Q How does one compute the noise power to simulate real and complex GNSS signals?

MARK PETOVELLO is a Professor in the Department of Geomatics Engineering at the University of Calgary. He has been actively involved in many aspects of positioning and navigation since 1997 including GNSS algorithm development, inertial navigation, sensor integration, and software development. Email: mark.petovello@ucalgary.ca

Simulation of GNSS signals is very important to test and validate algorithms. With some algorithms, such as acquisition or tracking, we do not need to simulate a realistic constellation with actual ranges, Doppler or navigation data. We can simply simulate the signal after reception at a receiver's front-end and only take into account and fix to desired values certain parameters, such as the intermediate and the sampling frequencies, the Doppler and Doppler rate, or the signal and the noise power. This last parameter requires some caution, because the noise power depends on the type of sampling (real or complex).

In this article we show how to simulate noisy GNSS signals after a front-end. We will first present the model considered for the received signal and review the constraints on the intermediate and sampling frequencies for real and complex sampling. Then, we will introduce the noise, review the properties of a white noise, and discuss the sampling of a band-limited white noise to determine the expression of the noise power.

Signal model

The GNSS signal received at the antenna can be modeled as

$$s_r(t) = a_I \, x_I(t) \cos(2\pi f_r t + \varphi_r) \\ + a_Q \, x_Q(t) \sin(2\pi f_r t + \varphi_r), \quad (1)$$

where t is the time; a_I and a_Q are amplitudes of the in-phase (I) and quadrature-phase (Q) components; $x_I(t)$ and $x_Q(t)$ are the baseband signals of the I and Q components consisting of a spreading code and possibly a secondary code, a sub-carrier or data; f_r includes the carrier and Doppler frequencies; and φ_r is the carrier phase.

It is also possible to have only one component, e.g., $a_Q = 0$ for the GPS L1 C/A signal. The model could be more complete and include the Doppler effect on the code or the Doppler rate, but that would not change the following discussion and is thus omitted. An example of the spectrum for $s_r(t)$ — denoted $s_r(f)$— is given in **Figure 1**, where B is the bandpass bandwidth of the signal.

Let us first consider the front-end depicted in **Figure 2** (top), which performs real sampling. This is a very simplified representation, but enough for our discussion. This front-end is composed of:

- a bandpass filter (BPF) for image-rejection
- a mixer to bring the signal to a lower frequency
- a low pass filter (LPF) for anti-aliasing (aliasing is explained next); this filter can also be a bandpass filter
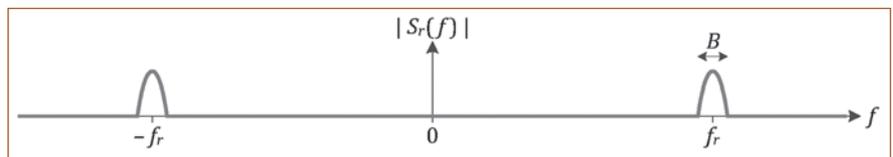- an analog-to-digital converter (ADC)

Of course, there are also a few

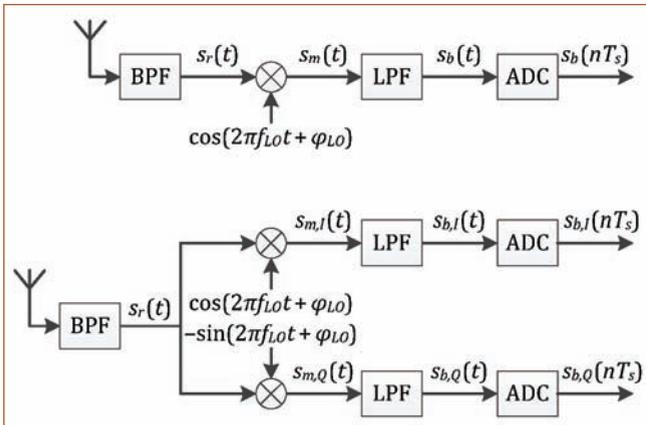**FIGURE 1** Example of incoming signal spectrum

FIGURE 2 Simplified front-end, with a real sampling (top) and with complex sampling (bottom)

amplifiers in a front-end, not shown in Figure 2. See the Additional Resources section for details about real and complex front-end designs.

