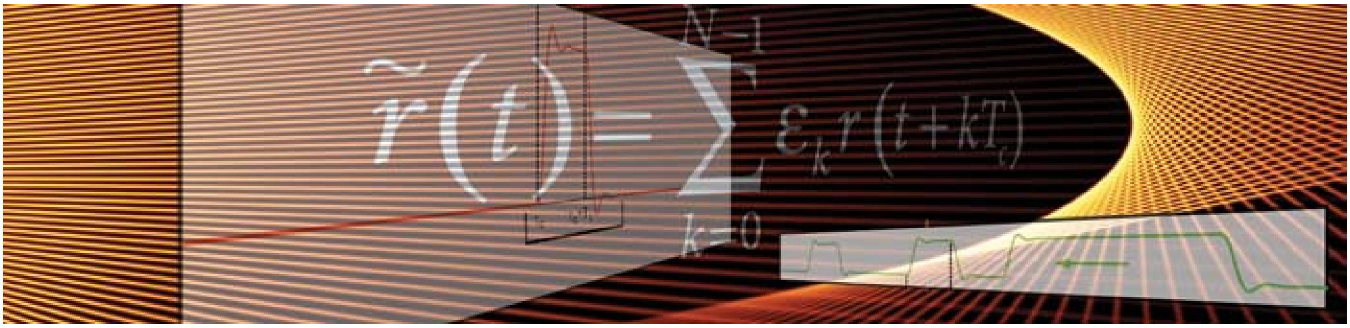
Lightening the Data Processing Load

Signal Compression in GNSS Receivers

LAWRENCE R. WEILL
CALIFORNIA STATE UNIVERSITY, FULLERTON, AND COMM SCIENCES CORPORATION



The emergence of increasingly sophisticated GNSS signal processing techniques over the past 20 years is dramatically increasing the computational load on GNSS receivers. These techniques include new methods for acquiring and tracking a wide variety of second-generation signal structures, advanced multipath mitigation techniques, and the development of high-sensitivity receivers for reliable operation indoors and in urban canyons. This article by a long-time GNSS signal expert describes a process of signal compression that can substantially reduce the amount of computation a receiver must perform.

The last two decades have seen the evolution of increasingly sophisticated GNSS signal processing technology. These advances include such things as improved methods for acquiring and tracking a wide variety of new signal structures, advanced multipath mitigation techniques, the development of high-sensitivity receivers for reliable operation indoors and in urban canyons, high-speed processing to reduce time to first fix, and algorithms for improved ranging accuracy and attitude estimation.

Common to many of the new signal processing methods is the need to process the millions of chips in a GNSS signal in unconventional ways, which can dramatically increase the amount of computation the receiver must perform. For example, in typical GNSS receivers the computation of correlation functions is not difficult because the correlator reference waveform can be an ideal chipping sequence with only the values ±1, and multiplications become trivial.

However, techniques such as the high performance of the Multipath Mitigation Technology (MMT) algorithm, developed by the author and a colleague, requires correlator reference waveforms that include the effects of filtering in the receiver and the satellite. Thus, in MMT many millions of multibit multiplications would be needed to compute the correlation function for just one delay value, notwithstanding that many high-resolution delay values are actually needed.

The computational demands in this example and in many other advanced processing techniques can be dramatically reduced by first implementing a process called *signal compression*. In the signal compression technique discussed in the article, for which a patent is pending, a large number of raw digitized baseband signal samples (typically on the order of 10^7-10^8) is reduced to a small vector having only a few tens of samples (the exact number depends on the type of GNSS signal being processed). The compressed signal then has the appearance of a single chip of the received signal (or two separate chips in some of the newer chip-multiplexed signals considered for GPS L1C and Galileo).

This compression technique requires only simple additions to generate, and preserves all signal range and phase information. Subsequent processing of any type is dramatically simplified because of the extremely small size of

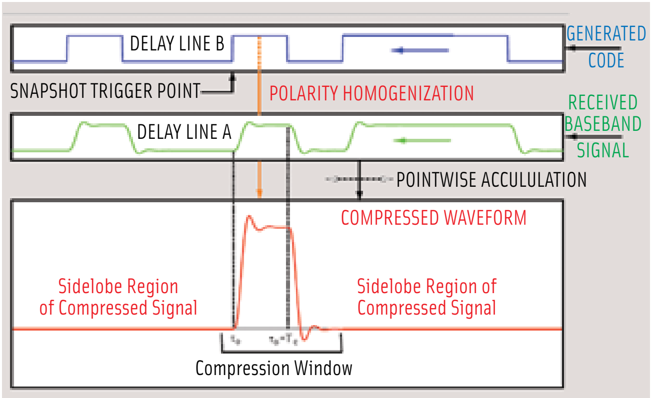


FIGURE 1 Visualization of signal compression

the compressed signal. Therefore, substantial advantages can accrue in a wide variety of endeavors, such as range and phase extraction, phase ambiguity resolution, multipath mitigation, and integrity monitoring among others.

Compression Mathematics

Before describing the design and operation of our technique for GNSS signal compression, we should first look at some of the key aspects of signal processing and review the mathematical concepts for implementing a compression scheme.

Received Signal Model. An interesting characteristic of compression is that it imposes essentially no restrictions on the form of the received signal $r(t)$ that is to be compressed and can accommodate a wide variety of signal structures. However, for conceptual simplicity in this discussion we will assume that $r(t)$ is complex-valued and is at baseband with Doppler and navigation data modulation removed.

For example, the received signal might have the simple form

$$r(t) = a\exp(j\phi)\, f(t-\tau) + n(t) \qquad (1)$$

where a is the signal amplitude, ϕ is the phase, $f(t)$ is modulation (usually a PN code), τ is the signal delay, and $n(t)$ is complex-valued Gaussian noise.

