, the noise generation (GenerateNoiseVector) simply consists of simulating a one dimensional complex Gaussian white sequence with independent and identically distributed real and imaginary parts with variance [9]

$$\sigma_n^2 = \frac{1}{C/N_0 T_c}. \tag{29}$$

When simulating a standard PLL alone, perfect code synchronization is assumed and (26) simplifies to

$$\sqrt{C}\frac{\sin\left(\pi \Delta f_d T_c\right)}{\pi \Delta f_d T_c} \exp\left\{j\Delta\varphi\right\} + \eta_c, \tag{30}$$

where Δf_d is obtained by comparing the true Doppler frequency against the loop filter output. $\Delta\varphi$ is the phase error obtained as the difference between the true phase and the phase estimate produced by the NCO.

In SATLSim, the function UpdateDiscriminator implements a standard Costas discriminator. Different phase discriminators, as indicated in [21], can be easily implemented by changing this function.

3.3. Double estimator (DoubleEstimator.m)

In the Double Estimator (DE) case, i.e. when DLL and SLL are jointly used, the function GenerateNoiseVector, responsible for the generation of the correlator noise, produces a $5 \times N_{sim}$ matrix, where N_{sim} is the number of simulation runs. The five rows of this matrix correspond to the five correlators required by the DE that are characterized by the following correlation matrix

$$C_n = \begin{bmatrix} 1 & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & R_l\left(\frac{d_s}{2}, 0\right) & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & R_l\left(d_s, 0\right) \\ R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & 1 & R_l\left(0, \frac{d_{sc}}{2}\right) & R_l\left(0, d_{sc}\right) & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) \\ R_l\left(\frac{d_s}{2}, 0\right) & R_l\left(0, \frac{d_{sc}}{2}\right) & 1 & R_l\left(0, \frac{d_{sc}}{2}\right) & R_l\left(\frac{d_s}{2}, 0\right) \\ R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & R_l\left(0, d_{sc}\right) & R_l\left(0, \frac{d_{sc}}{2}\right) & 1 & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) \\ R_l\left(d_s, 0\right) & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & R_l\left(\frac{d_s}{2}, 0\right) & R_l\left(\frac{d_s}{2}, \frac{d_{sc}}{2}\right) & 1 \end{bmatrix}, \qquad (31)$$

where d_{sc} is the subcarrier Early-minus-Late spacing.

The NCO update (`UpdateNCO`) is performed on both code and subcarrier loops and the estimated errors, $\Delta\tau_d$ and $\Delta\tau_s$, are used to compute new correlator signal components (`GenerateSignalCorrelation`). Two nonlinear discriminators (`UpdateDiscriminator`) and loop filters (`UpdateFilter`) are run in parallel to determine the rate of change of both code and subcarrier delay.

The DE provides an example of how several tracking loops, operating in parallel, can be easily coupled in order to provide more realistic simulations accounting for the interaction of different tracking algorithms [9].

3.4. Sample results

In this section, sample results obtained using the SATLSim toolbox are shown for the DE case. Results for the analysis of the PLL can be found in [10]. Specific focus is devoted to the analysis of the tracking jitter, which is one of the main metrics used for the analysis of digital tracking loops. The tracking jitter quantifies the amount of noise transferred by the tracking loop to the final parameter estimate [29]. The tracking jitter is the standard deviation of the final parameter estimate normalized by the discriminator gain. The non-linear discriminator is usually a memoryless device characterized by an input/output function relating the parameter estimation error to the discriminator output. The discriminator gain is the slope of this function in the neighborhood of zero (hypothesis of small estimation error).

Tracking jitter results obtained using non-coherent discriminators [21] for both DLL and SLL are shown in Figure 11. The figure is divided into three parts:

a) Tracking jitter of the DLL alone

b) Tracking jitter of the SLL alone

c) Jitter of the combined delay estimate.

This is due to the fact that the DE jointly uses a DLL, for estimating the code delay, and a SLL, for determining the subcarrier delay. Subcarrier and code delay are then combined to obtain the final estimate of the travel time of the transmitted signal [27]. Thus, three different jitters are evaluated for the different estimates produced by the system. Tracking jitter has been expressed in meters by multiplying the standard deviation of the delay estimates by the speed of light.

In addition to this, the curves are shown in Figure 11a) and Figure 11b). More specifically, three different methodologies have been employed for determining the tracking jitter.