After the mixer and the low pass filter, the signal is

$$s_b(t) = \frac{a_I}{2} x_I(t) \cos(2\pi f_i t + \varphi_i) + \frac{a_Q}{2} x_Q(t) \sin(2\pi f_i t + \varphi_i), \quad (2)$$

where $f_i = f_r - f_{LO}$ is the intermediate frequency and $\varphi_i = \varphi_r - \varphi_{LO}$, with f_{LO} and φ_{LO} the frequency and phase of the local oscillator. **Figure 3** shows the spectrum of this baseband signal. To avoid aliasing, i.e., an overlapping of the positive and negative sides of the spectrum, we require $f_{min} \geq 0$ or, equivalently, $f_i \geq B/2$.

After the ADC, the signal is sampled and becomes

$$s_b(nT_s) = \frac{a_I}{2} x_I(nT_s) \cos(2\pi f_i nT_s + \varphi_i) + \frac{a_Q}{2} x_Q(nT_s) \sin(2\pi f_i nT_s + \varphi_i) \quad (3)$$

where T_s is the sampling period equal to $1/f_s$, with f_s being the sampling frequency. The spectrum of the sampled signal is the spectrum of the continuous-time signal duplicated around each multiple of f_s, as shown in Figure 3. To avoid aliasing, we require $f_s \geq 2f_{max}$, or $f_s \geq 2f_i + B$. This is the well-known Nyquist-Shannon sampling criterion.

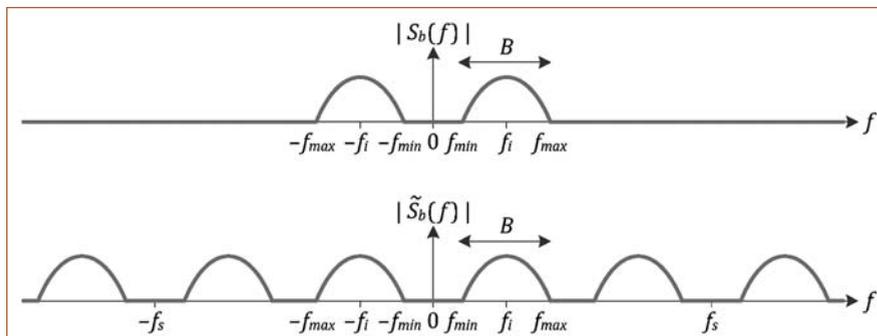Now, let us consider the front-end depicted in Figure 2 (bottom), which performs a complex sampling. After the

mixer and the low pass filter, the baseband signals are

$$s_{b,I}(t) = \frac{a_I}{2} x_I(t) \cos(2\pi f_i t + \varphi_i) + \frac{a_Q}{2} x_Q(t) \sin(2\pi f_i t + \varphi_i), \quad (4)$$

and

$$s_{b,Q}(t) = \frac{a_I}{2} x_I(t) \sin(2\pi f_i t + \varphi_i) - \frac{a_Q}{2} x_Q(t) \cos(2\pi f_i t + \varphi_i). \quad (5)$$

The complex baseband signal is then

$$s_b(t) = s_{b,I}(t) + j s_{b,Q}(t) = \left(\frac{a_I}{2} x_I(t) - j \frac{a_Q}{2} x_Q(t) \right) e^{j(2\pi f_i t + \varphi_i)}, \quad (6)$$

with $j = \sqrt{-1}$. **Figure 4** shows the corresponding spectrum (assuming $\varphi = 0$ rad, else the spectrum would not be exactly on the real and imaginary axes), remembering that

$$\cos u = \frac{e^{-ju} + e^{ju}}{2}$$

and

$$\sin u = \frac{-e^{-ju} + e^{ju}}{2j} = j \frac{e^{-ju} - e^{ju}}{2}$$

(see Additional Resources section for a reference about this). In this case, even if there can be an aliasing in each branch, when considering the signal as complex, aliasing is not possible because the negative side of the spectrum is canceled as shown in Figure 4 (second from bottom). Therefore f_i can take any value, whatever the signal bandpass bandwidth B.