If multipath is present, the signal might have the more complicated structure

$$r(t) = \sum_{k=1}^{N} a_k \exp(j\phi_k)\, f(t-\tau_k) + n(t) \qquad (2)$$

The signal might also be subject to distortion in passing through a dispersive medium, such as the ionosphere. The point being made here is that signal compression does not require any particular received signal structure.

Signal Cross-Correlation. A fundamental operation in GNSS receivers is cross-correlation of the received signal $r(t)$ with a receiver-generated waveform $m(t)$. The cross-correlation has the form

$$R_{rm}(\tau) = \int_0^T r(t)\, m^*(t-\tau)\, dt \qquad (3)$$

where $[0, T]$ is the time interval of signal observation and the asterisk denotes complex conjugate.

Definition of Signal Compression. For conceptual simplicity in defining signal compression, the cross-correlation in equation (3) is assumed to be a circular correlation of period T. However, with appropriate modifications and approximations, we can extend the theory to include non-circular cross-correlation.

In defining the basic form of signal compression, the reference waveform $m(t)$ in (3) is required to have the form

$$m(t) = \sum_{k=0}^{N-1} \varepsilon_k\, c(t-kT_c), \quad 0 < t \le T \qquad (4)$$

where $c(t)$ is an arbitrary chip waveform (which may be complex-valued and may include the effects of filtering) and the ε_k are real-valued weights (although an extension to complex values is possible).

Here we can see that $m(t)$ is the sum of N weighted time-shifted (rotated) versions of $c(t)$, where the time shifts are integral multiples of the constant T_c. All GNSS pseudo-noise (PN) codes) are of this form, except for chip-multiplexed codes which we will deal with later. It is important to understand that the replicas of $c(t)$ are allowed to overlap.

The *compressed signal* $\tilde{r}(t)$ is defined by

$$\tilde{r}(t) = \sum_{k=0}^{N-1} \varepsilon_k\, r(t+kT_c) \qquad (5)$$

In this expression $\varepsilon_k r(t+kT_c)$ is $r(t)$ weighted by ε_k and left-shifted (rotated) by kT_c.

Visualizing Compression. Figure 1 is a helpful visualization of the compression process that was conceived by my colleague, Dr. Ben Fisher. For simplicity we assume that the signal's PN code has simple real-valued chips, such those in the GPS C/A or P codes. The received baseband signal passes through delay line A, which is several chips in length (for clarity, the noise is omitted).

The signal enters the delay line from the right and moves to the left (this permits the waveform within the delay line to be seen as if it were displayed on an oscilloscope, with later parts of the waveform on the right). The receiver's reference PN code, which is tracking the received code, simultaneously passes through an identical delay line B, the center of which is called the *trigger point*.

As the leading edge of each chip of the receiver's reference code reaches the trigger point of delay line B, a snapshot is taken of the entire waveform in delay line A. If the triggering chip has negative polarity, the polarity of the entire snapshot waveform is inverted. The polarity-homogenized snapshots (one for each arriving chip of the received signal) are pointwise accumulated to build up the compressed signal shown at the bottom of Figure 1.

Compression as a Correlation. We can equivalently define the compressed signal as the circular cross-correlation of the received signal $r(t)$ with the impulsive sequence

$$s(u) = \sum_{k=0}^{N-1} \varepsilon_k\, \delta(u-kT_c) \qquad (6)$$

as shown pictorially in Figure 2. Then,

$$\begin{aligned}\tilde{r}(t) &\triangleq \int_0^T r(u)\, s(u-t)\, du \\ &= \int_0^T r(u) \sum_{k=0}^{N-1} \varepsilon_k \delta(u-kT_c-t)\, du \\ &= \sum_{k=0}^{N-1} \int_0^T \varepsilon_k r(u)\, \delta(u-kT_c-t)\, du \\ &= \sum_{k=0}^{N-1} \varepsilon_k\, r(t+kT_c)\end{aligned} \qquad (7)$$

which shows that this alternate definition is equivalent to that given by (5).

Properties of the Compressed Signal

The compressed signal $\tilde{r}(t)$ has a number of very useful properties. Among the most important are the following.

1. Small Size. In GNSS applications the compressed signal has the very nice property that essentially all of its energy (excluding noise) is concentrated into a pulse with a width on the order of T_c. To make this evident, assume that the received signal has the simple form

$$r(t) = am(t-\tau) + n(t) = a\left[\sum_{j=0}^{N-1} \varepsilon_j c(t-\tau-jT_c)\right] + n(t) \qquad (8)$$

where a is the signal amplitude, τ is the signal delay, $n(t)$ is zero-mean wide-sense stationary noise, the weights ε_j have values of ±1, and all time shifts are circular rotations over the period T. Substitution of this expression into (5) and manipulating the resulting double summation yields

$$\tilde{r}(t) = a\sum_{k=0}^{N-1}\sum_{j=0}^{N-1} \varepsilon_k \varepsilon_j c\left[t-\tau+(k-j)T_c\right] + \tilde{n}(t) \qquad (9)$$

where the double summation is the *compressed noise-free signal* and

$$\tilde{n}(t) = \sum_{k=0}^{N-1} \varepsilon_k n(t+kT_c) \qquad (10)$$

is the *compressed noise function*.

The terms in the double summation of (9) can be grouped into N groups such that each group contains N terms having the same value of $k-j$ modulo N. Thus, $\tilde{r}(t)$ will be the summation of N group sums. The group sum corresponding to a particular value p of $k-j$ modulo N is $c[t-\tau+pT_c]$ weighted by the sum of terms $\varepsilon_j \varepsilon_k$, which satisfy $k-j=p$ modulo N. Ignoring the noise, we can see that $\tilde{r}(t)$ consists of a concatenation of N weighted and translated (rotated) copies of $c(t)$.