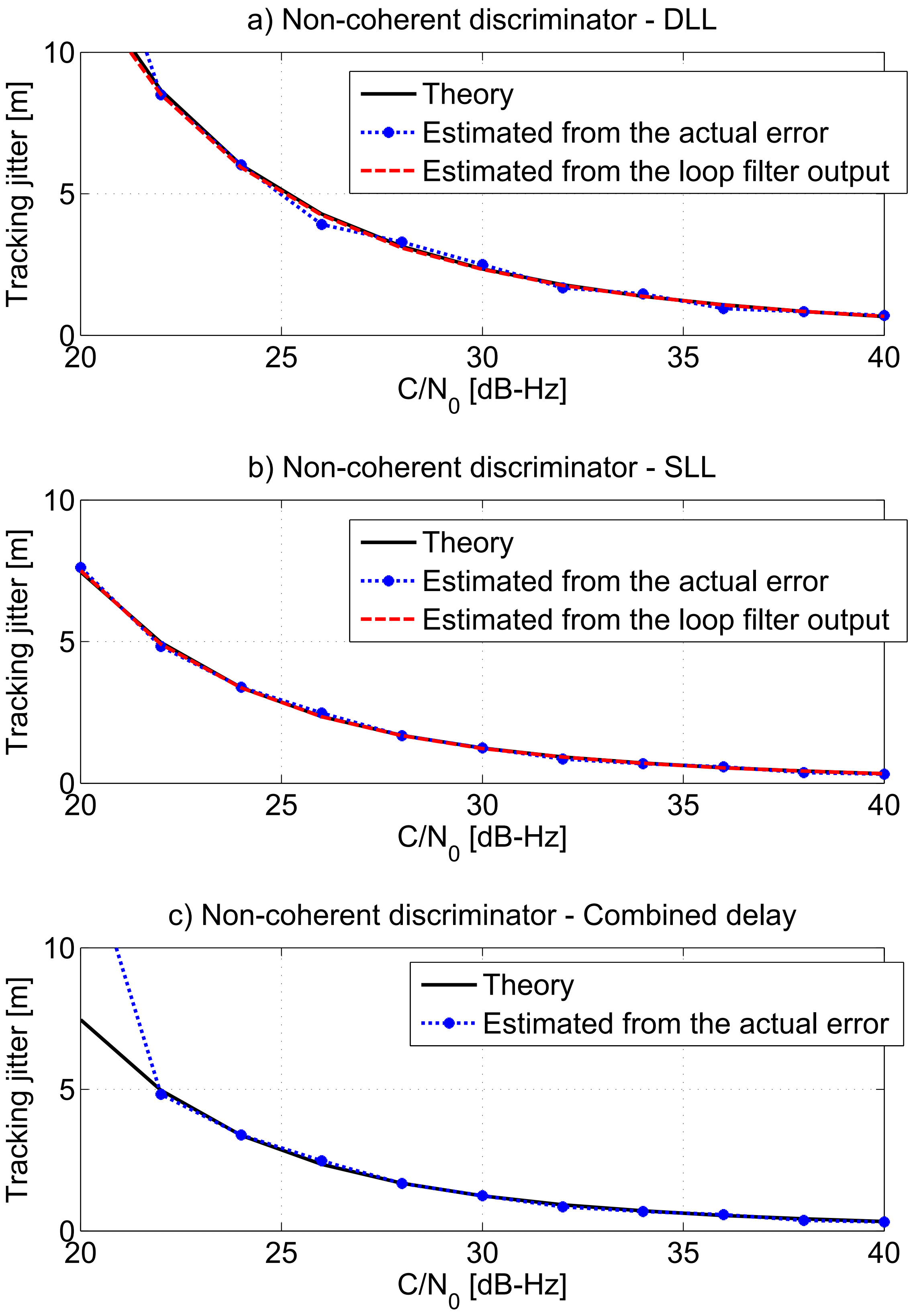

Figure 11. Tracking jitter obtained using the SATLSim toolbox. a) Tracking jitter of the DLL alone. b) Tracking jitter of the SLL alone. c) Jitter of the combined delay estimate.

Parameter	Value
Sampling Frequency	$f_s = 8$ MHz
Integration Time	$T_c = 4$ ms
DLL Early-minus-Late spacing	0.1955 μs (0.2 chips)
SLL Early-minus-Late spacing	0.1955 μs (0.2 chips)
DLL Loop Order	1
SLL Loop Order	1
DLL Loop Bandwidth	0.5 Hz
SLL Loop Bandwidth	0.5 Hz
Modulation type	BOC(1, 1)

Table 3. Parameters used for the evaluation of the tracking jitter shown in Figure 11.