After sampling, the signal is

$$s_b(nT_s) = \left(\frac{a_I}{2} x_I(nT_s) - j \frac{a_Q}{2} x_Q(nT_s) \right) e^{j(2\pi f_i nT_s + \varphi_i)}. \quad (7)$$

The bottom portion of Figure 4 shows the signal spectrum at this point, and, to avoid aliasing, we should have

$$f_s + f_{min} \geq f_{max} \Leftrightarrow f_s \geq f_{max} - f_{min} \quad (8)$$
$$\Leftrightarrow f_s \geq B.$$

Therefore, the sampling frequency now depends only on the signal bandpass bandwidth.

Noise Power Computation

In reality, an element of noise is present in addition to the received signal. This noise, called thermal noise, is induced by the antenna and the front-end themselves and is assumed to be an additive white Gaussian noise (AWGN). So, we next explain how to compute the power of this noise at the output of the front-end.

A white noise is a noise whose power spectral density (PSD) is equal at all frequencies, as shown in **Figure 5**, top. As the PSD is the Fourier transform of the auto-correlation (Wiener-Khinchin theorem), the autocorrelation of a white noise is null everywhere except at zero, as shown in lower plot in Figure 5. This implies that the values of a white noise at different instants are uncorrelated, and, therefore,
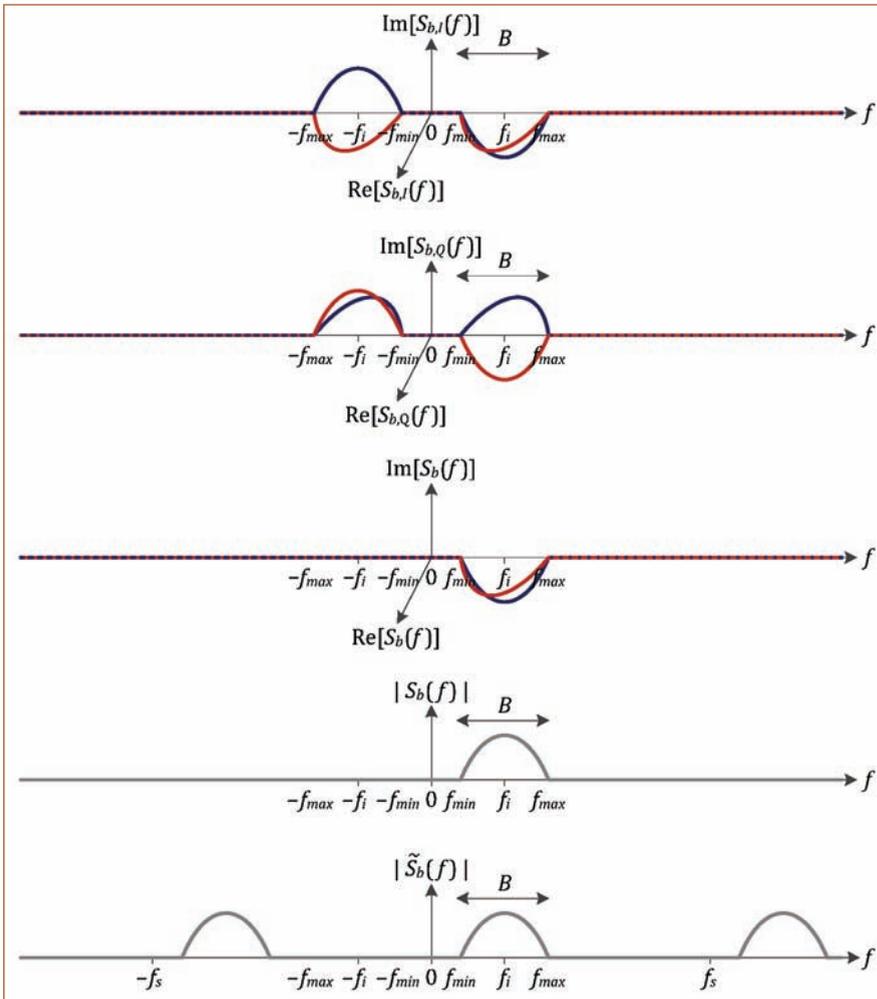