If the number of chips N is sufficiently large (on the order of 10^3 or more), the autocorrelation function of the chipping sequence has the property that the group sums in which $k-j \neq 0$ modulo N are negligible compared to the group sum in which $k-j=0$ modulo N. Furthermore, the sum of all of these small group sums is also negligible because the translations of the weighted copies of $c(t)$ prevent the small group sums from accumulating to large values. Thus, to a very good approximation, the double summation in (9) is just the sum of the terms where $k-j=0$ modulo N:

$$\tilde{r}(t) \cong a\sum_{k=0}^{N-1} \varepsilon_k^2 c(t-\tau) + \tilde{n}(t) = aNc(t-\tau) + \tilde{n}(t) \qquad (11)$$

This is a very significant result, because it tells us that the compressed received signal is essentially just the single weighted chip waveform $aNc(t-\tau)$, with small "sidelobe" chips to either side. Thus, the significant part of the compressed signal is contained in a short window just long enough to contain this single chip and its delay uncertainty, and the sidelobe chips as well as all noise outside this window can be rejected.

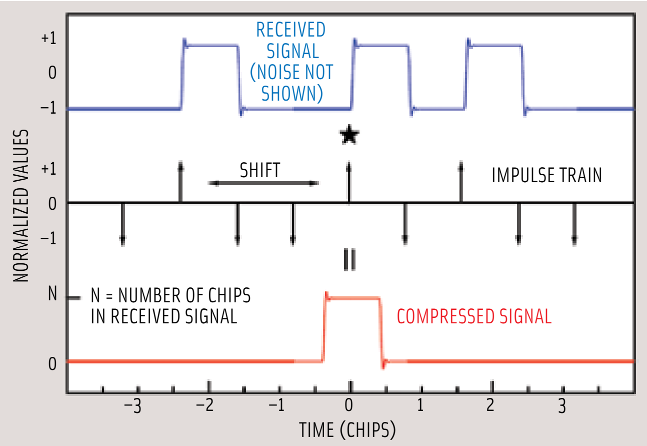


FIGURE 2 Compression as a correlation

The required length of the window is $T_c + \delta$, where δ is large enough to accommodate the measurement uncertainty of τ, the trailing transient due to filtering, and any multipath components with delays larger than τ (almost always within 1 chip of the direct path delay). Thus the window length is somewhat larger than a one-chip duration of the code, a quantity much smaller than the length T of the observed signal $r(t)$, which must include all N chips of the code. Because of this result, $\tilde{r}(t)$ can justifiably be called a compressed signal.

2. Compressed Noise Statistics. Assuming the signal model given by (8), the mean of the compressed noise function (10) is

$$\mathrm{E}\left[\tilde{n}(t)\right] = \mathrm{E}\left[\sum_{k=0}^{N-1} \varepsilon_k n(t+kT_c)\right] = \sum_{k=0}^{N-1} \varepsilon_k \mathrm{E}\left[n(t+kT_c)\right] = 0 \qquad (12)$$

and its circular covariance function is

$$\begin{aligned} R_{\tilde{n}\tilde{n}}(t,u) &= \mathrm{E}\left[\sum_{j=0}^{N-1} \varepsilon_j n(t+jT_c) \sum_{k=0}^{N-1} \varepsilon_k n^*(u+kT_c)\right] \\ &= \sum_{j=0}^{N-1}\sum_{k=0}^{N-1} \varepsilon_j \varepsilon_k \mathrm{E}\left[n(t+jT_c)\, n^*(u+kT_c)\right] \\ &= \sum_{j=0}^{N-1}\sum_{k=0}^{N-1} \varepsilon_j \varepsilon_k R_{nn}\left[t-u+(j-k)T_c\right] \\ &\cong NR_{nn}(t-u) \end{aligned} \qquad (13)$$

Some explanation is needed to understand the derivation (13). In the third line we have used the fact that the autocovariance function R_{nn} of the received noise $n(t)$ depends only on the difference of its two arguments, since $n(t)$ is a wide-sense stationary process. To obtain the fourth line, we have grouped the terms in the third line double summation into N groups such that each group contains N terms having the same value of $j-k$ modulo N (the groups are not displayed).

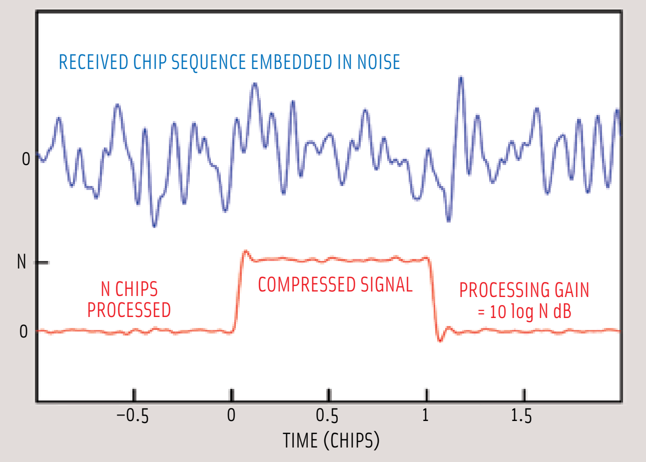


FIGURE 3 C/A-code signal visibility via processing gain

Consider the sum of the terms in the group for which $j - k = p$ modulo N, where p is one of the integers 0, 1, … , N–1. The sum of the terms for this group is

$$\sum_{k=0}^{N-1}\varepsilon_k\varepsilon_{k+p}R_{nn}\left[t-u+pT_c\right] = R_{nn}\left[t-u+pT_c\right]\sum_{k=0}^{N-1}\varepsilon_k\varepsilon_{k+p} \qquad (14)$$

If $p = 0$, the second summation in (14) is simply N. On the other hand, if N is sufficiently large and the weights ε_k are those associated with a typical pseudorandom sequence of chips, the second summation in (14) for $p \neq 0$ will be negligible compared to N. Thus the result (13) is simply (14) with $p = 0$, namely $NR_{nn}(t - u)$.

Expression (13) gives us the following very useful result: *When the weights ε_k have the values ±1, the covariance function of the compressed noise is simply the covariance function of the received noise scaled up by N.*

3. Processing Gain. As can be seen from (11), compression scales the noise-free signal power by N^2, and (13) shows that the noise variance gets scaled by N. Thus, the processing gain G is

$$G = 10\log\left(\frac{N^2}{N}\right) = 10\log\ N\ \text{dB} \qquad (15)$$

which is essentially the same as the processing gain obtained by correlation.