The theoretical curve corresponds to approximate formulas obtained by linearizing the input/output function of the non-linear discriminator. These formulas are valid only for small tracking errors or equivalently for high C/N_0. The jitter obtained from the actual error has been obtained by evaluating the variance of the code phase error. It is noted that in a real tracking loop the code phase error is not directly accessible since the true code phase is unknown. Thus, the tracking error can be evaluated by measuring the error at the loop filter output, which is an observable point, and propagating its variance through the loop. The tracking jitter obtained by propagating the variance at this measurable point corresponds to the curve denoted by “Estimated from the loop filter output”. The relationship between the variances of the discriminator output and the true tracking error can easily be evaluated when the loop is working in its linear region. The measured curve was introduced to further validate the theoretical model and test the correctness of the simulation methodology. This latest curve is not available for the combined delay estimate.

The parameters used for the evaluation of the tracking jitter, shown in Figure 11, are provided in Table 3.

From the results shown in Figure 11, it is observed that the developed semi-analytic technique is able to effectively capture the behavior of the system. For high C/N_0 values, a good agreement between theoretical and simulation results is found. However, for C/N_0 lower than 22 dB-Hz theoretical and simulation results start diverging. This is more clear in parts a) and c) of the figures. For such low C/N_0 values, the loop is no longer working in the linear region of the discriminator input/output function. Thus, the theoretical model is unable to capture the behavior of the loop that is losing lock. The semi-analytic technique implemented in the SATLSim MATLAB toolbox is able to effectively describe the non-linear behavior of the loop requiring only limited computation resources.

4. Conclusions

In this chapter, the development of fast semi-analytic techniques using MATLAB has been analyzed. In the semi-analytic framework, the knowledge of the system under analysis is exploited to reduce the computational load and complexity that full Monte Carlo simulations would require. In this way, the strengths of both analytical and Monte Carlo methods are effectively combined.

Two examples of semi-analytic techniques have been thoroughly analyzed and used to illustrate the two main configurations developed within the semi-analytic framework. The first example illustrates the sequential configuration where simulations and the analytical model are used sequentially. This type of configuration provides increased precision with respect to full Monte Carlo simulations, particularly when the quantities to be estimated assume small values. In addition to this, a significant reduction in terms of computational complexity is achieved. In the example considered, full Monte Carlo simulations require the implementation of a full transmission/reception chain including the interaction between two different systems, DVB-T and GNSS. This requirement led to a significant computational and development complexity. The considered semi-analytic approach is an effective solution for alleviating those requirements.

The second example considered the closed-loop approach and specific focus was devoted to the SATLSim MATLAB toolbox. This toolbox has been developed for the analysis of digital tracking loops and fully exploits the flexibility of the MATLAB programming language. The code has been organized in functions that can be easily replaced by different MATLAB modules. In this way, different loop components such as discriminators, loop filters and NCO models can be integrated in the SATLSim toolbox. The efficiency of semi-analytic techniques and the reduced development time enabled by the MATLAB language are an effective tool for the analysis of complex communications systems.

Author details

Daniele Borio and Eduardo Cano
EC Joint Research Centre, Institute for the Protection and Security of the Citizen, Italy

5. References

[1] W. H. Tranter, K. S. Shanmugan, T. S. Rappaport, and K. L. Kosbar. *Principles of Communication Systems Simulation with Wireless Applications*. Prentice Hall, January 2004.

[2] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan. *Simulation of communication systems*. Kluwer Academic/Plenum Publishers, new york edition, 2000.

[3] F.M. Gardner and J. D. Baker. *Simulation Techniques: Models of Communication Signals and Processes*. John Wiley & Sons, 2003.

[4] M. Jeruchim. Techniques for estimating the bit error rate in the simulation of digital communication systems. *IEEE Journal on Selected Areas in Communications*, 2(1):153 – 170, January 1984.

[5] M. Pent, L. Lo Presti, G. D'Aria, and G. De Luca. Semianalytic BER evaluation by simulation for noisy nonlinear bandpass channels. *IEEE Journal on Selected Areas in Communications*, 6(1):34 –41, January 1988.

[6] M.T. Core, R. Campbell, P. Quan, and J. Wada. Semianalytic BER for PSK. *IEEE Transactions on Wireless Communications*, 8(4):1644 –1648, April 2009.

[7] A. R. Golshan. Post-correlator modeling for fast simulation and joint performance analysis of GNSS code and carrier tracking loops. In *Proc. of the ION/NTM (National Technical Meeting)*, pages 312 – 318, Monterey, CA, January 2006.