FIGURE 3 Spectrum of the baseband signal in real sampling front-end, after mixing and filtering (top), after sampling (bottom).

FIGURE 4 Spectrum of the baseband signal in complex sampling front-end, after mixing and filtering (top), after sampling (bottom).
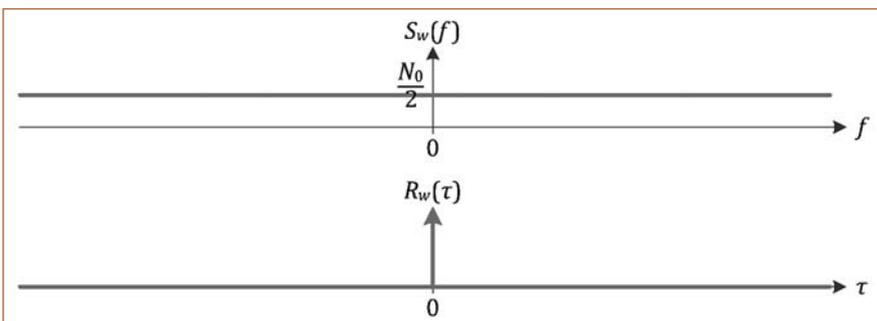

FIGURE 5 PSD (top) and autocorrelation (bottom) of a white noise

the mean of a white noise is zero. This is true whatever the probability density function of the noise, which can be Gaussian, Laplace, uniform, and so forth (as long as the mean is zero).

The two-sided PSD of the thermal noise is equal to $N_0/2$, where N_0 is the noise power density and depends on the effective temperature of the front-

end. However, the front-end filters the incoming signal; therefore, the thermal noise is filtered, too, and the noise just before the ADC is not white. We will thus see the conditions to obtain a white noise after the ADC.

With a real sampling, assuming an ideal brick wall filter of baseband bandwidth B_F, the noise just before the ADC

will be a band-limited white noise, as shown in **Figure 6**, and the noise power (which is equal to the integral of the noise PSD) will be $\sigma^2 = N_0 B_F$.

The autocorrelation function of this band-limited white noise is

$$R_w(\tau) = B_F N_0 \frac{\sin(2\pi B_F \tau)}{2\pi B_F \tau} \quad \text{(9)}$$
$$= B_F N_0 \operatorname{sinc}(2\pi B_F \tau),$$

and is also illustrated in Figure 6. This autocorrelation is equal to zero when $\tau = \frac{k}{2B_F}$ with k an integer different than zero, which means that the values of the band-limited white noise that are distant by $\frac{k}{2B_F}$ are uncorrelated. Therefore, if we sample this band-limited white noise with a time interval of $\frac{k}{2B_F}$, i.e., with a frequency of $f_s = \frac{k}{2B_F}$, the autocorrelation of the sampled noise will be null everywhere except at zero, and thus the sampled noise will be white.

This can also be seen using the PSD. After the sampling, the noise spectrum is duplicated around each multiple of f_s. **Figure 7** (top) shows the spectrum of the band-limited white noise sampled with a sampling frequency higher than $2B_F$, and the noise is clearly not white. Figure 7 (middle and bottom) shows the spectrum of the band-limited white noise sampled with sampling frequencies of $2B_F$ and B_F, respectively. In both cases, the noise is white and the noise power (which is equal to the integral of the noise PSD in the interval $\left[-\frac{f_s}{2}; \frac{f_s}{2}\right]$) is $N_0 B_F$.