4. Signal Visibility. If the number of chips N is sufficiently large, the processing gain of compression is great enough to make the compressed signal visible with very little noise. As a result, small subtleties in the chip waveshape due to filtering, multipath, ionospheric distortion, or other causes can easily be detected. This property is very beneficial for signal integrity monitoring and has been put to practical use in some models of GPS receivers.

Figure 3 illustrates how compression can make the received chip waveform visible, even when the signal is below the noise level. The top of the figure is a small portion of a one-second observation of a received baseband GPS L_1 C/A-coded signal at 45 dB-Hz, with a video bandwidth of 12 MHz. The signal is mostly noise and the individual C/A-code chips are not visible.

The bottom portion of the figure shows the result of compressing this signal received during a $T = 1$ second interval. The chip waveform is clearly visible, including the effects of filtering in the satellite and receiver. We should point out that the compression process permits the trailing transient of the chip to be clearly seen, even though the trailing transients of chips in the raw signal extend into following chips. In comparison to a compressed C/A-code at L1, the structure of a compressed BOC(1,1) signal at E1 (noise omitted) is shown at the top of Figure 9.

5. Time-Invariant Linearity. Compression is a time-invariant transformation, which means that the compression of the time-shifted signal $r(t-\tau)$ is simply $\tilde{r}(t-\tau)$. This follows directly from definition (5).

The compressed signal also enjoys a *linearity property*: If

$$r(t) = a_1 r_1(t) + a_2 r_2(t) \qquad (16)$$

then

$$\tilde{r}(t) = a_1\tilde{r}_1(t) + a_2\tilde{r}_2(t) \qquad (17)$$

Proof:

$$\begin{aligned}\tilde{r}(t) &= \sum_{k=0}^{N-1}\varepsilon_k r(t+kT_c)\\ &= \sum_{k=0}^{N-1}\varepsilon_k\left[a_1 r_1(t+kT_c) + a_2 r_2(t+kT_c)\right]\\ &= a_1\sum_{k=0}^{N-1}\varepsilon_k r_1(t+kT_c) + a_2\sum_{k=0}^{N-1}\varepsilon_k r_2(t+kT_c)\\ &= a_1\tilde{r}_1(t) + a_2\tilde{r}_2(t)\end{aligned} \qquad (18)$$

The time-invariant linearity property is essential for various advanced signal-processing operations, such as extracting multipath parameters from the signal.

6. Information Preservation and the Exact Reproduction of Cross-Correlation. The Compression Theorem, to be presented in the following section, shows that the compressed signal can be used to compute the cross-correlation (3) with a very small amount of arithmetic. Because cross-correlation is known to be a sufficient statistic for important parameters such as signal amplitude, delay, and phase, it follows that the compressed signal retains all of this information.

The Compression Theorem

Most importantly, the compressed signal can be used to drastically reduce the amount of computation needed by a GNSS receiver to accurately obtain the correlation function $R_{rm}(\tau)$ in (3). The basis for this assertion is the *Basic Compression Theorem*.

The Basic Compression Theorem. The correlation function

$$R_{rm}(\tau) = \int_0^T r(t)\, m^*(t-\tau)\, dt \qquad (19)$$

can be computed by the alternate method

$$R_{rm}(\tau)=\int_0^T \tilde{r}(t)\,c^*(t-\tau)\,dt \qquad (20)$$

Proof:

$$R_{rm}(\tau)=\int_0^T r(t)\,m^*(t-\tau)\,dt$$

$$=\int_0^T r(t)\left[\sum_{k=0}^{N-1}\varepsilon_k c^*(t-kT_c-\tau)\right]dt$$

$$=\sum_{k=0}^{N-1}\int_0^T \varepsilon_k r(t)\,c^*(t-kT_c-\tau)\,dt$$

$$=\sum_{k=0}^{N-1}\int_0^T \varepsilon_k r(u+kT_c)\,c^*(u-\tau)\,du$$

$$=\int_0^T\left[\sum_{k=0}^{N-1}\varepsilon_k r(u+kT_c)\right]c^*(u-\tau)\,du$$

$$=\int_0^T \tilde{r}(u)\,c^*(u-\tau)\,du \qquad (21)$$

This theorem shows that $R_{rm}(\tau)$ can be calculated by cross-correlating the compressed signal $\tilde{r}(t)$ with the very short function $c(t)$. Furthermore, since we have already noted that the significant portion of $\tilde{r}(t)$ also spans a short time interval, the region surrounding the correlation peak of $R_{rm}(\tau)$ can be obtained with orders of magnitude less computation than would otherwise be required.

Generalizing the Compression Theorem. The Compression Theorem requires that the receiver-generated correlator reference waveform $m(t)$ be the concatenation of weighted and time-shifted chips, each chip having the same waveform $c(t)$, as stated by (4). However, some of the newer codes being considered for GPS and Galileo are the sum of two chipping sequences having distinct chip waveforms, such as the Galileo E5ab signal and several proposed GPS/Galileo L1 code modulations.

These codes have the form

$$m(t)=m_1(t)+m_2(t) \qquad (22)$$

where

$$m_1(t)=\sum_{k=0}^{N-1}\varepsilon_{1k}c_1(t-kT_c)$$

$$m_2(t)=\sum_{k=0}^{N-1}\varepsilon_{2k}c_2(t-kT_c) \qquad (23)$$

Here $m_1(t)$ is a sequence of chips characterized by the waveform $c_1(t)$, and $m_2(t)$ is a sequence of chips characterized by another waveform $c_2(t)$. With chip-multiplexed codes, such as CBOC(6,1,1/11), the weights ε_{1k} and ε_{2k} serve as "chip selectors" with values of +1, 0, and –1, and are never both nonzero. This restriction does not exist for codes such as Galileo E5ab.