[8] J. S. Silva, P. F. Silva, A. Fernandez, J. Diez, and J. F. M. Lorga. Factored correlator model: A solution for fast, flexible, and realistic GNSS receiver simulations. In *Proc. of ION/GNSS*, pages 2676 – 2686, Forth Worth, TX, September 2007.

[9] D. Borio, P. B. Anantharamu, and G. Lachapelle. Semi-analytic simulations: An extension to unambiguous BOC tracking. In *Proc. of the ION/ITM (International Technical Meeting)*, pages 1023–1036, San Diego, CA, January 2010.

[10] Daniele Borio, Pratibha Anantharamu, and Gérard Lachapelle. SATLSim: a semi-analytic framework for fast GNSS tracking loop simulations. *GPS Solutions*, 15:427–431, 2011.

[11] J.M. Geist. Computer generation of correlated gaussian random variables. *Proceedings of the IEEE*, 67(5):862 – 863, may 1979.

[12] A. R. Golshan. Loss of lock analysis of a firstorder digital code tracking loop and comparison of results to analog loop theory for BOC and NRZ signals. In *Proc. of the ION/NTM (National Technical Meeting)*, pages 299 – 305, San Diego, CA, January 2005.

[13] J. W. Betz. Effect of narrowband interference on GPS code tracking accuracy. In *Proc. of ION/NTM*, pages 16–27, Anaheim, CA, January 2000.

[14] D. Borio. GNSS acquisition in the presence of continuous wave interference. *IEEE Transactions on Aerospace and Electronic Systems*, 46(1):47 – 60, January 2010.

[15] T. Van Slyke, W. Kuhn, and B. Natarajan. Measuring interference from a UWB transmitter in the GPS l1 band. In *Proc. of the IEEE Radio and Wireless Symposium*, pages 887 – 890, Orlando, FL, March 2008.

[16] A. Simsky, T. De Wilde, D. Mertens, E. Koitsaly, and J.-M. Sleewaegen. First field experience with L5 signals: DME interference reality check. In *Proc. of ION/GNSS*, pages 29 – 37, Savannah, GA, September 2009.

[17] D. Borio, S. Savasta, and L. Lo Presti. On the DVB-t coexistence with galileo and GPS system. In *Proc. of the* 3rd *ESA Workshop on Satellite Navigation User Equipment Technologies (NAVITEC)*, Noordwijk, The Netherlands, December 2006.

[18] M. Wildemeersch, A. Rabbachin, E. Cano, and J. Fortuny. Interference assessment of DVB-t within the GPS l1 and galileo e1 band. In *Proc. of the* 5th *ESA European Workshop on GNSS Signals and Signal Processing (NAVITEC)*, pages 1 – 8, Noordwijk, The Netherlands, December 2010.

[19] Steven M. Kay. *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*, volume 2. Prentice Hall, 1rt edition, February 1998.

[20] ETSI. Digital video broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television, 2006. EN 300 744.

[21] E. D. Kaplan and C. Hegarty, editors. *Understanding GPS: Principles and Applications*. Artech House, 2nd edition, November 2005.

[22] H. L. Van Trees. *Detection, Estimation, and Modulation Theory - Part 1*. Wiley-Interscience, 1st edition, September 2001.

[23] Daniele Borio. *A statistical theory for GNSS signal acquisition*. Phd thesis, Politecnico di Torino, April 2008.

[24] John W. Betz. Effect of partial-band interference on receiver estimation of C/N_0. In *Proc. of the ION/NTM*, pages 817 – 828, Long Beach, CA, January 2001.

[25] J. Fortuny-Guasch, M. Wildemeersch, and D. Borio. Assessment of DVB-T impact on GNSS acquisition and tracking performance. In *Proc. of the ION/GNSS*, pages 347–356, San Diego, CA, January 2011.

[26] P. Anantharamu, D. Borio, and G. Lachapelle. Sub-carrier shaping for BOC modulated GNSS signals. *EURASIP Journal on Advances in Signal Processing*, 2011(1):133, 2011.

[27] M. S. Hodgart and P. D. Blunt. A dual estimate receiver of binary offset carrier (BOC) modulated signals global navigation satellite systems. *Electronics Letters*, 43(16):877–878, August 2007.

[28] S. A. Stephens and J. B. Thomas. Controlled-root formulation for digital phase-locked loops. *IEEE Transactions on Aerospace and Electronic Systems*, 31(1):78 –95, january 1995.

[29] A. J. V. Dierendonck, P. Fenton, and T. Ford. Theory and performance of narrow correlator spacing in a GPS receiver. *NAVIGATION: the Journal of The Institut of Navigation*, 39(3):265 – 283, Fall 1992.