On the other side, to respect the Nyquist-Shannon theorem, we should have $f_s \geq 2B_F$. Therefore, the only possibility to satisfy both conditions is $f_s \geq 2B_F$, and thus the noise power must be $\sigma^2 = N_0 f_s / 2$.

Similar developments can be done for the case of complex sampling. The difference is that the sampling frequency must be equal to $f_s = B_F/k$ to get a discrete white noise, and the Nyquist-Shannon theorem imposes $f_s \geq B_F$. Therefore, the only possibility to satisfy both conditions is $f_s = B_F$, and thus the noise power on each branch must be $\sigma^2 = N_0 f_s$, i.e., twice the noise power that exists with real sampling.

Application to GNSS signals

Using the previous models, at the output of the front-end, a real GNSS signal can be modeled as

$$s_b(nT_s) = a_I x_I(nT_s) \cos(2\pi f_i nT_s + \varphi_i) \quad (10)$$
$$+ a_Q x_Q(nT_s) \sin(2\pi f_i nT_s + \varphi_i)$$
$$+ w(nT_s),$$

and a complex GNSS signal can be modeled as

$$s_b(nT_s) = \quad (11)$$
$$\left(a_I x_I(nT_s) - j a_Q x_Q(nT_s)\right) e^{j(2\pi f_i nT_s + \varphi_i)}$$
$$+ w_I(nT_s) + j\, w_Q(nT_s),$$

where w, w_I and w_Q are white Gaussian noises of power σ^2 (the value of σ^2 for w_I and w_Q is twice the one for w, as mentioned previously). The values for a_I and a_Q can correspond to the amplitudes after the antenna, and σ can be computed using the expressions of the previous section. However, a normalization is usually performed because only the ratio between the signal power and the noise power matters, not their absolute value.

To describe the level of a received GNSS signal, two parameters can be used: the signal power, or the carrier power to noise power density ratio (or carrier-to-noise ratio). The power of the baseband signals $x_I(t)$ and $x_Q(t)$ is equal to 1 because they are based on binary waveforms of amplitude ±1. Therefore, the power of an I/Q component is

$$P = a_{I/Q}^2/2 \Leftrightarrow a_{I/Q} = \sqrt{2P}.$$

The carrier-to-noise ratio is simply the ratio between the signal power and the noise power density, i.e., $C/N_0 = P/N_0$, thus

$$a_{I/Q} = \sqrt{2C/N_0\, N_0}.$$

Both signal amplitude and noise standard deviation (σ) have very low values. Consequently, in simulation, it is convenient to perform a normalization, usually to have a noise standard deviation (std) of 1. For that, it is sufficient to divide both signal amplitude and noise std by the noise std.
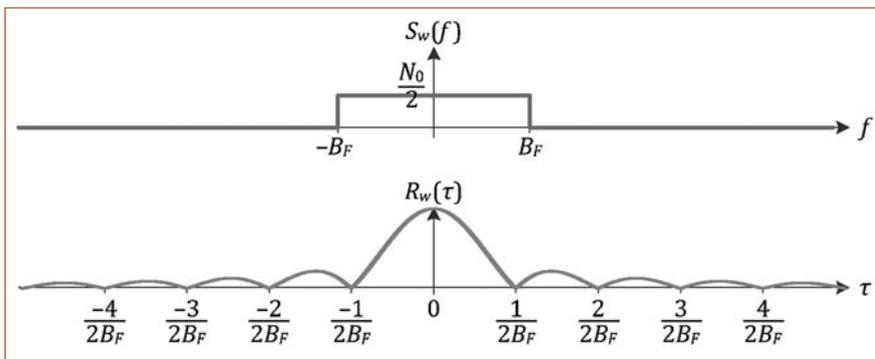
Previously we have seen that the