By forming the two compressed signals

$$\tilde{r}_1(t)=\sum_{k=0}^{N-1}\varepsilon_{1k}r(t+kT_c)$$

$$\tilde{r}_2(t)=\sum_{k=0}^{N-1}\varepsilon_{2k}r(t+kT_c) \qquad (24)$$

and applying the compression theorem twice, the correlation $R_{rm}(\tau)$ can be calculated as

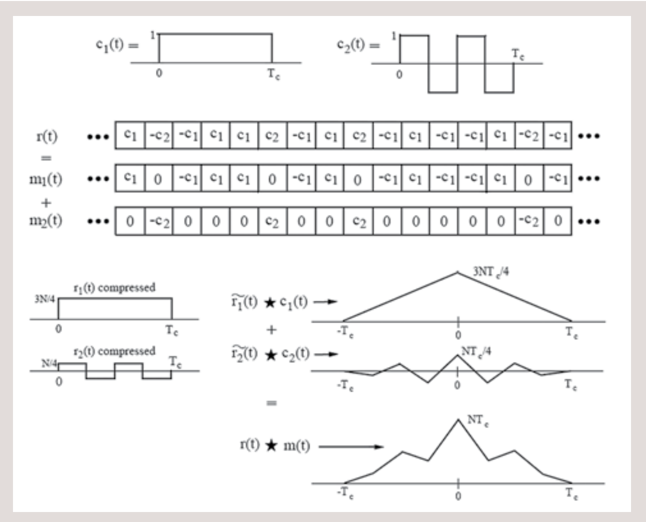


FIGURE 4 Compression of a TMBOC(2,1,1/4) signal

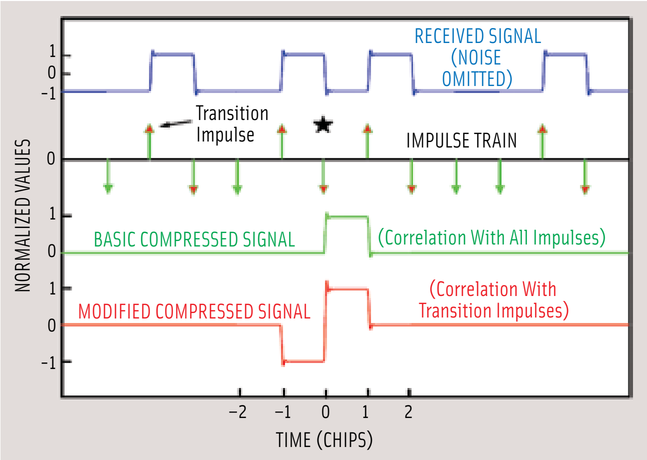


FIGURE 5 An example of modified compression

$$
\begin{aligned}
R_{rm}(\tau) &= \int_0^T r(t)\, m^*(t-\tau)\, dt \\
&= \int_0^T r(t)\left[m_1^*(t-\tau) + m_2^*(t-\tau)\right] dt \\
&= R_{rm_1}(\tau) + R_{rm_2}(\tau) \\
&= R_{\tilde{r}_1 c_1}(\tau) + R_{\tilde{r}_2 c_2}(\tau) \qquad (25)
\end{aligned}
$$

In this case $R_{rm}(\tau)$ is calculated efficiently as the sum of two correlations using the two compressed signals $\tilde{r}_1(t)$, $\tilde{r}_2(t)$ and the two single-chip waveforms $c_1(t)$, $c_2(t)$.

As an example of the application of the Generalized Compression Theorem, **Figure 4** illustrates the compression of a hypothetical TMBOC(2,1,1/4) signal.

Alternate Forms of Compression

In some applications we might find it useful to modify the structure of the compressed waveform by changing the weights ε_k and/or the shape of the chip waveform $c(t)$ in the reference waveform $m(t)$. An example is shown in **Figure 5**, where we consider basic compression as a correlation of $m(t)$ with the sequence of impulses whose weights are +1 for positive chips and –1 for negative chips.

In this figure the compression is altered by deleting those impulses that correspond to chips with no polarity change relative to the preceding chip. The deletion of an impulse is equivalent to setting its weight to zero. The resulting compressed waveform now has the appearance of a negative chip followed by a positive chip, instead of a single positive chip. This type of compressed signal is ideal for use as a discriminator in a delay-locked loop (DLL) for code tracking.

Applications

Certain types of signal-processing techniques and applications lend themselves to treatment using the Compression Theorem. Indeed, some are not practical without a signal-compression approach.

Calculation of High-Resolution Cross-Correlation Functions. In standard receivers the replica code $m(t)$ in the cross-correlation (3) is an ideal binary-valued (±1) waveform. Multiplications in the integrand become trivial and can be done at very high speed.

However, optimal processing theory requires the replica code to include the effects of filtering experienced by the received signal. In this case the replica code must have a multi-bit representation. This leads to millions of multi-bit multiplications in computing the cross-correlation function, which in some applications is computationally infeasible. These applications could include such things as ircraft landing, automated machine control, and any real-time kinematic application, especially if theoretically optimum performance is needed.

Furthermore, some new advanced signal processing methods require that the cross-correlation be computed for hundreds of high-resolution delay values τ. For example, consider a sampled P-code received baseband signal $r(t)$ with 12 MHz bandwidth sampled at 40 MHz. This sampling rate, which falls within the range used in typical GNSS receivers, essentially preserves all information in the received signal. However, the sample spacing of 7.5 meters is much too coarse for the calculation of a cross-correlation that might require a delay resolution of 10 centimeters or less.

A standard approach to solving this problem would be to interpolate between the samples of the received signal $r(t)$ prior to calculating the cross-correlation. However, this would require millions of multi-bit computations, exacerbated by the need to calculate the cross-correlation for many closely spaced values of delay τ.

Figure 6 shows how compression can drastically reduce the computational load in accurately computing such high-resolution cross-correlation functions. At the top of the figure are samples of a compressed P-code signal with 12 MHz baseband bandwidth. The sampling rate is 40 MHz. Instead of many millions of samples, the compressed signal requires at most 10-15 samples in its representation.

By the Compression Theorem, the cross-correlation of the received signal $r(t)$ and the replica code $m(t)$ can be accomplished by cross correlation of the compressed signal $\tilde{r}(t)$ and the fil-

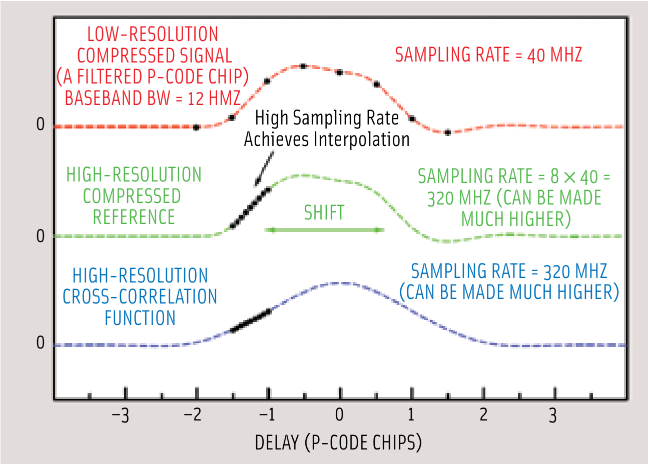


FIGURE 6 Computation of high-resolution cross-correlation functions

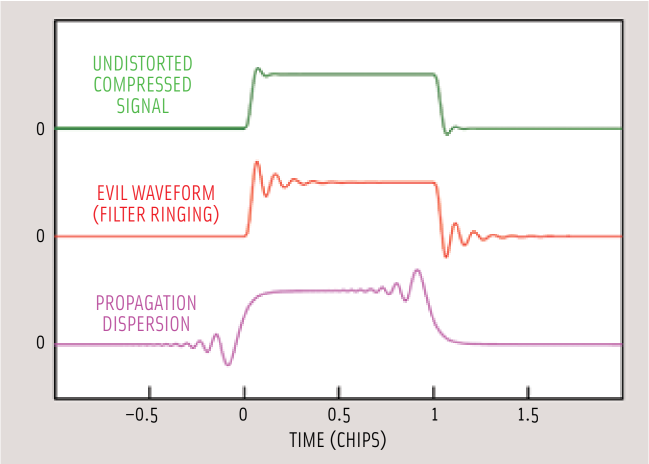


FIGURE 7 Observation of waveform anomalies

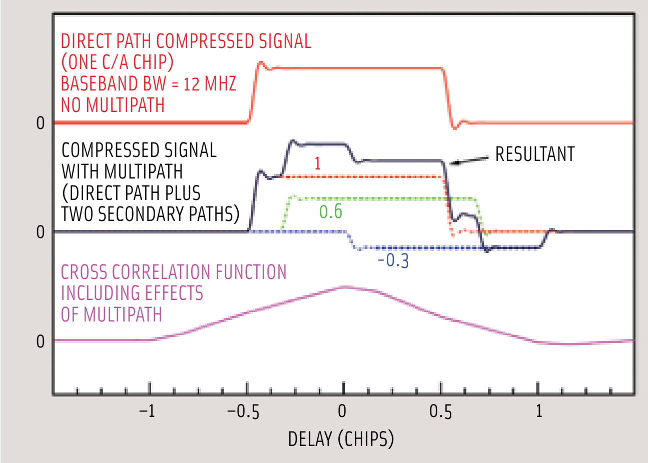


FIGURE 8 Extraction of multipath parameters

tered chip $c(t)$, a process requiring orders of magnitude fewer computations.

Furthermore, as indicated in the center of Figure 6, the interpolation required to obtain fine delay resolution can be automatically accomplished by using a fine resolution version of $c(t)$, which can be precalculated and stored in memory. The sampling rate for $c(t)$ is an integer multiple of the sampling rate for the compressed signal $\tilde{r}(t)$. The resulting cross-correlation has the desired fine resolution, as shown at the bottom of Figure 6.

Observation of Waveform Anomalies. The ability of compression to lift a chip waveform out of the noise makes anomalous ("evil") signals relatively easy to detect. **Figure 7** shows two examples. In the first, the signal contains RF ringing, which might be due to the failure of a filter in the satellite or receiver. In the second, the signal has experienced dispersion, which might occur as it travels through the ionosphere (the amount of dispersion has been exaggerated for clarity).

Ease of Analysis and Simulation. In signal processing research and development, analysis and simulation can often be made much more efficient by using a compressed signal instead of the received signal itself. This can improve work productivity in such areas as — but not limited to — signal waveform design, signal cross-correlation properties, code tracking techniques, effects of signal distortion, and advanced multipath mitigation methods.

Observation/Extraction of Multipath Parameters. **Figure 8** illustrates how the effects of multipath can be made clearly discernible by observation of the compressed signal. In this example the received signal is C/A coded and, in addition to the direct path signal, two multipath components are present, one in-phase with relative amplitude 0.6 and another 180 degrees out of phase with amplitude 0.3. The effects of multipath are clearly seen on the compressed signal, but are much harder to see in the cross-correlation function at the bottom of the figure.

Multipath Mitigation of Galileo Signals

In contrast to the simpler, legacy C/A-code, let's look at how signal compression works to mitigate the effects of multipath involving more complex GNSS signals that will begin appearing soon. For example, signal compression is essential for computational feasibility in recent multipath mitigation work by the author using a method called Multipath Mitigation Technology (MMT) on Galileo signals.

To illustrate the benefits of using compression, we will present comparative performance curves for MMT using the Galileo E5ab pilot signal. But first a description of the E5ab signal is needed.