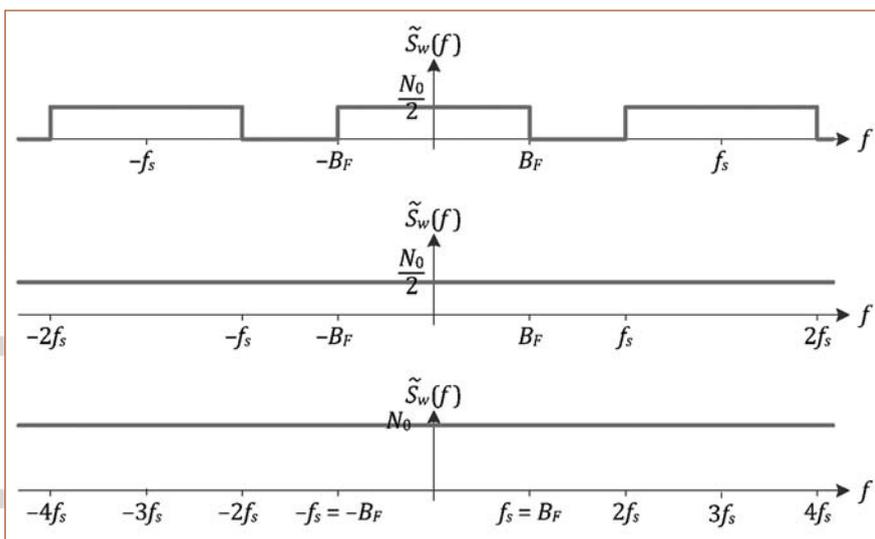**FIGURE 6** PSD (top) and autocorrelation (bottom) of a band-limited white noise

FIGURE 7 PSD of a sampled band-limited white noise, with $f_s = 3B_F$ (top), $f_s = 2B_F$ (middle), $f_s = B_F$ (bottom)

		Not normalized		Normalized	
Sampling		real	complex	real	Complex
Noise std (σ)		$\sqrt{\dfrac{N_0 f_s}{2}}$	$\sqrt{N_0 f_s}$	1	1
Signal amplitude ($a_{I/Q}$)	From P	$\sqrt{2P}$		$2\sqrt{\dfrac{P}{N_0 f_s}}$	$\sqrt{\dfrac{2P}{N_0 f_s}}$
	From C/N_0	$\sqrt{2C/N_0\, N_0}$		$2\sqrt{\dfrac{C/N_0}{f_s}}$	$\sqrt{\dfrac{2C/N_0}{f_s}}$

TABLE 1 Expressions of the noise std and amplitude for one I/Q component. Parameters are in linear scale not log scale.

noise power is not the same depending on the type of sampling. With the normalization, the noise std is the same for both real and complex sampling; so, it is the signal amplitude that will be different according to the type of sampling. **Table 1** summarizes the expressions to compute the noise std and the component amplitude from the component power and noise power density, without and with normalization. Note that in the normalized case, the signal amplitude can be determined directly from the carrier-to-noise ratio and the

sampling frequency, thus the computation of the noise power density N_0 is not required.

Let's consider two examples to illustrate all this. First, we want to simulate a GPS L1 C/A signal of $C/N_0 = 25$ dBHz, sampled at $f_s = 5$ MHz with real sampling. With normalized values, the noise std will be 1, and the signal amplitude will be

$$2\sqrt{\frac{10^{2.5}}{5 \cdot 10^6}} \approx 1.591 \times 10^{-2}$$

Second, we want to simulate a GPS L5 signal with a power of -174 dBW, sampled at $f_s = 524$ MHz with complex sampling, and $N_0 = -204$ dBW/Hz. Thus, the power of each I/Q component is -177 dBW since both components have the same power. With normalized values, the noise std will be 1, and the signal amplitude will be

$$\sqrt{\frac{2 \times 10^{-17.7}}{10^{-20.4} \times 20 \cdot 10^6}} \approx 6.463 \times 10^{-3}$$

To conclude, **Table 2** provides a concise summary with all the parameters discussed in the article.

Acknowledgments

Additional Resources

For more information about parallel acquisition architectures, refer to:
Leclère, J., "Resource-efficient parallel acquisition architectures for modernized GNSS signals," Ph.D. thesis, EPFL, Switzerland, 2014.