Perhaps the most exciting of all GNSS signals, the Galileo E5ab signal is centered at 1191.795 MHz. The carrier is modulated by two orthogonal codes running at 10.23 chips/second, each of which is frequency-shifted by modulating it with a complex-valued subcarrier phasor rotating at 15.345 MHz. A negative angular rotation shifts the first code spectrum downward by 15.345 MHz to form the E5a signal. A positive angular rotation shifts the second code spectrum upward by 15.345 MHz to form the E5b signal.

Because the two codes are orthogonal, receivers can use just the E5a signal or just the E5b signal, or the combined signals, which we denote by E5ab. In particular, the RF front end of

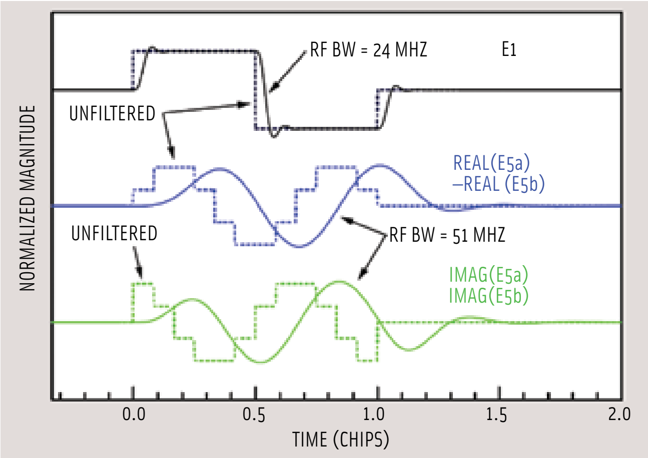


FIGURE 9 E1 and E5 compressed baseband signals

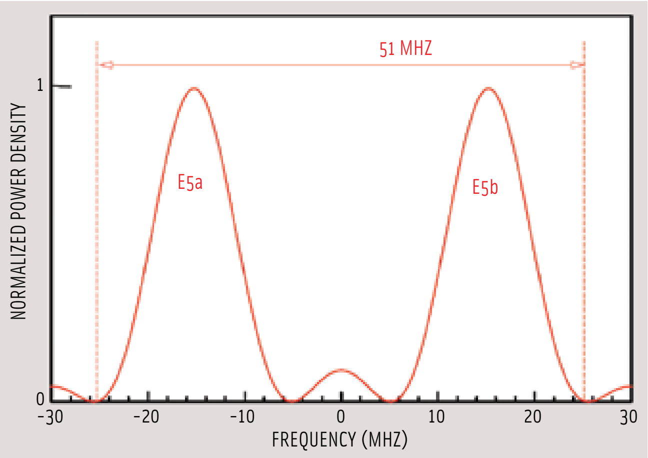


FIGURE 10 E5ab baseband power spectral density

GPS receivers designed for the L5 GPS signal can also receive the E5a signal because its center frequency is 1176.45 MHz, the same as GPS L5. We confine our attention to the E5ab signal because its wide bandwidth will produce the smallest errors due to thermal noise and multipath.

Unlike any other existing or proposed GNSS signal, the E5ab pilot signal has equal amounts of power in its real (I) and imaginary (Q) parts. As two orthogonal codes are used to generate the signal, two compressed signals must be formed in the receiver, one for E5a and one for E5b. Therefore, the generalization of the Compression Theorem applies.

Signal compression enables precise determination of high-resolution cross-correlation functions with orders of magnitude fewer calculations. Any GNSS signal can be compressed.

The bottom two panels of **Figure 9** show the structure of a single E5a chip and a single E5b chip, including the complex subcarrier modulation. Blue and green *dashed* lines show the respective real and imaginary parts of the unfiltered E5a chip; the *solid* lines represent the corresponding waveforms as they would be affected by restricting the video bandwidth to 25.5 MHz with a 4-pole Butterworth lowpass filter, corresponding to a 51 MHz RF bandwidth. The E5b chip has exactly the same waveforms, except the polarity of the real part is opposite that of the E5a waveform.

The filtered complex E5a and E5b chips in Figure 9 are the two functions $c_1(t)$ and $c_2(t)$ that are respectively cross-correlated with the compressed signals $\tilde{r}_1(t)$ and $\tilde{r}_2(t)$ to compute the correlation function $R_{rm}(\tau)$ in (25). The compressed signals are generated as indicated in (24), where the weights ε_{1k} and ε_{2k} are the respective polarity sequences of the E5a and E5b codes and T_c is the common chip duration. Except for scaling, noise, delay, and distortions due to multipath, the significant portions of $\tilde{r}_1(t)$ and $\tilde{r}_2(t)$ will respectively resemble $c_1(t)$ and $c_2(t)$.

The power spectrum of the E5ab signal is shown in **Figure 10**. The signal has a bandwidth of 51 MHz, which is the largest bandwidth of any current GNSS signal.

MMT Performance Using the Galileo E5ab Signal. Figure 11 compares the code-based multipath mitigation performance of MMT with the popular "double-delta" technique used in many current GNSS receivers. The signal consists of a direct path component and a secondary path one-half the amplitude of the direct path, with the secondary path delay on the horizontal axis.

The peak RMS ranging error using MMT is only 20 centimeters, while the peak error using a second-derivative correlator (equivalent to a double-delta correlator) is 92 centimeters. Furthermore, the significant errors for MMT span a much smaller multipath delay range. Overall, multipath errors using MMT are an order of magnitude smaller than with double-delta.

Using representative signal levels, **Figure 12** compares simulated multipath-induced phase errors for the two methods under the same conditions. The performance improvement with MMT is similar to that using code.

Implementation Issues

Although compression as defined by (5) and equivalently by (6) involves operations on a continuous waveform to produce a continuous compressed signal, practical compression is done digitally on a sampled waveform to produce samples of a compressed signal. It would be ideal if the received signal could be sampled with exactly N equally spaced samples per chip, where N is a positive integer. Then corresponding samples in each chip could be summed to produce a compressed signal also having N samples across the chip width.

However, Doppler on the signal and frequency error of the receiver clock cause the sampling points to be asynchronous relative to the received chips. To deal with this situation, each chip is divided into N

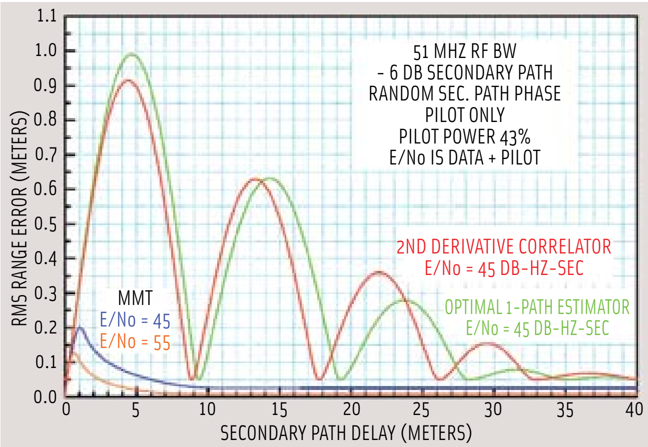


FIGURE 11 E5ab MMT comparative range error

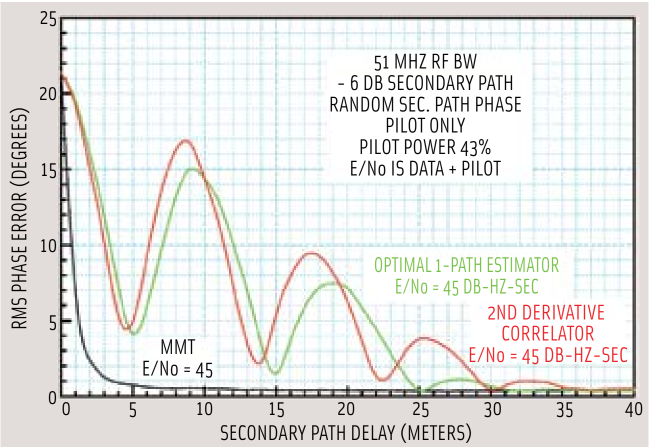


FIGURE 12 E5ab comparative phase error

equal-length segments, or bins. The bin into which each clocked sample point falls can be accurately calculated using both the code phase from the receiver's delay-locked loop and the integrated Doppler output from the receiver's phase-locked loop. The sample value can then be summed into that bin.

This process generally has the effect of "smearing" the compressed signal and slightly reducing its bandwidth. However, in applications requiring the reference chip *c(t)* to match the shape of the received chips, *c(t)* can be similarly pre-smeared prior to its storage in the receiver.

Summary

Signal compression is a linear time-invariant transformation that preserves signal amplitude, delay, and phase information, and makes the received signal structure visible.

It enables precise determination of high-resolution cross-correlation functions with orders of magnitude fewer calculations.

Any GNSS signal can be compressed.

Applications include advanced multipath mitigation, observation of received waveform anomalies, high-efficiency signal processing simulations, and others.

Acknowledgment

This article is based on a paper presented September 26, 2007, at the Institute of Navigation ION GNSS 2007 conference in Fort Worth, Texas.

Manufacturers

The Vision Correlator used in some GNSS receivers from **NovAtel, Inc.**, Calgary, Alberta, Canada, employs signal compression techniques based on the author's work described here.

Additional Resources

[1] "Binary-Valued Signal Modulation Compression for High Speed Cross-Correlation," U.S. Patent Application No. 10/336,103, filed 1/3/2003

[2] Fenton P. C., and Jason Jones, "The Theory and Performance of NovAtel Inc.'s Vision Correlator," *Proceedings of ION GNSS-2005*, Long Beach, CA, September 2005, pp. 2178-2186

[3] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, Second Edition, Wiley & Sons, New York, 2007, pp. 177-182

[4] Jones, J., P. Fenton, and B. Smith, "Theory and Performance of the PulseAperture Correlator," NovAtel Technical Report, NovAtel, Inc., Calgary, Alberta, Canada, June 9, 2004

[5] Weill, L. R., "Achieving Theoretical Accuracy Limits for Pseudoranging in the Presence of Multipath," *Proceedings of ION GPS-95*, Palm Springs, CA, September 1995, pp. 1521-1530

[6] Weill, L. R. "Achieving Theoretical Bounds for Receiver-Based Multipath Mitigation Using Galileo OS Signals," *Proceedings of ION GPS-2006*, Fort Worth, TX, September 2006, pp. 1035-1047

[7] Weill, L. R., "Conquering Multipath: The GPS Accuracy Battle," *GPS World*, Advanstar Communications, Eugene, OR, April 1997, pp. 59-66

[8] Weill, L. R,. "GPS Multipath Mitigation by Means of Correlator Reference Waveform Design," *Proceedings of the ION 1997 National Technical Meeting*, Santa Monica, CA, January 1997, pp. 197-206

[9] Weill, L. R. "Multipath Mitigation: How Good Can It Get with New Signals?," *GPS World*, Advanstar Communications, Eugene, OR, June 2003, pp. 106-113

[10] Weill, L. R., "Multipath Mitigation Using Modernized GPS Signals: How Good Can It Get?," *Proceedings of ION GPS-2002*, Portland, OR, September 2002, pp. 493-505

Author

Lawrence R. Weill received B.S. and M.S. degrees in electrical engineering from the California Institute of Technology. He earned the M.S. degree in mathematics at San Diego State University, and was awarded the Ph. D. in mathematics at the University of Idaho. He is professor of mathematics emeritus at California State University, Fullerton, and has operated his own consulting firm for 28 years. Weill is also one of the three technical founders of Magellan Systems Corporation, which in 1989 produced the world's first low-cost handheld GPS receiver for the consumer market. As an active researcher, Weill has published numerous papers on signal processing for GNSS, radar, sonar, optical sensor, and satellite communication systems, including substantial contributions to both the theoretical foundations and practical aspects of GNSS signal compression and multipath mitigation. IG