For information about GNSS receivers, including real and complex front-end designs, refer to:
Borre, K., et al., *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach,* Birkhäuser Boston, 2007

Chastellain, F., and C. Botteron, and P.-A. Farine, "Looking Inside Modern Receivers," *IEEE Microwave Magazine,* vol. 12, no. 2, pp. 87–98, 2011

van Diggelen, F., *A-GPS: Assisted GPS, GNSS, and SBAS,* Artech House, 2009.

For details about quadrature signals, please see:
Lyons, R., *Understanding Digital Signal Processing,* Prentice Hall, 2010.

Authors

JÉRÔME LECLÈRE is a postdoctoral researcher at École de Technologie Supérieure (ÉTS), Canada. He works on efficient acquisition algorithms and integration of GNSS and other sensors for automotive applications.
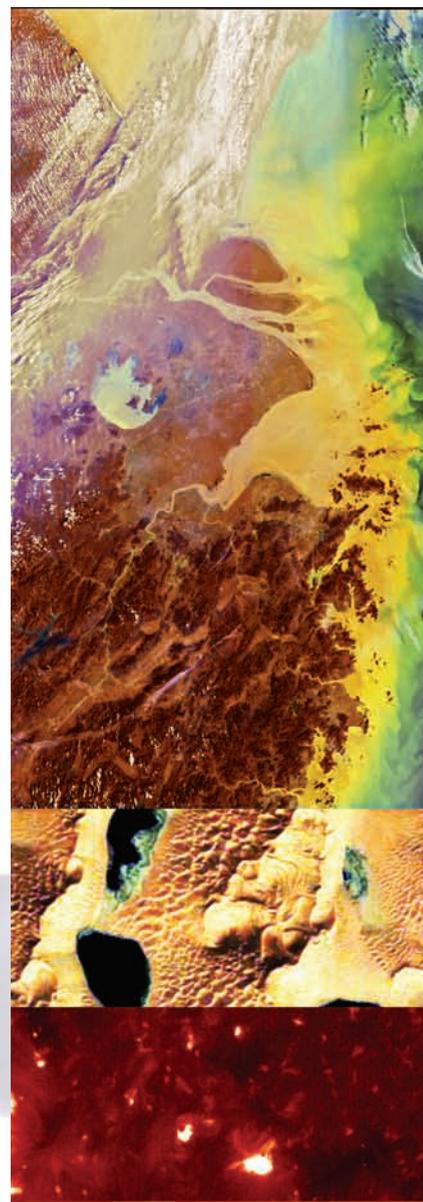
RENÉ JR. LANDRY is a professor in the department of electrical engineering at École de Technologie Supérieure (ÉTS), Canada and the director of the LASSENA laboratory. His expertise in embedded systems, navigation, and avionics is applied notably in the field of transport, aeronautic, and space technologies.

CYRIL BOTTERON leads, manages, and coaches the research and project activities of the Global Navigation Satellite System and ultra-wideband and millimeter-wave groups at École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. **IG**

Effective temperature of the front-end	$T_E = T_{ANT} + (F-1)T_{AMB}$	
Noise power density (N_0)	$k_B T_E$	
Sampling	Real	Complex
Intermediate frequency constraint	$f_i > B/2$	none
Sampling frequency constraint	$f_s > 2f_i + B$	$f_s > B$
Component noise standard deviation (σ)	1	1
Component amplitude ($a_{I/Q}$) — From P	$2\sqrt{\dfrac{P}{N_0 f_s}}$	$\sqrt{\dfrac{2P}{N_0 f_s}}$
Component amplitude ($a_{I/Q}$) — From C/N_0	$2\sqrt{\dfrac{C/N_0}{f_s}}$	$\sqrt{\dfrac{2C/N_0}{f_s}}$

TABLE 2 Expressions to use with Equations (12) and (13). F is the linear noise figure of the front-end, TANT is the effective temperature of the antenna, and TAMB is the ambient temperature.